



**UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE CIENCIAS Y SISTEMAS**

Trabajo monográfico para optar por el Título de Ingeniero de Sistemas.

**“CORE GERENCIAL DEL SISTEMA DE INFORMACIÓN LABORAL DEL MINISTERIO
DEL TRABAJO (CORE-SIL)”**

ELABORADO POR:

Br. Miguel Ángel Siu Hernández.	2005-20968
Br. Guillermo Alfredo González Vera.	2009-29858
Br. Fregdy Uriel Torrez Montenegro.	2011-39245

TUTOR:

MsC. Evelyn Espinoza Aragón.

Managua, Nicaragua. Octubre 2017.

AGRADECIMIENTO

La culminación de esta tesis nos inspira gratitud y agradecimiento hacia personas que hicieron posible esta coronación, agradecemos su participaron en el trabajo realizado, ya que invirtieron su tiempo y conocimiento para ayudarnos a completar este proyecto. A las personas que vivieron con nosotros la realización de esta tesis, que compartieron pláticas y conocimientos sin esperar nada a cambio.

De manera especial, a los (as) compañeros (as) del Ministerio del Trabajo; directores generales, dirección superior, dirección de fortalecimiento institucional, área de planificación, área de proyectos y dirección informática. Gracias por compartir sus conocimientos, experiencias, aportes, además de la confianza brindada en el manejo de la información, por facilitarnos todos datos institucionales los cuales procesamos y son la base del contenido en este documento.

De igual manera a nuestra tutora Msc. Evelyn Espinoza Aragón quien estuvo siempre atenta a contestar nuestras interrogantes y fue una guía metodológica para realizar este trabajo.

Y por último al personal de la Universidad, ya que dentro del ámbito de su competencia colaboraron. Siempre nos atendieron con cortesía y solidaridad.

DEDICATORIA

Lo más importante a nuestras respectivas familias que nos apoyaron con su comprensión, transmitiéndonos emociones y alentando en todo el proceso de diseño y elaboración de nuestra Tesis, que con sus elogios nos invitaban a brillar con luz propia.

De igual manera dedicamos este trabajo de tesis a aquellas compañeras y compañeros del Ministerio del Trabajo, que día a día trabajan con amor, con ahínco, con ética, con voluntad, con vocación de servicio, con alma, con espíritu, con inteligencia, con tolerancia, con valores, con solidaridad, con visión de país. Los cuales ponen todos sus esfuerzos en la promoción, tutela y restitución del derecho al trabajo digno en nuestro país y son fuentes de inspiración para realizar este tipo de trabajo.

RESUMEN EJECUTIVO

Las instituciones de Gobierno en Nicaragua, requieren de mecanismos de soporte que las asistan en el diseño e implementación de sus modelos de negocio. Este proyecto plantea satisfacer dicho requerimiento para el Ministerio del Trabajo (MITRAB) a través del desarrollo del “Core Gerencial del Sistema de Información Laboral” habilitando una herramienta para una mejor dirección institucional, orientado al cumplimiento de misión institucional a través de la implementación del modelo de negocio enfocado a procesos para asegurar un resultado eficiente, cumpliendo con los requisitos institucionales sin caer en vicios tecnológicos.

La iniciativa se sustenta en la planificación y control en base a modelos de desempeño, considerando el modelo de referencia para el desempeño creado por la arquitectura empresarial federal (FEA) de los Estados Unidos de Norteamérica, que tiene un enfoque al valor y utiliza la arquitectura de servicios para configurar a la institución, de manera de orientarla a los resultados del negocio. El proyecto se sustenta en la Ingeniería de Negocios, disciplina que provee una metodología para el diseño e implementación de soluciones empresariales desde la definición de la visión y la estrategia, hasta la implantación de procesos de negocio y sus respectivas aplicaciones computacionales de apoyo.

para llevar a cabo este sistema se efectuaron los siguientes estudios presentados por capítulo:

En el primer capítulo destinado al estudio organizacional, se exhibe un análisis del sistema organizacional del MITRAB.

En el segundo capítulo trata del estudio preliminar, aquí se expresan todos los lineamientos requeridos para dar pase al siguiente nivel que es el análisis del sistema, para ello engloba todos los aspectos necesarios tales como: la inmersión en el entorno de desarrollo del futuro sistema y del MITRAB, la obtención de los requerimientos, restricciones y el análisis de factibilidad técnica, operativa y económica para conocer la viabilidad de la implementación.

En el tercer capítulo aborda el proceso de análisis fundamental que conlleva la creación de un sistema de este tipo de una forma concisa y congruente para el entendimiento del cliente de manera que refleje lo que debe realizar el sistema de información.

El cuarto capítulo es un esbozo del diseño de cómo se va a desarrollar el sistema de información, valiéndose para ello del recurso gráfico-estático que conllevan los diagramas y que son un potente instrumento para guiar el desarrollo de la versión operativa del sistema mediante un lenguaje de programación y un sistema de administración de base de datos.

El quinto capítulo hablaremos sobre el conjunto de pruebas del sistema, con el objetivo de garantizar el cumplimiento de los requisitos y requerimientos establecidos en el análisis del sistema.

Y por último en el sexto capítulo abordaremos los patrones para una correcta adopción del sistema por parte de los usuarios, y un fácil proceso de adición de funcionalidad y actualizaciones futuras del sistema.

INDICE

I.	Introducción.....	1
II.	Antecedentes	2
III.	Justificación.....	3
IV.	Objetivos.....	4
4.1.	OBJETIVO GENERAL	4
4.2.	OBJETIVOS ESPECÍFICOS.....	4
V.	Marco teórico.....	5
5.1.	ARQUITECTURA EMPRESARIAL DE SERVICIOS	5
5.1.1.	Proveer servicios.....	6
5.1.2.	Orientación al servicio.....	6
5.2.	MODELOS DE REFERENCIA EN ARQUITECTURA EMPRESARIAL	7
5.3.	SISTEMAS DE INFORMACIÓN WEB	11
5.3.1.	Descripción de entorno web.....	11
5.4.	GESTIÓN DE SEGURIDAD DE LA INFORMACIÓN.....	17
5.4.1.	Principios de la seguridad	17
5.4.2.	Riesgo.....	20
5.5.	TECNOLOGÍAS Y HERRAMIENTAS DE DESARROLLO	21
5.5.1.	Capa de presentación	22
5.5.2.	Capa de negocio	22
5.5.3.	Capa de datos.....	22
5.6.	METODOLOGÍA PARA EL DESARROLLO DE SISTEMAS DE INFORMACIÓN WEB ..	23
5.6.1.	Proceso unificado de rational (rup)	23
5.6.2.	Lenguaje unificado de modelado (uml)	25
5.6.3.	Modelo constructivo de costes (cocomo ii)	28
CAPITULO I:	Estudio Organizacional.....	29
1.1.	ENTORNO GENERAL DE LA INSTITUCIÓN.	29
1.1.1.	Misión del mitrab	29
1.1.2.	Visión del mitrab.....	30

1.1.3.	Objetivos institucionales del mitrab	30
1.1.4.	Funciones generales del mitrab	30
1.1.5.	Estructura organizacional del mitrab	32
1.2.	DESCRIPCIÓN DEL MODELO DE NEGOCIO	33
1.2.1.	Análisis de los procesos institucionales	34
1.2.2.	Análisis operativo de ti	36
1.2.3.	Análisis de los sistemas de información.....	42
1.2.4.	Análisis de la seguridad	49
CAPITULO II: Estudio Preliminar.....		57
2.1.	FACTIBILIDAD TÉCNICA	57
2.1.1.	Hardware (análisis de la arquitectura informática)	57
2.1.2.	Diseño lógico de red	59
2.1.3.	Propuesta de adquisición de hardware	61
2.1.4.	Software	68
2.2.	FACTIBILIDAD OPERATIVA	70
2.2.1.	Impacto en los procesos	70
2.2.2.	Impacto en el personal.....	71
2.2.3.	Conclusión de la factibilidad operativa.....	71
2.3.	FACTIBILIDAD ECONÓMICA	71
CAPITULO III: Análisis del Sistema.....		72
3.1.	DESCRIPCIÓN DEL SISTEMA DE INFORMACIÓN.....	72
3.2.	OBJETIVOS DEL SISTEMA DE INFORMACIÓN.....	72
3.2.1.	Definición y descripción de la información de entrada y salida	73
3.3.	ARQUITECTURA DEL SISTEMA DE INFORMACIÓN.	75
3.4.	TIPIFICACIÓN DEL SISTEMA DE INFORMACIÓN.....	76
3.5.	ALINEAMIENTO ENTRE EL PROCESO Y EL SISTEMA DE INFORMACIÓN.	77
3.6.	REQUERIMIENTOS FUNCIONALES.....	77
3.7.	REQUERIMIENTOS NO FUNCIONALES.....	78
3.8.	REQUERIMIENTOS DE RESTRICCIONES.....	79
CAPITULO IV: Diseño del Sistema.....		80

4.1.	DEFINICIÓN DE ACTORES.....	80
4.2.	DIAGRAMA DE PAQUETES.....	82
4.3.	MODELAMIENTO DE CASOS DE USO UML.....	82
4.4.	PLANTILLAS DE COLEMAN.....	84
4.1.	DIAGRAMAS DE SECUENCIA.....	85
4.2.	DIAGRAMAS DE COMUNICACIÓN.....	85
4.3.	DIAGRAMA DE CLASES.....	85
4.6.	MODELO DE DATOS.....	87
4.7.	DIAGRAMAS DE ESTADOS.....	88
4.8.	DIAGRAMAS DE DESPLIEGUE.....	88
CAPITULO V: Pruebas del Sistema.....		90
5.1.	PRUEBAS FUNCIONALES.....	90
5.2.	PRUEBAS NO FUNCIONALES.....	91
CAPITULO VI: Documentación del Sistema.....		93
6.1.	DISEÑO DEL SISTEMA.....	93
6.2.	CODIGO FUENTE.....	93
6.3.	MANUAL DE USUARIO.....	94
VI.	Conclusiones.....	95
VII.	Recomendaciones.....	97
VIII.	Bibliografía.....	98
IX.	Anexos.....	100

INDICE DE FIGURAS

Figura 01: Diagrama de modelos componentes de arquitectura FEA	9
Figura 02 Diagrama de descomposición jerárquica del PRM	10
Figura 03: Diagrama de áreas de medición del PRM	11
Figura 04: Estadísticas de uso de navegadores web	12
Figura 05: Estadísticas de uso de servidores web.	14

Figura 06: Estadísticas de uso de lenguajes de programación.	15
Figura 07: Seguridad de la información y seguridad informática.	18
Figura 08: Arquitectura Multicapa.	21
Figura 09: Organigrama MITRAB.	32
Figura 10: Método IDEF0	34
Figura 11: Evaluación por Dominio. Aplicación de COBIT 4.1 al MITRAB	37
Figura 12: Evaluación de la Documentación de Sistemas. MITRAB	43
Figura 13: Resultados de % de calidad de uno de los sistemas evaluados.	47
Figura 14: Controles Aplicados VS No Aplicados. ISO 27001-MITRAB	49
Figura 15: Resultados por dominio. ISO 27001-MITRAB	50
Figura 16: Cantidad de Dominios vs Cumplimiento. ISO 27001-MITRAB	53
Figura 17: Red de Datos MITRAB	55
Figura 18. Diagrama Logico de Red Actual	59
Figura 19. Diagrama Logica de Red Propuesto	60
Figura 20. Claves de Implementacion	60
Figura 21. Enfoque Modelo	75
Figura 22. Arquitectura del Sistema	76
Figura 23. Diagrama de Paquete	82
Figura 24. Diagrama de Casos de uso	84
Figura 25. Diagrama de Clases	86
Figura 27. Modelo de Datos	87
Figura 26. Diagrama de despliegue Actual	88
Figura 28. Diagrama de despliegue Propuesto	89
Figura 29. Postman	90
Figura 30. Gatling	92
Figura 31. Manual de Usuario del API	94

INDICE DE TABLAS

Tabla 1: Datos básicos de peticiones GET y POST.	16
Tabla 2: Nivel de Automatización de Procesos del MITRAB.	35
Tabla 3. Tecnologías del Software	68
Tabla 4. Definición de Actores	80

I. INTRODUCCIÓN

Los gobiernos alrededor del mundo están buscando nuevas formas de mejorar la calidad de los servicios que brindan a los ciudadanos a través del empleo de *tecnologías de información*. En particular en las Instituciones de Gobierno de Nicaragua en virtud de la iniciativa de Buen Gobierno, del entorno competitivo, del crecimiento exponencial de sus operaciones, de cara a dar respuestas a las demandas de la población y maximizar los beneficios hacia los ciudadanos están conscientes de la necesidad de contar con herramientas que les permitan a las instituciones estandarizar, gestionar y garantizar información de sus operaciones.

En este contexto MITRAB desde el año 2007, dio inicio a las transformaciones en sus operaciones como parte de las estrategias impulsadas por el Gobierno de Reconciliación y Unidad Nacional (GRUN). En este sentido la institución ha avanzado en la elaboración y cumplimiento de una nueva política laboral acorde a los principios que rigen el Plan Nacional de Desarrollo Humano, fortaleciendo su marco jurídico-administrativo y organizacional, así como redefiniendo sus políticas y estrategias, es decir robusteciendo sus operaciones en plano general, orientando sus resultados en beneficio de la población nicaragüense.

Como parte de esta evolución el MITRAB ha venido automatizando procesos claves en la institución de forma descentralizada, consientes que los sistemas de información representan un instrumento gerencial, que contribuyen a predecir el comportamiento de la institución, facilitando que las entidades sean más eficientes y transparentes.

En este sentido se propone el desarrollo del CORE GERENCIAL DEL SISTEMA DE INFORMACIÓN LABORAL (CORE-SIL) el cual proporcionará un marco que centralice las operaciones del MITRAB, interconectando los distintos sistemas de información de la institución, gestionando los procesos claves, monitoreando el comportamiento de los indicadores Socio-Laborales a nivel nacional y a su vez implementando mecanismos de seguridad a nivel de aplicación, proveyendo un API de integración para los módulos existentes y a nuevos módulos.

II. ANTECEDENTES

El MITRAB en el marco de su quehacer Institucional ha venido realizando avances significativos en el tiempo de cara a ejecutar sus operaciones de manera más eficiente. Desde el 2008 ha participado en proyectos de fortalecimiento técnico / tecnológico, entre los cuales se encuentran: El Sistema de Información Laboral (SISLAB), Sistema de Conciliaciones Laborales (SISCOLAB), Sistema de Condiciones de Higiene y Seguridad del Trabajo dirigidas a las MiPymes (SICHYSTRAB), Sistema Automatizado de Administración de Empresas (SAES), entre otros. Todos estos en busca de fortalecer los procesos críticos involucrados directamente en los servicios que brindan esta institución a nivel Nacional.

Los desarrollos de los sistemas se han dado mediante consultorías externas, estos proyectos se formularon y presentaron bajo la Metodología General para la Preparación y Evaluación de Proyectos de Inversión Pública del Sistema Nacional de Inversión Pública (SNIP)¹. No se ha definido una metodología propia para el desarrollo de software, aunque el Consejo Nicaragüense de Ciencia y Tecnología (CONICYT) ha estado trabajando en una normativa de desarrollo de sistemas, pero esto aún no ha sido oficializado ni transmitido como norma de desarrollo a ningún ente gubernamental, ya que aún está en desarrollo. Por otro lado, los procesos no automatizados son gestionados través de formatos impresos. Todos los datos son analizados por las áreas correspondientes para luego proporcionar al área de fortalecimiento institucional información y determinar el comportamiento de los indicadores socio-laborales a nivel nacional, cabe destacar que el punto expuesto anteriormente da origen al retraso en el manejo de los indicadores estadísticos a nivel nacional hasta de 1 año.

¹ Metodología General para la Preparación y Evaluación de Proyectos de Inversión Pública, Ministerio de Hacienda y Crédito Público, Dirección General de Inversiones Públicas.

III. JUSTIFICACIÓN

La implementación del CORE GERENCIAL DEL SISTEMA DE INFORMACIÓN LABORAL es una herramienta versátil, el cual proporcionará un marco que centraliza las operaciones del MITRAB. El uso de un sistema de esta magnitud al cumplimiento de las estrategias impulsadas por Gobierno de Reconciliación y Unidad Nacional. Además de dotar de una herramienta que proveerá rápidamente información vital para la toma de decisiones a nivel de la unidad Superior, elevando el nivel de eficiencia y transparencia de la institución.

El principal beneficio obtenido es la creación de un marco de trabajo que garantice la integración y seguridad:

- La interconexión de los sistemas con la posibilidad de poder manejar información de interés para la unidad superior de la institución, gracias a la creación de un API de integración.
- Con la incorporación de este sistema se podrá agilizar el procesamiento de los indicadores socio-laborales, esto se traduce directamente a que la unidad superior tendrá mayor criterio en la toma de decisión gracias a contar con información de manera más rápida y oportuna. Además de aportar al decremento de los tiempos de procesamiento en los procesos de gestión de estadísticas labora nacional los cuales aproximadamente son de 1 año de defese.
- En cuanto a la Seguridad de la información se garantizara la confidencialidad, integridad y disponibilidad de la información, un ejemplo concreto es la incorporación de un punto de acceso por medio de Inicio de sesión único (por sus siglas en inglés SSO Single-Sign On) que le permitirá trabajar con las autorizaciones correspondientes en los sistemas, es decir Se tendrá un sólo punto de control para la gestión de los usuarios, así como las autorizaciones correspondientes a cada sistema informático.

IV. OBJETIVOS

4.1. Objetivo General

Desarrollar Core gerencial del sistema de información laboral del ministerio del trabajo (CORE-SIL), un sistema de Información que centralice las operaciones del MITRAB, interconectando los distintos sistemas de información de la institución, gestionando los procesos claves, monitoreando el comportamiento de los indicadores Socio-Laborales a nivel nacional y a su vez implementando mecanismos de seguridad nivel de aplicación, proveyendo un API de integración para nuevos módulos.

4.2. Objetivos Específicos

- Realizar un estudio organizacional para determinar el entorno operativo-legal del MITRAB.
- Realizar un análisis de la factibilidad técnica, operativa y económica de la implementación del CORE-SIL.
- Recopilar la información necesaria de las fuentes de información que permita especificar los requisitos funcionales y no funcionales que debe cumplir el CORE-SIL.
- Definir los requisitos del CORE-SIL a partir de los resultados obtenidos en la etapa de Análisis de Requerimientos, a fin de tener obtener la estructura y los componentes del sistema.
- Generar el código fuente de los componentes especificados en la fase de Diseño del Sistema con el objetivo de crear un producto que cumpla los requisitos funcionales y no funcionales del CORE-SIL.
- Aplicar pruebas al código fuente generado en la etapa de Codificación para asegurar su correcto funcionamiento.
- Establecer la infraestructura necesaria para el uso y funcionamiento de CORE-SIL.
- Elaborar el manual de usuario correspondiente para la debida utilización del CORE-SIL.

V. MARCO TEÓRICO

A continuación, se entrega un marco teórico, como base fundamental para la comprensión del proyecto. En general, el marco teórico que se presenta a continuación se refiere a la arquitectura de servicios, como base para la arquitectura empresarial del CORE-SIL, implementada a través del modelo de desempeño de referencia obtenido de la Arquitectura Empresarial Federal (FEA)² del Gobierno de los Estados Unidos de Norteamérica, que finalmente se concretó utilizando algunos indicadores aceptados como buena práctica por empresas de clase mundial agrupadas en el Centro de Productividad y Calidad de Norteamérica (APQC).³

5.1. Arquitectura Empresarial de Servicios

La meta de instaurar una arquitectura de servicios para la organización interna es asegurar que se comporte correctamente en la interacción con sus clientes.

Los procesos agregan consistencia y proporcionan la dirección en cómo las actividades de entrega del servicio deben ser realizadas, pero ¿cómo asegurar que las actividades estén siendo realizadas con un alto nivel de la calidad? Las pruebas de calidad se podrían utilizar para proporcionar un nivel de la garantía de calidad, pero son limitadas en tipo y número que es posible diseñar en un proceso antes de que pierda eficiencia. La verdadera función entonces de las pruebas de calidad, es que los responsables de ellas obliguen a los responsables de los procesos a realizar un buen trabajo.

Establecer *buenos procesos* proporciona los cimientos para la calidad de la entrega del servicio, pero los procesos solamente no garantizan calidad, debido a que la gente que los realiza, muchas veces no está motivada. Existen variadas herramientas de gestión que facilitan el trabajo (entrenamiento, reestructuración, incentivos económicos, etc.) pero no son suficientes para mejorar la motivación

² Office of Management and Budget's (OMB), 2006. FEA Consolidated Reference Model Version 2.1: www.egov.gov.

³ American Productivity & Quality Center (APQC), Open Standards Benchmarking Collaborative (OSBC). www.apqc.org/osbc.

de manera sustentable. La satisfacción profesional se cita a menudo como motivador fuerte para muchos y es conducida, entre otras cosas, por la percepción de dignidad y significado en las tareas y deberes del trabajo. Un empleado motivado, es aquel que tiene claro por qué su trabajo es importante y cómo se enmarca y colabora con los propósitos de la organización.

5.1.1. Proveer Servicios

La calidad de los productos obedece a un conjunto de criterios que evalúan qué tan bien construido está, lo que puede ser independiente de la satisfacción del cliente. Al contrario, la inseparabilidad del servicio implica que el cliente está involucrado en su producción y, por lo tanto, la última medida de calidad de servicio es la satisfacción del cliente. La diferencia está entonces en ser enfocado al interior (a la calidad de producto, proceso, etc.) y enfocado al exterior (a la calidad del servicio, satisfacción del cliente).

5.1.2. Orientación al Servicio

Por todo lo que ya se ha mencionado, es muy importante para una organización de Gobierno recordar siempre por qué existe y qué entrega, más que enfocarse solamente a cómo lo entrega.

La orientación al servicio ayuda a la organización a recordar qué entrega a sus clientes y por qué. La gran particularidad de las organizaciones Gubernamentales es que necesitan de un alto grado de flexibilidad en sus operaciones y no es posible codificar cada actividad que es necesario llevar a cabo para proveer un servicio en particular. Una orientación solamente al proceso puede generar comportamientos limitados a actividades definidas incluso en desmedro de la calidad de la entrega del servicio. Al contrario, la orientación al servicio genera un comportamiento orientado a alcanzar una calidad de servicio determinada, sin importar si las actividades a realizar están definidas por procesos, situación que claramente generará mayor satisfacción al cliente.

En resumen, la orientación a procesos enfatiza la efectividad y eficiencia, y la

orientación al servicio enfatiza los entregables de la organización (sus resultados). Entonces, una combinación eficiente de orientación a procesos y al servicio asegurará la entrega eficiente de servicios de calidad.

5.2. Modelos de Referencia en Arquitectura Empresarial

El modelo de desempeño a considerado por este proyecto, se basa en el modelo Performance Reference Model (PRM) contenido en el Federal Enterprise Architecture (FEA)⁴, así como también en el modelado Integration Definition for Function Modeling (IDEF0). Establecido, en conjunto, por la Oficina de Gestión y Presupuestos (OMB), la Oficina de Gobierno Electrónico (E-Gov) y Tecnologías de Información (IT), con el apoyo de la Administración de Servicios Generales (GSA) y el Consejo de Jefaturas de Información Federal (Federal CIO Council), el Programa de Arquitectura Empresarial Federal (FEA Program), comprende un modelo orientado al negocio del Gobierno Federal de los EEUU, que genera un lenguaje común y un marco para describir y analizar inversiones en TI, promover la colaboración y finalmente transformar el Gobierno Federal en una organización centrada en el ciudadano, orientada a los resultados y basada en el mercado, como ha sido dispuesto en la Agenda de Gestión del Presidente (PMA).

Los modelos de referencia contenidos en el FEA han sido diseñados para facilitar el análisis inter agencias y la identificación de la duplicidad de inversiones, brechas y oportunidades de colaboración entre las agencias. En conjunto, los modelos de referencia comprenden un marco para describir elementos importantes de la Arquitectura Empresarial Federal, de manera consistente. A continuación, se introducen de manera simple los modelos de referencia.

- **Performance Reference Model (PRM):** es el modelo de referencia de desempeño que este proyecto utiliza. Es un marco para medir el desempeño a través de todo el Gobierno Federal de EEUU, que le permite a las agencias gestionar el negocio del gobierno a nivel estratégico, entregando formas de

⁴ Office of Management and Budget's (OMB), 2006. FEA Consolidated Reference Model Version 2.1: www.egov.gov.

utilizar la EA para medir el éxito de las inversiones en TI y su impacto sobre los resultados estratégicos. El PRM cumple estas metas estableciendo un lenguaje común, a través del cual las EA's de las agencias describen sus resultados y métricas utilizados para alcanzar los objetivos del programa y del negocio. La estructura del PRM está diseñada para expresar claramente las relaciones causa – efecto entre entradas y salidas. Esta “Línea de Visibilidad” se implementa a través del uso de una estructura jerárquica: Área de Medición, Categoría de Medición, Grupo de Medición e indicador. Más adelante en el informe se abordará una descripción en detalle del PRM.

- **Integration Definition for Function Modeling (IDEF0):** es un método diseñado para modelar decisiones, acciones y actividades de una organización o sistema. Estos modelos consisten en una serie de diagramas jerárquicos junto con unos textos y referencias cruzadas entre ambos que se representan mediante unos rectángulo o cajas y una serie de flechas. Uno de los aspectos de IDEF0 más importantes es que como concepto de modelización va introduciendo gradualmente más y más niveles de detalle a través de la estructura del modelo. De esta manera, la comunicación se produce dando a lector un tema bien definido con una cantidad de información detallada disponible para profundizar en el modelo.
- **Business Reference Model (BRM):** es el modelo de referencia del negocio, que provee de un marco para facilitar una perspectiva funcional (en vez de una organizacional) de las líneas de negocio del gobierno federal, incluyendo sus operaciones internas y servicios para ciudadanos, independientemente de las agencias y oficinas que las realizan. Esto promueve la colaboración entre agencias y oficinas y sirve como fundación base para la Arquitectura Empresarial Federal y la estrategia E-Gov.
- **Service Component Reference Model (SRM):** es el modelo de referencia para los componentes de servicio. El SRM es un marco enfocado al negocio, que clasifica los componentes de servicio de acuerdo a cómo soportan al negocio y a los objetivos de desempeño. Sirve para identificar y clasificar componentes de servicio horizontales y verticales que soportan agencias

federales y sus inversiones en TI y activos. El modelo ayuda en recomendar características de servicio que puedan soportar el re uso de los componentes de negocio y servicios a través del gobierno federal.

- **Technical Reference Model (TRM):** es el modelo de referencia técnico. El TRM es un marco que categoriza los estándares y tecnologías para soportar y habilitar la entrega de los Componentes de Servicios y capacidades. Además, unifica los modelos técnicos existentes por agencia y la propuesta del E-Gov, entregando una fundación para avanzar en la reutilización y estandarización de la tecnología y Componentes de Servicio desde una perspectiva global del gobierno.
- **Data Reference Model (DRM):** Es el modelo de referencia de datos. El DRM ha sido diseñado con la intención de promover la identificación, uso e intercambio apropiado de los datos y la información a través del gobierno federal por medio de la estandarización de los datos en contexto, intercambio y descripción.

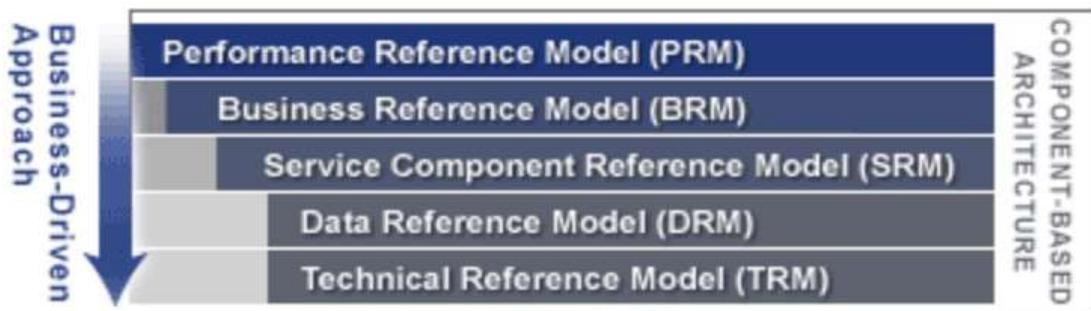


Figura 01 Diagrama de modelos componentes de arquitectura FEA

Fuente: FEA Consolidated Reference Model Version 2.1: www.egov.gov.

Como se enuncia anteriormente, el modelo de desempeño para el MITRAB. se basa en el modelo de referencia de desempeño (PRM) de la Arquitectura Empresarial Federal (FEA). Sobre este modelo se basa el proyecto ya que determina el formato y la entrega de la información que soportará dicho proceso.

Para profundizar en el PRM, es necesario revisar con mayor detalle su estructura y en particular sus áreas de medición. Como se ha mencionado, el PRM tiene una

estructura jerárquica compuesta por áreas de medición, categorías de medición, grupos de medición e indicadores de medición, que se pueden ver en el diagrama de descomposición jerárquica y es caracterizadas de manera genérica a continuación.

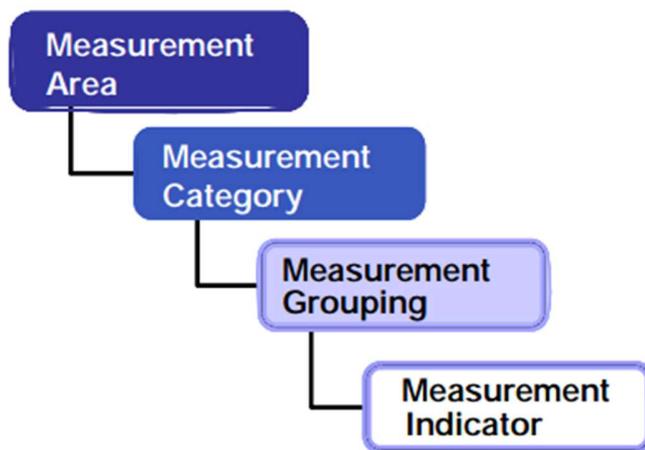


Figura 02 Diagrama de descomposición jerárquica

Fuente: FEA Consolidated Reference Model Version 2.1, www.egov.gov.

- **Áreas de Medición:** representan el marco organizativo de alto nivel del PRM y captura los aspectos de desempeño a nivel de salidas o resultados. Esta capa está directamente relacionada a los objetivos de desempeño establecidos a nivel institucional. El PRM incluye seis áreas de medición: Resultados de la misión y del Negocio, Resultados del Cliente, Procesos y Actividades, Capital Humano, Tecnología y Otros Activos Fijos.
- **Categorías de Medición:** Conjuntos dentro de cada área de medición que describen los atributos o características a medir.
- **Grupo de Medición:** Refinamiento de detalle de las categorías en tipos específicos de indicadores de medición por ámbito de cada categoría.
- **Indicadores de Medición:** Las mediciones específicas implementadas a través de métricas.

Como se ha mencionado, las seis áreas de medición del PRM representan el marco organizativo de alto nivel, capturando los aspectos del desempeño a nivel

de salidas o resultados; esto permite analizar tanto el enfoque a la creación de valor del PRM, como su capacidad de reflejar relaciones causa efecto y mostrarlo de manera gráfica, a través del diagrama que las representa.

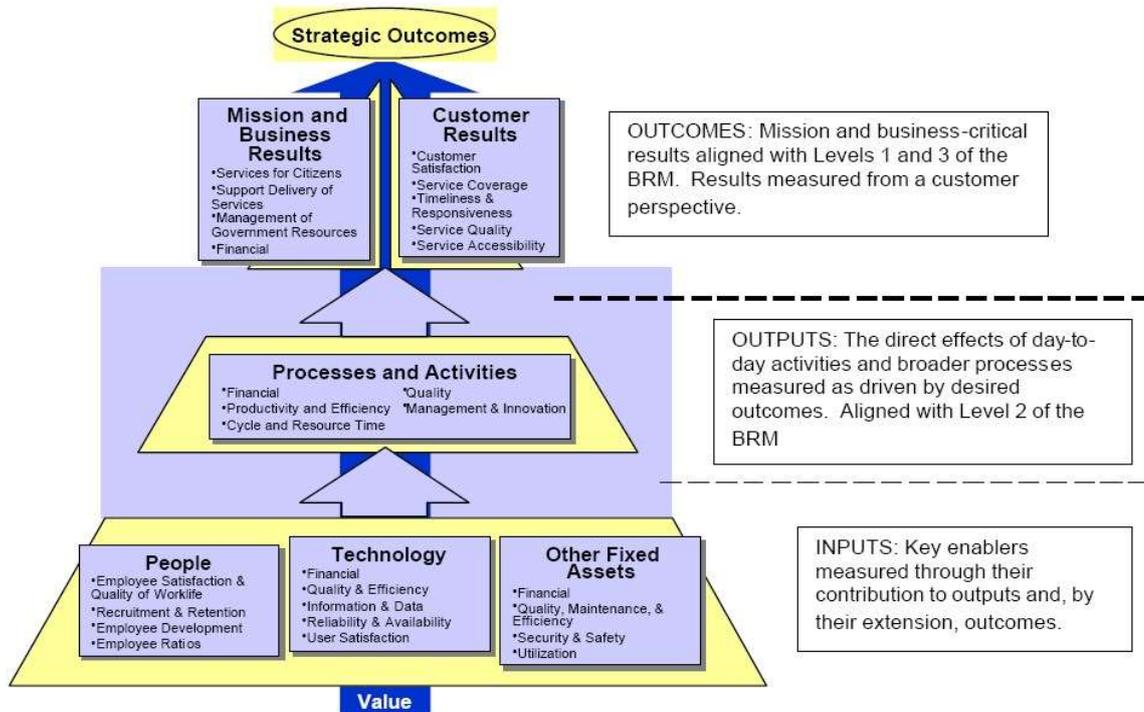


Figura 03: Diagrama de áreas de medición del PRM

Fuente: FEA Consolidated Reference Model Version 2.1, www.egov.gov

5.3. Sistemas de Información Web

Las aplicaciones web son un tipo de aplicación cliente- servidor donde el usuario accede a través de internet o una intranet mediante un navegador, el protocolo mediante el cual se comunican (HTTP) están estandarizados y no han de ser creados por el programador de aplicaciones.

5.3.1. Descripción de entorno web

Describimos un entorno web como aquel en el que un cliente utiliza un protocolo de transporte para llegar a interactuar con un servidor web. Si bien originalmente los entornos de web (World Wide Web) se basaban en recuperar elementos sencillos de textos que se iban relacionando entre sí por medio de hiperenlaces, estos entornos han ido ganando en complejidad y funcionalidad llegando a permitir

en la actualidad procesos y transacciones complejas.

5.3.1.1. Tecnologías

5.3.1.1.1. Cliente web

Entendemos como cliente web aquel que inicia la comunicación con el servidor con el fin de realizar algún tipo de transacción de información.

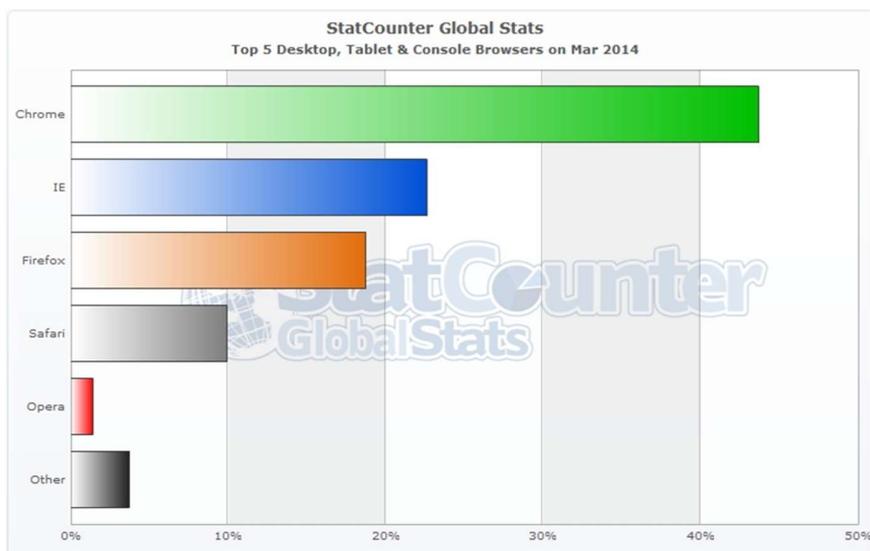


Figura 04: Estadísticas de uso de navegadores web

Fuente: <http://gs.statcounter.com>

Como punto de contacto entre el contenido de la aplicación y usuario final es claramente una potencial vulnerabilidad a la hora de realizar cualquier tipo de ataque.

Lo más común a la hora de pensar en un cliente web suele ser un navegador (Firefox, Chrome, Opera, Internet Explorer...) pero la estandarización de los entornos web y sus protocolos ha hecho que muchos de los clientes que se conectan a las aplicaciones web no sean navegadores como tal, como es el caso de las aplicaciones móviles, televisores (Smart-tv), etc.

5.3.1.1.2. Conexión

Para realizar la interconexión entre cliente y servidor debe hacerse uso de

diversos protocolos de distinto nivel. Si bien podrían utilizarse distintas opciones, los que se han convertido en estándares oficiales o de facto son los siguientes:

- **HTTP:** Protocolo a nivel de aplicación que está establecido como estándar en la comunicación web. Especialmente pensado para transferencia del conocido como hipertexto se compone de diversas peticiones posibles, así como de una serie de funcionalidades e informaciones asociadas como pueden ser, entre otros, códigos de respuesta, sesión asociada, conexiones persistentes y versión segura del protocolo, llamado HTTPS basado en protocolos SSL/TLS.
- **TCP:** Protocolo a nivel de transporte encargado de transportar las peticiones HTTP mediante el envío de paquetes TCP. Se trata de un protocolo confiable, que garantiza la recepción y ordenación correcta de los paquetes enviados. Si bien se utiliza como estándar en la mayor parte de aplicaciones asociadas a Internet como web, correo electrónico o FTP, en ocasiones puede ser sustituido por el protocolo UDP, similar al protocolo TCP, pero más sencillo y que no dispone de tantos sistemas para la detección de errores.
- **IP:** Protocolo a nivel de red no orientado a conexión, utilizado para el envío de información entre nodos.
- **Otros:** A nivel de enlace y físico existen numerosos protocolos disponibles para la transmisión de información, Ethernet, Wifi, Token Ring... Cualquiera de los anteriores protocolos es susceptible de recibir ataques, así como de adolecer de vulnerabilidades de seguridad. Dado que queda fuera del alcance de este proyecto no nos adentraremos más allá.

5.3.1.1.3. Servidor web

Un servidor web será aquel que recoja y procese la petición del cliente, bien devolviendo información, bien llevando a cabo algún proceso. Existe infinidad de software distinto para procesar las distintas peticiones. Podemos diferenciar dos tipos de respuestas a una petición:

- **Estática:** Que consiste en devolver el elemento solicitado por el cliente sin más. Por ejemplo, el servidor recibe una petición para entregar un archivo HTML almacenado en el disco, que será recuperado y devuelto sin más. Para todas las peticiones que se reciban, la respuesta será la misma.
- **Dinámica:** La más común en la actualidad. La petición es recogida por el servidor que hará uso de un conjunto de instrucciones definidas en algún lenguaje de programación para generar la respuesta. Por ejemplo, se solicita al servidor una página HTML con la información meteorológica del día actual en Managua, al tratarse de una información variante en el tiempo (dinámica) requerirá de un proceso consulta y posterior generación del HTML por parte del programa. Nos encontraríamos entonces ante una aplicación web. En el caso de estas últimas, es necesario un lenguaje de programación que procese la petición debidamente

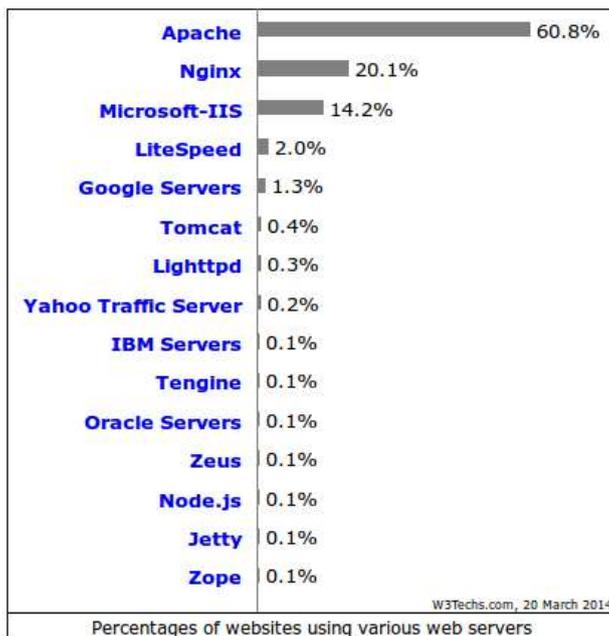


Figura 05: Estadísticas de uso de servidores web.
Fuente <http://w3techs.com>

Es importante señalar que la utilización de uno u otro lenguaje de programación o servidor, no tiene en sí implicaciones en la seguridad de las aplicaciones más

allá del mantenimiento de productos actualizados y, sobre todo, de la gestión y utilización correctas de los mismos.

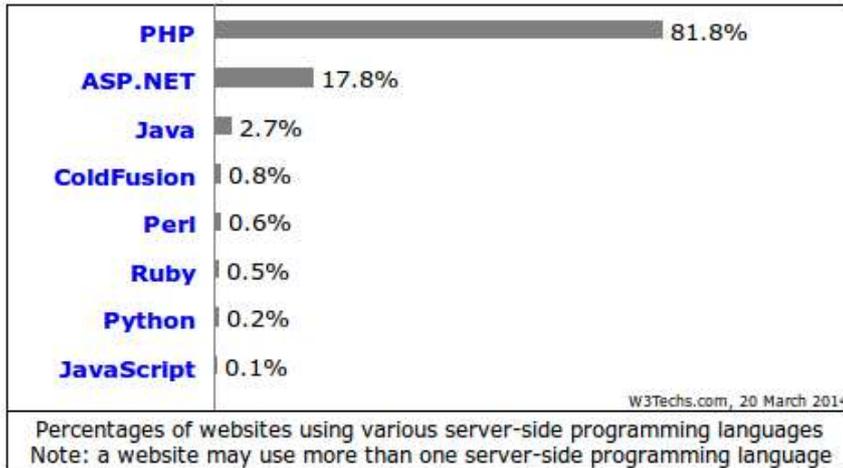


Figura 06: Estadísticas de uso de lenguajes de programación.

Fuente <http://w3techs.com>

5.3.1.1.4. Protocolo de una petición

Dentro del protocolo HTTP existen distintas peticiones posibles a un servicio. Las más comunes son las peticiones GET y POST que, si bien son similares, difieren en la forma en que se envía información desde el cliente al servidor web. Veremos el funcionamiento genérico de una petición y luego diferenciaremos las dos posibilidades anteriormente descritas. Tomaremos como ejemplo una petición realizada desde un navegador web, que utilice el protocolo HTTP y para simplificar el ejemplo obviaremos algunas de las distintas posibilidades existentes (redirecciones, datos cacheados, otros protocolos.)

- a) Inicio de la petición: El usuario introduce una URL en la interfaz del navegador y ejecuta la petición.
- b) Resolución de direcciones IP: Habitualmente la URL introducida es una dirección alfanumérica que debe ser convertida a una dirección IP. Para ello se dispone del protocolo DNS, mediante sus servidores y sistemas de caché se realiza esa traducción para que el protocolo HTTP sea capaz de realizar la petición correctamente.

- c) Establecimiento de conexión con dirección IP: Apoyándose en el protocolo TCP se realiza la apertura de un socket a la dirección IP correspondiente.
- d) Envío de la petición: En este momento se realiza la petición en sí, sea de tipo GET o POST con todos los datos asociados a la misma que son incluidos en la petición
- e) Envío de la respuesta: Una vez procesada la petición por parte del servidor web se realiza la respuesta con el contenido devuelto como parte del contenido HTTP.
- f) Mostrado de la información: El navegador web muestra en su interfaz la información suministrada en la respuesta.

La diferencia fundamental entre peticiones GET y POST es el modo en que se adjuntan los datos a la petición, aunque también existen otras diferencias:

	GET	POST
Cacheo	Se cachea	No se cachea
Tipo de codificación	application/x-www-form-urlencoded	application/x-www-form-urlencoded o multipart/form-data (si ficheros)
Historial	Almacenadas en los navegadores	No almacenadas en los navegadores
Límite de tamaño	Datos enviados a través de la URL, su Sin restricciones de tamaño, aunque longitud máxima es 2048 caracteres	Sin restricciones de tamaño, aunque los servidores suelen limitarlo
Tipo de contenido	Exclusivamente caracteres ASCII	Sin restricciones. Se acepta incluso contenido binario

Visibilidad/Seguridad	Datos visibles en la URL, envío no seguro	Datos no visibles. No obstante, tampoco se puede considerar seguro
------------------------------	-------------------------------------------	--------------------------------------------------------------------

Tabla 1: Datos básicos de peticiones GET y POST.

Fuente: <http://www.w3schools.com>

5.4. Gestión de Seguridad de la Información

5.4.1. Principios de la Seguridad

Partiendo de la base de que nunca es posible alcanzar la seguridad absoluta, se entiende la seguridad como un equilibrio entre los riesgos asumibles frente a las expectativas y necesidades establecidas.

Así mismo, debe entenderse la seguridad como un proceso que de forma cíclica debe ir revisando y mejorando las políticas y medidas establecidas.

Si ahondamos en el contenido propio de este proyecto, podemos ver que durante mucho tiempo los conceptos de seguridad informática y seguridad de la información han sido presentados como análogos en muchas referencias, cuando no debe ser así y más bien se pueden definir como conceptos complementarios.

5.4.1.1. Seguridad de la Información

El propósito de la seguridad de la información es construir un sistema que tenga en cuenta todos los posibles riesgos en la administración de la información, estén o no relacionados con la seguridad informática.

En concreto, el principio de la seguridad informática es prevenir el acceso, modificación, uso o destrucción no autorizados de la información, independientemente del soporte en el que se encuentre ésta.

Se definen unos principios básicos que deben cumplirse cuando nos referimos a la información:

- **Confidencialidad:** Debe prevenirse el revelado de información a personas o sistemas no autorizados.

- **Integridad:** La información debe mantenerse libre de cualquier modificación o alteración de su contenido que no sea autorizada.
- **Disponibilidad:** Así mismo, la información deberá estar accesible y disponible para las personas o sistemas autorizados cuando así lo requieran.
- **Autenticidad/Autenticación:** Es necesario poder garantizarse que el emisor de la información es quien dice ser.
- **No repudio:** En determinados entornos, no debe haber posibilidad por parte del emisor de rechazar la autoría de la información.

5.4.1.2. Seguridad Informática

Esta disciplina es la encargada de implementar las medidas técnicas de protección de la información. Es decir, el despliegue de las tecnologías

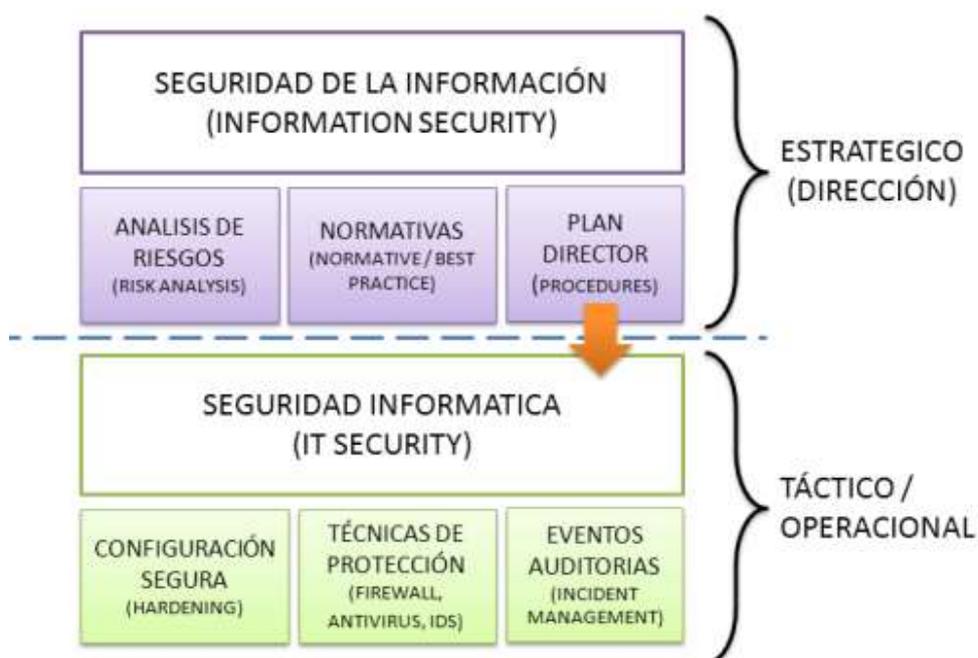


Figura 07: Seguridad de la información y seguridad informática.

Fuente: <http://www.owasp.com>

necesarias (desde el control de acceso, antivirus, cortafuegos, alertas, etc.) que, articuladas junto con prácticas de dirección de las tecnologías de información, establecen las formas de actuación y defensa frente a situaciones de fallos y brechas de seguridad.

Así la seguridad informática debe proteger una serie de activos en lo que no sólo se encuentran los datos o la información, sino que pueden incluirse:

- **Datos:** Normalmente, el principal valor a proteger, la información contenida en el sistema suele ser lo más valioso y principal objetivo de los atacantes.
- **Usuarios:** Las personas que utilizan el sistema y acceden a la información contenida en el mismo.
- **Infraestructura:** El entorno físico que da cabida a todo el sistema. Servidores, elementos de red, edificios, etc.

Además de las distintas técnicas para asegurar el sistema, uno de los aspectos más importantes en la seguridad informática de una organización es la puesta en marcha de una política de seguridad. Así, se deberán definir los distintos procedimientos, acciones en caso de emergencia, fomento de la importancia de la seguridad dentro de los integrantes de la organización, revisión de los puntos establecidos, puesta en marcha de mejoras, etc.

5.4.1.3. Vulnerabilidad

“Dado que una aplicación posee un conjunto de activos (recursos de valor como los datos en una base de datos o en el sistema de archivos), una vulnerabilidad es una debilidad en un activo que hace posible a una amenaza. Así que una amenaza es un caso potencial que puede dañar un activo mediante la explotación de una vulnerabilidad. Un test es una acción que tiende a mostrar una vulnerabilidad en la aplicación” [OWASP, 2008].

“Estimación de la exposición efectiva de un activo a una amenaza. Se determina por dos medidas: frecuencia de ocurrencia y degradación causada” [Magerit, 2006].

“La vulnerabilidad puede definirse como la falta de una garantía que causa una debilidad y que podría ser explotada. Las vulnerabilidades pueden surgir desde el

diseño, implementación y configuración de hardware, software o procesos “[Scambray et al., 2010, p. 26].

5.4.1.4. Vulnerabilidad de las Aplicaciones Web

Son los fallos que se pueden presentar en el análisis, diseño e implantación de las aplicaciones web, así como también en la correcta configuración una vez puesta en marcha, estos fallos no pueden ser resueltos íntegramente y sólo se puede minimizar su efecto tomando las políticas de seguridad adecuadas y estando atento a los problemas de seguridad o a las nuevas vulnerabilidades que puede aparecer en cualquier momento [OWASP, 2008].

5.4.2. Riesgo

Efecto de la incertidumbre en los objetivos [ISO/IEC27000, 2014].

Estimación del grado de exposición a que una amenaza se materialice sobre uno o más activos causando daños o perjuicios a la organización [Magerit, 2006].

Combinación de la probabilidad de un evento y sus consecuencias [NTP-ISO/IEC17799, 2007].

5.4.2.1. Análisis de Riesgo

El análisis de riesgo es un proceso sistemático para estimar la magnitud de los riesgos al que está expuesta una organización. El análisis de riesgo permite determinar cómo es, cuánto vale y como de protegidos se encuentran los activos [Magerit, 2006].

Proceso de comprender la naturaleza del riesgo para determinar el nivel del riesgo [ISO/IEC27000, 2014].

5.4.2.2. Sistema de Gestión de Seguridad de la Información

Es un sistema de gestión que comprende la política, la estructura organizativa, los procedimientos, los procesos y los recursos necesarios para implantar la gestión

de la seguridad de la información. Este sistema es la herramienta de que dispone la dirección de las organizaciones para llevar a cabo las políticas y los objetivos de seguridad (integridad, confidencialidad, disponibilidad, asignación de responsabilidad, autenticación, etc.) [AENOR, 2004].

“Debido a la complejidad de llevar a cabo un plan de seguridad, es necesario una metodología. Por este motivo aparecieron los sistemas de gestión de la seguridad de la información (SGSI). En general, cualquier sistema de gestión de la seguridad, tendrá que comprender la política, la estructura organizativa, los procedimientos, los procesos y los recursos necesarios para implantar la gestión de la seguridad de la información dentro de una organización” [Coloran, 2008].

5.5. Tecnologías y Herramientas de Desarrollo

Se propone trabajar con tecnologías basados en la web para un despliegue multiplataforma, en el caso de los dispositivos móviles el sistema será adaptativo según las dimensiones la pantalla del dispositivo. Además, la arquitectura será multicapa se garantiza un producto estable, escalable y robusto.

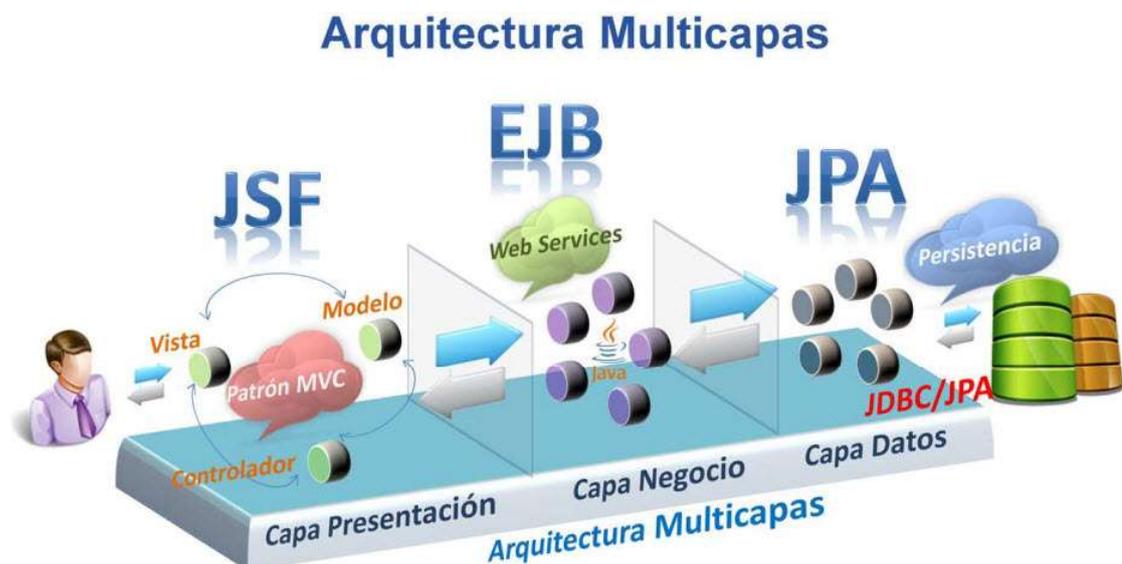


Figura 08: Arquitectura Multicapa.
Fuente: <http://www.oracle.com>

5.5.1. Capa de Presentación

- **HTML5:** última versión de HTML, que se utilizará para la estructura y presentación de las páginas web.
- **CSS3:** Hojas de estilo de cascada que se usará para el aspecto y la forma de las páginas web.
- **JavaScript:** Lenguaje interpretado que se embeberá en las páginas web, provocando que las instrucciones sean analizadas y procesadas por el navegador en el momento que deben ser ejecutadas.

5.5.2. Capa de Negocio

- **Java:** Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Lo realmente impresionante de esta tecnología es que se alza sobre todas las plataformas de hardware y ejecuta los códigos de bits que, en vez de ser específicos de una arquitectura, incluyen instrucciones para ser ejecutados en una máquina virtual, siendo así uno de los componentes fundamentales de esta plataforma: la Máquina Virtual Java (JVM). La JVM es el intérprete Java y la Interfaz de Programación de Aplicaciones (API). El API Java es un conjunto de clases ya desarrolladas que ofrecen un gran abanico de posibilidades al programador; por lo que ofrece un conjunto de librerías estándar, que contiene mucha de las funciones reutilizables disponibles en los sistemas operativos actuales.

5.5.3. Capa de Datos

- **Mongo DB:** MongoDB forma parte de la nueva familia de sistemas de base de datos NoSQL. En lugar de guardar los datos en tablas como se hace en las bases de datos relacionales, MongoDB guarda estructuras de datos en documentos similares a JSON con un esquema dinámico (MongoDB utiliza una especificación llamada BSON), haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

- **Ehcache:** Ehcache es una caché de código abierto, basado en estándares que aumenta el rendimiento, repercute en la base de datos, y simplifica la escalabilidad. Es el más usado caché basada en Java porque es robusta y probada, con todas las funciones, y se integra con otras bibliotecas y los marcos populares. Ehcache escala desde el almacenamiento en caché en proceso, todo el camino hasta mixtos en proceso de entrada / salida de proceso implementaciones con cachés con terabytes.

5.6. Metodología para el Desarrollo de Sistemas de Información Web

5.6.1. Proceso Unificado de Rational (RUP)

RUP es una metodología completa de análisis y diseño orientado a objetos que unifica tres metodologías: el método Booch de Grady Booch, la ingeniería del software orientada a objetos (OOSE, Object-Oriented Software Engineering) de Ivar Jacobson et al. y la técnica de modelado de objetos OMT (Object Modeling Technique: Técnica de Modelado de Objetos) de James Rumbaugh et al. El término “proceso unificado” se utiliza en la actualidad para abreviar.⁵

El proceso unificado es más que una serie de pasos que, si se siguen, resultarán en la construcción de un sistema de información. El proceso unificado puede verse como una metodología adaptable. Es decir, tiene que ser modificada para el sistema de información específico que se va a desarrollar.⁶

El proceso unificado es una técnica para elaborar modelados. Éste utiliza un lenguaje de modelado para crear modelos, tal como UML (Unified Modeling Language: Lenguaje Unificado de Modelado). Un modelo es un conjunto de diagramas UML que representan un o más aspectos del sistema de información que queremos desarrollar. Es decir, UML es la herramienta que usamos para representar (modelar) el sistema de información objetivo. Una razón importante para utilizar una representación gráfica como es UML se expresa con el viejo

⁵ Schach, Stephen. Análisis y diseño orientado a objetos con UML y el proceso unificado. P. 56

⁶ Ibidem. P. 56

proverbio, a saber, una imagen dice más que mil palabras. Los diagramas UML permiten a los profesionales en sistema de información comunicarse entre sí más rápidamente y de una manera más precisa que si solo se utilizaran descripciones verbales.⁷

El proceso unificado es una metodología iterativa y por incrementos. Cada workflow o flujo de trabajo consta de una serie de pasos y con el fin de llevar a cabo ese workflow, los pasos se repiten hasta que los miembros del equipo de desarrollo están seguros de tener un modelo UML exacto del sistema que quieren desarrollar.⁸ Según el modelo del ciclo de vida iterativo y por incrementos de Jacobson, Booch y Rumbaugh, los workflows corresponden a las fases del paradigma tradicional y se dividen en cinco workflows básicos:

- **Workflow de requisitos:** se extraen los requisitos del cliente. Es decir, el cliente y los futuros usuarios del sistema de información por desarrollar interactúan con el equipo de desarrollo de sistemas de información con el fin de determinar las necesidades del cliente.⁹
- **Workflow de análisis:** El objetivo de esta fase es preparar el documento de especificaciones, que plantea lo que debe hacer el sistema de información o lo que es capaz de hacer.¹⁰
- **Workflow de diseño:** aquí los miembros del equipo de desarrollo describen cómo se va a desarrollar el sistema de información.¹¹
- **Workflow de implementación:** es la traducción del diseño del sistema a un lenguaje de programación apropiado.¹²
- **Workflow de pruebas (mantenimiento):** después que se ha instalado el sistema de información, necesitará modificarse, ya sea para eliminar cualquier falla o porque necesita ampliarse de alguna manera.¹³

⁷ *Ibidem*. P. 56-57

⁸ *Ibidem*. P. 57

⁹ *Ibidem*. P. 7

¹⁰ *Ibidem*. P. 7

¹¹ *Ibidem*. P. 8

¹² *Ibidem*. P. 9

¹³ *Ibidem*. P. 9

- De hecho, el proceso unificado tiene tanto workflows como fases, pero las fases del proceso unificado no corresponden a las fases del paradigma tradicional.¹⁴

Las fases del proceso unificado son los incrementos. Además, aun cuando en teoría el desarrollo de un sistema de información podría llevarse a cabo en tres, cinco o dieciséis incrementos, en la práctica parece consistir de cuatro: iniciación, elaboración, construcción y transición.¹⁵

En la fase de iniciación el objetivo principal es determinar si el sistema de información propuesto es económicamente viable y obtener la versión inicial del caso de negocios. Si el sistema de información es de uso interno, el caso de negocios incluirá el análisis de costo-beneficio inicial.¹⁶

El objetivo de la fase de elaboración es refinar o explicar los requisitos iniciales (casos de usos) de la etapa anterior.¹⁷

El propósito de la fase de construcción es producir la primera versión operativa de calidad del sistema de información, a veces llamada versión beta.¹⁸

El objetivo de la fase de transición es asegurar que los requisitos del cliente se hayan cumplido. Esta fase es dirigida por la retroalimentación desde los sitios en los cuales se han instalado las versiones beta.¹⁹

5.6.2. Lenguaje Unificado de Modelado (UML)

*“El lenguaje unificado de modelado es el lenguaje gráfico universal para describir un sistema de información.”*²⁰ Un lenguaje es simplemente una herramienta para expresar ideas. Al ser un lenguaje UML puede usarse para describir los sistemas

¹⁴ *Ibíd.* P. 207

¹⁵ *Ibíd.* P. 210-211

¹⁶ *Ibíd.* P. 211, 214

¹⁷ *Ibíd.* P.214

¹⁸ *Ibíd.* P.214

¹⁹ *Ibíd.* P.215

²⁰ *Ibíd.* P. 62

de información desarrollados mediante el paradigma tradicional o cualquiera de las muchas versiones del paradigma orientado a objetos, incluyendo el proceso unificado. En otras palabras, el UML es una notación, no una metodología, además es una notación que puede usarse junto con cualquier metodología y se ha convertido en un estándar a nivel mundial.²¹

UML representa una unificación y evolución de diversos métodos de análisis y diseño orientados a objetos que aparecieron a finales de la década de 1980 y principios de los 1990 de Grady Booch, James Rumbaugh e Ivan Jacobson. UML ha sido adaptado como estándar por el OMG (Object Management Group: Grupo de Gestión de Objetos) y ha sido aceptado por la comunidad software como principal notación para el modelado de objetos y de componentes.²²

UML define una serie de diagramas, los cuales pueden dividirse en las siguientes categorías.

Diagramas estructurales, que describen las relaciones entre los componentes. Entre éstos se incluyen²³:

- **Diagramas de clases:** es un modelo de clases que muestra las relaciones estáticas entre ellas, incluyendo la asociación y la generalización.²⁴
- **Diagramas de objetos:** Se puede considerar un caso especial de un diagrama de clases en el que se muestran instancias específicas de clases (objetos) en un momento particular del sistema.²⁵
- **Diagramas de componentes:** muestra las dependencias entre los componentes de software.²⁶
- **Diagramas de Comportamiento:** que describen las relaciones dinámicas entre los componentes. Entre estos se incluyen:

²¹ Ibídem. P. 222

²² Ibídem. P. 760

²³ Connolly, Thomas. Sistemas de bases de datos: un enfoque práctico para diseño, implementación y gestión. P. 760

²⁴ Schach, Stephen. Análisis y diseño orientado a objetos con UML y el proceso unificado. P. 238

²⁵ Fuente: http://es.wikipedia.org/wiki/Diagrama_de_objetos

²⁶ Schach, Stephen. Análisis y diseño orientado a objetos con UML y el proceso unificado. P. 237

- **Diagramas de casos de usos:** diagrama individual que incorpora una serie de casos de usos, los cuales modelan la interacción entre los actores y el sistema de información mismo.²⁷
- **Diagramas de interacción (secuencia y colaboración):** representa la realización de un escenario específico del caso de uso.²⁸
- **Diagramas de estados:** es un modelo dinámico porque refleja el comportamiento de los objetos, es decir, la manera en que reaccionan a eventos específicos.²⁹
- **Diagramas de actividades:** destacan el flujo de control entre actividades.³⁰

Un caso de uso describe qué hace un sistema, pero no especifica cómo lo hace. El comportamiento de un caso de uso se puede especificar describiendo un flujo de eventos de forma textual, lo suficientemente claro para que alguien ajeno al sistema lo entienda fácilmente. Conviene separar el flujo principal de los flujos alternativos, porque un caso de uso describe un conjunto de secuencias, no una única secuencia, y sería imposible expresar todos los detalles de un caso de uso no trivial en una única secuencia. Primero se describe el flujo de eventos de un caso de uso mediante texto. Sin embargo, conforme se mejora la comprensión de los requisitos del sistema estos flujos se pueden especificar gráficamente mediante diagramas de interacción. Cada secuencia representa un posible flujo a través de todas las variantes. Cada secuencia se denomina escenario. Un escenario es una secuencia específica de acciones que ilustra un comportamiento. Los escenarios son a los casos de uso lo que las instancias a las clases, es decir, un escenario es básicamente una instancia de un caso de uso.³¹

El proceso unificado se maneja mediante casos de usos. Durante el workflow del análisis se describen los casos de usos en términos de las clases del sistema de

²⁷ *Ibídem.* P. 238

²⁸ *Ibídem.* P. 134

²⁹ *Ibídem.* P. 238

³⁰ Booch G, Rumbaugh J., Jacobson I. El lenguaje unificado de modelado. Pág. 225

³¹ Booch G, Rumbaugh J., Jacobson I. El lenguaje unificado de modelado. Pág. 195-196.

información. En el proceso unificado hay tres tipos de clases: entidad, borde y control. Una clase entidad es un modelo de la información perdurable. Una clase borde modela la interacción entre el sistema de información y sus actores. Las clases borde, por lo general, se asocian con la entrada y la salida. Cada pantalla de entrada, de salida e informe impreso se modela mediante una clase borde. Una clase control es un modelo de los cálculos y algoritmos complejos. La notación del UML para los tres tipos de clases son los siguientes:

- **Clase entidad:** círculo sobre línea horizontal.
- **Clase borde:** círculo con el símbolo “T” de forma vertical en el lateral izquierdo.
- **Clase control:** círculo con saeta en sentido anti horario.

Estos son estereotipos, es decir, extensiones del UML. Una de las muchas ventajas del UML es que permite definir constructos adicionales que no son parte del UML, pero que son necesarios para hacer un modelo de un sistema con precisión.³²

5.6.3. Modelo Constructivo de Costes (COCOMO II)

A pesar de las dificultades para estimar el tamaño, es esencial que los desarrolladores de Sistemas de Información simplemente hagan lo que esté a su alcance para obtener estimaciones precisas tanto de la duración del proyecto como del costo del mismo³³. Los modelos algorítmicos de estimación de costo híbridos incorporan ecuaciones matemáticas, modelos estadísticos y el juicio de expertos.

El modelo híbrido más importante es el COCOMO de Bohema. COCOMO ha probado ser el método de estimación más confiable para los SI tradicionales, los valores reales tienen una discrepancia de 20 por ciento con respecto a los valores pronosticados, alrededor de dos de cada tres veces. COCOMO se presentó en 1981. COCOMO II es una revisión importante de la revisión de 1981. Éste puede

³² Schach, Stephen. Análisis y diseño orientado a objetos con UML y el proceso unificado. P. 115-117, 129

³³ Schach, Stephen. Análisis y diseño orientado a objetos con UML y el proceso unificado. P. 341-342

lidar con una variedad de técnicas modernas para el desarrollo de sistemas, aunque el modelo aún es demasiado nuevo para conocer el grado en el cual es mejor que su predecesor, el COCOMO original.³⁴

CAPÍTULO I: ESTUDIO ORGANIZACIONAL

1.1. Entorno General de la Institución.

El Ministerio de Trabajo (MITRAB), es la instancia encargada de formular, rectorar y dirigir la política laboral del estado a fin de promover, asegurar y restituir los derechos laborales con prácticas de género de la población económicamente activa a través del sistema de inspección, diálogo, alianza y consenso, entre personas trabajadoras y empleadores, en correspondencia a la legislación laboral y a convenios internacionales ratificados.

Además de la sede central en Managua, tiene presencia en los 16 Departamentos del país, a través de las delegaciones departamentales, teniendo las oficinas en las cabeceras departamentales y en las dos regiones autónomas, Atlántico Norte y Atlántico Sur, en la Costa del Caribe, con el objetivo de estar más cerca de la ciudadanía y poder brindarles un mejor servicio y dar respuestas inmediatas a las necesidades de cada uno de sus territorios.

El actuar del MITRAB, está sustentado en la constitución política de Nicaragua y el conjunto de leyes que rigen y definen sus actividades, tales como la ley del código del trabajo (185), ley de adición de riesgos y enfermedades profesionales (456), entre otras.

1.1.1. Misión del MITRAB

“Promoción, tutela y restitución del derecho al trabajo digno”

Formular, normar, evaluar y dirigir la política laboral del gobierno a fin de promover, asegurar y restituir los derechos laborales para el mejoramiento

³⁴ *Ibíd.* P. 341-345.

económico, social, ambiental y humano con planes y estrategias de desarrollo que permita la generación de empleos justos y dignos para las personas trabajadoras del campo y la ciudad, el respeto a la legislación laboral vigente y a los tratados y convenios internacionales ratificados en la materia.

1.1.2. Visión del MITRAB

Establecimiento de un modelo de relaciones laborales justas basado en la protección y desarrollo de los trabajadores del campo y la ciudad, en un ambiente socio laboral democrático, con equidad y justicia social, que contribuya a elevar la calidad de vida de la población y el desarrollo sostenido de la nación, en un esfuerzo común de generación de empleos dignos y promoción de las inversiones que lleve al mejoramiento del nivel de vida de las familias trabajadoras.

1.1.3. Objetivos Institucionales del MITRAB

- Promover, tutelar y garantizar el cumplimiento de las disposiciones legales en materia de derechos laborales y las condiciones de Higiene y Seguridad del Trabajo; impulsando el diálogo, alianza y consenso entre las personas trabajadoras, empleadoras y el gobierno.
- Fortalecer la promoción del empleo, auto empleo, productividad laboral y mejoras al empleo con integración de la calidad
- Garantizar la restitución de los derechos laborales de las personas trabajadoras con la efectividad en la solución de los conflictos laborales, a través de los métodos alternos de resolución de conflictos.

1.1.4. Funciones Generales del MITRAB

Al Ministerio del Trabajo según la Ley 290 Ley de Organización, Competencia y Procedimientos del Poder Ejecutivo y sus reformas le han correspondido las funciones siguientes:

Ley 290, arto. 27, publicada en la Gaceta, Diario Oficial No.35 del 22 de febrero del año 2013:

- A.** Proponer al presidente de la República, coordinar y ejecutar la política del Estado en materia laboral, de empleos, salarios, de higiene y seguridad ocupacional y de capacitación de la fuerza de trabajo.
- B.** Ejercer, ejecutar y cumplir las funciones, atribuciones y obligaciones que le confieren y establecen la legislación laboral, la Constitución Política y los compromisos internacionales suscritos por Nicaragua y vigentes en materia laboral y sindical, particularmente las normas y convenios internacionales de la OIT.
- C.** Formular, en coordinación con las entidades pertinentes, las normas relativas a condiciones de seguridad, higiene y salud ocupacional y supervisar su aplicación en los centros de trabajo.
- D.** Derogado.
- E.** Intervenir en la solución de conflictos laborales a través de la negociación, conciliación, arbitraje o cualquier otro procedimiento establecido por la ley;
- F.** Formular la política de formación técnica y capacitación continua a la fuerza laboral.
- G.** Brindar asesoría legal gratuita a los trabajadores involucrados en conflictos laborales individuales o colectivos y promover programas de capacitación a trabajadores y empleadores sobre los derechos, deberes, normas y procedimientos en la materia de su competencia.
- H.** Proporcionar a los empleadores procedimientos para la organización científica del trabajo y los salarios.
- I.** Dirigir estudios e investigaciones específicas en el campo laboral;
- J.** Derogado;
- K.** Incluir en las políticas de empleo, las disposiciones necesarias para garantizar las medidas en el ámbito laboral que se derivan de la Ley No.

648, Ley de Igualdad de Derechos y Oportunidades, así como las acciones correctivas para lograr la igualdad de derechos laborales entre hombres y mujeres.

1.1.5. Estructura Organizacional del MITRAB

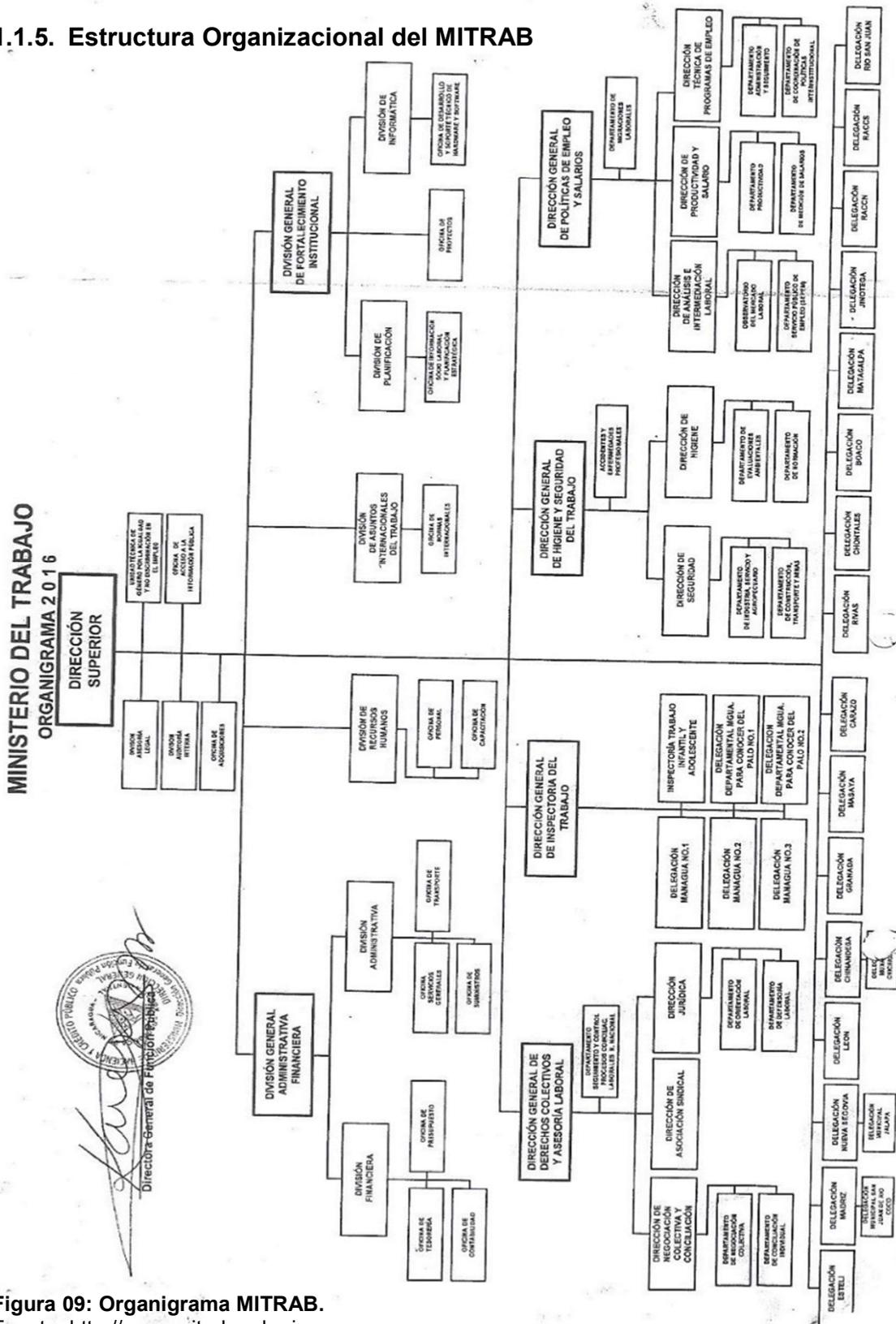


Figura 09: Organigrama MITRAB.
Fuente: <http://www.mitrab.gov.ni>

1.2. Descripción del Modelo de Negocio

A como mencionamos anteriormente el MITRAB es la instancia encargada de formular, rectorar y dirigir la política laboral en el país, tiene presentación en a nivel nacional atreves de sus 16 delegaciones y centraliza sus operaciones en su sede en Managua. Da respuestas a las Solicitudes de trámites, denuncias y consultas de los ciudadanos a través de los procesos institucionales administrados por las 4 direcciones sustantivas; Dirección General de Inspectoría del Trabajo, Dirección General de Higiene y Seguridad del Trabajo, Dirección General de Derechos Colectivos y Asesoría Laboral, Dirección General de Políticas de Empleo y Salario.

Cada dirección procesa estadísticamente los datos de sus operaciones asociados a los diversos procesos internos. Toda la información estadística resultante es suministrada a la unidad de Planificación en la Dirección General de Fortalecimiento Institucional la cual es la encargada de Gestionar los Indicadores Socio-laborales a nivel nacional, Generando los informes correspondientes a las estrategias de presidencia e institucionales (Informe de Gestión diarios, semanales, mensuales y anuales). Además, es la responsable de gestión estratégica de la Institución.

Cabe destacar que los procesos de negocio Internos no están aprobados institucionalmente, más bien se ha venido acentuando los buenos resultados en la práctica y la rutina. Estos son fluctuantes y pueden evolucionar o desaparecer según su nivel de relevancia en las operaciones. Para efecto de estandarización utilizaremos el modelamiento IDEF0³⁵, para representar el comportamiento de los procesos de negocios del MITRAB (modelamiento estático que entrega gradualmente el detalle de un proceso mientras va avanzando a través de niveles, forma de modelamiento llamada “descomposición jerárquica”).

³⁵ Método de modelamiento estándar creado por el Instituto nacional de estándares y tecnología de EE.UU. (National Institute of Standard and Technology of U.S.A.)

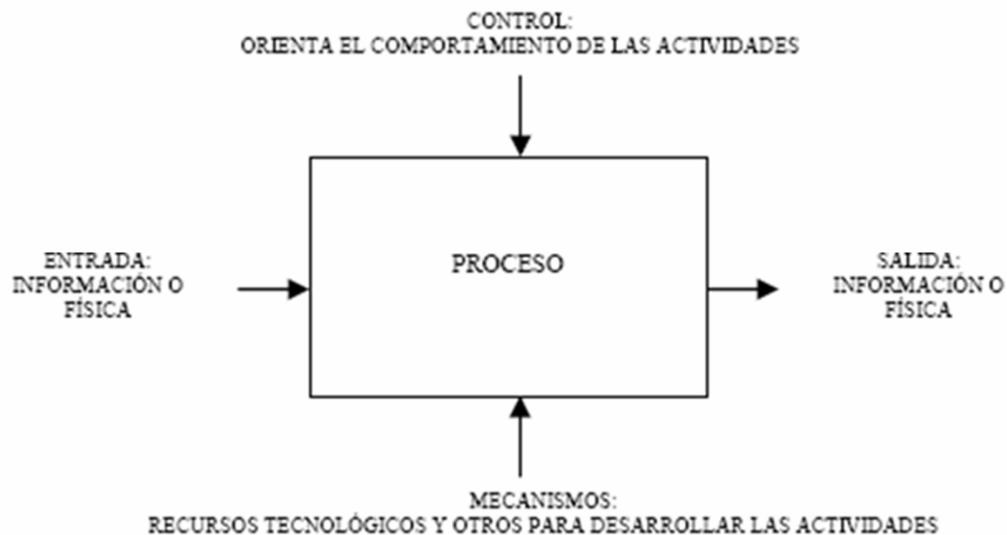


Figura 10: Método IDEF0

Fuente: FEA Consolidated Reference Model Versión 2.1: www.egov.gov.

1.2.1. Análisis de los procesos institucionales

1.2.1.1. Recopilación y Revisión de los procesos de la institución

El análisis de los procesos institucionales se realizó tomando en cuenta la documentación de cada una de las Direcciones Sustantivas, es importante señalar que esta información fue corroborada a través de entrevistas con los involucrados en los procesos, para el cálculo del porcentaje de automatización se tomó en cuenta el total de tareas que los servidores públicos realizan por cada proceso de la institución, entre el número de tareas que están automatizadas por alguno de los sistemas de la institución o mediante el uso de programas de ofimática como Excel, como se observa en la siguiente formula:

$$\text{Porcentaje de Automatizacion} = \frac{\text{Numero de Tareas Automatizadas}}{\text{Total de Tareas del Proceso}} \times 100$$

Este ejercicio permitió conocer el quehacer de la institución en sus diferentes Direcciones Sustantivas, así como identificar aquellos procesos que se encuentran automatizados y aquellos que presenta debilidades o la métrica de automatización es totalmente nula.

A continuación, se detalla el porcentaje de automatización de los procesos de las cuatro Direcciones Sustantivas:

Dirección Sustantiva	%
Dirección General de Inspectoría del Trabajo	34.27%
Dirección General de Higiene y Seguridad del Trabajo	23.67%
Dirección General de Derechos Colectivos y Asesoría Laboral	20.3%
Dirección General de Políticas de Empleo y Salario	30.48%
Total, de Automatización	27.18%

Tabla 2: Nivel de Automatización de Procesos del MITRAB

Fuente: Elaboración Propia

En la tabla anterior se puede observar con claridad que el porcentaje de automatización total de los procesos en las cuatro Direcciones Sustantivas del Ministerio es apenas el 27.18% en promedio, es decir, las tareas automatizadas representan menos de 1/3 de todas las tareas llevadas a cabo por la Institución.

La Dirección General de Derechos Colectivos y Asesoría Laboral obtiene un 20.3% de automatización de sus tareas hace uso de un sistema llamado "File Master", que durante su concepción e implementación incorporó los requerimientos principales del área, sin embargo, el contexto actual y los nuevos requerimientos institucionales hace que el sistema no responda a las exigencias del área.

La Dirección General de Higiene y Seguridad Laboral posee un 23.67% de automatización, en el que hacen uso del "Sistema de Reporte de Accidentes y Enfermedades" y también del SICHYS, en relación a este último está aún en prueba piloto para entrar en funcionamiento total el próximo año.

La Dirección General de Políticas de Empleo y Salario con un 30.48% de automatización de sus procesos hace uso de los Sistemas SISEPEM y SAES. Sin embargo, queda una brecha importante para tener un porcentaje de automatización adecuado.

La Dirección General de Inspectoría Laboral es la Dirección con mayor porcentaje de automatización, con el 34.27% de automatización de los procesos. Cabe destacar que no hacen uso de ningún Sistema Especializado, sino que a través de formatos en Excel han llegado a este nivel. No obstante, los procesos de esta forma se hacen más difíciles y menos eficiente que si se tuviera un sistema especializado acorde a las necesidades de la Dirección.

Por tanto, es importante señalar que el 27.18% incluye el uso de formatos en Excel que contribuyen a la automatización de información, por lo que el porcentaje anteriormente mencionado representa un estimado, es decir que si solo se tomaran en cuenta los Sistemas Informáticos Especializados el porcentaje de automatización de procesos sería mucho menor (*Ver Anexo 1: Análisis de Procesos MITRAB*).

1.2.2. Análisis Operativo de TI

1.2.2.1. Evaluación de la estructura organizacional

Para conocer el estado actual y el nivel de avance de las operaciones TI en la División de Informática, se utilizó el marco de referencia de gobernanza TI COBIT 4.1, según las recomendaciones de ISACA (Asociación de Auditoría y Control de Sistemas de Información) cuando se desconoce el nivel de madurez de una organización TI o este nivel es menor a 2 se debe de implementar esta versión.

A nivel general la División, cuenta con 25.5% de cumplimiento de los estándares propuesto por COBIT 4.1, este nivel de cumplimiento es no aceptable y ubica las operaciones en un nivel 1 (Inicial). Cada uno de estos dominios cuentan con procesos y estos se conforman por varios objetivos de control, para obtener la evaluación total de la organización TI, de los dominios, procesos y objetivos de control se utilizó una matriz de evaluación (*Ver DVD – Evaluación Cobit*).

De los resultados obtenidos se realizó el análisis de los dominios (Ver DVD - Evaluación Organizacional TI): Planear y Organizar (PO), Adquirir e Implementar (AI), Entregar y Dar Soporte (DS) y Monitorear y Evaluar (ME), cada uno obtuvo el siguiente nivel de cumplimiento:

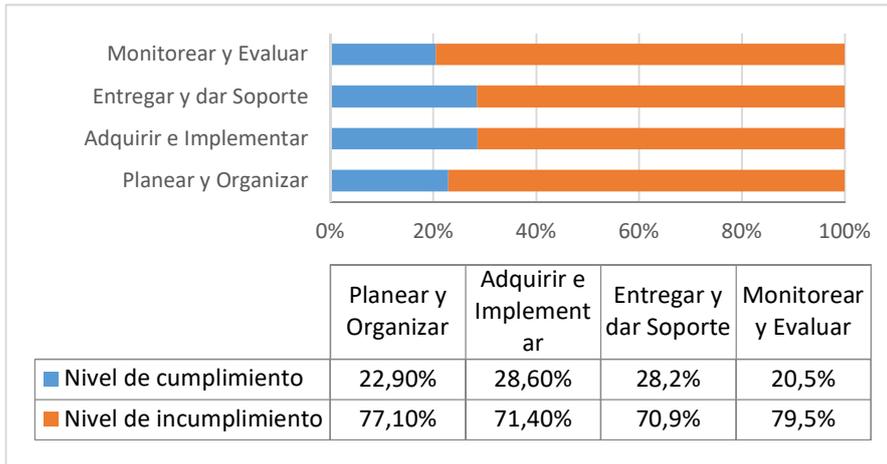


Figura 11: Evaluación por Dominio. Aplicación de COBIT 4.1 al MITRAB
Fuente: Elaboración Propia

El porcentaje de aceptación del nivel de cumplimiento debe ser mayor a 60%, de acuerdo a los resultados obtenidos se verifica que los cuatro dominios no cumplen con el nivel mínimo, lo que indica que la División de Informática debe fortalecer cada uno de los procesos que componen dichos dominios.

Resultados por Dominio:

1. Planear y Organizar: dentro de este dominio se tiene 10 procesos cada uno tiene el siguiente nivel de cumplimiento: Definir un plan estratégico 6.7%; Definir la arquitectura de la información 22.7%; Determinar la dirección tecnológica 0%; Definir los procesos, organización y relaciones TI 24.4%; Administrar las inversiones TI 39.3%; Comunicar las aspiraciones y la dirección de la gerencia 0%; Administrar los recursos humanos de TI 43.8%; Administrar la calidad 0%; Evaluar y administrar los riesgos TI 0%; Administrar proyectos 46.3 %. De la evaluación de este dominio se obtuvieron los siguientes resultados:

- Este dominio evalúa la definición e implementación de un plan estratégico

TI, en este sentido la División Informática no cuenta con uno, actualmente no tienen tecnologías que se acoplen a las necesidades de la institución, la administración de los recursos TI con los que cuentan están más enfocados a la rendición de cuentas y control de los costos, pero la División como tal no tiene un lineamiento convergente con la institución.

- En lo que respecta a la Arquitectura de la Información solo se cuenta con la clasificación de los datos, todos los demás temas relacionados son deficientes, tales como el modelo de información, diccionario de datos corporativo y la administración de seguridad de los datos, que son aspectos a tomar en cuenta para tener una arquitectura de datos corporativa robusta.
- La División de Informática carece de un plan de dirección tecnológica y un plan de infraestructura, actualmente sus operaciones aún no están estandarizadas.
- En relación a los procesos, organización y relaciones TI, la División cuenta con las funciones por cargo, pero no han creado los indicadores de desempeño para la debida supervisión y evaluación de los roles, además no se han definido y estandarizado las responsabilidades de buenas prácticas, los controles de calidad y el análisis de riesgo, para todo esto se hace necesario establecer un marco de trabajo.
- Otro proceso muy importante es la comunicación de las aspiraciones y la dirección de la gerencia, esta se ha visto afectada debido a que aún no se presenta un ambiente de políticas y de control.
- En la administración de la calidad a lo interno de la División Informática, no existe ningún nivel de cumplimiento, no hay monitoreo continuo y revisiones internas - externas del desempeño, esto se debe principalmente a la carencia de un SGC (Sistema de Gestión de la Calidad).
- En el marco de trabajo en la División de Informática aún no se tiene en cuenta la administración de los riesgos, esto implica realizar evaluación de

riesgos y vulnerabilidades que evite exponer el cumplimiento de los objetivos estratégicos de la institución.

2. Adquirir e implementar: este dominio tiene 7 procesos a evaluar cada uno tiene el siguiente nivel de cumplimiento: Identificar soluciones automatizadas 16.7%; Adquirir y mantener software aplicativo 20.3%; Adquirir y mantener infraestructura tecnológica 34.3%; Facilitar la operación y el uso 36.1%; Adquirir recursos TI 83.3%; Administrar cambios 14.0%; Instalar y acreditar soluciones y cambios 21.1%. De la evaluación de este dominio se obtuvieron los siguientes resultados:

- El proceso de identificar soluciones automatizadas, la participación de la División de Informática en cuanto al desarrollo de sistemas de información se limita a apoyar en la evaluación del impacto con respecto a la implantación de los requerimientos funcionales y técnico.
- En la adquisición y mantenimiento de software aplicativo la División se ha visto involucrada sobre todo en la configuración e implantación de software adquirido, pero no se lleva un control de estas configuraciones, aún no se ha realizado auditorías a los sistemas existentes, durante la implementación no se realiza aseguramiento de la calidad y una vez ya implementado no se da mantenimiento.
- Administración de cambios, actualmente no tienen estándares, pero si se aseguran de que no se realice ningún cambio sin previa autorización.
- El proceso de instalar, acreditar soluciones y cambios refleja que no se adopta una metodología de prueba en las aplicaciones e infraestructura con el propósito de disminuir los errores.

3. Entregar y dar Soporte: Este dominio tiene 13 procesos en cada uno se tiene el siguiente nivel de cumplimiento: Definir y administrar los niveles de servicio 61.1%; Administrar los servicios de terceros 66.7%; Administrar el desempeño y la capacidad 60%; Garantizar la continuidad del servicio 0%; Garantizar la

seguridad de los sistemas 12.7%; Identificar y asignar costos 57.1%, Educar y entrenar usuarios 0%; Administrar la mesa de servicios y los incidentes 13.3%; Administrar la configuración 38.3%; Administrar los problemas 50%; Administrar los datos 0%; Administrar el ambiente físico 26.7%; Administrar las operaciones 36.7%.

- La División de informática cuenta con marco de trabajo para la administración de los servicios, pero aún no brindan todos los servicios, actualmente no están llevando a cabo: desarrollo de sistemas, monitoreo de la seguridad TI y reporte de los servicios ofrecidos. Es decir, el personal de la División este concentrado únicamente a brindar servicios de soporte técnico.
- Respecto a administrar los servicios de terceros tales como consultores externos, proveedores, etc., se identifican las relaciones y se monitorea el desempeño de los proveedores, pero no se administran los posibles riesgos de estos.
- En referencia a administrar el desempeño y la capacidad, actualmente se revisan la capacidad y el desempeño de los recursos de TI, se identifican las tendencias de las cargas de trabajo, pero no se tiene en cuenta los riesgos que puedan interrumpir el desempeño y la capacidad de los recursos, no se cuenta con planes de contingencias.
- No se cuenta con un marco de trabajo para garantizar la continuidad de TI, es decir no se garantiza continuidad de los procesos en caso de interrupción.
- La Seguridad de los Sistemas se encuentra por debajo del límite de aceptación no se salvaguarda la información contra uso no autorizado, divulgación, modificación, daño o pérdida, por otro lado, los sistemas se encuentran expuestos a ataques.
- Se identifican y se registran todos los costos de TI, alineado con los

procedimientos de contabilización de la institución, pero se garantiza que los cargos por servicios se identifiquen en el modelo, este se debe revisar constantemente.

- La educación y entrenamiento a los usuarios es uno de los objetivos de la División y es una función dentro de los roles³⁶, sin embargo, no se está brindando información necesaria en el uso efectivo de la tecnología, un posible motivo puede ser la falta de personal para atender todos los aspectos técnicos de la institución.
- Administrar la mesa de servicio y los incidentes, los servicios requeridos por los usuarios de la institución no se atienden eficientemente, debido a esto la División ha iniciado a implementar un proceso básico de colas, en el que, al llegar una solicitud de soporte por parte de los usuarios de TI, ésta se enumera en función del orden de llegada y así se asignan entre los miembros de la División TI, con el fin de que dichas solicitudes sean atendidas.
- Administrar la configuración, se registra toda la información relevante sobre los elementos de configuración, se monitorean todos los activos y sus cambios, los procedimientos de configuración están definidos, pero no se tiene integrado la gestión de los cambios, incidentes y de problemas.
- Existen procesos para reportar problemas que han sido identificados como parte de la administración de incidentes, pero aún no se cuenta con un sistema de administración de problemas.
- La División de Informática atiende algunos requerimientos de la institución en la administración de los datos, hasta donde le es posible el manejo de la información cuentan con el procedimiento para el respaldo y restauración de los datos.

³⁶ Información extraída del documento División de informática de Sistemas pag.60, proporcionado por la división informática del MITRAB

- Administración del ambiente físico, para el centro de datos existen posibles riesgos naturales y algunos causados por los usuarios, aun no se tienen regulaciones, estándares, medidas de seguridad, controles y monitoreo del ambiente.
- Se tienen algunos procedimientos e instrucciones de operaciones y se documentan sus salidas, pero no se tiene programación de las tareas y monitoreo de la infraestructura TI.

4. Monitorear y Evaluar: Este dominio tiene 4 procesos en cada uno se tiene el siguiente nivel de cumplimiento: Monitorear y evaluar el desempeño 43.8%; Monitorear y evaluar el control interno 0%; Garantizar el cumplimiento regulatorio 0%; Proporcionar gobierno TI 35.7%.

- No se ha establecido un marco de trabajo, monitoreo general que definan el alcance, la metodología y el proceso a seguir para medir la solución y la entrega de servicios de TI, el enfoque de monitoreo dentro de la División es bajo y no se mide el nivel de desempeño de los involucrados bajo indicadores.
- No se tiene definido el monitoreo continuo del ambiente de control de TI y el marco de trabajo de control de TI para satisfacer los objetivos organizacionales.
- No presentan medidas o políticas que permitan regular el cumplimiento de la dirección.
- No se tiene un alineamiento estratégico de TI y a su vez no se están tomando decisiones para la obtención de los beneficios provenientes de las inversiones habilitadas con las tecnologías de información.

1.2.3. Análisis de los Sistemas de Información

1.2.3.1. Revisión Documental

De la documentación presentada por sistema se tienen la siguiente evaluación:

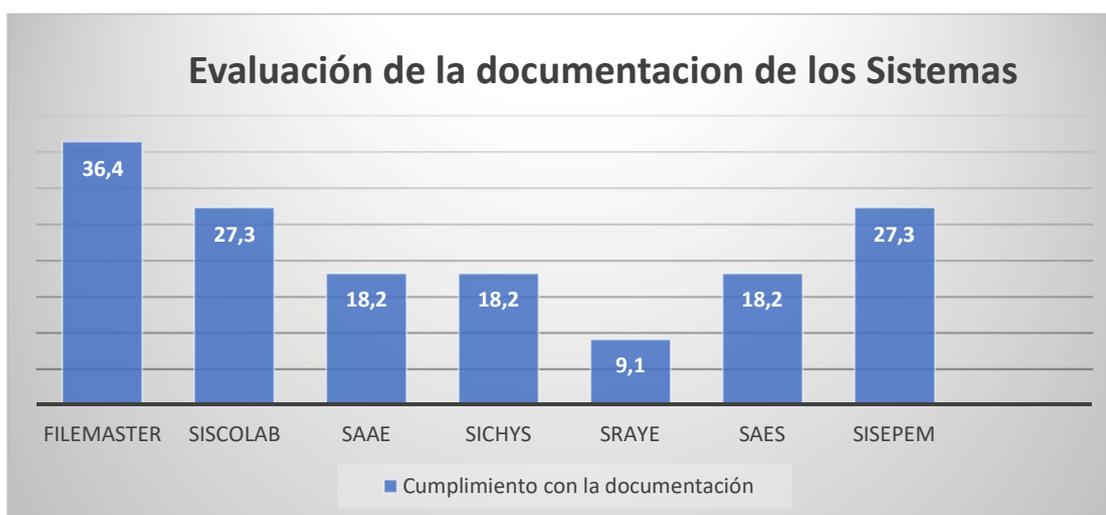


Figura 12: Evaluación de la Documentación de Sistemas. MITRAB

Fuente: Elaboración Propia

Estos resultados se obtuvieron de la matriz de evaluación de documentos existentes de sistema de información desarrollados. (*Ver Anexo 2: Evaluación de documentación de Sistema*)

- FILEMASTER:** Este sistema cuenta con un manual de usuario, donde se hace referencia a la descripción del sistema, su funcionamiento y como se puede utilizar, además contiene el manual de capacitación que permite tener una visión clara del proceso donde se define el diagrama general del sistema. Incluye el manual de la implementación del sistema y documentación que sirve como medio de comunicación para la organización. El sistema no muestra una metodología de desarrollo, ni los requerimientos funcionales y no funcionales, ni el script de la base datos, el diagnóstico de la situación, ni la normalización de la programación, por lo que el sistema se encuentra en un porcentaje de funcionamiento del 36% aproximadamente ya que no cumple con toda la documentación necesaria.
- SISCOLAB:** En la información del sistema, hay una descripción actual de la situación de los procesos en el funcionamiento de los servicios, detalla una guía de orientación para los usuarios, es decir una explicación clara y breve de los módulos que se presentan en el sistema.

No muestra el manual administrativo ni el de la implementación del sistema, tampoco la metodología de desarrollo que se utilizó, la definición de los requerimientos funcionales y no funcionales, los casos de uso de las tareas, el script de la base de datos, el diccionario de datos y las normas de programación, por lo observado el sistema presenta información incompleta, por lo cual tiene un 27% de cumplimiento en la documentación del sistema.

- **SAAE:** En la información del sistema solamente contiene el manual de usuario y manual de captación donde se exponen los procesos que el usuario puede manipular a través del sistema implantado con los detalles y características del sistema.

La documentación del sistema es inconsistente debido que no contiene inicialmente el diagnóstico de la situación, el manual de la implementación, la documentación necesaria como medio de comunicación para el personal administrativo, la metodología de desarrollo, el script de la base de datos, los casos de uso del sistema, los requerimientos funcionales y no funcionales, el diccionario de datos y las normas de programación lo que el sistema obtiene un 18% aproximadamente en el nivel de cumplimiento en la documentación del sistema.

- **SICHYS:** Este sistema aún no está en funcionamiento total, debido a que se encuentra en pilotaje y migración de la información, lo que significa que aún no cumple con todas las normas de implementación, manual de usuario con el detalle del funcionamiento del sistema, el manual administrativo como medio de comunicación con el área administrativa de la institución, la definición de los requerimientos funcionales y no funcionales.

No hay detalle del diagnóstico de la situación ni el contenido del diccionario de datos, el sistema contiene un nivel de cumplimiento muy bajo por la inconsistencia de la documentación clasificada en un 18.2%.

- **SRAYE:** Este sistema no cumple con la mayoría de la documentación necesaria, ya que solamente se detalla el manual de captación que es un

instructivo que permite al usuario la captación del sistema. Según la evaluación este sistema tiene el nivel más bajo de rendimiento cumpliendo con el 9.1% de la documentación, estos avances se deben a que el consultor a cargo de dicho modulo aun no finaliza.

- **SAES:** La documentación del sistema solamente encontramos el manual de usuario y de captación de la aplicación donde se especifica el contenido a nivel funcional del sistema. No contiene documentación de desarrollo y de programación, así como hace falta los requerimientos funcionales y no funcionales del sistema, no obstante, el consultor de este módulo aún no termina con su desarrollo.
- **SISEPEM:** La documentación del sistema describe el diagnóstico de la situación, así como contiene información de manual de usuario y de captación que sirven como medio de comunicación con los usuarios que utilicen el sistema. No obstante, presenta ciertas inconsistencias puesto que no posee documentación de la metodología de desarrollo, casos de uso, requerimientos funcionales y no funcionales, entre otros, obteniendo así un 27.3% del nivel de cumplimiento.

1.2.3.2. Evaluación del diseño y las funcionalidades de los módulos existentes (Calidad).

Para el análisis de los sistemas de información en el MITRAB se utilizó la ISO 9126 la cual es un estándar internacional para la evaluación de la calidad del software. Esta norma indica las características de calidad y los lineamientos para su uso. De acuerdo a la ISO la definición de calidad es “Un conjunto de propiedades y características de un producto o servicio que le confieren sus aptitudes para satisfacer necesidades expresadas o implícitas”, por lo tanto, esos serán los aspectos a tomar en cuenta en dicho análisis.

De tal manera que la ISO 9126 nos servirá para definir un marco conceptual de calidad del producto de software (calidad interna y calidad externa).

Se crearon dos instrumentos para realizar dicho análisis, el primero es la ISO propiamente dicha la cual fue aplicada, y el otro instrumento se basa en los aspectos de control interno de la ISO 9126, tales como funcionalidad, fiabilidad, eficiencia y usabilidad, el proceso incluyó la apreciación de los usuarios finales que interactúan a diario con los diferentes sistemas de información. (*Ver Anexo 3: Matriz de evaluación de los sistemas*)

Los sistemas de información que se analizaron son:

- SISTEMA DE REPORTE DE ACCIDENTES Y ENFERMEDADES.
- SICHYS
- SISEPEM
- FILE MASTER (RAC)
- SAAES

El nivel mínimo aceptable de calidad es del 70%, cabe destacar que el nivel de calidad debe ser determinante en la etapa de mejoras, puesto que permitirá evaluar su incorporación al SISLAB.

Como puede apreciarse en el gráfico siguiente, dos sistemas de información obtuvieron un nivel aceptable de calidad. SICHYS 76%, y siendo Higiene y Seguridad el que sobresalió con 77%, sin embargo, todos los sistemas del MITRAB deben mejorar en cada uno de aspectos evaluados en la ISO 9126

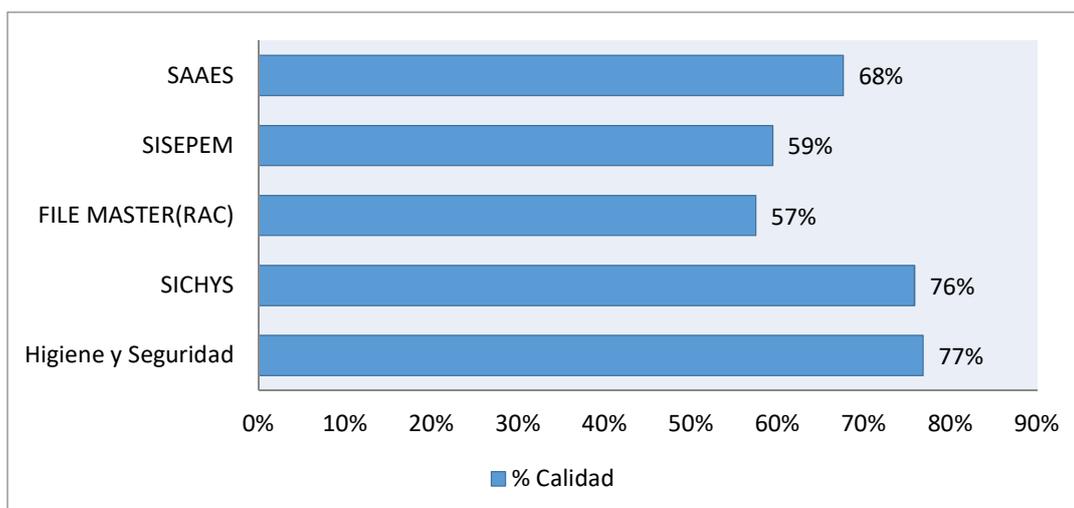


Figura 13: Resultados de % de calidad de uno de los sistemas evaluados.

Fuente: Elaboración Propia

A continuación, se analiza cada una de las características propuesta por la ISO 9126, para de esta manera poder apreciar en qué punto tienen mayor debilidad cada uno los sistemas de información en el MITRAB.

Es importante señalar que el sistema SICHYS todavía no se ha implementado en su totalidad, el análisis realizado por parte de los consultores fue con una versión beta que tenía el área de higiene y seguridad, por tal razón en la encuesta con los usuarios finales no se incluyó.

1. Funcionalidad: Esta característica nos ayuda a evaluar si los requisitos funcionales están disponibles en el software, incluye una serie de características que permiten calificar si un producto de software maneja en forma adecuada ese conjunto de funciones que satisfacen las necesidades para las cuales fue desarrollado.

Los sistemas de información que tuvieron un nivel aceptable en funcionalidad son: Sistema de Reporte de Accidentes y Enfermedades con 73%, SAAES con 73% y SICHYS con 90% siendo este último el que posee mejor calificación. (Ver anexo 4: Resultados de la Evaluación de los sistemas – Grafico N°1)

2. Usabilidad: Esta característica nos indica que, si el software es fácil de usar, posee un conjunto de atributos que permiten evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el sistema.

En la parte de usabilidad el único sistema de información que no pasó el nivel mínimo fue el FILE MASTER (RAC) y el de mejor calificación fue SICHYS con 85%. (Ver anexo 4: Resultados de la Evaluación de los sistemas – Grafico N°2)

3. Fiabilidad: Esta característica evaluó la confiabilidad y frecuencia de falla por errores en el software. También la tolerancia a fallas, es decir se refiere a la capacidad de restablecer el nivel de operación y recobrar los datos que hayan sido afectados directamente por una falla, así como el tiempo y esfuerzo necesarios para lograrlo. Ninguno pasó el nivel mínimo de calificación, es decir todos ellos tienen problema en esta característica. (Ver anexo 4: Resultados de la Evaluación de los sistemas – Grafico N°3)

4. Eficiencia: Evalúa que tan eficiente es el software, esta característica permite evaluar la relación entre el nivel de funcionamiento del software y la cantidad de recursos usados.

Todos los sistemas de información evaluados cumplieron con el nivel mínimo de aceptación. Siendo el SAAES el que obtuvo mejor calificación con 85%. (Ver anexo 4: Resultados de la Evaluación de los sistemas – Grafico N°4).

5. Mantenibilidad: Esta característica analiza que tan fácil es el software para resistir cambios y que atributos permiten medir el esfuerzo necesario para realizar modificaciones al software, ya sea por la corrección de errores o por el incremento de funcionalidad.

De todos los sistemas de información, dos cumplieron con el nivel de aceptación, siendo SICHYS el que obtuvo mayor calificación con 90%, el cual es el último sistema que se le ha dado mantenimiento, modificando las funcionales para que sean acordes a los requisitos establecidos, el otro es el Sistema de Reporte de Accidentes y Enfermedades. (Ver anexo 4: Resultados de la Evaluación de los sistemas – Grafico N°5)

6. Portabilidad: Este evalúa la habilidad del software para transferirse a otro ambiente (sistema operativo, servidor, etc.)

Tres sistemas de información cumplieron con el nivel de aceptación, son SAES con 90%, SICHYS con 90% y el que obtuvo mayor calificación fue Higiene y

Seguridad con 95%. (Ver anexo 4: Resultados de la Evaluación de los sistemas – Grafico N°6)

Es importante destacar que de acuerdo a las tecnologías usadas en los diferentes sistemas se puede determinar su nivel de portabilidad en los diferentes ambientes, en este sentido, es descartado completamente el File Master, puesto que es un que esté ligado a tecnologías privativas de pago tales como asp.net y SQL server, esto significa que solo podría correr bajo ambiente Windows. Los demás sistemas hicieron uso de tecnologías cómo PHP, MySQL y PostgreSQL que son de código abierto y pueden ser instaladas en cualquier Sistema Operativo, incluyendo Windows, esto garantiza una fácil integración.

1.2.4. Análisis de la seguridad

1.2.4.1. ISO 27001

A fin de evaluar la gestión de la seguridad de la información se utilizó la norma ISO 27001, así mismo se aplicaron los diferentes controles establecidos en la ISO 27002, en este sentido el MITRAB tiene un nivel de cumplimiento del 35.49%, esto significa que la institución tiene un riesgo en seguridad de la información bastante alto.

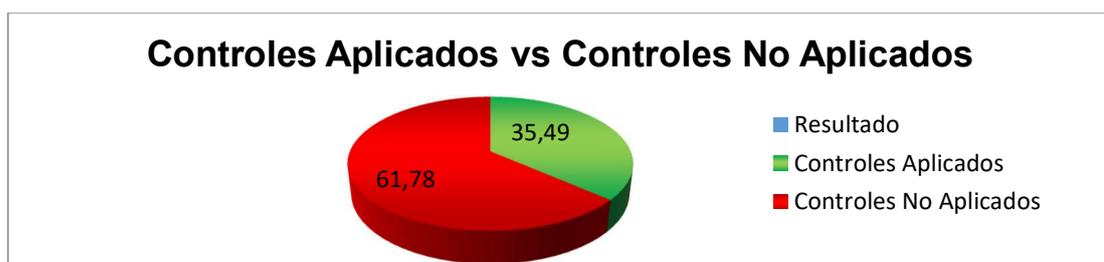


Figura 14: Controles Aplicados VS Controles No Aplicados. ISO 27001

Fuente: Elaboración Propia

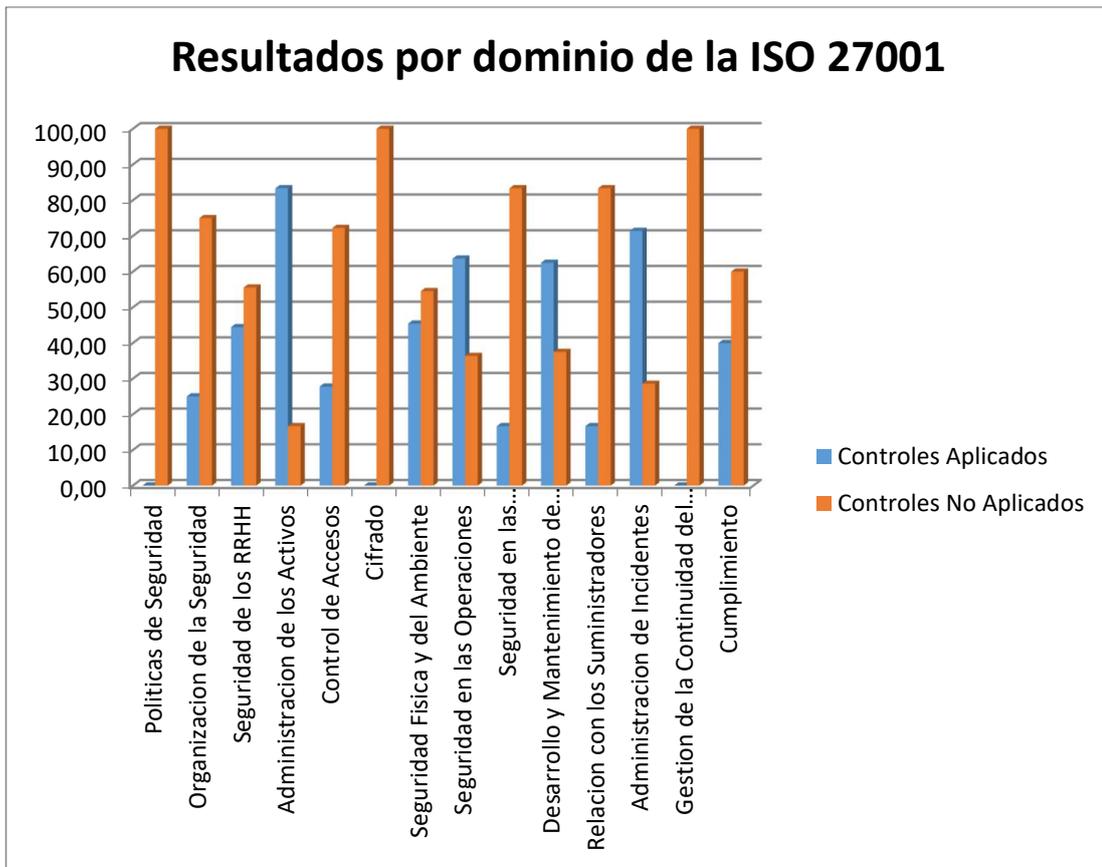


Figura 15: Resultados por dominio. ISO 27001

Fuente: Elaboración Propia

Condensado por Dominios:

Los dominios que están totalmente nulos son:

- Políticas de Seguridad: No existen políticas, normas, ni procedimientos enfocados a la parte de seguridad, algunas se dicen verbalmente, pero no se garantiza el cumplimiento. Cabe destacar que, en la entrevista, el director de informática Ing. Ricardo Ruiz hizo mención que se está trabajando en este tema, pero aún está en etapa de elaboración.
- Cifrado: No existen normas, políticas ni gestión de cifrado, este dominio es totalmente nulo y significa un alto riesgo para la confidencialidad e integridad de la información.

Gestión de la continuidad del negocio: Si bien es cierto, se tiene un plan de continuidad frente a riesgos, este plan no contempla la continuación de la parte

TI, es necesario que el plan de continuidad de negocio institucional esté basado en criterios de diferentes especialidades, con un comité multidisciplinario para que se tenga un plan completo, en el que se incluya el aspecto TI, puesto que la información es un recurso primordial.

Dominios con cumplimiento entre 1% a 50%:

- Organización de la Seguridad: Existen muchos elementos que convergen en este dominio, entre ellos tenemos recursos humanos, planificación, recursos tecnológicos, administración de los mismos, etc. Sólo el 25% de este dominio se cumple, esto representa apenas $\frac{1}{4}$ de cumplimiento, lo que es bastante peligroso y deficiente.
- Seguridad de los RRHH: El personal que está en constante uso de los recursos y activos de TI no se puede ignorar en este análisis, puesto que son estos quienes dan entrada, procesan y dan salida correspondiente a la información, tener un 44.44% de cumplimiento significa que el personal no está claro de la importancia de la seguridad en el manejo de la información, aspecto que es contraproducente para alcanzar la seguridad de la información.
- Control de accesos: Tener una buena gestión de los accesos es vital para establecer un buen alcance de la seguridad, un 27.78% de cumplimiento es totalmente riesgoso, puesto que no se garantiza conocer que usuarios tuvieron acceso a cierta información y si estos tienen permisos para hacer uso de ella.
- Seguridad física y del medio ambiente: Un buen modelo de seguridad toma en cuenta aspectos del entorno físico. Un 45.45% de cumplimiento aún está por debajo de la media del dominio, no hay una buena gestión de la seguridad física, esto podría fácilmente llevar a daños físicos en los recursos TI, daños a los servidores y por ende falta de disponibilidad de servicios, etc.
- Seguridad en las telecomunicaciones: Las telecomunicaciones nos garantizan que la información viajará de un punto a otro, de forma efectiva, pero también debería ser de forma segura. Se tiene 16.67% de cumplimiento en estos controles, lo que significa que la información no viaja de manera segura e incluso la efectividad de que la información llegará al destino.

- Relación con los proveedores: Los servicios de terceros también tienen un aporte en la consecución de la seguridad de la información, puesto que deben existir políticas respecto a estos, acuerdos de confidencialidad, etc. En este dominio se tiene un 16.67% de cumplimiento, no se hace un buen seguimiento de los proveedores, no se tienen políticas respecto a los mismos ni se hace supervisión periódica y efectiva.
- Cumplimiento: El cumplimiento de aspectos legales y políticas de seguridad arrojó un 40% de cumplimiento, principalmente en los controles definidos en materia legal, tal como propiedad intelectual y cumplimiento de la legislación en los sistemas. Todos los sistemas y procesos están debidamente respaldados con artículos de ley, sin embargo, el cumplimiento de esto no es suficiente, sino que gran parte influye la no existencia de políticas de seguridad.

Dominios con cumplimiento entre 51% a 70%:

- Seguridad en las Operaciones: El nivel de cumplimiento de los controles en este dominio es del 63.64%, un poco por encima de la media, sin embargo, no garantiza la seguridad en las operaciones de TI, se deben hacer mejoras sustanciales en el objetivo de control referido a auditorías. Entre otros.
- Desarrollo y Mantenimiento de los Sistemas: Con un cumplimiento del 62.50%, debido a que se da el seguimiento correspondiente, a pesar de que el desarrollo y parte del mantenimiento de los sistemas no se hagan a lo interno de la institución. Aunque existen algunas situaciones que se deben mejorar en cuanto a la definición de los requerimientos de seguridad de los sistemas y la creación de políticas para el desarrollo.

Dominios con cumplimiento mayor a 70%:

- Administración de activos: El mejor calificado de todos los dominios, con un 83.33%, debido a que se tienen clasificados los activos TI con sus respectivos responsables. El aspecto que debe mejorarse es el etiquetado de la información.

- Administración de incidentes: Con un cumplimiento del 71.43% de los controles, ronda un porcentaje aceptable. Se tiene una gestión y seguimiento de incidentes, sin embargo, se tienen que elaborar procedimientos de recopilación de evidencia más formales.

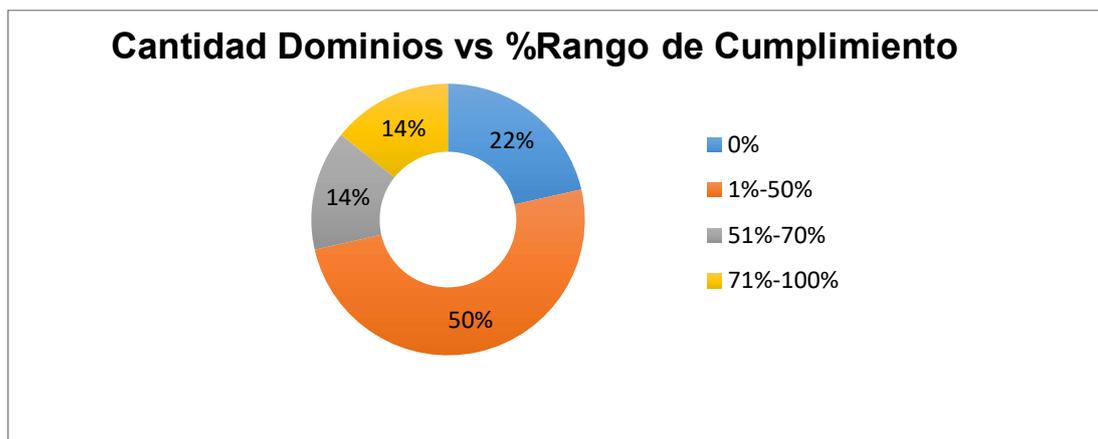


Figura 16: Cantidad de Dominios vs Rango de cumplimiento. ISO 27001

Fuente: Elaboración Propia

Solo existen 2 dominios con un cumplimiento mayor al 70%, porcentaje que se considera aceptable, de los 14 dominios establecidos en la norma ISO 27001. Entre 0% al 50% se tienen 10 dominios de 14, un nivel totalmente bajo en relación a la importancia de la seguridad de la información (ver DVD: Inciso A. Informe ISO 27001).

Como puede verse, en materia de la seguridad de la información hay mucho trabajo por hacer, en el pentesting aplicado (Ver DVD: Inciso B. Informe de Pentesting) se consiguió tener un informe de vulnerabilidades bastante grande (Ver DVD: *Nessus Scan*), donde los principales hosts con vulnerabilidades críticas son aquellos con sistema operativo Windows Server 2003 o XP. Se explotó una de estas vulnerabilidades consiguiendo así la denegación de servicios. Se ha decidido tener estos sistemas operativos debido a la compatibilidad de un sistema que está desarrollado en asp.net y usa SQL server 2000 como motor de persistencia, sin embargo, es un riesgo latente mantener este tipo de sistemas operativos aún en uso, principalmente luego que se anunció que quedaban discontinuados por el proveedor "Microsoft", lo que significa que no se seguirán

emitiendo parches de seguridad para estos sistemas operativos.

Existen 2 recomendaciones sumamente importantes que son indispensables y que de acuerdo al análisis de seguridad que se hizo repercutirá considerable y positivamente en la seguridad:

- Establecer políticas de seguridad de la información y activos TI con sus respectivas normas y/o procedimientos.
- Llevar a cabo una actualización de los Sistemas Operativos que han quedado sin soporte (Windows XP o Windows Server 2003) a una versión más reciente (Windows 7 o Windows Server 2008), en todas las terminales y prioritariamente en los servidores de aplicación.

Estas 2 recomendaciones mejorarán considerablemente la seguridad en la institución y servirán de preámbulo para el desarrollo e implementación de la plataforma SISLAB bajo un marco de seguridad mucho más estable. Otras recomendaciones y planes de mejoras se darán a conocer en el siguiente entregable de la asistencia actual.

1.2.4.2. Evaluación de los estándares de aplicación

Se han evaluado los estándares de seguridad en las aplicaciones instaladas en el MITRAB, retomando los principales insumos de la seguridad en los aplicativos tales como: “Base de datos”, “Transferencia segura de datos”, “Roles y permisos”, “Codificación”. A continuación, se detalla dicho análisis:

- **Sistema File Master:** El Sistema File Master posee una asignación de “Roles” en la que se establecen las acciones permitidas por cada Rol. Estos roles identificados son:
 - Secretarias
 - Abogados Conciliadores
 - Jefaturas

Sin embargo, no se tiene mayor documentación técnica de dicho sistema, solo cuenta con un manual de usuario.

Por otro lado, no se tiene conocimiento de los estándares de codificación debido a que es una aplicación “enlatada”, de tal manera que no se tiene código fuente. No obstante, se tiene conocimiento que fue desarrollado en “.net” y usa “SQL Server” como gestor de Base de datos. Esto significa que el entorno de producción es un Windows Server.

- **SISCOLAB:** fue desarrollado en el 2010, bajo una consultoría, en la que se pretende migrar el uso del RAC a dicho sistema. Posee una gestión de seguridad (Gestión de usuarios, autenticación, roles, permisos).

1.2.4.3. Evaluación de los estándares de redes

Las delegaciones del MITRAB se encuentran interconectadas con el edificio central a través de una Red de Datos

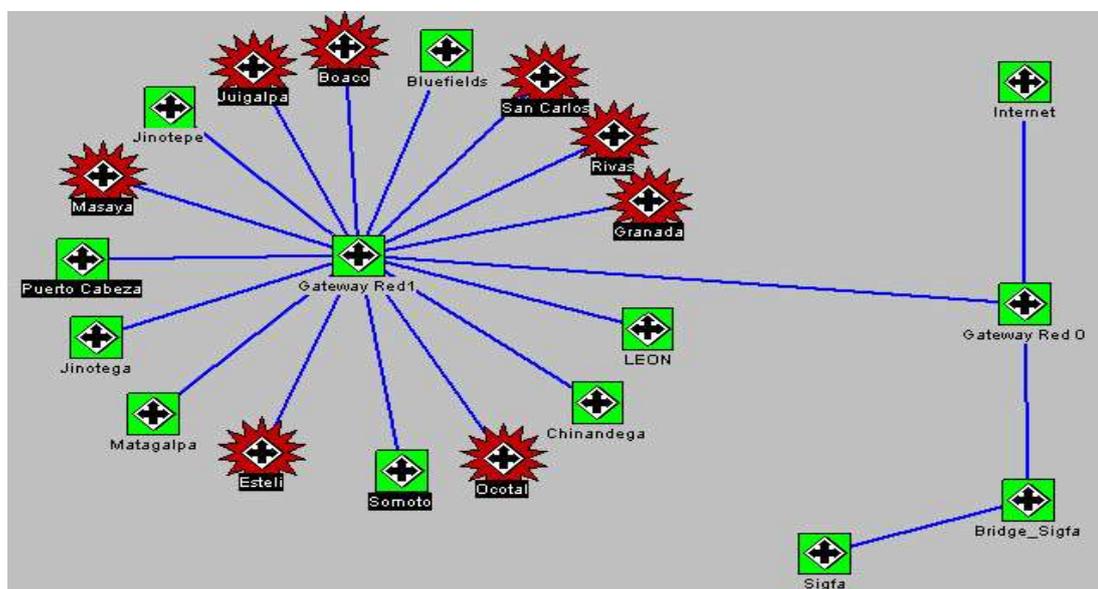


Figura 17: Red de Datos MITRAB

Fuente: Elaboración Propia

Cabe destacar que la red empresarial del MITRAB, no tiene VLANs, VPN, etc. Esto desencadena un sin número de brechas de seguridad, puesto que desde cualquier terminal se puede acceder a todos los sistemas, a los servicios, etc.

1.2.4.4. Evaluación de los estándares físicos

Respecto a los estándares físicos, se encuentran muchas deficiencias, principalmente en la sala de servidores, donde no existen las suficientes medidas de seguridad. Basta con ingresar al área de la División Informática para tener total acceso a dicha sala. Ahí se encuentran todos los servidores, centrales de redes, etc.

Cabe destacar que, durante la inspección visual, la puerta de los servidores siempre permaneció abierta, lo que significa una fácil y rápida entrada a cualquier persona. Además, no posee ninguna cámara de seguridad a lo interno de la sala, por lo que no se sabe quiénes entran. Los servidores se encuentran expuestos físicamente. La ISO 27001 Evalúa este ITEM a mayor detalle. (ver anexo 8: Análisis de Seguridad).

CAPÍTULO II: ESTUDIO PRELIMINAR

2.1. *Factibilidad Técnica*

2.1.1. Hardware (Análisis de la Arquitectura Informática)

La RED MITRAB es de topología de estrella, de alta disponibilidad dado el soporte del proveedor de servicio y de la dedicación del personal TI a las tareas orientadas red, la cual no tiene una estrategia para el almacenamiento masivo de datos definida, contando con 176 terminales de trabajo en la oficina de Managua, 112 en las delegaciones departamentales de los cuales se puede destacar que son terminales en su mayoría obsoletas (un 78.4% de los ordenadores en uso en Managua y un 70.5% de los ordenadores en los departamentos, deja un 70.4% de equipos en obsolescencia a nivel nacional que rebasan los 5 años de uso) y que operan con sistemas sin soporte, siendo puntos de desventaja para la institución. Se cuenta con una sala de servidores con 6 quipos físicos los cuales ponen a disposición los distintos servicios y aplicaciones a todos usuarios de la institución incluyendo las delegaciones departamentales (*Ver anexo 5: Análisis de arquitectura*). Podemos destacar las siguientes características:

- **Normalización, separación de funciones:** En la actualidad existe alta disponibilidad de los servicios de red, sin embargo, hay un vacío al momento de hablar de perfiles de usuarios que debiesen estar accediendo a estos. En otras palabras, no hay normalización de la red, ni separación de funciones de red.
- **Amplia conectividad:** RED MITRAB cuenta actualmente con una red de datos proporcionada por el proveedor de servicio ENITEL la cual interconecta las delegaciones departamentales con la oficina central de Managua. Dado este escenario los tiempos de respuestas son eficientes puesto que existe una amplia conectividad. A lo interno de cada delegación no existe redundancia en los equipos pasivos de red originando la movilización de inmediata del personal TI, siendo esta una actividad de alta prioridad.

- **Recursos compartidos.** Mediante las arquitecturas de red se pueden compartir recursos tales como impresoras y bases de datos, y con esto a su vez se consigue que la operación de la red sea eficiente y económica.
- **Administración de la red.** La División Informática del MITRAB se encarga de definir, operar, actualizar y dar mantenimiento a la RED. Interfaces, aplicaciones y facilidad de uso. Dado el despliegue de cableado estructurado (cantidad de interfaces de red por usuario) y las configuraciones básicas necesarias, es muy fácil que cualquier usuario tenga acceso a los servicios y aplicación en la red, sin embargo, no existe administración adecuada de los usuarios originando un problema de seguridad de la información.
- **Administración de datos.** Se toma en cuenta la administración de los datos y la necesidad de interconectar los diferentes sistemas de administración de bases de datos en los servidores, sin embargo, la administración de los datos almacenados es casi nula, generando datos basura en muchas ocasiones que afectan la eficiencia de los procesos de la institución, así como también la efectividad al realizar consultas y reportes con los datos exactos y precisos de un determinado intervalo de tiempo.

2.1.2. Diseño Lógico de Red

2.1.2.1. Diagrama Lógico de Red Actual

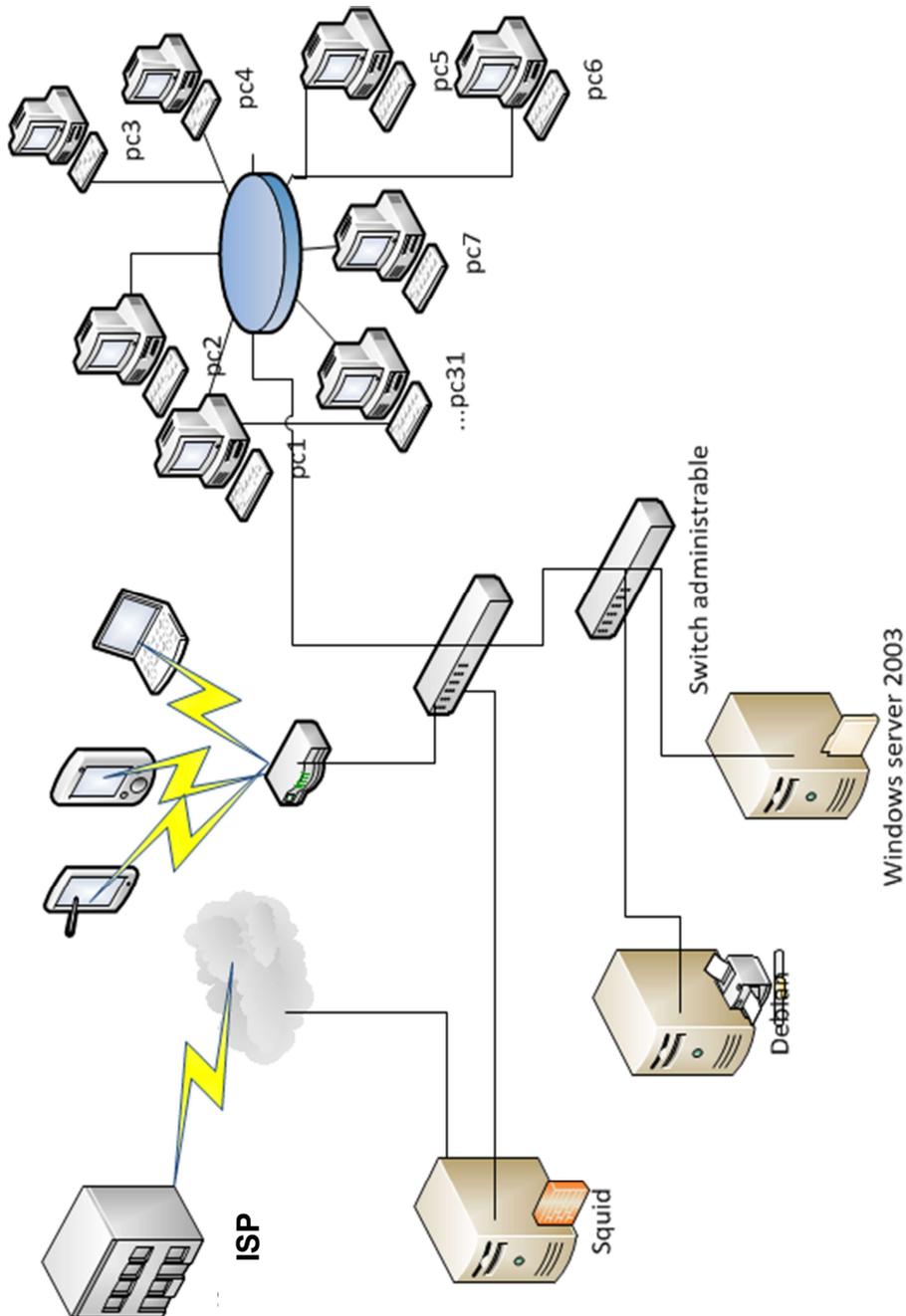


Figura 18. Diagrama Lógico de Red Actual
Fuente: Elaboración Propia

2.1.2.2. Diagrama Lógico de Red Propuesto

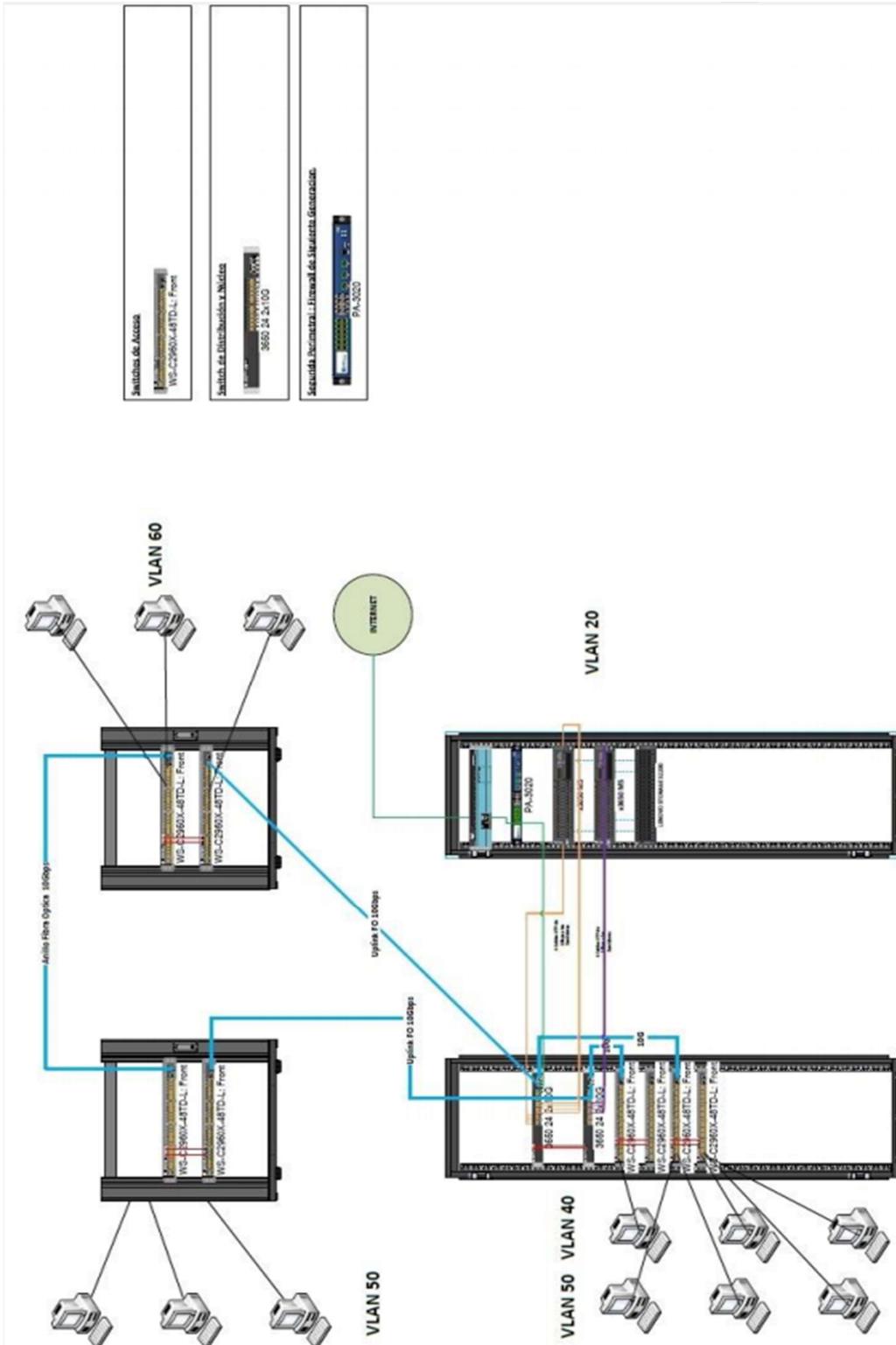


Figura 19. Diagrama Lógico de Red Propuesto
Fuente: Elaboración Propia

2.1.3. Propuesta de adquisición de Hardware

Con el fin de definir la mejor tecnología a implementar en el proyecto, se ha tomado como criterio que la misma debe garantizar sostenibilidad y mayores beneficios en el largo plazo; en este sentido se han comparado técnicamente dos opciones: Adquisición de equipos de cómputos convencionales (CPU, monitor, teclado y Mouse) o virtualización de escritorios (Terminales con las cuales se virtualiza el escritorio de los servidores Públicos) .

los equipos de cómputos actuales del MITRAB están obsoletos, esta situación afecta negativamente la funcionalidad y operatividad de las gestiones institucionales:

- Se cuenta con espacios de almacenamiento reducido.
- Procesador y memoria con baja velocidad lo cual no permite el procesamiento de datos, consultas e información de forma ágil y rápida.
- Equipos sin garantía vigentes, cada vez que se dañan la División de Informática del MITRAB implementa acciones correctivas en dependencia de la disponibilidad de repuestos, sin embargo por el tiempo de vida de los mismos hay dificultad en encontrar piezas en el mercado y no siempre son de la mejor calidad.
- Si se daña un equipo provoca estar fuera de servicio al usuario de 2 a 3 días mínimo y hasta un máximo de 15 a 20 días hábiles, dado que la adquisición de repuestos es tardada, hay que tomar en cuenta que tendríamos menos recursos para la atención a los usuarios que asisten al Ministerio solicitando un servicio institucional.

Bajo este escenario lo más factible es la adquisición de equipamiento nuevo y moderno. En este sentido se comparan los beneficios y sostenibilidad de dos alternativas.

Desde el punto de vista de beneficios técnicos funcionales si se compara la virtualización de escritorios con equipos de cómputo convencionales, es evidente

que las estaciones de trabajo usualmente ofrecen más alto rendimiento del que normalmente encontramos en las computadoras personales, mayor agilidad, reducción de costos de mantenimiento, la fiabilidad, compatibilidad, versatilidad, eficacia para entornos multiprocesos son otros de los beneficios de esta alternativa.

Una implementación de virtualización de escritorios ofrece las siguientes ventajas:

- **Centralización de la información**, Esto permite tener mayor seguridad de la información, evita fuga de datos y siempre está disponible las políticas de uso y acceso.
- **Movilidad de aplicaciones**, los usuarios podrán tener acceso a las aplicaciones relacionadas a su puesto de trabajo en todo lugar y momento con dispositivos móviles, con la debida autorización mediante políticas de seguridad y de su uso.
- **Equipos de usuario actualizados y controlados**, permitirá contar con equipos actuales a los usuarios pero con mayor control de uso de aplicaciones, únicamente permitirá usar aplicaciones relacionadas a sus funciones. Esto permitirá el uso únicamente gestiones de trabajo.
- **Sustitución y garantía**: Todos los equipos, servidores, estaciones de trabajo y de comunicaciones a ser parte del este proyecto contará con una garantía de 3 años, lo cual permitirá tener soporte y reposición de los equipos durante ese tiempo sin incurrir en compra de repuesto, así mismo se garantiza la sustitución del equipo en caso de ser necesario durante este periodo. Para que esta solución siga su duración y éxito es necesario la renovación del soporte a 3 años consecutivamente.
- **Alta disponibilidad**, Los servidores y equipos de comunicación tendrán configuraciones que les permitirán estar disponibles a todos los usuarios 100% de tiempo sin interrupciones.

- **Gestión inmediata:** Los equipos de comunicación y redes tendrán conexiones de alta velocidad entre ellos y entre edificios a 10 Gigabit Ethernet³⁷, así mismo con las estaciones de trabajo que sean parte del proyecto tendrán mayor velocidad de conexión a 1 Gigabit Ethernet, esto permitirá tener mayor fluidez en las conexiones, y gestión rápida de su información y aplicaciones.

En cuanto a los equipos de cómputos convencionales son equipos que a los 5 años de uso es necesario su reemplazo dado a la vida útil de los mismos; en cambio las estaciones de virtualización tienen una vida útil de 10 años, lo que se debe garantizar es la actualización del servidor, así como ampliación del soporte a los tres años de su adquisición.

Desde el punto de vista económico, si bien es cierto el costo de la inversión inicial de las estaciones virtualizadas es mayor a la compra de equipos de cómputos convencionales; a largo plazo los efectos de mantenimiento, sustitución, así como gasto energético tienen mayor impacto positivo de ahorro con las estaciones virtualizadas, podemos concluir que en la vida del proyecto tenemos los siguientes datos importantes para la toma de decisiones:

1. El costo de inversión de equipos de cómputos convencionales es de C\$ 4,980,035.42; esto incluye: monitor, teclado, mouse y licencia de sistema operativo; sin embargo el mantenimiento de esta alternativa conlleva gastos energéticos hasta por un monto de C\$ 7,151,510.48; así mismo por efectos de mantenimiento un gasto de C\$ 752,146.58; a este gasto se debe adicionar que en el año 5 se debe sustituir la flota de equipos de cómputos que ya dieron su vida útil, tomando en cuenta el crecimiento de los servidores públicos de cada área; en este sentido el gasto al finalizar el proyecto con esta alternativa asciende a C\$17,001,804.96
2. En cuanto a la adquisición de escritorios virtualizados tenemos que la inversión para adquisición de las mismas es de C\$ 6,927,895.69; su gasto de

³⁷ Gigabit Ethernet: también conocida como GigaE, es una ampliación del estándar Ethernet que consigue una capacidad de transmisión de 1 gigabit por segundo, correspondientes a unos 1000 megabits por segundo de rendimiento contra unos 100 de Fast Ethernet.

mantenimiento es de C\$ 95,931.11 apartir del tercer año, actualización de servidores requeridos en el quinto año asciende a C\$ 1,827,000.00 y gasto por consumo de energia es de C\$ 79,461.23; en este sentido el gasto al finalizar el proyecto es de C\$ 8,930,288.02

Comparando ambos costos vemos que durante la vida del proyecto, son mayores los costos incurridos al adquirir equipos de cómputos convencionales; 90% más costoso en comparación a la compra de equipos para la virtualización de escritorios; a estos beneficios debemos agregar todos los beneficios técnicos de la implementación de estaciones de trabajo virtualizada.

Basando en los criterios antes mencionados se opta por la tecnología que ofrece la virtualización de escritorios, esto significará un gran avance al proceso de actualización y renovación de la infraestructura informática del Ministerio del Trabajo, que se materializa en una mejor gestión al momento de atender a los usuarios que requieren de los servicios institucionales.

Esta Figura muestra los componentes claves de una implementación típica de de estaciones vistualizadas:

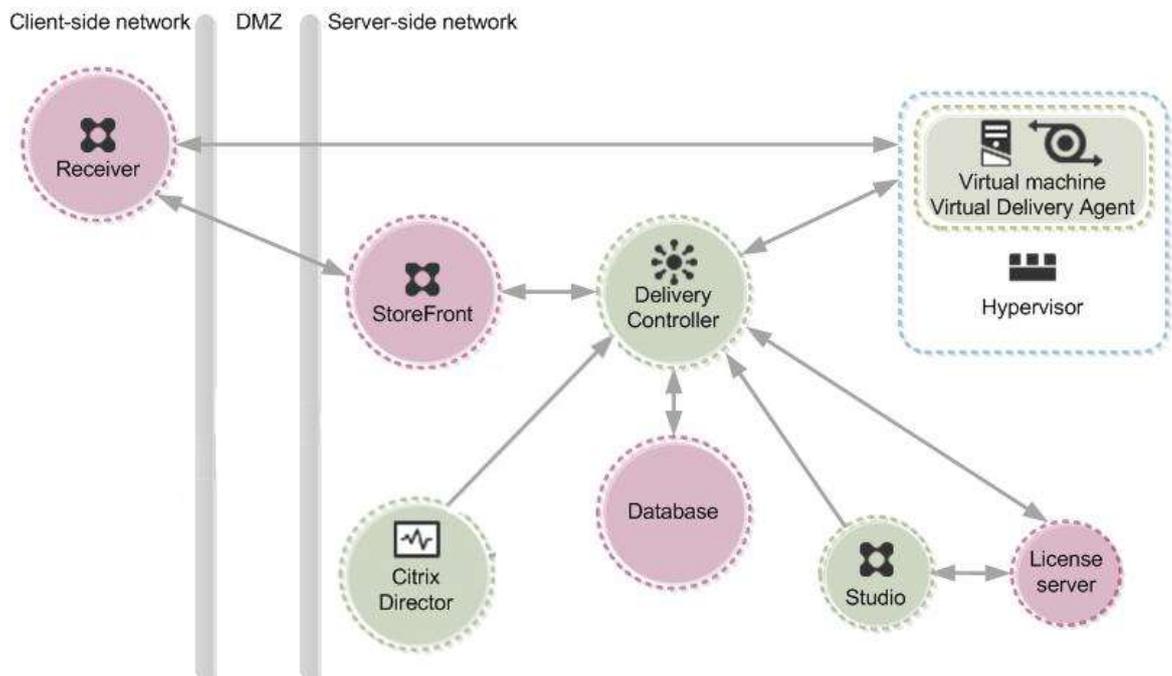


Figura 20. Claves de Implementación

Fuente: Elaboración Propia

Los componentes de esta Figura son:

- **Delivery Controller:** El Delivery Controller es el componente de administración central de las estaciones. Se compone de servicios que se comunican con el Hypervisor para distribuir aplicaciones y escritorios, autenticar y administrar el acceso de los usuarios, y actuar como intermediarios de las conexiones entre los usuarios y sus escritorios y aplicaciones virtuales, así como optimizar el uso y equilibrar la carga de dichas conexiones. El Controller administra el estado de los escritorios, iniciándolos y deteniéndolos, según la demanda existente y la configuración administrativa.
- **Base de datos:** Se necesita al menos una base de datos Microsoft SQL Server para almacenar toda la información de configuración y sesiones. Esta base de datos almacena los datos recopilados y administrados por los distintos servicios que conforman el Controller.
- **Virtual Delivery Agent (VDA):** El VDA se instala en cada equipo físico o máquina virtual del sitio que quiere poner a disposición de los usuarios. Permite que la máquina se registre con el Controller, que, a su vez, permite que la máquina y sus recursos alojados estén disponibles para los usuarios. Los VDA establecen y administran la conexión entre la máquina y el dispositivo del usuario, verifican que haya una licencia de Citrix disponible para el usuario o para la sesión, y aplican las directivas que se hayan configurado para la sesión. El VDA comunica la información de la sesión al Broker Service en el Controller a través del Broker Agent incluido en el VDA.
- **StoreFront:** StoreFront autentica a los usuarios en los sitios donde se alojan los recursos, y administra almacenes de escritorios y aplicaciones a los que acceden los usuarios. Aloja el almacén de aplicaciones necesarias para las gestiones operativas, que permite el autoservicio de los usuarios al acceder a los escritorios y aplicaciones disponibles para ellos. También hace un seguimiento de las suscripciones a aplicaciones de los usuarios, los nombres

de los accesos directos y otros datos, para asegurarse de que existe coherencia de uso entre los distintos dispositivos.

- **Receiver:** Se instala en los dispositivos de usuario y otros dispositivos de punto final, como, por ejemplo, escritorios virtuales. Da a los usuarios un acceso rápido, seguro y de autoservicio a los documentos, las aplicaciones y los escritorios, desde cualquier dispositivo del usuario.
- **Studio:** es la consola de administración que permite configurar y administrar la implementación, lo que elimina la necesidad de usar consolas distintas para administrar la entrega de aplicaciones y de escritorios. Studio incluye varios asistentes que funcionan como guía para la configuración del entorno, la creación de cargas de trabajo para alojar escritorios y aplicaciones, y la asignación de éstos a los usuarios. También puede usar Studio para asignar licencias de Citrix y realizar un seguimiento de las mismas en el sitio.
- **Director:** Es una herramienta basada en Web que permite a los equipos de asistencia técnica y TI supervisar un entorno, solucionar problemas antes de que se vuelvan graves, y realizar tareas de asistencia técnica para los usuarios finales.

Esta propuesta va acompañada de renovar la Infraestructura Tecnológica del Ministerio del Trabajo para contar con la tecnología necesaria que permita satisfacer las demandas de nuevas aplicaciones y tendencias tecnológicas, se trata de adquirir equipo tecnológico para mejorar la gestión del centro de datos de la Institución como estrategia de mejora en la atención e interacción en sus servicios tecnológicos.

Con esta propuesta el Ministerio del Trabajo fortalecerá su plataforma de comunicación permitiéndoles:

- Mejorar los accesos de los usuarios
- Aumentar la escalabilidad
- Mejorar la comunicación
- Mejorar la flexibilidad
- Aumentar el rendimiento de las Aplicaciones

- Analizar el comportamiento de la RED LAN

La solución está compuesta por Switches 3650 de 24 puertos con 4 puertos UpLinks para fibra, cada uno en Stack para proporcionar redundancia a nivel del Core y conexión de los gabinetes situados en la Institución específicamente en Local I Sector Construcción y Oficina de Información Pública OAIP.

Este par de Switches estarán realizando la labor de sustitución del Switch DELL de 48 puertos que actualmente tiene la Institución.

Específicamente la propuesta de solución después de un estudio de selección y proyección de 10+ años en su vida útil está compuesta por :

- Switches Cisco Catalyst 3650 24 Port Data 2x10G Uplink IP Base WS-C3650-24TD-S, cada uno en stack para proporcionar redundancia a nivel de la parte core y conexión de los gabinetes ubicados en la Institución.
- Reemplazo de los switches actuales por switches de mayor performance y velocidad de puertos ya que estos actualmente operan en velocidad Fast Ethernet (10/100), este tipo de performance está quedando obsoleto debido a que la mayoría de las aplicaciones, servidores e infraestructura de red está operando en 1GE, por esto se propone 4 switches Cisco Catalyst 2960-X 48 GE Port Data, 2 x 10G SFP+, LAN Base WS-C2960X-48TD-L, estos serán ubicados en el gabinete de comunicaciones del Centro de Datos principal del MITRAB.
- Reemplazo de los switches que actualmente operan en velocidades de 10/100, por 4 switches Cisco Catalyst 2960-X 48 GE Port Data, 2 x 10G SFP+, LAN Base WS-C2960X-48TD-L, en la operatividad de los puertos funcionando a 1GE.
- Creación de Anillo de Fibra Óptica a 10Gbps.
- Instalación y Configuración de Equipos de Acceso Cisco 2960x.
- Instalación y Configuración de Equipos de Core CISCO 3650.
- Configuración de la Alta Disponibilidad y Tolerancia de Fallos en Anillo de Fibra, VLAN, Spanning Tree, Ruteo entre VLANS, rutas por Defecto.

- Instalación y Configuración de Firewall, Equipo Palo Alto PA3020 NGFW como Firewall de Seguridad Perimetral.
- Instalación y Configuración de Firewall L7, Equipo F5 como Firewall de Seguridad a nivel de Aplicaciones.

2.1.4. Software

Se seleccionan estas herramientas ya que estas están basadas en software libre, razón por la cual no se necesita invertir en la adquisición de licencias de software cerrados. Así mismo por la funcionalidad y confiabilidad que estas proporcionan.

Como resultado de la evaluación técnica se determinó que el negocio posee la infraestructura tecnológica (hardware y software) necesaria para la puesta en funcionamiento del sistema

TECNOLOGIA	LOGO	DESCRIPCIÓN
HTML 5		última versión de HTML, que se utilizará para la estructura y presentación de las páginas web.
CSS3		Hojas de estilo de cascada que se usará para el aspecto y la forma de las páginas web
JAVASCRIPT		Lenguaje interpretado que se embeberá en las páginas web, provocando que las instrucciones sean analizadas y procesadas por el navegador en el momento que deben ser ejecutadas.

<p>JAVA</p>		<p>Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Lo realmente impresionante de esta tecnología es que se alza sobre todas las plataformas de hardware y ejecuta los códigos de bits que, en vez de ser específicos de una arquitectura, incluyen instrucciones para ser ejecutados en una máquina virtual, siendo así uno de los componentes fundamentales de esta plataforma: la Máquina Virtual Java (JVM).</p>
<p>MONGODB</p>		<p>MongoDB forma parte de la nueva familia de sistemas de base de datos NoSQL. En lugar de guardar los datos en tablas como se hace en las bases de datos relacionales, MongoDB guarda estructuras de datos en documentos similares a JSON con un esquema dinámico (MongoDB utiliza una especificación llamada BSON), haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.</p>
<p>EHCACHE</p>		<p>Ehcache es una caché de código abierto, basado en estándares que aumenta el rendimiento, repercute en la base de datos, y simplifica la escalabilidad. Es el más usado caché basada en Java porque es robusta y</p>

		probada, con todas las funciones, y se integra con otras bibliotecas y los marcos populares.
--	--	----------------------------------------------------------------------------------------------

Tabla 3. Tecnologías del Software

Fuente: Elaboración Propia

2.2. Factibilidad Operativa

Para determinar si el sistema a desarrollar será de utilidad para la institución se requiere evaluar el impacto que provocará su implementación en el entorno institucional, tanto a nivel del personal como de los procesos institucionales, identificando así la factibilidad operativa del sistema.

2.2.1. Impacto en los Procesos

Según el análisis de los procesos institucionales realizado, la institución cuenta con un porcentaje del 27.18% de automatización de procesos incluyendo el uso de formatos en Excel, el cual es un indicador muy bajo dada la importancia de la institución y las tareas que realiza, cabe destacar que este porcentaje de automatización se ha alcanzado mediante el uso de múltiples sistemas de información que no están interconectados generando aún más tareas a las áreas sustantivas, ya que están tienen que recopilar información de distintas fuentes(sistemas, formatos Excel, formatos físicos) procesar esta información y transmitirla al área de fortalecimiento institucional encargada de procesar las estadísticas de la institución, dicho esto los beneficios que traería consigo el sistema de información laboral del MITRAB son de vital importancia para la institución ya que no solo permitiría un nivel de automatización del casi 100%, por el nivel de flexibilidad del sistema, sino que también permitirá a las áreas sustantivas enfocarse en su quehacer institucional y no en el procesamiento de información.

2.2.2. Impacto en el Personal

Actualmente el personal del MITRAB tiene que interactuar con hasta 7 sistemas informáticos los cuales no permiten un nivel de automatización aceptable lo que obliga al personal a utilizar formatos físicos o en Excel para solventar esta falta, aumentando así la carga laboral al personal de la institución, es por esta razón que se observó durante el estudio de la situación actual, la necesidad del personal de un sistema que permitiese agilizar y reducir la complejidad de las tareas que realizan además de liberar al personal de tareas que no son parte de los objetivos de su área, como es la recopilación de información y su tratamiento para generación de estadísticas, además durante entrevistas al personal de la institución se manifestó la necesidad de un sistema en línea, que permita al personal el acceso desde cualquier dispositivo, ya que muchas tareas son realizadas fuera de la institución, por ejemplo las inspecciones laborales.

2.2.3. Conclusión de la factibilidad Operativa.

Con respecto a los puntos anteriores se concluye que el sistema será operativamente factible debido a que permita la automatización de los procesos institucionales, la integración de las distintas áreas sustantivas y contara con la aceptación de los usuarios ya que supone una herramienta vital para apoyo a las actividades que realizan y el acceso a la información, a la vez que reduce su carga laboral.

2.3. Factibilidad Económica

Para conocer el costo asociado al desarrollo del Core Gerencial del Sistema de Información Laboral, se procedió a calcular su valor a través de la implementación del Modelo Constructivo de Costos.

Partiendo del análisis de los procesos, formatos y variables a considerar en la automatización, así como del tiempo definido para la implementación de la iniciativa (19 meses), se determina un aproximado de 82,919.2 líneas de código,

para ello se requerirá de 9 personas con un salario de mercado para este tipo de especialidad.

En este sentido la propuesta para el desarrollo del “CORE GERENCIAL DEL SISTEMA DE INFORMACIÓN LABORAL DEL MINISTERIO DEL TRABAJO (CORE-SIL)” asciende a C\$ 8,095,128.5119 (Ver Anexo 6: COCOMO). Además, se propone fortalecer la infraestructura tecnológica del MITRAB bajo el esquema de la terna de seguridad de la información garantizando integridad, disponibilidad y confiabilidad de la información. Para ello se propone la adquisición de los equipos de seguridad, procesamiento, almacenamiento, comunicación y respaldo de energía con una inversión de C\$ 7,530,000 (*Ver Anexo 7: Adquisición de Equipos*)

Por otro lado, se propone la adquisición de estaciones de trabajo virtualizadas con una inversión de adquisición de C\$ 6,927,895.69; su gasto de mantenimiento es de C\$ 95,931.11 apartir del tercer año (Ver Anexo 8: Costos de equipos para virtualización de escritorios)

Se prevee actualización de servidores requeridos en el quinto año asciende a C\$ 1,827,000.00 y gasto por consumo de energía es de C\$ 79,461.23.

CAPÍTULO III: ANÁLISIS DEL SISTEMA

3.1. Descripción del Sistema de Información

A través del análisis de la problemática existente se formularon los objetivos del sistema informático.

3.2. Objetivos del Sistema de Información

- Generar Marco de Trabajo el cual permita la gestión de la información resultante de los procesos institucionales, habilitando el aprendizaje continuo, alineando al MITRAB al cumplimiento de su misión.
- Reducir los tiempos dedicados por parte la Dirección Superior del MITRAB

a la elaboración de informes de gestión.

- Aumentar la visibilidad por parte la Dirección Superior del MITRAB, habilitando criterios para la toma de decisión en tiempo oportunos.
- Crear un esquema de seguridad nivel de aplicación, que permita garantizar la confidencialidad, integridad y disponibilidad de la información, generando un único punto de control para la gestión de los usuarios, así como las autorizaciones correspondientes a cada sistema informático que integran el SIL
- Crear un API de integración para la interconexión del sistema informático que integran el SIL

3.2.1. Definición y descripción de la información de entrada y salida

El Modelo de Desempeño del MITRAB, está basado en el PRM de FEA, presentado anteriormente en el marco teórico y ha sido diseñado para ilustrar las relaciones causa – efecto entre entradas, salidas y resultados de los servicios, procesos internos que los soportan y recursos que participan, con el fin de comprender en qué grado cada entrada o intervención está colaborando a conseguir los resultados estratégicos de la institución, agrupados en las seis perspectivas de objetivos estratégicos mencionadas en el levantamiento del planteamiento estratégico.

El modelo de desempeño de referencia, PRM, ha sido adaptado al contexto y al Modelo de negocio del MITRAB, ya que el PRM persigue en primer lugar ser una poderosa herramienta de evaluación de iniciativas de inversión para la elaboración de presupuestos.

El Modelo de Desempeño del MITRAB, presenta un enfoque a la generación de valor a través de los resultados, definiendo como resultados, los obtenidos respecto a los objetivos estratégicos de manera directa, los obtenidos respecto a la misión del negocio, los obtenidos por los clientes (empleadores / trabajadores), el desempeño de los procesos y actividades y el desempeño y valor de los recursos y mecanismos.

El Modelo de Desempeño del MITRAB es transversal a la organización y se estructura de manera jerárquica alrededor de áreas de medición, categorías de medición y grupos de medición.

- Áreas de medición: pertenecen a la capa jerárquica más alta y capturan los aspectos de desempeño a nivel de resultados. El Modelo de Desempeño contempla cuatro distintas áreas de medición:
 - Resultados para la Misión y Resultados del Negocio
 - Resultados para el Cliente
 - Desempeño y valor de los Procesos y Actividades
 - Desempeño y valor de los Recursos (Financieros, Humanos, Instalaciones y Tecnología).
- Categorías de Medición: Diferentes actividades o servicios que se llevan a cabo dentro de cada área de medición, que representan objetos de medición.
- Grupos de Medición: atributos a medir para cada categoría de medición. Representan potenciales objetivos estratégicos en su máximo nivel de especificidad.
- Indicador: la medición genérica.

Utilizando un diagrama se puede especificar de mejor forma el enfoque hacia la generación de valor que genera la estructura del Modelo de Desempeño del MITRAB

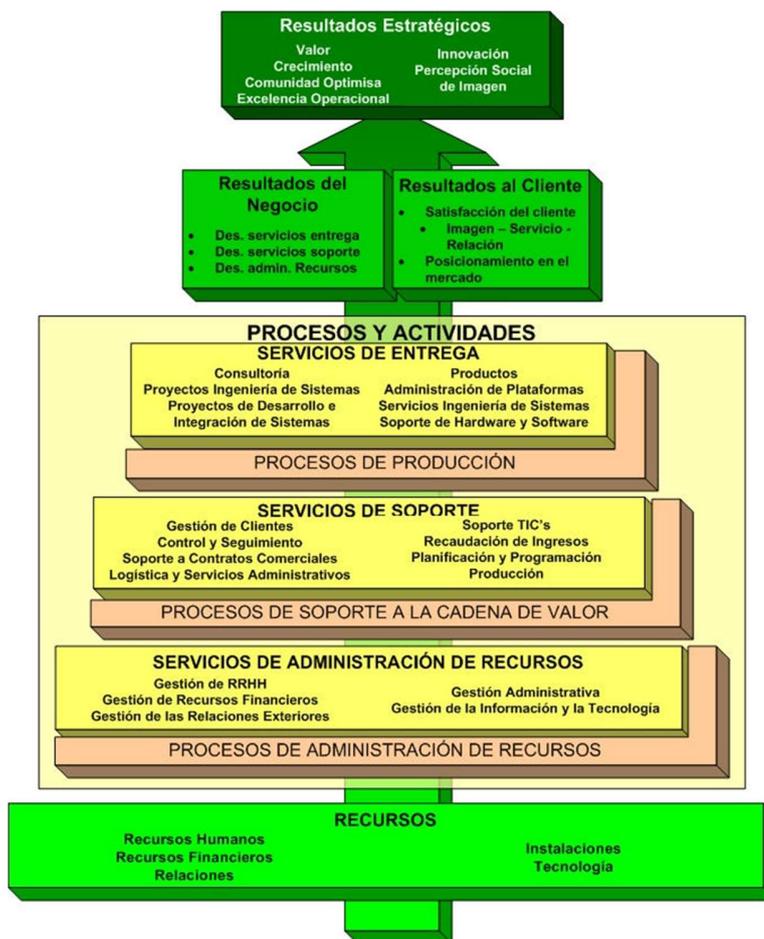


Figura 22. Enfoque Modelo

Fuente: FEA Consolidated Reference Model Versión 2.1: www.egov.gov

3.3. Arquitectura del Sistema de Información.

Las actividades apoyadas por aplicaciones Web que permitirán crear, navegar y editar las diferentes estructuras del modelo de desempeño y su arquitectura, tendrán un diseño genérico basado en la estructura detallada en el punto de Diseño del Modelo de Desempeño, utilizando la arquitectura de estado. Si la arquitectura de estado no fuera capaz de soportar algún requerimiento de información por parte del planteamiento estratégico generado, será responsabilidad del administrador de la información generar una estructura que satisfaga este requerimiento. Las actividades enunciadas, implicarán entonces la factibilidad de navegar a través de los niveles jerárquicos del Modelo de Desempeño, compuesto por Modelo – Área de Medición – Categoría de Medición

– Indicador – Plan de Acción – Actividad – Asignación, con la funcionalidad de crear, editar y borrar. A continuación, se presenta la arquitectura tecnológica estándar que soporta estas actividades, basada en tecnología Web.

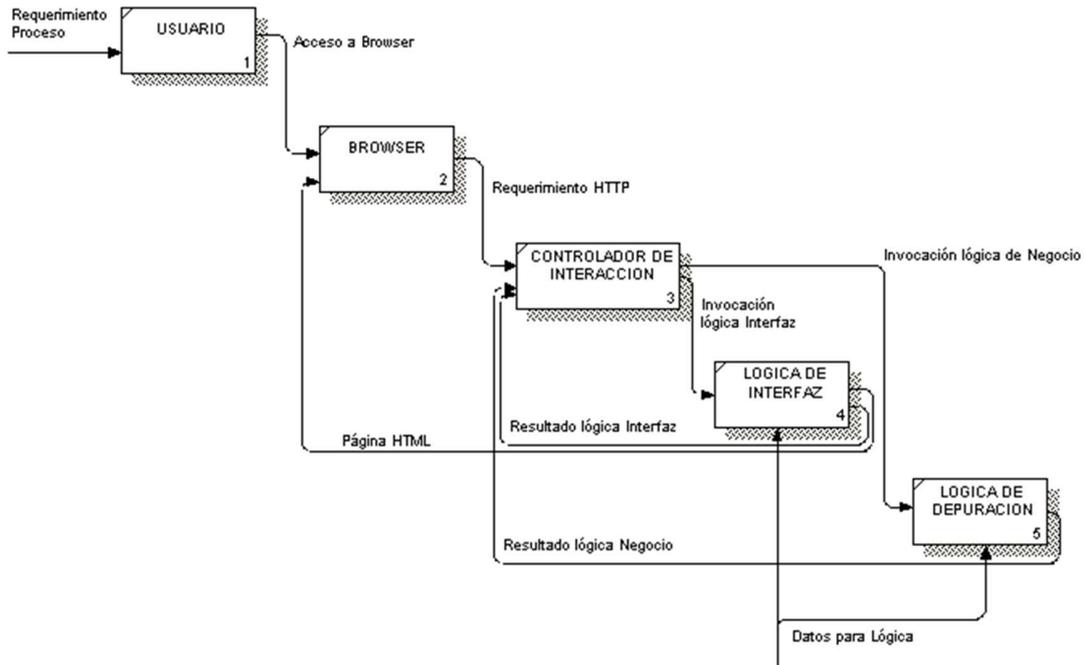


Figura 23. Arquitectura del Sistema

Fuente: <http://www.oracle.com>

La arquitectura genérica expuesta anteriormente, deja entrever el modelo MVC (Model – View – Controller), o modelo de tres capas, donde la primera capa, es la de acceso a datos; la segunda capa es la de presentación o interfaz; y la tercera capa es la de lógica de negocio y depuración, que es representada en el modelo como un recurso. Cabe destacar que esta arquitectura genera un paso trazable desde las actividades del proceso a la ejecución de la lógica del negocio y su interfaz con la actividad, sea un usuario u otra.

3.4. Tipificación del Sistema de Información.

El sistema de información a crear es de tipo transaccional, ya que automatiza gestión de la información resultante de los procesos institucionales. Además, es

de tipo multiusuario porque lo utiliza más de un actor. Su arquitectura es de dos niveles: cliente –servidor.

3.5. Alineamiento entre el proceso y el Sistema de Información.

Con la finalidad de transparentar el paso del diseño del proceso a el diseño del sistema de información de apoyo, se indicarán las aplicaciones que soportarán a cada actividad del proceso con apoyo, luego se mostrará el esquema de solución planteado y finalmente como estas aplicaciones cumplen con la funcionalidad que se hace explícita en el punto siguiente de modelamiento de casos de uso. A continuación, se describe brevemente cada componente del apoyo expuesto.

- **Administración de Seguridad a nivel de aplicación:** Marco de referencia que provee seguridad a nivel de aplicación, se encarga del análisis de peticiones de clientes para verificar su autenticidad y validez mediante el uso de políticas y reglas. Provee métodos para la correcta gestión de usuarios, niveles de autorización, creación y manejo de sesiones mediante el uso de buenas prácticas y granularidad de permisos. Además, proporciona un API de interoperabilidad entre distintas aplicaciones.
- **Administración de Modelos de Negocio:** Componente que otorga al administrador de uno o más modelos de desempeño: crear, modificar o borrar modelos y todas sus entidades jerárquicas (contenedores y variables). Claramente esta componente apoya a la actividad de generación de la estructura del Modelo de Desempeño del MITRAB.
- **Administración de Inteligencia de Negocio:** El Procesamiento, monitoreo y gestión de los indicadores estadísticos socio-labórales a nivel nacional, a través de la utilización de Balanced Scorecard para la definición de objetivos, indicadores e iniciativas estratégicas en cuatro perspectivas base; financiera, clientes, procesos internos y aprendizaje-crecimiento.

3.6. Requerimientos Funcionales.

Los requerimientos funcionales hacen referencia a detalles del sistema que debe de cumplir para que este sea funcional. Son las tareas fundamentales con las

cuales el Sistema trabajará y funcionará. Con el CORE-SIL, los requerimientos funcionales son:

- CORE-SIL debe de permitir a los usuarios potenciales interactuar con un API Rest-Full.
- CORE-SIL debe de proveer Servicios para manejo de Modelos.
- CORE-SIL debe de proveer Servicios para manejo Instancias de Modelos.
- CORE-SIL debe de proveer Servicios para manejo de Contenedores de Modelos.
- CORE-SIL debe de proveer Servicios para manejo de Variables de los Modelos.
- CORE-SIL debe de proveer Servicios para manejo y generación de Indicadores y Reportes.
- CORE-SIL debe de proveer Servicios para manejo de Cuentas de Usuario.
- CORE-SIL debe de proveer Servicios para manejo de Roles de seguridad.
- CORE-SIL debe de proveer Servicios para manejo de Aplicaciones que se conectan con el sistema.
- CORE-SIL debe de proveer Servicios para manejo de Módulos de aplicaciones.
- CORE-SIL debe de proveer Servicios para manejo de Roles de seguridad.
- CORE-SIL debe de proveer Servicios para manejo de Políticas de seguridad.
- CORE-SIL debe de proveer Servicios para manejo de Servicios web del propio CORE-SIL.
- CORE-SIL debe de proveer Servicios para visualización de Bitácora.
- CORE-SIL debe de proveer Servicios para visualización de Lista negra.
- CORE-SIL debe de proveer Servicios para visualización de sesiones.

3.7. Requerimientos No Funcionales.

Los requerimientos no funcionales hacen referencia a características del sistema de manera general, requerimientos que son adicionales a los funcionales pero que deben de cumplirse de igual manera. Estos aspectos son tales como: disponibilidad, flexibilidad, mantenibilidad, seguridad, facilidad de uso, etc.

En ese sentido, CORE-SIL, tendrá como requerimientos no funcionales los siguientes:

- Debe de garantizarse la confiabilidad, seguridad y disponibilidad para los diferentes actores que interactuaran con el sistema.
- La seguridad del sistema será un requisito no funcional sumamente importante, ya que se manejarán datos muy delicados de índole corporativos de las diferentes empresas-ciudadanos.
- Los tiempos de respuesta deben de ser aceptables, de acuerdo a las posibilidades tecnológicas de las que disponga CORE-SIL por medio del MITRAB.
- CORE-SIL debe de estar construido de tal forma que cualquier cambio pueda ser agregado sin afectar en gran manera el funcionamiento anterior.
- Debe de ser de fácil uso, para cada uno de los usuarios, muy intuitivo y práctico.
- Los mensajes de error deben de ser específicos, para que el usuario identifique fácilmente el tipo de error cometido.
- Todo el sistema deberá estar debidamente documentado, tanto el código fuente como los manuales de administración y usuario.

3.8. Requerimientos de Restricciones.

Para un funcionamiento adecuado de CORE-SIL, las siguientes restricciones serán de vital importancia:

- El actor Administrador será el primer usuario registrado en la Base de Datos.
- Solo puede existir un usuario Administrador.
- Solo el Administrador puede dar de baja, habilitar y asignar roles de usuario a los usuarios desarrolladores
- El Administrador solo puede ver datos relevantes y a groso modo del registro de las operaciones de los procesos de negocio.
- El usuario desarrollador solo podrá administrar las aplicaciones que el administrador les de permisos.

CAPÍTULO IV: DISEÑO DEL SISTEMA.

4.1. Definición de Actores.

Un actor es una idealización de un proceso que interactúa con el sistema, se caracteriza por tener interacciones con el sistema. Un usuario físico puede estar limitado a los actores múltiples dentro del sistema.

Los actores para el sistema son:

	NOMBRE	ROL
	Administrador de Sistema	<p>Persona el cual posee todos los privilegios dentro del sistema (administración de Modelos de Negocio, administración de Inteligencia de Negocios, administración de Seguridad).</p> <p>Es el único responsable crear aplicaciones y gestionar las autenticaciones pertinentes para los desarrolladores.</p> <p>Además de tomar medidas de seguridad a nivel de aplicación de una anomalía en las sesiones de los usuarios finales.</p>
	Desarrollador de Sistema	<p>Persona con conocimientos en desarrollo de sistema de Información, el cual crea aplicaciones implementando los modelos de negocio los cuales son autorizados por un administrador del sistema. Este puede editar o borrar formularios, administrar sus variables y crear Indicadores de desempeño en base a</p>

		<p>estas.</p> <p>Estas aplicaciones pueden ser de tipo implementación externa el cual puede ser desarrollada bajo cualquier tecnología y puede ser integrada a través del API del sistema gerencia o bien de tipo implementación interna la cual puede ser accedida a través del cliente base del sistema.</p> <p>Además, puede administrar el esquema de seguridad de las aplicaciones creadas (sesiones, usuarios, roles, políticas, bitácora)</p>
	<p>Usuario de Sistema</p>	<p>Perfil que gestiona la información operativa de la institución, la cual mediante el uso de los formularios de los distintos procesos de negocio suministra información a las aplicaciones, alimentando las variables operativas y a su vez los indicadores claves de desempeño.</p> <p>Este puede también puede ver el desempeño institucional.</p>

Tabla 5. Definición de Actores
Fuente: Elaboración Propia

4.2. Diagrama de Paquetes.

En este diagrama los elementos son organizados en grupos separados por especialidad de cada módulo como son, gestión de Inteligencia de Negocios, Gestión de Modelos de Negocio y Administración de la Seguridad, especificando en cada paquete las diferentes entidades que los componen, se muestran en el siguiente diagrama:

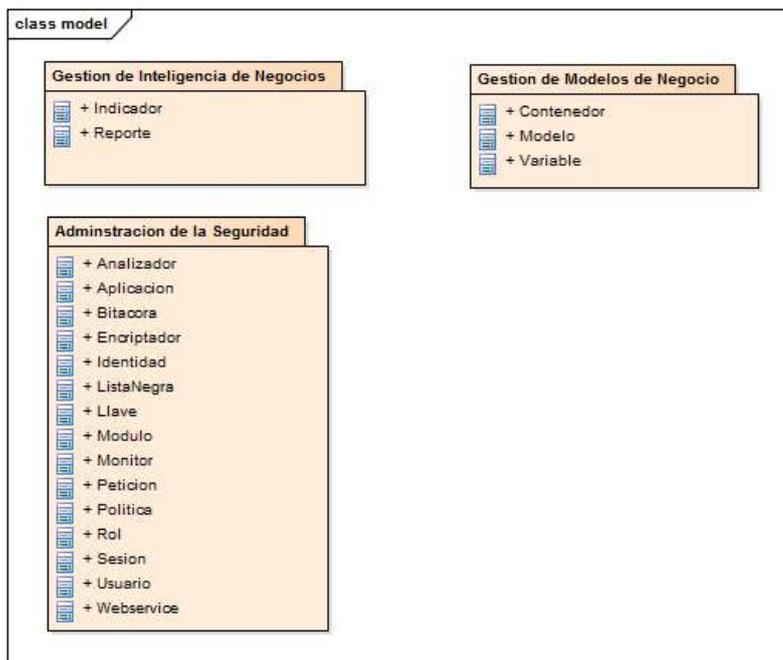
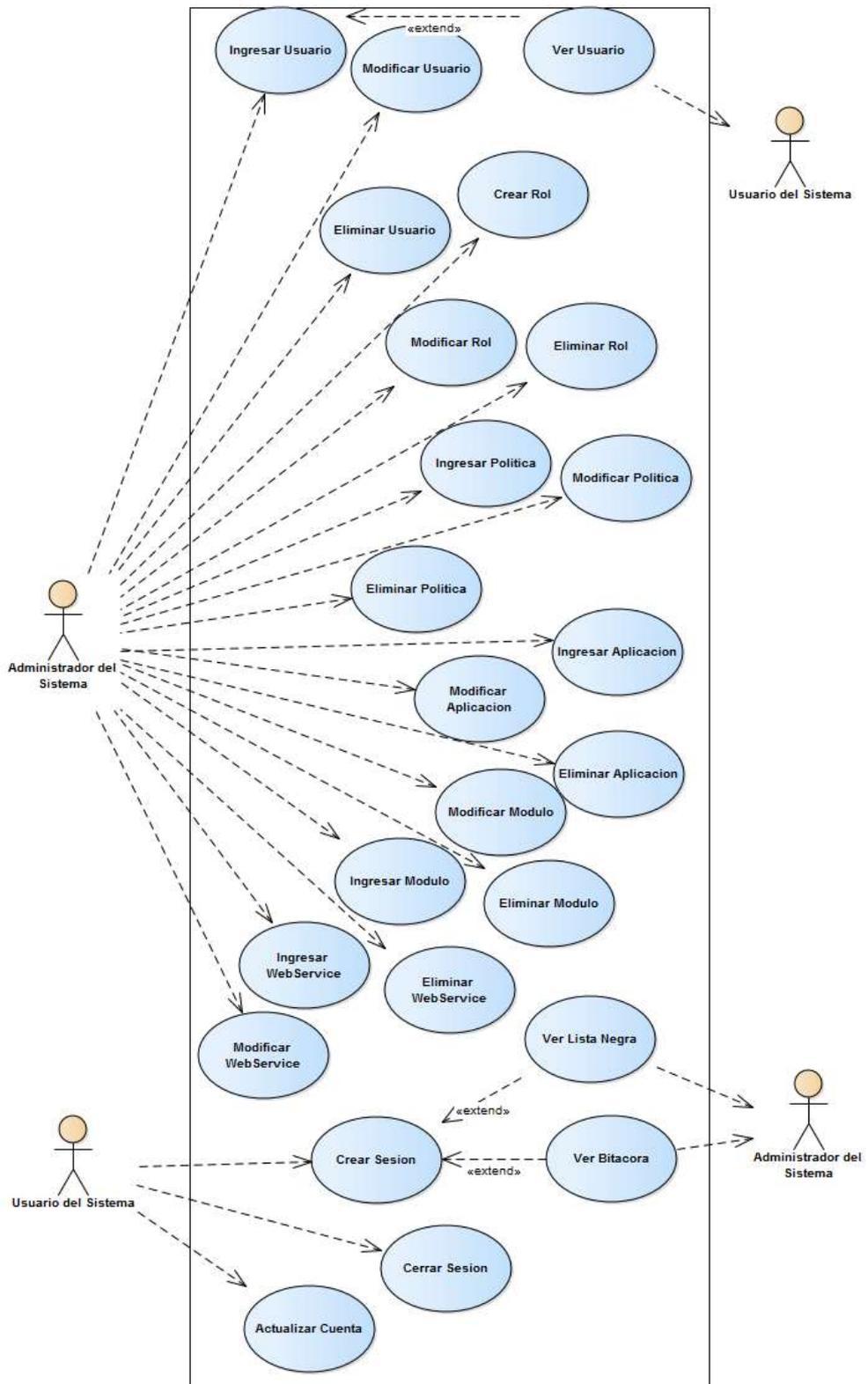


Figura 24. Diagrama de Paquete

Fuente: Elaboración Propia

4.3. Modelamiento de Casos de Uso UML.

Mediante el modelamiento y especificación de casos de uso, se pretende derivar la funcionalidad requerida por el proceso a la aplicación de apoyo que se plantea diseñar y construir. Cabe destacar que la práctica de modelamiento y especificación de casos de uso UML (Unified Modeling Language) está inserta en la mayoría de las metodologías de diseño de ingeniería de software más modernas y exitosas. A continuación, se presentan los Casos de Uso del sistema:



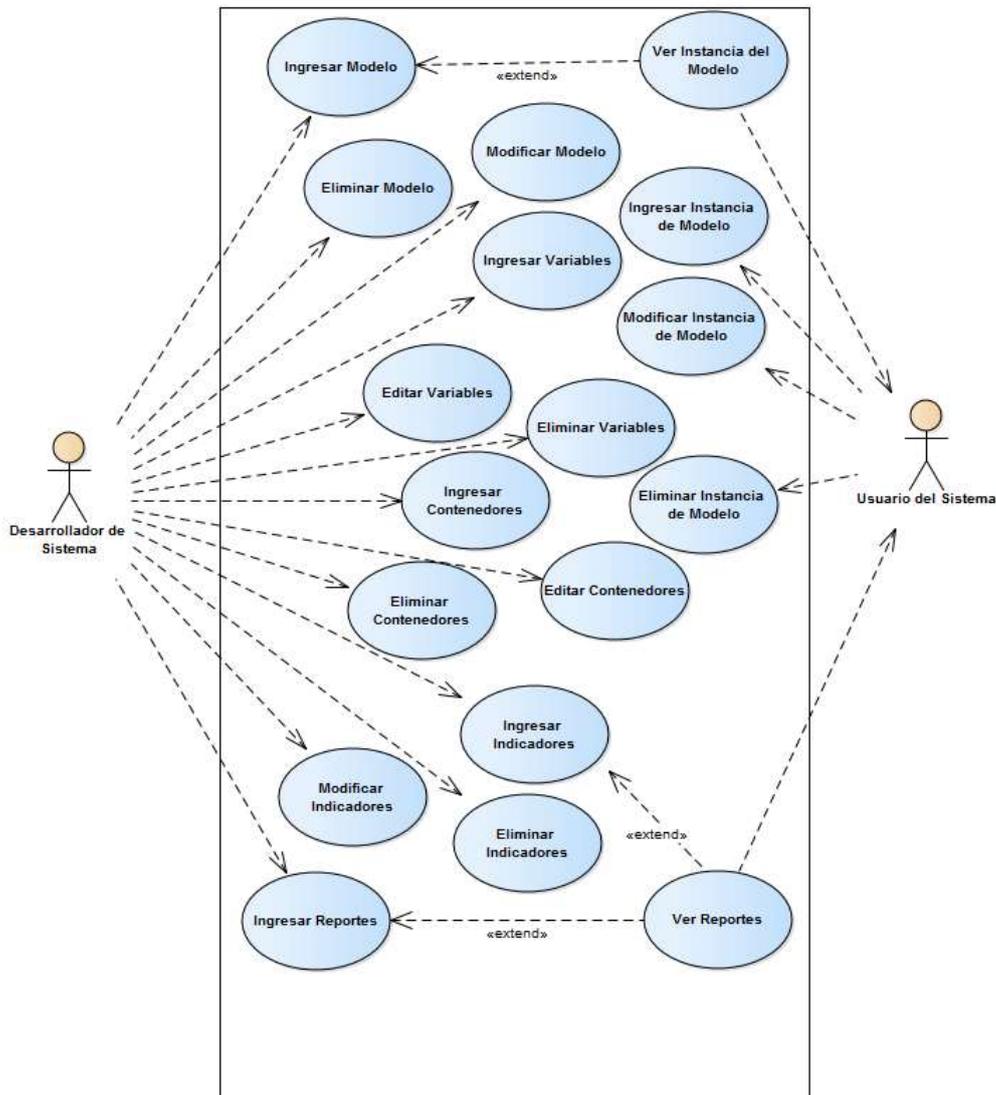


Figura 25. Diagrama de Casos de uso
 Fuente: Elaboración Propia

4.4. Plantillas de Coleman.

Las especificaciones de los casos de uso que presentan las principales funciones o tareas realizadas por el actor, así como información de las operaciones y excepciones de cada interacción del usuario con el sistema. A continuación, se muestran las plantillas de Coleman del sistema. (Ver Anexo 9: Plantillas de Coleman.)

4.1. Diagramas de Secuencia.

Son un tipo de diagramas de interacción que se utilizan para modelar aspectos dinámicos del sistema, se describe el flujo de orden temporal (que pasa primero y que después), en estos se refleja el responsable de cada actividad y su relación con el sistema, por cada acción realizada por el usuario. Una vez que el usuario realiza una solicitud de información, esta se ejecuta mediante una consulta en la base de datos, según el tipo de información y permisos de usuario, el sistema se comporta de cierta manera para dar salida a la solicitud enviada. A continuación, se muestran cada diagrama de secuencia, se muestra en ambos aspectos de salida del caso, es decir, de manera satisfactoria e insatisfactoria. (Ver Anexo 10. Diagramas de Secuencia.)

4.2. Diagramas de Comunicación.

Otro tipo de diagramas de interacción son los de comunicación, destacados por la organización estructural de los objetos que envían y reciben mensajes, mostrando la relación de comunicación que hay entre las entidades involucradas para realizar una tarea. De igual manera cada caso de uso genera un diagrama de comunicación. A continuación, se muestran los diagramas de comunicación, se muestra en ambos aspectos de salida del caso, es decir, de manera satisfactoria e insatisfactoria. (Ver Anexo 11. Diagramas de Comunicación.)

4.3. Diagrama de Clases.

Este diagrama muestra la estructura de las clases del sistema, así como también las relaciones que hay entre ellas. Se define la solución de un diseño y ayuda a construir componente de software orientados a objetos.

Los diagramas de clases son los más útiles para ilustrar las relaciones entre clases e interfaces. A continuación, se muestra el diagrama de clases elaborado para el sistema.

4.6. Modelo de Datos.

Un modelo de datos puede ser pensado como un diagrama de flujo que ilustra las relaciones entre los datos, el siguiente diagrama determina la estructura lógica de la base de datos utilizada para el sistema:

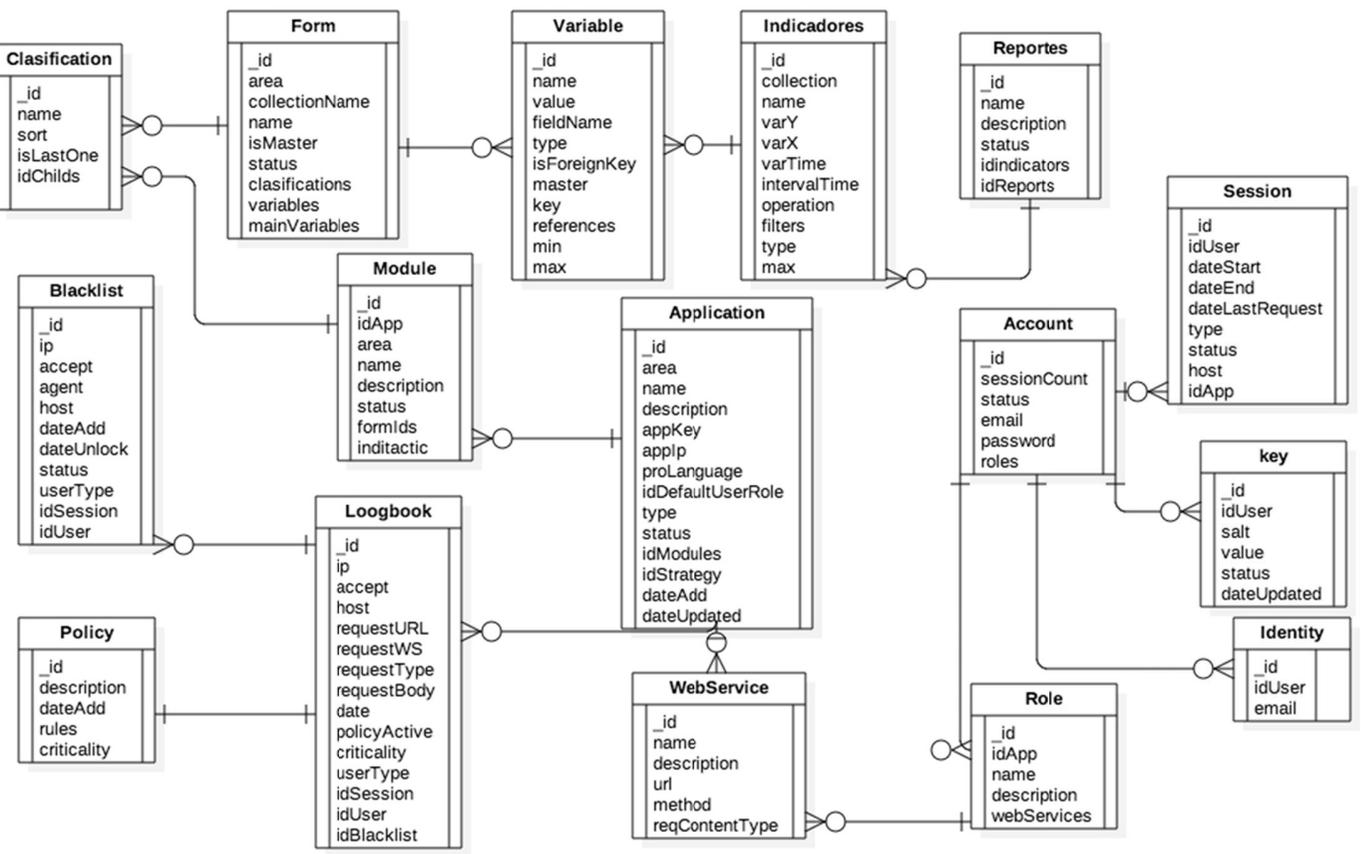


Figura 27. Modelo de Datos
Fuente: Elaboración Propia

4.7. Diagramas de estados.

El diagrama de estados muestra los diferentes estados por los que puede pasar un objeto, así como las transiciones y eventos asociados, muestra el ciclo de vida de un objeto desde el momento en que este es creado. A continuación, se presentan los diagramas de estado por cada objeto dentro del sistema. (Ver Anexo 12. *Diagramas de Estados*).

4.8. Diagramas de despliegue.

El diagrama de despliegue muestra la configuración de nodos de procesamiento en tiempo de ejecución y los componentes que residen en ellos, muestran la configuración de elementos del proceso.

4.8.1 Diagrama Despliegue Actual.

Actualmente el MITRAB posee una estructura de despliegue simple, sin dispositivos de seguridad avanzados que eviten la vulnerabilidad de la red de trabajo local

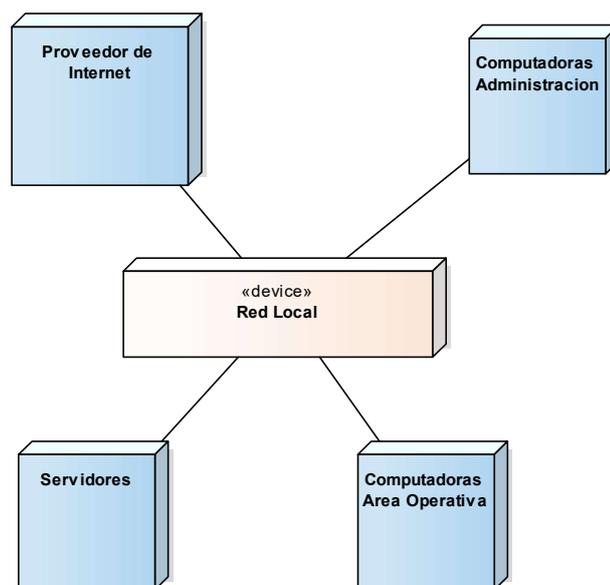


Figura 28. Diagrama de despliegue Actual
Fuente: Elaboración Propia

4.8.2 Diagrama Despliegue Propuesto.

Dentro del diagrama propuesto se plantea conectar la red local con el proveedor de internet a través de dispositivos con altos índices de seguridad, como son un Core cisco, seguido de un Firewall a nivel de capa 3, que distribuya la red a través de Switches para cada uno de los sectores de computo, los cuales son para el área administrativa, el área operativa y los servidores, este último tendrá una seguridad con un firewall a nivel de capa 7.

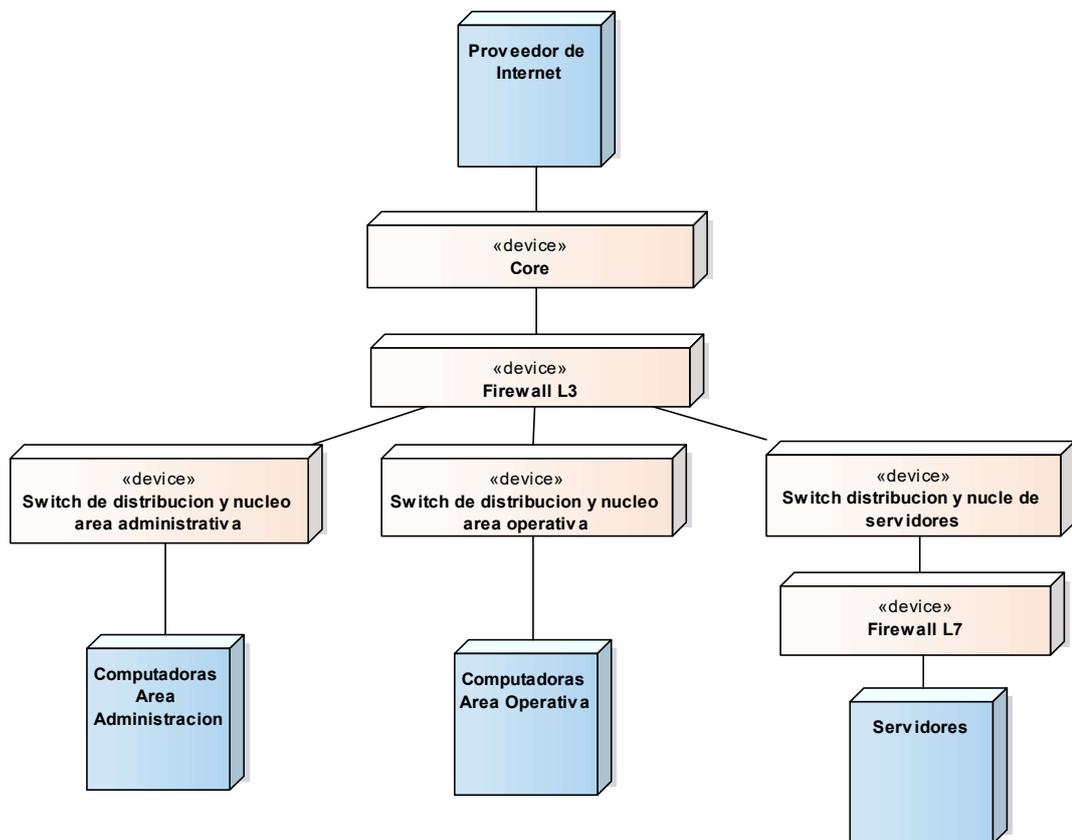


Figura 29. Diagrama de despliegue Propuesto

Fuente: Elaboración Propia

CAPÍTULO V: PRUEBAS DEL SISTEMA.

Con el fin de garantizar una alta calidad del sistema desarrollado y una ejecución acorde a los requisitos y requerimientos establecidos en el análisis del sistema, se elaboraron pruebas tanto funcionales así como no funcionales, mediante la aplicación de distintos softwares especializados.

5.1. Pruebas Funcionales.

Para asegurar el pleno funcionamiento del sistema acorde a las funciones anteriormente descritas en el diseño del sistema, se hizo uso de la herramienta POSTMAN³⁸ diseñada para el proceso de pruebas de APIs Restfull la cual utilizamos para verificar la funcionalidad de los 65 servicios web que conforman el core gerencial del MITRAB.

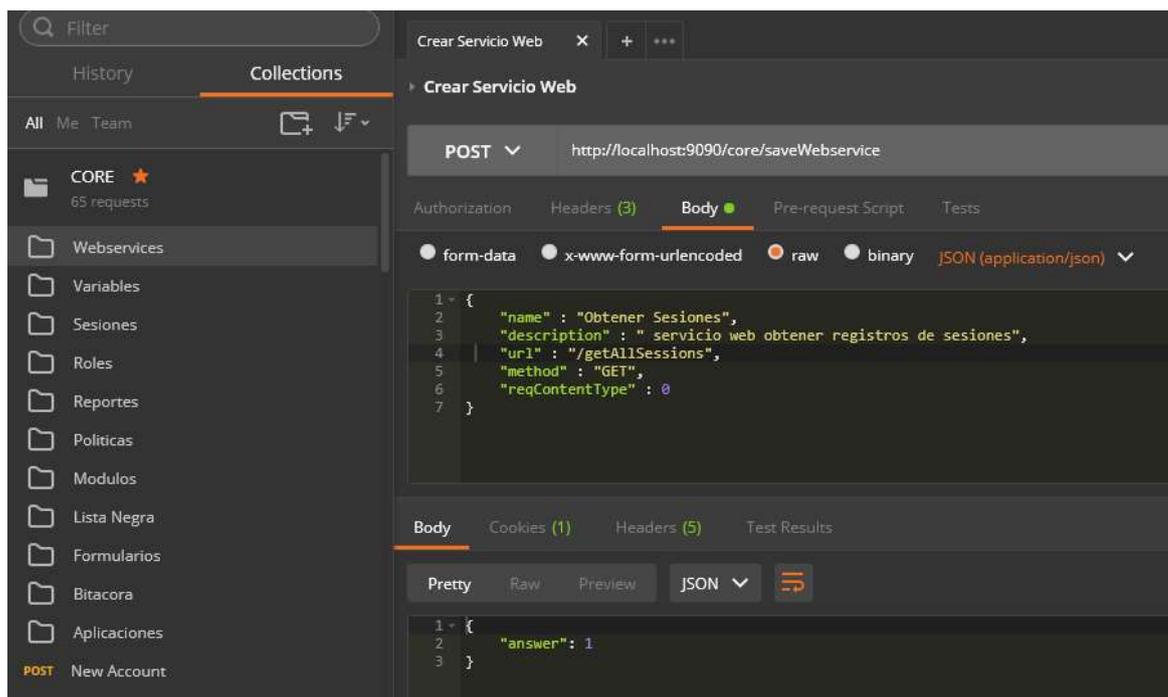


Figura 30. POSTMAN
Fuente: Elaboración Propia

³⁸ Postman - <https://www.getpostman.com/>

En la figura 30 podemos observar una captura de la herramienta POSTMAN la cual nos permite verificar que el resultado obtenido de una petición a un determinado servicio web del sistema, nos devuelve el resultado esperado, asegurando de esta manera la correcta funcionalidad establecida del sistema.

La herramienta utilizada además de ser multiplataforma, también permite guardar los escenarios de prueba, los cuales pueden volverse a ejecutar luego de realizada una actualización al sistema, para verificar que ninguna funcionalidad ha sido afectada luego de un cambio de versión, también la herramienta permite que los escenarios de prueba puedan ser exportados para ser ejecutados desde distintos ordenadores, por tal razón se estará brindando una copia de estos escenarios de prueba junto con este trabajo.(Ver DVD: Pruebas Funcionales)

5.2. Pruebas No funcionales.

Desde el aspecto no funcional, se debe comprobar que el sistema presente un alto rendimiento y capacidad de soportar mucho trabajo debido a la gran necesidad de generación de estadísticas en tiempo real por parte de la institución y la cantidad de usuarios del sistema, por tal razón se decidió utilizar Gatling³⁹, herramienta especializada en pruebas de estrés y rendimiento, la cual está escrita en scala⁴⁰ y se ejecuta por medio de comandos produciendo reportes de cada prueba en formato HTML para ser visualizados en cualquier navegador.

³⁹ Gatling, Open-Source Load & Performance Testing Tool For Web Applications - <https://gatling.io/>

⁴⁰ Scala, Lenguaje de programación multi-plataforma basado en Java. - <https://www.scala-lang.org/>

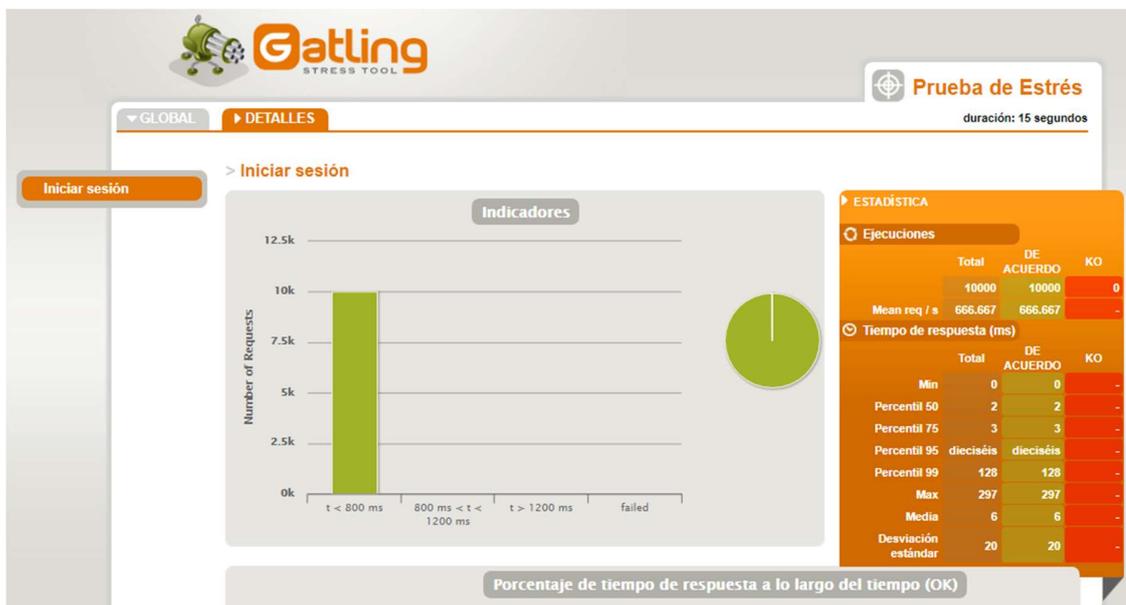


Figura 30. GATLING
Fuente: Elaboración Propia

Las pruebas de estrés se realizaron a los principales servicios web del sistema y con diferentes niveles de estrés(desde 100 hasta 10,000 peticiones simultaneas), en la figura 31, observamos el reporte HTML que presenta Gatling como resultado de una prueba de estrés realizada al Login del sistema en el cual el servicio web se sometió a 10,000 peticiones de inicio de sesión simultaneas, la prueba tuvo una duración de 15 segundos, y el sistema respondió satisfactoriamente un promedio de 666.67 peticiones/segundo, lo cual pone en evidencia la correcta optimización y rendimiento del sistema gracias al uso de métodos asíncronos y cache del sistema.

Los demás resultados de las pruebas de estrés se encuentran en DVD: Pruebas No Funcionales, aunque de manera general podemos decir que el sistema tardo un máximo de 2 segundos en responder a una petición, aun con un estrés de 10,000 peticiones simultaneas, cabe destacar que las pruebas fueron realizadas en un equipo de desarrollo, por tanto, se espera que en un entorno productivo el sistema se ejecute de manera más eficaz y eficiente.

CAPÍTULO VI: DOCUMENTACIÓN DEL SISTEMA.

Para una correcta adopción del sistema por parte de los usuarios, y un fácil proceso de adición de funcionalidad y actualizaciones futuras del sistema, es necesario contar con una sólida base documental, que parte desde el Análisis del sistema hasta el Manual de usuario y que permita el crecimiento del sistema a lo largo del tiempo.

6.1. Diseño del Sistema.

El diseño del sistema fue elaborado utilizando el software de modelado Enterprise Architect ⁴¹ lo que facilita el proceso de diagramación siguiendo los estándares del UML, y también permite la modificación del diseño del sistema según nuevos requerimientos, todo esto empaquetado en un solo archivo el cual se encuentra en DVD: Diseño del Sistema.

6.2. Código Fuente.

Muchos de los sistemas del MITRAB no cuentan con código fuente propiamente documentado, esta debilidad ha imposibilitado a tales sistemas a evolucionar a la par de la institución, ya que sin un código fuente debidamente estructurado, legible y sin documentación realizar actualizaciones a estos sistemas se vuelve muy difícil.

Por esta razón el código fuente del core gerencial utiliza nombre de métodos legibles bajo el estándar camelCase y se encuentra debidamente documentado mediante el uso de la herramienta Javadoc⁴², creada por Oracle como un estándar para documentar sistemas desarrollados utilizando Java.

Exportado a PDF el Javadoc del Core gerencial, cuenta con 304 paginas donde se detallan todos los paquetes, clases, objetos y métodos del sistema, facilitando

⁴¹ Enterprise Architect - Plataforma de Modelado integrada - <http://www.sparxsystems.com.ar/products/ea/index.html>

⁴² Javadoc Tool - <http://www.oracle.com/technetwork/articles/java/index-jsp-135444.html>

futuras modificaciones y correcciones que se deseen realizar al código fuente, dicho PDF se puede encontrar en DVD: JAVADOC CORE.

6.3. Manual de Usuario.

El core gerencial del sistema de información laboral como todo software requiere de un manual de usuario para sacar el máximo provecho a todas las funcionalidades que posee, pero al ser una API integral de tipo restful, su manual de usuario está desarrollado bajo la OpenApi Specification⁴³, esta especificación establece el uso de un archivo descriptor en formato YAML o JSON (en DVD: OPENAPI), para crear manuales de usuarios dinámicos⁴⁴ como aplicaciones web que pueden conectarse directamente al API y que además pueden generar SDK o código cliente en diferentes lenguajes para facilitar la integración del API con otros sistemas (**ver anexo 13: Manual de Usuario del API**).

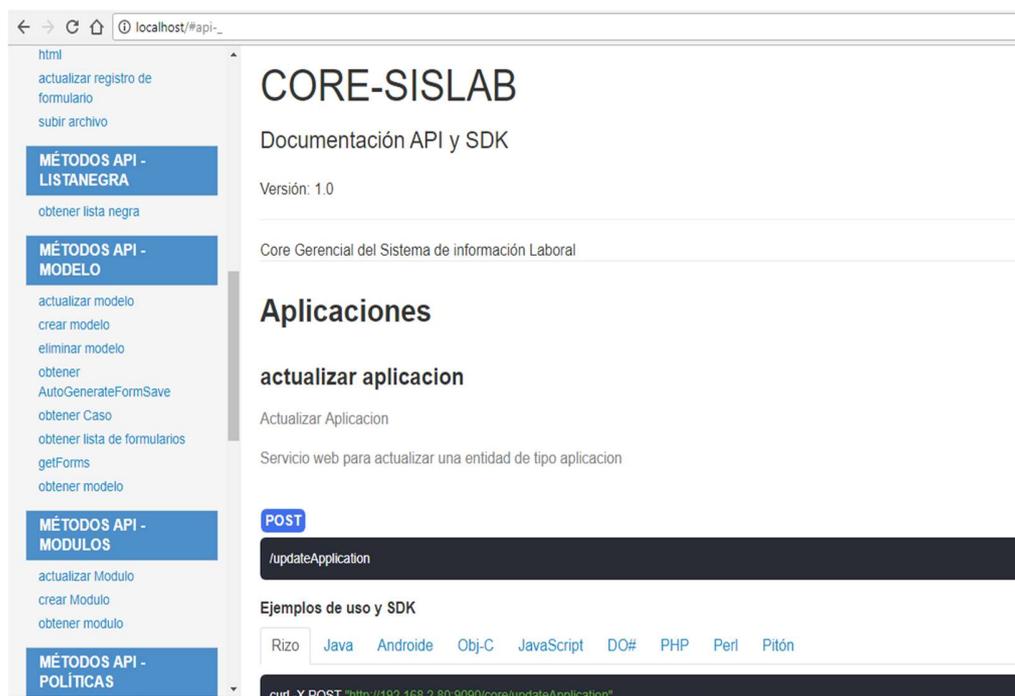


Figura 30. MANUAL DE USUARIO DEL API

Fuente: Elaboración Propia

⁴³ The OpenAPI Specification - <https://github.com/OAI/OpenAPI-Specification>

⁴⁴ Swagger Editor 2 - <http://editor2.swagger.io/#/>

VI. CONCLUSIONES.

- Por medio del estudio organizacional se encontró que la estructura actual y sus atribuciones son propicias para la implementación del sistema, pero que a la falta de automatización y seguridad de la información son una gran debilidad de la organización.
- A través del estudio de los requerimientos y un examen exploratorio preliminar se logró cimentar los fundamentos necesarios para la comprensión y concepción del diseño del sistema informático por medio de la abstracción de los procesos principales reflejados en los diagramas de actividades, los cuales forman parte de las funciones principales del sistema.
- El estudio de factibilidad operativa refleja que el elemento humano implicado en la utilización del sistema está debidamente capacitado para usarlo.

El estudio de factibilidad económica refleja que a través de las ventajas institucionales que el MITRAB presenta, éste puede asumirlos satisfactoriamente las siguientes inversiones:

- Partiendo del análisis de los proceso, formatos y variables a considerar en la automatización, así como del tiempo definido para la implementación de la iniciativa (19 meses), se determina un aproximado de 82,919.2 líneas de código, para ello se requerirá de 9 personas con un salario de mercado para este tipo de especialidad. En este sentido la propuesta para el desarrollo del “CORE GERENCIAL DEL SISTEMA DE INFORMACIÓN LABORAL DEL MINISTERIO DEL TRABAJO asciende a C\$ 8,095,128.5119
- Es necesario fortalecer la infraestructura tecnológica del MITRAB bajo el esquema de la terna de seguridad de la información garantizando integridad, disponibilidad y confiabilidad de la información. Para ello se propone la adquisición de los equipos de seguridad, procesamiento, almacenamiento, comunicación y respaldo de energía con una inversión de C\$ 7,530,000 (Ver Anexo 12: Adquisición de Equipos)
- De cara a optimizar recursos se propone la adquisición de estaciones de trabajo virtualizadas con una inversión de adquisición de C\$

6,927,895.69

- Mediante la utilización del lenguaje unificado de modelado y de un exhaustivo análisis, se obtuvo un diseño visual-conceptual que cumplía con los requerimientos de automatización expresados por los clientes y que sirvieron de un excelente marco referencial para el desarrollo del sistema. Dentro de lo que cabe mencionar, se desarrollaron ocho tipos de diagramas fundamentales de UML: casos de usos, clases, actividades, componentes, secuencia, colaboración, despliegue y paquetes que cumplen con los estándares necesarios. En total se obtuvieron cincuenta y cinco casos de uso, treinta clases y cinco diagramas de actividades.
- Se logró desarrollar un producto software que cumpliera con los estándares de diseño, requisitos funcionales y no funcionales del CORE-SIL.
- Mediante el uso de herramientas de software especializadas se realizaron las pruebas al CORE-SIL verificando así, su correcto funcionamiento, rendimiento y capacidad para cumplir con sus requerimientos establecidos.
- Basándose en estándares globalmente aceptados se realizó la documentación del CORE-SIL desde su diseño (UML), su código fuente (JAVADOC), y el API como tal (OPENAPI Specification) para obtener un sistema fácil de entender y modificar según el crecimiento de la institución.

VII. RECOMENDACIONES

Primeramente, se recomienda al Ministerio del Trabajo implementar el Core gerencial del Sistema de Información Laboral planteado en este documento, para centralizar el flujo de información, interconectar de manera segura sistemas y usuarios, y generar estadísticas de los indicadores socio-laborales, además se recomienda:

- Implementar esta propuesta metodológica en el Ministerio del Trabajo con vistas a tomar de referencia su resultado, para facilitar la generalización en otras instituciones gubernamentales.
- Incorporar a nivel de implementación web e implementación en dispositivos móviles el CORE-SIL, así de este modo, podría estar disponible desde cualquier punto que se solicite.
- Considerar la migración a herramientas de software libre para el desarrollo y/o conversión de los sistemas.
- Mudar ambiente de producción a uno que tenga actualizaciones regularmente y mayor interactividad.
- Realizar un análisis de los perfiles laborales de las Instituciones Gubernamentales, a fin de definir los perfiles requeridos ya que esto no formó parte del estudio y se hace necesario para un trabajo futuro.

VIII. BIBLIOGRAFÍA

- Gobierno de Nicaragua. (s.f.). Recuperado el 06 de 07 de 2014, de <http://www.gobenic.gob.ni/comision/QS/estrategia>, 01 de diciembre de 2016.
- Gobierno de Nicaragua. (s.f.). Gobierno de Nicaragua. Recuperado el 01 de diciembre de 2016, de <http://www.gobenic.gob.ni/comision/QS/estrategia>
- Gobierno de Nicaragua. (2013). Plan Nacional de Desarrollo Humano.
- American Productivity & Quality Center (APQC), Open Standards Benchmarking Collaborative (OSBC). (s.f.). Recuperado el 01 de diciembre de 2016, de <http://apqc.org/osbc>
- Office of Management and Budget's (OMB). FEA Consolidated Reference Model Versión 2.1. (s.f.). Recuperado el 01 de diciembre de 2016, de <http://www.egov.gov>
- Baca, G. (2006). Formulación y evaluación de proyectos informáticos. México D.F.: McGraw-Hill Interamericana.
- Cohen Karen, D., & Asín Lares, E. (2005). Sistemas de información para los negocios: un enfoque para la toma de decisiones. México D.F.: McGraw-Hill Interamericana.
- Connolly, T., & Begg, C. (2005). Sistemas de bases de datos: un enfoque práctico para el diseño, implementación y gestión. Madrid: Pearson Education.
- Post, G. (2006). Sistemas de administración de bases de datos. México D.F.: McGraw-Hill Interamericana.
- Schach, S. (2005). Análisis y diseño orientado a objetos con UML y el proceso unificado. México D.F.: McGraw-Hill Interamericana.
- Schach, S. (2006). Ingeniería de software clásica y orientada a objetos 6ta edición. México D.F.: McGraw-Hill
- Booch, G., Rumbaugh, J., & Jacobson, I. (1999). El lenguaje unificado de modelado. México D.F.: Addison Wesley.
- Statcounter. (s.f.). Recuperado el 01 de diciembre de 2016, de <http://gs.statcounter.com>

- W3techs.com. (s.f.). Recuperado el 01 de diciembre de 2016, de <http://w3techs.com>
- W3schools. (s.f.). Recuperado el 01 de diciembre de 2016, de <http://www.w3schools.com>
- Owasp. (s.f.). Recuperado el 01 de diciembre de 2016, de <http://www.owasp.com>
- Oracle. (s.f.). Recuperado el 01 de diciembre de 2016, de <http://www.oracle.com>
- MITRAB. (s.f.). Recuperado el 01 de diciembre de 2016, de <http://www.mitrab.gob.ni>

IX. ANEXOS

9.1. ANEXO 1: Análisis de Procesos MITRAB

Área	Procesos
Inspectoría Laboral	16

No	Proceso	Sistema que lo automatiza	Número de tareas por proceso	Número de tareas automatizadas	Porcentaje de automatización
1	Inspección ordinaria	File master	34	5	14.70 %
2	Re-inspección	File master	31	9	29.03%
3	Inspección especial	File master	52	14	26.92%
4	Inspección por Comparencia	File master	42	17	40.47%
5	Ilegalidad de Huelga	File master	32	12	37.5%
6	Reglamentos internos del trabajo	File master	54	24	44.45%
7	Multas por incumplimiento en Materia de Derechos Laborales	File master	89	42	47.20%
8	Procedimiento para trámites conciliatorio en las inspectorías	File master	24	19	79.17%

	departamentales de Managua.				
9	Solicitud de cancelación de contrato de trabajo	File master	75	37	49.34%
10	Suspensión Colectiva de Contratos Individuales de Trabajo	File master	45	19	42.22%
11	Cierre de Empresa	File master	43	17	39.54%
12	Pliegue de Peticiones	File master	43	26	60.47%
13	Proceso de Apelación	File master	33	19	57.58%
14	Cambio de horarios de trabajo	File master	54	25	46.30%
15	Comisión Tripartita	File master	26	3	11.53%
16	Constancia para personas adolescentes que cumplen con la edad mínima de admisión al empleo	File master	19	4	21.05%

Porcentaje de automatización del área	40.5%
----------------------------------------------	--------------

El área consta con todos los procesos que se involucran en las tareas de la inspectoría laboral, sus procesos principales son las Inspecciones ordinarias, re-inspecciones e inspecciones especiales esto es debido a que son la base de la mayoría de los demás procesos y/o sub-procesos como el proceso de Apelación y Multas. El objetivo de esta área es la de verificar que tanto los empleadores y trabajadores cumplan con las leyes laborales establecidas en la constitución laboral, así como la de hacerlas cumplir, siempre y cuando en la inspección se encuentren alguna infracción o anomalía.

Área	Procesos
Higiene y Seguridad	27

No	Proceso	Sistema que lo automatiza	Número de tareas por proceso	Número de tareas automatizadas	Porcentaje de automatización
1	Inspección en materia de higiene y seguridad ocupacional	SISCHYSTRAB	41	8	19.52%
2	Re-inspección en materia de higiene y seguridad ocupacional	SISCHYSTRAB	39	5	12.82%
3	Proceso de Suspensión Temporal o Total y	SISCHYSTRAB			

	Levantamiento de Riesgos				
4	Proceso de la inspección por comparecencia	SISCHYSTRAB			
5	Proceso para investigaciones de accidentes laborales (graves, muy graves y mortales)	SISCHYSTRAB			
6	Proceso para la solicitud de prórroga para el cumplimiento de plazos de las disposiciones establecidas en inspección de higiene y seguridad	SISCHYSTRAB			
7	Proceso sancionador en materia de higiene y seguridad (multas)	SISCHYSTRAB			
8	Proceso para otorgar y extender licencia para funcionamiento de equipos generadores de vapor (calderas)	SISCHYSTRAB			
9	Proceso para cancelación de	SISCHYSTRAB			

	licencia de equipos Generadores de vapor (calderas)				
10	Proceso para instalación y reubicación de sala de caldera y equipo generador	SISCHYSTRAB			
11	Proceso para la verificación de inspección técnica (inspección interior, externa y prueba hidrostática) a los equipos generadores de vapor (caldera)	SISCHYSTRAB			
12	Proceso para autorización de libros de caldera (control diario, La bitácora y trabajos de reparación y mantenimiento)	SISCHYSTRAB			
13	Proceso para extender licencia a establecimientos para Comercializar equipo	SISCHYSTRAB			

	de protección personal				
14	Trámite de licencia en materia higiene y seguridad para la empresa ya instalada	SISCHYSTRAB			
15	Trámite de Licencia en Materia de Higiene y Seguridad para la empresa por iniciar operaciones	SISCHYSTRAB	15	7	46.67%
16	Cuando la persona trabajadora no es asegurada	SISCHYSTRAB	21	10	47.62%
17	Cuando la denuncia verse sobre la obligación de mantener la relación laboral posterior a riesgo laboral (restablecer o reubicar puesto de trabajo)	SISCHYSTRAB			
18	Cuando la denuncia sea por riesgo laboral de personas trabajadoras aseguradas que el empleador no le	SISCHYSTRAB			

	haya llenado el formato de notificación de accidentes de trabajo (NAT) del INSS				
19	Denuncia de la comisión mixta de higiene y seguridad en el trabajo	SISCHYSTRAB			
20	Procedimiento para el registro de comisión mixta de higiene y seguridad del trabajo	SISCHYSTRAB	16	1	6.25%
21	Procedimiento para aprobar el plan de trabajo anual de la comisión mixta de	SISCHYSTRAB			
22	Procedimiento para aprobar el reglamento interno de funcionamiento de la	SISCHYSTRAB			
23	Proceso para abrir el libro de acta de la comisión mixta de HST	SISCHYSTRAB			
24	Procedimiento para aprobar los reglamentos técnicos	SISCHYSTRAB			

	organizativos en materia de HST				
25	Proceso para brindar asesoría en materia de higiene y seguridad del trabajo	SISCHYSTRAB	11	1	9%
26	Procesos relativos al otorgamiento de carné a Operadores de equipos generadores de vapor (caldera)	SISCHYSTRAB			
27	Proceso de Notificación de los riesgos profesionales	SISCHYSTRAB			
Porcentaje de automatización del área					23.67%

La siguiente área es muy parecida a la de inspección laboral con la diferencia que en esta se hace el resguardo a la salud e integridad de los trabajadores en el ambiente laboral, esto cubre las condiciones de la empresa verificando todas y cada una de las posibles amenazas a los trabajadores, así como hacer cumplir los certificados de seguridad y estar al día con las licencias de los equipos de trabajo. También hace las investigaciones de los accidentes laborales ocurridos en el lugar y el levantamiento de las actas de infracción si acaso existe alguna.

Área	Procesos
Derechos Colectivos y Asesoría Laboral	24

No	Proceso	Sistema que lo automatiza	Número de tareas por proceso	Número de tareas automatizadas	Porcentaje de automatización
1	Negociación de Convenios Colectivos a través del MITRAB.	File master	27	8	29.62%
2	Negociación de Convenios Colectivos por arreglo directo.	File master	17	5	29.41%
3	Prórroga de un convenio colectivo.	File master	16	8	50%
4	Adendum en un convenio colectivo.	File master	16	9	56.25
5	Adhesión a Convenios Colectivos.	File master	18	11	61.1%
6	Interpretación de Cláusulas de Convenio Colectivo.	File master	18	11	61.1%
7	Solicitud de cancelación de contrato de trabajo de un trabajador firmante de pliego petitorio	File master	17	5	29.41%
8	Conciliación individual.	File master	34	7	20.58%
9	Inscripción de nuevas organizaciones sindicales (Otorgar personalidad jurídica)	File master	48	8	16.67%
10	Actualización, reestructuración de las	File master	32	5	15.625%

	Organizaciones Sindicales.				
11	Prórroga de periodo de duración de la junta Directiva Sindical.	File master	32	4	12.5%
12	Reforma total o parcial del Acta constitutiva y estatutos de las Organizaciones Sindicales	File master	32	4	12.5%
13	Apertura de libros por segunda vez ya sea por perdida y solicitudes de cierres de libros	File master	21	2	9.52%
14	Certificación de Acta Constitutiva y /o Estatutos.	File master	21	2	9.52%
15	Disolución de las Organizaciones Sindicales.	File master	6	3	50%
16	Copia simple de Acta Constitutiva, Estatutos, listado de asistencia a las asambleas, actas de la asamblea	File master	7	1	14.28%
17	Acreditación de Procuradores Laborales y de Seguridad Social.	File master	22	2	9.1%
18	Asesoría Sindical.		8	0	0%
19	Consulta laboral jurídica presencial.		27	0	0%

20	Consulta laboral jurídica por escrito.		27	0	0%
21	Consulta laboral jurídica por correo electrónica.		27	0	0%
22	Consulta laboral jurídica por el Centro de Llamadas.		27	0	0%
23	Juicios orales en la vía judicial.		27	0	0%
24	Juicios escritos en la vía judicial.		27	0	0%
Porcentaje de automatización del área					20.3%

El objetivo de esta área es la de hacer que Empleadores y Trabajadores resuelvan sus conflictos de una manera alterna como lo son las conciliaciones, de esta manera todos los conflictos presentes se resolverán en la misma empresa por medio de charlas junto a los acuerdos conciliados por cada una de las partes para hacer cumplir sus derechos. Todo lo anterior es debido a que el área de Derechos Colectivos y Asesoría laboral se encarga de realizar las asesorías laborales para hacerlo posible.

Área	Procesos
Empleo y Salarios	28

No	Proceso	Sistema que lo automatiza	Número de tareas por proceso	Número de tareas automatizadas	Porcentaje de automatización
1	Elaboración de Matriz de Generación de Empleo.	SAES	7	1	14.29%

2	Sistematización de los datos generados por INSS, Zona Franca y Gobierno Central.	SAES	6	2	33.3%
3	Elaboración del informe de los resultados obtenidos por la encuesta mensual de establecimientos del sector formal	SAES	8	7	87.5%
4	Atención a Usuarios demandantes de empleos	SISEPEM	9	3	33.3%
5	Atención a usuarios oferentes de plazas vacantes (empleadores)	SISEPEM	7	3	42.85%
6	Intermediación Laboral entre Demandantes de empleo y oferentes de plazas vacantes (empleadores)	SISEPEM	13	6	46.15%
7	Visitas a empresa para promocionar el SEPEM.		9	0	0
8	Realización de Encuentros Empresariales		13	0	0
9	Talleres de Orientación Laboral		13	0	0
10	Acreditación de Agencias de Empleos Privadas		9	0	0
11	Procedimiento para realizar la investigación estadística de salarios y el	SAES	14	3	21.42%

	empleo por sector de actividad económica.				
12	Procedimiento para la elaboración del informe de los resultados obtenidos por la encuesta mensual de salarios en los establecimientos del sector formal.	SAES	4	3	75%
13	Procesamiento y análisis de las remuneraciones generadas por INSS y Gobierno Central.	SAES	4	1	25%
14	Procedimiento para realizar estudios y/o análisis del mercado salarial, condiciones del mercado laboral e indicadores socioeconómicos nacional e internacional.	SAES	11	4	36.36%
15	Procedimiento para realizar intercambio de información en materia de salario, normativas agropecuarias y de alimentación.	SAAE	4	1	25%
16	Procedimiento para realizar monitoreo estadístico de rendimiento productivo y salario de los recolectores del café y recopilación primaria del comportamiento de	SAES	10	6	69%

	los precios de los productos básicos de la dieta del trabajador del campo.				
17	Procedimiento para la elaboración del informe de los resultados obtenidos del monitoreo del rendimiento productivo del café y del comportamiento de los precios de los productos de la dieta del trabajador del campo.	SAES	7	4	57.4%
18	Procedimiento para elaborar estudios del mercado salarial-productivo y condiciones socioeconómicas del sector café.	SAES	5	4	80%
19	Procedimiento para diseñar una metodología para la medición de la productividad laboral en la micro y pequeña industria artesanal.	SAES	6	4	66.67
20	Procedimiento de preparación e impartición de la capacitación de Productividad a miembros de microempresas productivas.	SAES	8	2	25%

21	Selección de los protagonistas de los programas de autoempleo.	SAAE	14	4	28.5%
22	Proceso de inducción a los protagonistas de los programas de autoempleo.		8	0	0%
23	Proceso de compra y transferencia de equipos básicos a los protagonistas de los programas de autoempleo.	SAAE	13	3	23%
24	Proceso narrativo del seguimiento a los protagonistas de los proyectos de auto empleo	SAAE	8	2	25%
25	Proceso de constitución y seguimiento a las mesas municipales de empleo a nivel nacional.		8	0	0%
26	Proceso de alianzas suscritas a nivel interinstitucional, con instancias de gobiernos locales, empresa privada, ong's, entre otros.	SAAE	9	1	11.1%
27	1-Proceso sobre gestión migratoria para la contratación de fuerza de trabajo migrante	SAES	13	1	7.69%

	nicaragüense para laborar en Costa Rica bajo el convenio binacional				
28	Asesorar, orientar e informar a usuarios en general en materia de migraciones laborales	SAES	5	1	20%
Porcentaje de automatización del área					30.48%

El área de Empleo y Salario tiene muchos objetivos los cuáles se resumen como el área encargada de la recopilación de información en materia de empleo, salario y migración; es decir que orienta y dirige las investigaciones respecto a lo antes mencionado. Entre los datos recopilados por el área son las vacantes disponibles, los trabajos colocados, los salarios mínimos y el porcentaje de aumento por la actividad económica...etc. Así como dar asistencia técnica con respecto a empleo y salario.

9.2. ANEXO 2: Evaluación de documentación de Sistema

	FILEMASTER	SISCOLAB	SAAE	SISCHYS	SISTEMA DE HIGIENE Y SEGURIDAD	SAES	SISEPEM	SISLAB
Documentación								
Manual de usuario								
Manual de Captación								
Manual de Implementación								
Manual Administrativo								
Metodología de desarrollo								
Requerimientos Funcionales y No Funcionales								
Casos de Uso								
Script de Base de datos								
Diagnóstico de la situación								
Diccionario de datos								
Normas de programación								
Porcentaje de Cumplimiento								

9.3. ANEXO 3: Matriz de evaluación de los sistemas

					SISTEMA A EVALUAR			
CARACTERÍSTICAS	SUB CATERÍSTICAS	PONDERACION	%	DESCRIPCION	PUNTAJE	%	ENCUESTA	TOTAL
Funcionalidad	Adecuación	5	25%	¿Puede el software realizar las tareas requeridas?				
	Exactitud	5	25%	¿Da los resultados como se esperaba?				
	Interoperabilidad	5	25%	¿Puede el sistema interactuar con otros sistemas?				
	Seguridad	5	25%	¿El software prevé acceso no autorizados?				
Total		20	17%					
Fiabilidad	Madurez	10	50%	¿Muchas de las fallas en el software han sido eliminadas en el tiempo?				
	Tolerancia a Fallos	5	25%	¿Es el software capaz de manejar errores?				
	recuperabilidad	5	25%	¿Puede el software resumir el trabajo y restaurar los datos perdido después de una falla?				
Total		20	17%					
Usabilidad	Compre nsibilidad	5	25%	¿Los usuarios comprenden como usar el sistema con facilidad?				
	Facilidad de Aprendizaje	5	25%	¿Puede el usuario aprender a utilizar el sistema con facilidad?				
	Operatividad	5	25%	¿Puede el usuario usar el sistema sin mucho esfuerzo?				
	Atractivo	5	25%	¿La interfaz luce bien?				

Total		20	17%					
Eficiencia	Tiempo de comportamiento	10	50%	¿Qué tan rápido el sistema responde?				
	Utilización de recursos	10	50%	¿El sistema utiliza los recursos eficientemente?				
Total		20	17%					
Mantenibilidad	analizabilidad	5	25%	¿Las fallas pueden ser fácilmente diagnosticadas?				
	Variabilidad	5	25%	¿Puede el software ser fácilmente modificado?				
	Estabilidad	5	25%	¿Puede el software continuar funcionando si los cambios se hacen?				
	comprobabilidad	5	25%	¿Puede el software ser testeado fácilmente?				
Total		20	17%					
Portabilidad	Adaptabilidad	5	25%	¿Puede el software ser movido a otro ambiente?				
	Instabilidad	5	25%	¿Puede el software ser instalado fácilmente?				
	Conformidad	5	25%	¿El software cumple con estándares de portabilidad?				
	Reemplazabilidad	5	25%	¿Puede el software fácilmente reemplazar a otro software?				
Total		20	17%					
Total		120	100%					
Promedio								

9.4. ANEXO 4: Resultados de la Evaluación de los sistemas

1. Funcionalidad

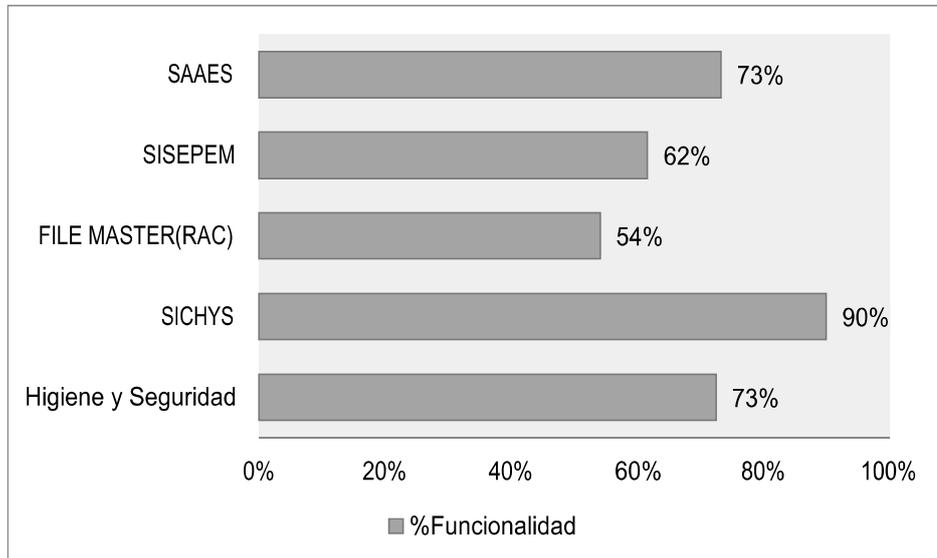


Grafico No 1: Porcentaje de cumplimiento en Funcionalidad.

2. Usabilidad

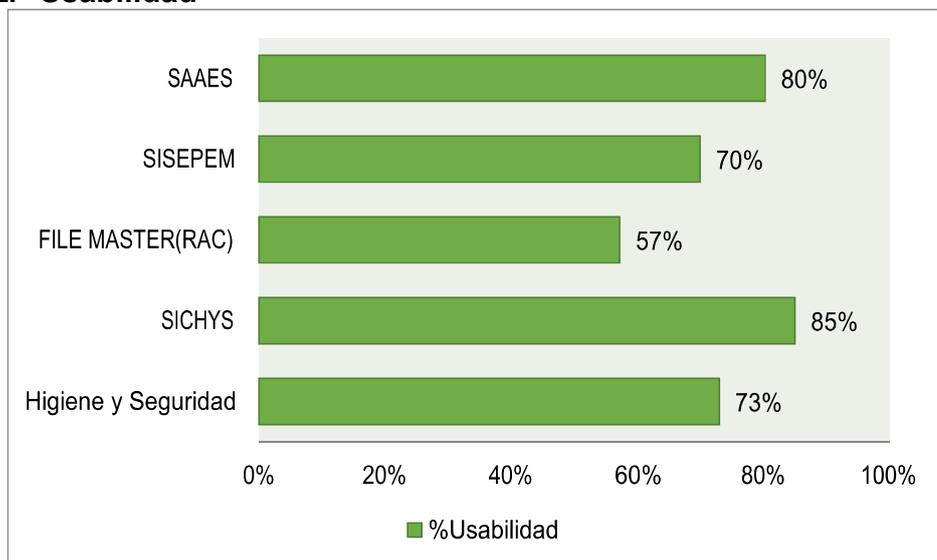


Grafico No 2: Porcentaje de cumplimiento en Usabilidad.

3. Fiabilidad

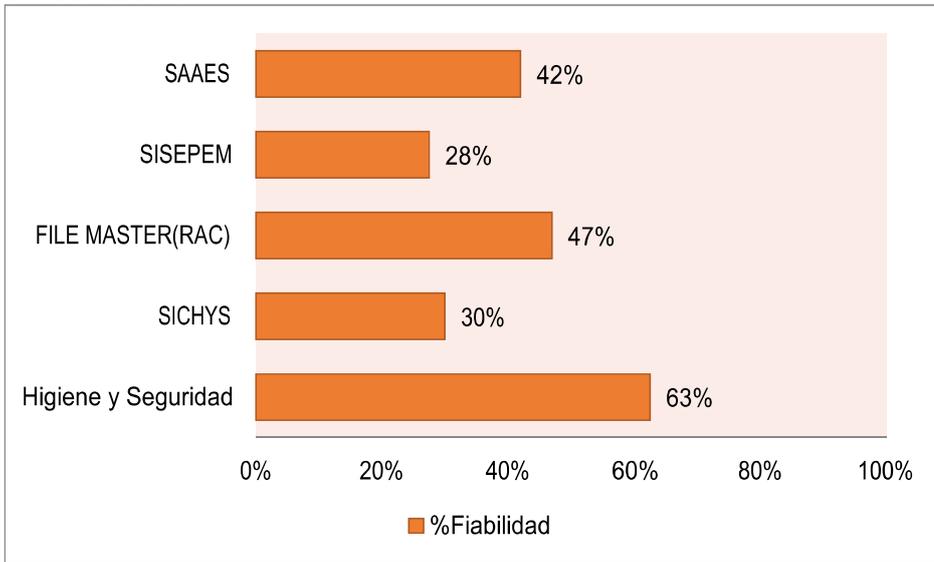


Grafico No 3: Porcentaje de cumplimiento en Fiabilidad.

4. Eficiencia

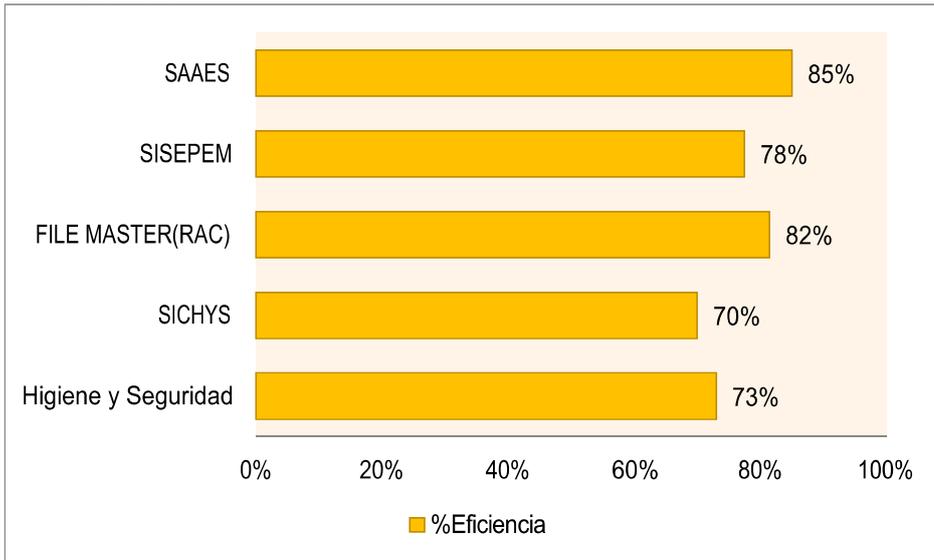


Grafico No 4: Porcentaje de cumplimiento en Eficiencia.

5. Mantenibilidad

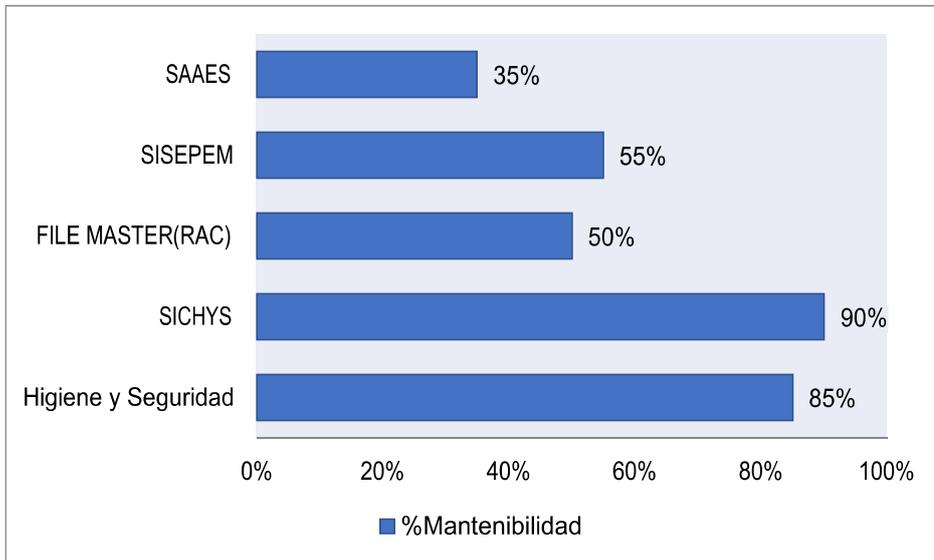


Grafico No 5: Porcentaje de cumplimiento en Mantenibilidad.

6. Portabilidad

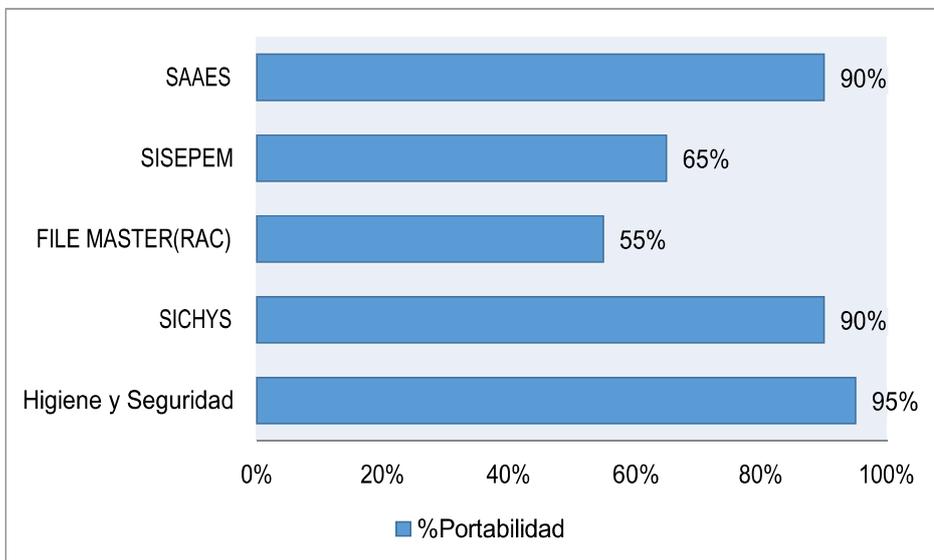


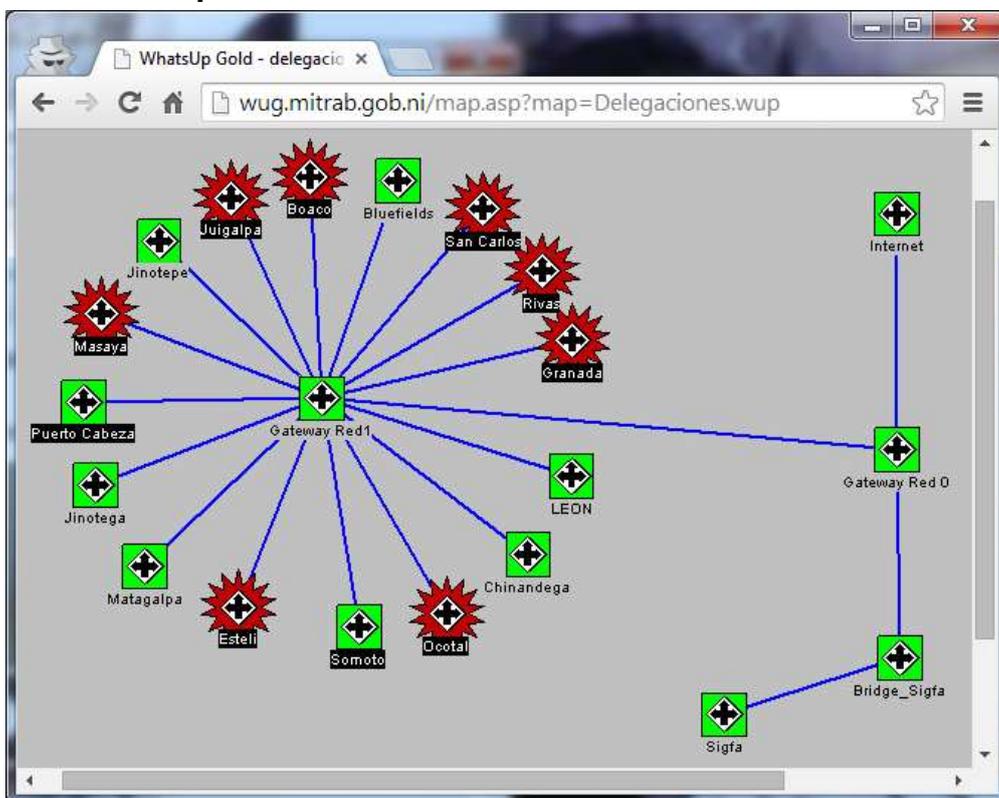
Grafico No 6: Porcentaje de cumplimiento en Portabilidad.

9.5. ANEXO 5: Análisis de arquitectura

9.5.1. Arquitectura del almacenamiento

No existe ninguna evidencia de una estrategia para el almacenamiento masivo de datos, Sobre la efectiva protección de los datos críticos, implementación de almacenamiento masivo tipo “storage”, preferentemente en un esquema de “Storage Area Network” (SAN) que permita integrar en una red separada discos y unidades de respaldo.

9.5.2. Arquitectura de comunicaciones



WAN MITRAB

9.5.3. Arquitectura de servidores

1. DELL POWEREDGE 1950	
Tipo	SERVIDOR FIREWALL
Configuración física	1U rack
Procesadores (Tipo y Velocidad)	Hasta dos procesadores de secuencia de doble núcleo Intel® Xeon® 5000 con 3.0 GHz de frecuencia de reloj; o hasta dos procesadores de secuencia de doble núcleo Intel Xeon 5100 con 3.0 GHz de frecuencia de reloj; o hasta dos procesadores de secuencia de cuatro núcleos Intel Xeon 5300 con 2.66 GHz de frecuencia de reloj
Capacidad de Memoria	Módulos DIMM de 4 GB con memoria intermedia completa (FBD) en pares coincidentes; 533 MHz o 667 MHz; 8 zócalos para admitir hasta 32 GB
Capacidad de almacenamiento	SAS de 3,5" (a 10.000 rpm): unidades de disco duro de 300 GB conectables en marcha;
Tipo de Conexión Pendiente	Independiente (standalone) - Cluster (Si el servidor está conectado con otros servidores que pueden ser vistos como un sistema único.)
Sistema Operativo/Versión	GNU/Linux CentOS 5.0

Uso	Como Firewall
Servicios	IPTABLES, versión 1.3.5-4/ Software libre
Estado del Uso	Instalado y en producción
Propiedad	Propio del MITRAB
Requisitos de hardware satisfechos	NO
Requisitos de software satisfechos	NO
Configuración necesaria	El firewall debería ser un gadget específicamente para la función de firewall y no un servidor con un sistema operativo el cual puede estar vulnerable.
Detalles	No hay un sistema de detección de intrusos.

2. DELL POWEREDGE 1950

Tipo	SERVIDOR secundario)	MAIL	(DNS
Configuración física	1U rack		
Procesadores (Tipo y Velocidad)	<p>Hasta dos procesadores de secuencia de doble núcleo Intel® Xeon® 5000 con 3.0 GHz de frecuencia</p> <p>de reloj; o hasta dos procesadores de secuencia de doble núcleo Intel Xeon 5100 con 3.0 GHz de</p> <p>frecuencia de reloj; o hasta dos procesadores de secuencia de cuatro núcleos Intel Xeon 5300</p> <p>con 2.66 GHz de frecuencia de reloj</p>		

Capacidad de Memoria	Módulos DIMM de 4 GB con memoria intermedia completa (FBD) en pares coincidentes; 533 MHz o 667 MHz; 8 zócalos para admitir hasta 32 GB
Capacidad de almacenamiento	SAS de 3,5" (a 10.000 rpm): unidades de disco duro de 300 GB conectables en marcha;
Sistema Operativo/Versión	GNU/Linux CentOS 5.0
Uso	Servicio de correo de protocolo SMTP
Servicios ejecutándose	Bind 9.3.6, Zimbra Mail 7
Estado del Uso	Instalado y en producción
Propiedad	Propio del MITRAB
Requisitos de hardware satisfechos	SI
Requisitos de software satisfechos	NO
Configuración necesaria	Actualizar el servicio de BIND
Detalles	Bind es un servicio altamente importante y su seguridad y estabilidad siempre debe estar en lo más. El servicio de correo se ejecuta sin fallos

3. DELL POWEREDGE 1950

Tipo	SERVIDOR DNS PRIMARIO
Configuración física	1U rack
Procesadores (Tipo y Velocidad)	Hasta dos procesadores de secuencia de doble núcleo Intel® Xeon® 5000 con 3.0 GHz de frecuencia

	<p>de reloj; o hasta dos procesadores de secuencia de doble núcleo Intel Xeon 5100 con 3.0 GHz de frecuencia de reloj; o hasta dos procesadores de secuencia de cuatro núcleos Intel Xeon 5300 con 2.66 GHz de frecuencia de reloj</p>
Capacidad de Memoria	<p>Módulos DIMM de 4 GB con memoria intermedia completa (FBD) en pares coincidentes; 533 MHz o 667 MHz; 8 zócalos para admitir hasta 32 GB</p>
Capacidad de almacenamiento	<p>SAS de 3,5" (a 10.000 rpm): unidades de disco duro de 300 GB conectables en marcha;</p>
Sistema Operativo/Versión	<p>GNU/Linux CentOS 5.0</p>
Uso	<p>Sistema de Nombre de Dominio (DNS) y Servidor web con proxy-cache</p>
Servicios ejecutándose	<p>Squid 2.6, Apache 2</p>
Estado del Uso	<p>Instalado y en producción</p>
Propiedad	<p>Propio del MITRAB</p>
Requisitos de hardware satisfechos	<p>SI</p>
Requisitos de software satisfechos	<p>NO</p>
Configuración necesaria	<p>Actualizar el servicio de proxy-cache de Squid a su última versión estable</p>
Detalles	<p>Servidor para la página web de la organización.</p>

4. DELL POWEREDGE 2950

Tipo	SERVIDOR DE BASES DE DATOS
Configuración física	Altura en rack de 2U

Procesadores (Tipo y Velocidad)	<p>Hasta dos procesadores de secuencia de doble núcleo Intel® Xeon® 5000 con 3.0 GHz de frecuencia de reloj</p> <p>o hasta dos procesadores de secuencia de doble núcleo Intel Xeon 5100 con 3.0 GHz de frecuencia de reloj</p> <p>o hasta dos procesadores de secuencia de cuatro núcleos Intel Xeon 5300 con 2.66 GHz de frecuencia de reloj</p>
Capacidad de Memoria	Módulos DIMM de 4 GB con memoria intermedia completa (FBD) en pares coincidentes; 533 MHz o 667 MHz; 8 zócalos para admitir hasta 32 GB
Capacidad de almacenamiento	500 GB
Sistema Operativo/Versión	GNU/Linux CentOS 6.0
Uso	Gestión de las bases de datos: MySQL 5.1.61 Community edition, PostgreSQL 8.4.13/ Software libre
Estado del Uso	Instalado y en producción
Propiedad	Propio del MITRAB
Requisitos de hardware satisfechos	SI
Requisitos de software satisfechos	NO
Configuración necesaria	Actualizar las versiones de PostgreSQL y MySQL instalados
Detalles	Se podrían actualizar los gestores de bases de datos para tener los últimos parches de seguridad y fix de bugs al día.

5. HP ProLiant DL380G6

Tipo	SERVIDOR DE VIRTUALIZACIÓN
Configuración física	Rack U2
Procesadores (Tipo y Velocidad)	X5670 2.93GHz 6C 95w
Capacidad de Memoria	32 GB DIMM DDR3-1333
Capacidad de almacenamiento	700 GB
Tipo de Conexión Pendiente	Independiente (standalone) - Cluster (Si el servidor está conectado con otros servidores que pueden ser vistos como un sistema único.)
Sistema Operativo/Versión	GNU/Linux CentOS 5.0
Uso	Virtualización del Windows server 2003(en el que se montó el RAC) y montaje de los demás Sistemas de Información.
Servicios ejecutándose	VMWare workstation 7.0, SQL Server 2005, RAC, SAAE, SAE, SYSCOLAB, etc.
Estado del Uso	Instalado y en producción
Propiedad	Propio del MITRAB
Requisitos de hardware satisfechos	NO
Requisitos de software satisfechos	NO
Configuración necesaria	Se necesita cambiar actualizar los softwares instalados debido a su obsolescencia y vulnerabilidad
Detalles	Al tener montado todos los sistemas de información provoca una carga general en el sistema causando ralentización en las consultas de las aplicaciones.

6. Kinetos Teleware	
Tipo	Servidor de Telefonía
Configuración física	
Procesadores (Tipo y Velocidad)	Intel 2.3 GHz
Capacidad de Memoria	1 GB
Capacidad de almacenamiento	150 GB
Sistema Operativo/Versión	GNU/Linux CentOS 5.0
Uso	Comunicaciones dentro de la organización con la aplicación de Kinetos
Estado del Uso	Instalado y en producción
Propiedad	Propio del MITRAB
Requisitos de hardware satisfechos	SI
Requisitos de software satisfechos	SI
Configuración necesaria	No se necesita configuración

9.5.4. Aplicaciones

BIND	
Tipo de Software	Software de aplicación
Versión / Tipo	Versión 9.3.6 / Software libre
Empresa fabricante	Internet System Consortium
Número de licencias	ilimitadas
Propietario de la licencia	Propio
Fecha de vencimiento	No tiene

Tipo de instalación	Instalado en el servidor DELL POWEREDGE 1950 (2)
Modelo de instalación	La instalación es un Daemon o servicio ejecutándose en el servidor
Número de Usuarios	No es una aplicación a la que accedan los usuarios, su interacción con los usuarios es nula.
Soporte	El soporte es brindado por la comunidad de software libre por medio de documentos en la red, pero para un soporte más especializado y profesional se necesita comprar la suscripción a la Empresa propietaria del software.
Uso	Se utiliza para asignar el nombre de Dominio de la organización
Responsable	Dirección de Informática
Información adicional	Necesita ser actualizada muy frecuente debido que las actualizaciones de seguridad de esta aplicación son críticas, la última de versión de esta aplicación es la 9.10 lo cual indica que debe ser actualizada urgentemente.

APACHE	
Tipo de Software	Software de aplicación
Versión / Tipo	Versión 2 / Software libre
Empresa fabricante	Apache Software Foundation
Número de licencias	ilimitadas
Propietario de la licencia	Propio
Fecha de vencimiento	No tiene
Tipo de instalación	Instalado en el servidor DELL POWEREDGE 1950(3)

Modelo de instalación	La instalación es un Daemon o servicio ejecutándose en el servidor
Número de Usuarios	No es una aplicación a la que accedan los usuarios, su interacción con los usuarios es nula.
Soporte	El soporte es brindado por la comunidad de software libre por medio de documentos en la red.
Uso	Se utiliza para crear el servidor web de la Organización y sobre el montar todas las páginas y/o aplicaciones web.
Responsable	Dirección de Informática
Información adicional	Necesita una configuración extensa para que sea segura y no provea ninguna vulnerabilidad. Está en su versión más estable.

SQUID	
Tipo de Software	Software de Aplicación
Versión / Tipo	Versión 2.6 / Software libre
Empresa fabricante	The Squid Software Foundation
Número de licencias	ilimitadas
Propietario de la licencia	Propio
Fecha de vencimiento	No tiene
Tipo de instalación	Instalado en el servidor DELL POWEREDGE 1950(3)
Modelo de instalación	La instalación es un Daemon o servicio ejecutándose en el servidor
Número de Usuarios	No es una aplicación a la que accedan los usuarios, su interacción con los usuarios es nula.

Soporte	El soporte es brindado por la comunidad de software libre por medio de documentos en la red.
Uso	Sus utilidades está la de mejorar el rendimiento de las conexiones de la Organización a Internet guardando en caché peticiones recurrentes a servidores web y DNS, acelerar el acceso a un servidor web determinado o añadir seguridad realizando filtrados de tráfico.
Responsable	Dirección de Informática
Información adicional	No está en su versión más estable dado que la última versión de la aplicación es la 3.4 y 3.3 para ciertas distribuciones GNU/Linux

IPTABLES	
Tipo de Software	Software de aplicación
Versión / Tipo	Versión 1.3.5-4/ Software libre
Empresa fabricante	Netfilter
Número de licencias	ilimitadas
Propietario de la licencia	Propio
Fecha de vencimiento	No tiene
Tipo de instalación	Instalado en el servidor DELL POWEREDGE 1950 (1)
Modelo de instalación	La instalación es un Daemon o servicio ejecutándose en el servidor
Número de Usuarios	No es una aplicación a la que accedan los usuarios, su interacción con los usuarios es nula.
Soporte	El soporte es brindado por la comunidad de software libre por medio de documentos en la red, también tiene

	soporte de pago, pero la Organización no la utiliza.
Uso	Controla y filtra los accesos de red en la organización, así como proveer un firewall a la empresa.
Responsable	Dirección de Informática
Información adicional	No está en su versión más estable dado que la última versión de la aplicación es la 1.4.21

MYSQL	
Tipo de Software	Software de aplicación
Versión / Tipo	Version 5.1.61 / Software libre (Community edition)
Empresa fabricante	Oracle Corporation
Número de licencias	ilimitadas
Propietario de la licencia	Propio
Fecha de vencimiento	No tiene
Tipo de instalación	Instalado en el servidor DELL POWEREDGE 2950
Modelo de instalación	La instalación es un Daemon o servicio ejecutándose en el servidor
Número de Usuarios	No es una aplicación a la que accedan los usuarios, su interacción con los usuarios es nula, a excepción de los Sys Admins
Soporte	El soporte es brindado por la comunidad de software libre por medio de documentos en la red, también tiene soporte de pago, pero la Organización no la utiliza.

Uso	Creación y almacenamiento de ciertas bases de datos para la organización.
Responsable	Dirección de Informática
Información adicional	No está en su versión más estable dado que la última versión de la aplicación es la 5.6.21 por lo cual necesita ser actualizada.

POSTGRESQL	
Tipo de Software	Software de aplicación
Versión / Tipo	Versión 8.4.13/ Software libre (Community edition)
Empresa fabricante	PostgreSQL Core Team
Número de licencias	ilimitadas
Propietario de la licencia	Propio
Fecha de vencimiento	No tiene
Tipo de instalación	Instalado en el servidor DELL POWEREDGE 2950
Modelo de instalación	La instalación es un Daemon o servicio ejecutándose en el servidor
Número de Usuarios	No es una aplicación a la que accedan los usuarios, su interacción con los usuarios es nula, a excepción de los Sys Admins
Soporte	El soporte es brindado por la comunidad de software libre por medio de documentos en la red, también tiene soporte de pago, pero la Organización no la utiliza.
Uso	Creación y almacenamiento de ciertas bases de datos para la organización.
Responsable	Dirección de Informática

Información adicional	No está en su versión más estable dado que la última versión de la aplicación es la 9.2 por lo cual necesita ser actualizada.
-----------------------	-------------------------------------------------------------------------------------------------------------------------------

SQL Server	
Tipo de Software	Software de Aplicación
Versión / Tipo	Versión 2005/ de pago(pirata)
Empresa fabricante	Microsoft Corporation
Número de licencias	Varias por lo que han sido pirateadas
Propietario de la licencia	Propio
Fecha de vencimiento	No tiene
Tipo de instalación	Instalado en el servidor HP ProLiant DL380G6 (virtualizado)
Modelo de instalación	Cliente-Servidor
Número de Usuarios	No es una aplicación a la que accedan los usuarios, su interacción con los usuarios es nula, a excepción de los Sys Admins
Soporte	Sin soporte
Uso	Para montar las bases de datos de los SI que utilizan tecnología Microsoft.
Responsable	Dirección de Informática
Información adicional	El sistema está muy obsoleto la última versión Windows server 2014. El soporte de actualizaciones ha finalizado en el año 2012 por lo que la seguridad de este sistema es crítica.

9.5.5. Sistemas Operativos

WINDOWS SERVER	
Tipo de Software	Sistema operativo
Versión / Tipo	Versión 2003/ de pago(pirata)
Empresa fabricante	Microsoft Corporation
Número de licencias	Varias por lo que han sido pirateadas
Propietario de la licencia	Propio
Fecha de vencimiento	No tiene
Tipo de instalación	Instalado en el servidor HP Proliant DL380G6 (virtualizado)
Modelo de instalación	Cliente-Servidor
Número de Usuarios	No es una aplicación a la que accedan los usuarios, su interacción con los usuarios es nula, a excepción de los Sys Admins
Soporte	Sin soporte
Uso	Para el hosting del sistema RAC y montar SQL Server 2005.
Responsable	Dirección de Informática
Información adicional	El sistema está muy obsoleto la última versión Windows server 2012. El soporte de actualizaciones ha finalizado en el año 2010 por lo que la seguridad de este sistema es crítica.

CentOS	
Tipo de Software	Sistema operativo
Versión / Tipo	Versión 5/ software libre
Empresa fabricante	Development Team Core

Número de licencias	libre
Propietario de la licencia	Propio
Fecha de vencimiento	No tiene
Tipo de instalación	Instalado en todos los servidores DELL POWEREDGE 1950
Modelo de instalación	Cliente-Servidor
Número de Usuarios	No es una aplicación a la que accedan los usuarios, su interacción con los usuarios es nula, a excepción de los Sys Admins
Soporte	Sin soporte
Uso	Para el uso y montaje de todas las tecnologías de software libre como bind, zimbra, kinetos, squid, apache 2.
Responsable	Dirección de Informática
Información adicional	Aunque ya existe la versión 7 la versión 5 tiene soporte hasta 2017 y solo se necesita actualizar sus paquetes en cuanto estén disponibles.

CENTOS	
Tipo de Software	Sistema operativo
Versión / Tipo	Versión 6/ software libre
Empresa fabricante	Development Team Core
Número de licencias	libre
Propietario de la licencia	Propio
Fecha de vencimiento	No tiene
Tipo de instalación	Instalado en el servidor POWEREDGE 2950
Modelo de instalación	Cliente-Servidor
Número de Usuarios	No es una aplicación a la que accedan los usuarios, su interacción con los

	usuarios es nula, a excepción de los Sys Admins
Soporte	Sin soporte
Uso	Para el uso y montaje de todas las tecnologías de software libre como PostgreSQL, MySQL.
Responsable	Dirección de Informática
Información adicional	Aunque ya existe la versión 7 la versión 6 tiene soporte hasta 2020 y solo se necesita actualizar sus paquetes en cuanto estén disponibles.

9.5.6. Inventarios de los equipos informáticos

Esta información la tenemos en dependencia de la fecha de fabricación, la siguiente tabla muestra el nivel de vejez de nuestros equipos:

OFICINAS	PC	Mayor a 5 años	Menor o igual a 4 Años	Total
Despacho Ministra	6	3	3	6
Despacho Viceministro	3	2	1	3
Secretaria General	2	2	0	2
OAIP	1	1	0	1
Igualdad de Genero	2	2	0	2
Biblioteca	3	2	1	3
Administración Financiera	2	2	0	2
Presupuesto	2	2	0	2
Caja	2	2	0	2
Contabilidad	6	4	2	6
Suministro	2	2	0	2
Adquisiciones	3	2	1	3

Auditoría Interna	2	2	0	2
Asuntos Internacionales	4	3	1	4
Fortalecimiento	3	2	1	3
Recursos Humanos	6	5	1	6
Dirección General de Empleo y Salario	2	2	0	2
Observatorio Laboral	7	5	2	7
Dirección de Empleo	2	1	1	2
Trabajo Infantil	2	2	0	2
Migraciones Laborales	4	3	1	4
Productividad y Salario	6	6	0	6
Control de Bienes	2	2	0	2
Transporte	1	1	0	1
Servicios Generales	3	3	0	3
Higiene y Seguridad	18	12	6	18
Oficina de Proyectos	2	2	0	2
Planificación	5	5	0	5
Asociaciones Sindicales	3	3	0	3
Sepem	9	6	3	9
Inspectoría General del Trabajo	5	5	0	5
PALO	5	5	0	5
Inspectoría Sector Construcción	5	4	1	5
Inspectoría Sector Agroindustria	5	5	0	5
Inspectoría Sector Servicio	8	6	2	8
Conciliación individual	6	6	0	6

Negociación Colectiva	6	3	3	6
Dirección Jurídica	5	3	2	5
Defensoría Laboral	7	5	2	7
CALL CENTER	3	3	0	3
Informática	5	1	4	5
Sindicato	1	1	0	1
Totales	176	138	38	

La siguiente tabla muestra la relación de dichas PC con la memoria RAM:

OFICINAS	PC	<=512	<1024	>=1024	Totales
Despacho Ministra	6			6	6
Despacho Viceministro	3	1		2	3
Secretaría General	2		1	1	2
OAIP	1		1		1
Igualdad de Género	2			2	2
Biblioteca	3		1	2	3
Administración Financiera	2			2	2
Presupuesto	2			2	2
Caja	2	1	1		2
Contabilidad	6	2	3	1	6
Suministro	2			2	2
Adquisiciones	3	1		2	3
Auditoría Interna	2			2	2
Asuntos Internacionales	4		1	3	4
Fortalecimiento	3	1		2	3
Recursos Humanos	6	2	1	3	6
Dirección General de Empleo y Salario	2		1	1	2
Observatorio Laboral	7	3	2	2	7
Dirección de Empleo	2		1	1	2
Trabajo Infantil	2			2	2
Migraciones Laborales	4	2	1	1	4
Productividad y Salario	6	2		4	6

OFICINAS	PC	<=512	<1024	>=1024	Totales
Control de Bienes	2			2	2
Transporte	1		1		1
Servicios Generales	3	1	1	1	3
Higiene y Seguridad	18	2	5	11	18
Oficina de Proyectos	2			2	2
Planificación	5			5	5
Asociaciones Sindicales	3	1		2	3
Sepem	9	2	3	4	9
Inspección General del Trabajo	5			5	5
PALO	5	1		4	5
Inspección Sector Construcción	5		1	4	5
Inspección Sector Agroindustria	5			5	5
Inspección Sector Servicio	8		1	7	8
Conciliación individual	6			6	6
Negociación Colectiva	6	3	3		6
Dirección Jurídica	5	2	2	1	5
Defensoría Laboral	7	3	3	1	7
CALL CENTER	3			3	3
Informática	5		2	3	5
Sindicato	1	1			1
Totales	176	31	36	109	

Los sistemas operativos que tenemos en Managua:

OFICINAS	PC	Windows XP	Windows 7	Total
Despacho Ministra	6	4	2	6
Despacho Viceministro	3	3	1	4
Secretaría General	2	2	0	2
OAIP	1	1	0	1
Igualdad de Género	2	1	1	2
Biblioteca	3	3	0	3
Administración Financiera	2	2	0	2
Presupuesto	2	2	0	2

OFICINAS	PC	Windows XP	Windows 7	Total
Caja	2	2	0	2
Contabilidad	6	6	0	6
Suministro	2	1	1	2
Adquisiciones	3	2	1	3
Auditoría Interna	2	2	0	2
Asuntos Internacionales	4	4	0	4
Fortalecimiento	3	2	1	3
Recursos Humanos	6	4	2	6
Dirección General de Empleo y Salario	2	1	1	2
Observatorio Laboral	7	6	1	7
Dirección de Empleo	2	2	0	2
Trabajo Infantil	2	1	1	2
Migraciones Laborales	4	3	1	4
Productividad y Salario	6	3	3	6
Control de Bienes	2	1	0	1
Transporte	1	1	0	1
Servicios Generales	3	2	1	3
Higiene y Seguridad	18	2	16	18
Oficina de Proyectos	2	2	0	2
Planificación	5	5	0	5
Asociaciones Sindicales	3	1	2	3
Sepem	9	3	6	9
Inspección General del Trabajo	5	0	5	5
PALO	5	1	4	5
Inspección Sector Construcción	5	1	4	5
Inspección Sector Agroindustria	5	0	5	5
Inspección Sector Servicio	8	6	2	8
Conciliación individual	6	3	3	6
Negociación Colectiva	6	1	5	6

OFICINAS	PC	Windows XP	Windows 7	Total
Dirección Jurídica	5	3	2	5
Defensoría Laboral	7	5	2	7
CALL CENTER	3	0	3	3
Informática	5	0	5	5
Sindicato	1	1	0	1
Totales	176	95	81	

En la siguiente tabla se muestra la caracterización de equipos en delegaciones por memorias RAM en megabytes:

DELEGACION	PC	<=512	Menos 1024	>=1024	Total
Somoto	6	2	3	1	6
Estelí	12	4	4	4	12
Jinotega	6	2	2	2	6
Matagalpa	8	2	4	2	8
Juigalpa	6	1	3	2	6
Boaco	6	1	3	2	6
Chinandega	10	3	5	2	10
León	11	4	3	4	11
Carazo	6	1	1	4	6
Rivas	5	1	4	0	5
Masaya	5	1	3	1	5
Granada	10	3	5	2	10
Rio San Juan	5	2	2	1	5
RAAN	5	2	3	0	5
RAAS	5	3	2	0	5
Nueva Segovia	6	1	5	0	6
Totales	112	33	52	27	112

Delegaciones en base a edad expresada en años de uso tomando en cuenta el tipo de procesador:

DELEGACION	PCs	Mayor a 5 años	Menor o igual a 4 Años	Total
Somoto	6	4	2	6
Estelí	12	10	2	12
Jinotega	6	2	4	6
Matagalpa	8	5	3	8
Juigalpa	6	4	2	6
Boaco	6	4	2	6
Chinandega	10	6	4	10
León	11	7	4	11
Carazo	6	5	1	6
Rivas	5	4	1	5
Masaya	5	3	2	5
Granada	10	6	4	10
Rio San Juan	5	5	0	5
RAAN	5	5	0	5
RAAS	5	5	0	5
Nueva Segovia	6	4	2	6
Total	112	79	33	112

Delegaciones por Sistema Operativo:

DELEGACION	PCs	WIN XP	WIN 7	Total
SOMOTO	6	6	0	6
Estelí	12	12	0	12
Jinotega	6	6	0	6
Matagalpa	8	6	2	8
Juigalpa	6	6	0	6
Boaco	6	6	0	6
Chinandega	10	9	1	10
León	11	8	3	11

Carazo	6	6	0	6
Rivas	5	5	0	5
Masaya	5	4	1	5
Granada	10	9	1	10
Rio San Juan	5	5	0	5
RAAN	5	5	0	5
RAAS	5	5	0	5
Nueva Segovia	6	6	0	6
Total	112	104	8	112

9.6. ANEXO 6: COCOMO

Para conocer el costo asociado al desarrollo del sistema se procedió a calcular su valor. Para ello es necesario conocer el tiempo de desarrollo y la cantidad de personal para realizarlo. Esto lo logramos a través de la implementación del **Modelo Constructivo de Costos II (o COCOMO II, por su acrónimo del inglés Constructive Cost Model versión II)** es un modelo matemático de base empírica utilizado para estimación de costos de software. Está orientado a la magnitud del producto final, midiendo el "tamaño" del proyecto, en función de la cantidad de líneas de código, principalmente.

COCOMO II basa su medición tomando en cuenta los siguientes elementos:

COMPONENTES FUNCIONALES BÁSICOS

- **INTERACCIÓN FUNCIÓN DE TRANSACCIÓN**
 - Entrada externa (**EI -> External input**)
(Pantallas donde el usuario ingresa datos)
 - Salida externa (**EO -> External output**)
(Informes, gráficos, Listados de datos)
 - Consulta externa (**EQ -> External query**)
(Recuperar y mostrar datos al usuario (Buscar))
- **ALMACENAMIENTO FUNCIÓN DE DATOS**
 - Archivo lógico interno (**ILF -> Internal Logical File**)
Archivo del punto de vista lógico, no como en un sistema operativo / Pueden ser tablas en la base de datos
 - Archivo de interfaz externo (**EIF -> External Interface File**)

Datos referenciados a otros sistemas / Datos mantenidos por otros sistemas, pero usados por el sistema actual

Podemos decir que la interacción y almacenamiento se resumen de la siguiente manera:

- Buscar (**EQ**)
- Actualizar (**EI**)
- Insertar (**EI**)
- Listar (**EO**)
- Eliminar (**EI**)
- Informes o reportes (**EO**)
- Tablas de BD (**ILF**)

También es importante mencionar que se definen funciones por su tipo y su complejidad. Valores estándar Según International Function Point Users Group (IFPUG).

Tipo / Complejidad	Baja	Media	Alta
(EI) Entrada externa	3 PF	4 PF	6 PF
(EO) Salida externa	4 PF	5 PF	7 PF
(EQ) Consulta externa	3 PF	4 PF	6 PF
(ILF) Archivo lógico interno	7 PF	10 PF	15 PF
(EIF) Archivo de interfaz externo	5 PF	7 PF	10 PF

9.6.1. PUNTOS DE FUNCIÓN.

PUNTOS DE FUNCIÓN SIN AJUSTAR

DESCRIPCIÓN	COMPLEJIDAD									TOTAL
	BAJA			MEDIA			ALTA			
ENTRADAS (EI)	0	X	3	0	X	4	20	X	6	120
SALIDAS (EO)	0	X	4	0	X	5	20	X	7	140
CONSULTAS (EQ)	0	X	3	0	X	4	20	X	6	120
ARCHIVOS (ILF)	0	X	7	0	X	10	20	X	15	300
INTERFACES DE PROGRAMA (EIF)	0	X	5	0	X	7	0	X	10	0
TOTAL, DE PUNTOS DE FUNCIÓN SIN AJUSTAR (FPB)										680

9.6.2. RESUMEN DE LAS CARACTERÍSTICAS GENERALES DEL SISTEMA

CARACTERÍSTICA	VALOR	RAZONAMIENTO
COMUNICACIÓN DE DATOS	1	La entrada de datos es mayormente remota.
FUNCIÓN DISTRIBUIDA	0	No hay requerimientos de funciones distribuidos para el sistema.
RENDIMIENTO	1	El análisis y diseño de las consideraciones de rendimiento son estándar.
CONFIGURACIÓN UTILIZADA MASIVAMENTE	3	La aplicación corre en un entorno operativo existente y fuertemente utilizado
TASAS DE TRANSACCIÓN	1	Las tasas son tales que las consideraciones de análisis de rendimiento son estándares.
ENTRADAS DE DATOS EN LÍNEA	5	La entrada de datos es totalmente en línea.
DISEÑO PARA LA EFICIENCIA DE USUARIO FINAL	3	No se especifican requerimientos especiales.

ACTUALIZACIÓN EN LÍNEA	4	La actualización de los ficheros internos debe ser en línea y debe haber protección contra la pérdida de datos.
COMPLEJIDAD DEL PROCESAMIENTO	2	Existe mucho procesamiento lógico y procesamiento de control sensitivo.
UTILIZABLE EN OTRAS APLICACIONES	0	Las exigencias de reusabilidad son nulas.
FACILIDAD DE INSTALACIÓN	2	Los requerimientos de conversión e instalación fueron descritos por el usuario y se proporcionaron guías de conversión e instalación.
FACILIDAD DE OPERACIÓN	2	Se requieren, proporcionan y prueban procesos específicos de arranque, backup y recuperación.
PUESTO MÚLTIPLES	2	Se incluyeron necesidades de varios puestos en el diseño.
FACILIDAD DE CAMBIO	0	No hay requerimientos especiales del usuario para minimizar o facilitar el cambio.
NIVEL DE INFLUENCIA		$\Sigma FI1 = 26$

9.6.3. FACTORES DE ESCALA

Factor	Nombre	Rango	Razonamiento	Valor
PREC	Desarrollos previos similares.	Nominal	Conlleva aspectos novedosos.	3.72
FLEX	Flexibilidad del desarrollo.	Muy bajo	La flexibilidad entre el sistema y los requerimientos para su desarrollo son rigurosos, pues se deben cumplir a cabalidad.	5.07
RESL	Manejo de riesgos y la arquitectura.	Nominal	Se toman en cuenta algunos riesgos y la arquitectura no es tan compleja.	4.24

TEAM	Cohesión de equipo.	Muy alto	Existe consistencia en el trabajo de equipo y apoyo del organismo	1.1
PMAT	Madurez del proceso.	Inicial	Indica que las áreas de proceso principales están en un estado incipiente dentro de la organización.	6.24
TOTAL \sumSFI =				20.37

9.6.4. FACTORES DE ESFUERZO COMPUESTO

Factor	Nombre	Rango	Razonamiento	Valor
INDICADORES DEL PRODUCTO				
RELY	Fiabilidad requerida del software.	Bajo	Ante un fallo del software sólo ocasionaría fallas técnicas mínimas.	0.88
DATA	Volumen de datos.	Nominal	La relación entre el tamaño de la base de datos y las líneas de código del sistema son medias.	1
CPLX	Complejidad del producto.	Bajo	Los anidamientos no son muy complejos y a pesar de llevar muchas operaciones lógicas son de nivel moderado.	0.88
RUSE	Reutilización requerida.	Muy bajo	No se requiere forzosamente construir componentes reusables.	1.00
DOCU	Documentación asociada a las necesidades del ciclo de vida.	Bajo	La documentación asociada se debe realizar a lo largo del proyecto.	0.95
INDICADORES DE LA PLATAFORMA				

TIME	Restricción del tiempo de ejecución.	Muy bajo	El tiempo necesario para ejecutar las operaciones del sistema es muy bajo con respecto al total disponible.	1
STOR	Restricción de almacenamiento	Muy bajo	Tanto el sistema como la base de datos ocupan un valor muy bajo de volumen de almacenamiento con respecto al total disponible.	1
PVOL	Volatilidad de la plataforma.	Muy bajo	La plataforma de operación (hardware y sistema operativo) puede cambiar en un período mayor o igual a 2 años.	1
INDICADORES DEL PERSONAL				
ACAP	Habilidad del analista.	Alto	Existe una capacidad alta por parte de los analistas para el análisis y diseño del sistema.	0.83
PCAP	Habilidad del programador.	Alto	Existe una capacidad alta por parte de los programadores para la programación del sistema.	0.87
PCON	Continuidad del personal.	Muy bajo	Indica que la rotación del personal durante el desarrollo del proyecto es prácticamente nula.	1.24
AEXP	Experiencia en las aplicaciones.	Alto	Existe una experiencia media de 2 años en el equipo del proyecto en el desarrollo de sistemas.	0.89
PEXP	Experiencia en la plataforma.	Muy alta	La experiencia media del equipo en la utilización de la plataforma del sistema operativo es de 6 años.	0.81

LTEX	Experiencia en la herramienta y en el lenguaje de desarrollo.	Alto	La experiencia media del equipo en este punto es de 3 años.	0.91
INDICADORES DEL PROYECTO				
TOOL	Uso de herramientas software.	Bajo	Se utilizan herramientas front end, back end y CASE.	1.12
SITE	Desarrollo multilugar.	Muy alto	El desarrollo del sistema se lleva a cabo en un mismo edificio o complejo.	0.84
SCED	Calendario de desarrollo requerido.	Nominal	El calendario de desarrollo requerido indica que siempre existe probabilidad de compresión o alargamiento del proyecto.	1
TOTAL, PEMI=				0.406557246

9.6.5. FACTOR DE AJUSTE

$$FA = [0.65 + 0.01 * \sum Fi1]$$

$$FA = [0.65 + 0.01 * 26]$$

$$\text{FACTOR DE AJUSTE (FA) = 0.91}$$

9.6.6. PUNTOS DE FUNCIÓN AJUSTADOS

$$PFA = FPB * FA$$

$$PFA = 680 * 0.91$$

$$\text{PUNTOS DE FUNCIÓN AJUSTADOS(PFA) = 618.8}$$

9.6.7. RELACIÓN ENTRE LAS LÍNEAS DE CÓDIGO Y LOS PUNTOS DE FUNCIÓN.

LANGUAGE	QSM SLOC/FP DATA			
	AVG	MEDIAN	LOW	HIGH
ABAP (SAP) *	28	18	16	60
ASP*	51	54	15	69
Assembler *	119	98	25	320
Brio +	14	14	13	16
C *	97	99	39	333
C++ *	50	53	25	80
C# *	54	59	29	70
COBOL *	61	55	23	297
Cognos Impromptu Scripts +	47	42	30	100
Cross System Products (CSP) +	20	18	10	38
Cool:Gen/IEF *	32	24	10	82
Datastage	71	65	31	157
Excel *	209	191	131	315
Focus *	43	45	45	45
FoxPro	36	35	34	38
HTML *	34	40	14	48
J2EE *	46	49	15	67
Java *	53	53	14	134
JavaScript *	47	53	31	63
JCL *	62	48	25	221
LINC II	29	30	22	38
Lotus Notes *	23	21	19	40

Natural *	40	34	34	53
.NET *	57	60	53	60
Oracle *	37	40	17	60
PACBASE *	35	32	22	60
Perl *	24	15	15	60
PL/1 *	64	80	16	80
PL/SQL *	37	35	13	60
Powerbuilder *	26	28	7	40
REXX *	77	80	50	80
Sabretalk *	70	66	45	109
SAS *	38	37	22	55
Siebel *	59	60	51	60
SLOGAN *	75	75	74	75
SQL *	21	21	13	37
VB.NET *	52	60	26	60
Visual Basic *	42	44	20	60

9.6.8. TOTAL, DE LÍNEA DE CÓDIGO FUENTE

LDC: Número promedio de líneas de código

LDC= 134

TLDC= LDC x PFA

TLDC = 134 * 618.8

TOTAL, DE LÍNEA DE CÓDIGO FUENTE (TLDC) = 82,919.2 LDC

Expresado en miles (MF)

$$TLDC = 82,919.2 / 1000$$

TOTAL, DE LÍNEA DE CÓDIGO FUENTE (TLDC) = 82.919 MF

9.6.9. AHORRO Y GASTO DE SOFTWARE DE ESCALA

$$B = 0.91 + (0.01 \times \Sigma SF_i)$$

$$B = 0.91 + (0.01 * 20.37)$$

AHORRO Y GASTO DE SOFTWARE DE ESCALA (B) = 1.1137

9.6.10. ESTIMACIÓN DEL ESFUERZO

$$E = A \times TLDC^B \times \pi EM_i$$

Donde:

- A: Constante de calibración = 2.94.
- TLDC: Total de línea de código fuente en miles.
- B: Ahorro y gasto de software de escala.
- πEM_i : Factor de esfuerzo compuesto.

$$E = 2.94 * 82.919^{1.1137} * 0.406557246 = 163.784$$

ESTIMACIÓN DEL ESFUERZO (E) = 164 (PERSONAS-MESES)

9.6.11. TIEMPO DE DESARROLLO

$$Tdes = 3.67 \times (E)^{0.28 + (0.002 * \Sigma SF_i)}$$

$$Tdes = 3.67 \times (163.784)^{0.28 + (0.002 \times 20.37)}$$

$$T_{des} = 18.830$$

TIEMPO DE DESARROLLO(TDES) = 19 MESES

9.6.12. CANTIDAD DE PERSONAL

$$CH = E / T_{des}$$

$$CH = 163.784 / 18.83 = 8.7 \text{ personas}$$

CANTIDAD DE PERSONAL(CH) = 9 PERSONAS

9.6.13. PRODUCTIVIDAD

$$P = (TLDC \times 1000) / T_{des}$$

$$P = \underline{82,919.2}$$

9

LÍNEAS DE CÓDIGO POR HOMBRE-MÁQUINA(P)= 9,213.24

9.6.14. CALCULO DE LOS COSTOS DEL PROYECTO

Para obtener el costo total del proyecto se debe tomar en consideración los costos directos e indirectos:

$$CTP = CD + CI$$

Donde:

- CTP: Costo total del proyecto
- CD: Costos Directos
- CI: Costos Indirectos
- CFT: Costos de la Fuerza de Trabajo
- CUMT: Costo de Utilización de Medios Técnicos
- CMAT: Costos de Materiales

CD = CFT + CUMT + CMAT

CI = 15% CD

DISTRIBUCIÓN Y TIEMPO DE ESFUERZO POR ETAPA

Cuando el tamaño de un proyecto no se ajusta al de los valores estándares de la siguiente tabla, la distribución del esfuerzo y el tiempo de desarrollo lo podemos obtener a través de interpolación. El total de líneas de código expresado en miles es de 82.919 MF, por lo que los porcentajes de interpolación se ubican en la tercera y cuarta columna de la siguiente tabla:

INDICADOR	FASES	PEQUEÑO 2MF	INTERMEDIO 8MF	MEDIANO 32MF	GRAN 128MF	MUY GRANDE 512MF
ESFUERZO						
Porcentajes (%)	Estudio Preliminar	7	7	7	7	7
	Análisis	17	17	17	17	17
	Diseño y desarrollo	64	61	58	55	52
	Diseño	27	26	25	24	23
	Desarrollo	37	35	33	31	29

	Prueba e implementaci ón	19	22	25	28	31
TIEMPO DE DESARROLLO						
Porcentajes (%)	Estudio Preliminar	16	18	20	22	24
	Análisis	24	25	26	27	28
	Diseño y Desarrollo	56	52	48	44	40
	Prueba e implementaci ón	20	23	26	29	32

Tabla: Esfuerzo y tiempo de desarrollo estándares por etapa del ciclo de vida del desarrollo del software

$$\% prog = \%MF_1 + \left(\frac{MF - MF_1}{MF_2 - MF_1} \right) * (\%MF_2 - \%MF_1)$$

Ecuación 1: Porcentaje de esfuerzo y tiempo de desarrollo para un proyecto con MF no estándar

CALCULO DEL PORCENTAJE DE ESFUERZO EN LA ETAPA DE DISEÑO Y DESARROLLO

MF = 82.919	$\% prog = 61 + \left(\frac{82.919 - 32}{128 - 32} \right) * (55 - 58)$ $\% prog = 58 + 0.5304 * (55 - 58)$ $\% prog = 56.4088 \%$
MF1 = 32	
MF2 = 128	
%MF1 = 58	
%MF2 = 55	

CALCULO DEL PORCENTAJE DE ESFUERZO EN LA ETAPA DE PRUEBA E IMPLEMENTACIÓN.

MF = 82.919	$\% prog = 61 + \left(\frac{82.919 - 32}{128 - 32} \right) * (55 - 58)$ $\% prog = 28 + 0.5304 * (25 - 28)$ $\% prog = 26.4088 \%$
MF1 = 32	
MF2 = 128	
%MF1 = 25	
%MF2 = 28	

CÁLCULO DEL PORCENTAJE DEL TIEMPO DE DESARROLLO EN LA ETAPA DE ESTUDIO PRELIMINAR

MF = 82.919	$\% prog = 61 + \left(\frac{82.919 - 32}{128 - 32} \right) * (55 - 58)$ $\% prog = 20 + 0.5304 * (22 - 20)$ $\% prog = 21.0608 \%$
MF1 = 32	
MF2 = 128	
%MF1 = 20	
%MF2 = 22	

CÁLCULO DEL PORCENTAJE DE ESFUERZO EN LA ETAPA DE ANÁLISIS

MF = 82.919	$\% prog = 61 + \left(\frac{82.919 - 32}{128 - 32} \right) * (55 - 58)$ $\% prog = 26 + 0.5304 * (27 - 26)$ $\% prog = 26.5304 \%$
MF1 = 32	
MF2 = 128	
%MF1 = 26	
%MF2 = 27	

CÁLCULO DEL PORCENTAJE DE DESARROLLO EN LA ETAPA DE DISEÑO Y DESARROLLO

MF = 82.919	$\% prog = 61 + \left(\frac{82.919 - 32}{128 - 32} \right) * (55 - 58)$ $\% prog = 48 + 0.5304 * (44 - 48)$ $\% prog = 45.8784 \%$
MF1 = 32	
MF2 = 128	
%MF1 = 48	
%MF2 = 44	

CÁLCULO DEL PORCENTAJE DE DESARROLLO EN LA ETAPA DE PRUEBA E IMPLEMENTACIÓN

MF = 82.919	$\% prog = 61 + \left(\frac{82.919 - 32}{128 - 32} \right) * (55 - 58)$ $\% prog = 26 + 0.5304 * (29 - 26)$ $\% prog = 27.5912 \%$
MF1 = 32	
MF2 = 128	
%MF1 = 26	
%MF2 = 29	

PARA CALCULAR ESF

$$ESF = E * \%ESF \quad \text{Donde: } E = 164$$

$$ESF (\text{Estudio preliminar}) = 164 * 7\% = 11.48$$

$$ESF (\text{Análisis}) = 164 * 17\% = 27.88$$

$$ESF (\text{Diseño y Desarrollo}) = 164 * 56.4088 \% = 92.5104$$

$$ESF (\text{Prueba e Implementación}) = 164 * 26.4088 \% = 43.3104$$

PARA CALCULAR TDES

$$des = TDES * \%TDES \quad \text{Donde: } TDES = 19$$

$$Tdes (\text{Estudio preliminar}) = 19 * 21.0608\% = 4.001$$

$$Tdes (\text{Análisis}) = 19 * 26.5304\% = 5.0407$$

$$Tdes (\text{Diseño y Desarrollo}) = 19 * 45.8784\% = 8.7168$$

$$Tdes (\text{Prueba e Implementación}) = 19 * 27.5912\% = 5.2423$$

Todos estos cálculos realizados anteriormente son parte de la Distribución de esfuerzo y tiempo de desarrollo del sistema en cada etapa:

ETAPA	ESF (%)	ESF	TDES (%)	TDES	CH
Estudio Preliminar	7	11.48	21.0608	4.001	2.86
Análisis	17	27.88	26.5304	5.0407	5.53
Diseño y Desarrollo	56.4088	92.5104	45.8784	8.7168	10.61
Prueba e Implementación	26.88	43.5104	27.5912	5.2423	8.29

Tabla: Distribución de esfuerzo y tiempo de desarrollo del sistema en cada etapa

DISTRIBUCIÓN DEL COSTO DE FUERZA DE TRABAJO POR ETAPA

Nota: Tomaremos el promedio salarial del mercado actualmente en Nicaragua para la especialidad tomando en cuenta las tecnologías y dimensión de proyecto.

- PROJECT MANAGER = \$3,000/mes
- SOFTWARE ENGINEER = \$1,500/mes
- JAVA BACKEND DEVELOPER (SENIOR)= \$1,000/mes
- FRONTEND DEVELOPER(SENIOR)= \$800/mes

CFT = Salario * Cantidad de Personas(CH) * TDES

DISTRIBUCIÓN DEL COSTO DE FUERZA DE TRABAJO ETAPA ESTUDIO PRELIMINAR

Donde: CH = 2.86 ~ 3 / TDES = 4.001 ~ 4

CFT = (\$3,000) (1 x PROJECT MANAGER) (4.001) = \$12,003

+

CFT = (\$1,500) (2 x SOFTWARE ENGINEER) (4.001) = \$12,003

CFT1 = \$24,006

DISTRIBUCIÓN DEL COSTO DE FUERZA DE TRABAJO ETAPA ANÁLISIS

Donde: CH = 5.53 ~ 6 / TDES = 5.0407

CFT = (\$3,000) (1 x PROJECT MANAGER) (5.0407) = \$15,122.1

+

CFT = (\$1,500) (2 x SOFTWARE ENGINEER) (5.0407) = \$15,122.1

+

CFT = (\$1,000) (1 x BACKEND DEVELOPER) (5.0407) = \$5,040.7

+

CFT = (\$800) (1 x FRONTEND DEVELOPER) (5.0407) = \$4,032.56

CFT2 = \$39,317.46

DISTRIBUCIÓN DEL COSTO DE FUERZA DE TRABAJO ETAPA DISEÑO Y DESARROLLO

Donde: CH = 10.61 ~ 11 / TDES = 8.7168

CFT = (\$3,000) (1 x PROJECT MANAGER) (8.7168) = \$26,150.4

+

CFT = (\$1,500) (2 x SOFTWARE ENGINEER) (8.7168) = \$26,150.4

+

CFT = (\$1,000) (4 x BACKEND DEVELOPER) (8.7168) = \$34,867.2

+

CFT = (\$800) (4 x FRONTEND DEVELOPER) (8.7168) = \$27,893.76

CFT3 = \$115,061.76

DISTRIBUCIÓN DEL COSTO DE FUERZA DE TRABAJO ETAPA PRUEBA E IMPLEMENTACIÓN

Donde: CH = 8.29 ~ 8 / TDES = 5.2423

CFT = (\$3,000) (1 x PROJECT MANAGER) (5.2423) = \$15,726.9

+

CFT = (\$1,500) (2 x SOFTWARE ENGINEER) (5.2423) = \$15,726.9

+

CFT = (\$1,000) (3 x BACKEND DEVELOPER) (5.2423) = 15,726.9

+

CFT = (\$800) (2 x FRONTEND DEVELOPER) (5.2423) = \$8,387.68

CFT4 = \$55,568.38

CFT = CFT1 + CFT2 + CFT3 + CFT4

CFT = \$24,006 + \$39,317.46 + \$115,061.76 + \$55,568.38

CFT = \$233,953.6

CFT = C\$7,018,608

Durante las 4 etapas del proceso de desarrollo del sistema se deberá realizar una inversión de **C\$7,018,608** (siete millones, dieciocho mil, seiscientos ocho córdobas).

DISTRIBUCIÓN DE LOS COSTOS DE UTILIZACIÓN DE LOS MEDIOS TÉCNICOS

Para calcular el costo de utilización de medios técnicos (CUMT), es necesario determinar:

- Cantidad de horas que trabajará cada consultor utilizará la computadora.

Cada consultor utilizará la computadora 40 horas a la semana, es decir, 160 horas al mes.

- Cantidad de energía que consume cada computadora durante el tiempo de trabajo

Según algunas instituciones de energía pública de países en América Latina, el consumo promedio de una computadora es de 300 watts, es decir, 0.3 KW. (Enre, s.f.), mientras que el consumo de un router es de 8 watts, es decir 0.008 KW.

DISPOSITIVOS	POTENCIA (WATTS)	POTENCIA (KW)
Computadora	300	0.3
Totales	300	0.3

Tabla 1: Consumo de energía promedio expresada en watt

Para calcular el costo del consumo de energía, se utiliza la siguiente fórmula

$$CCe = Ce * CKH * NoH$$

Donde:

CCe: Costo de consumo de energía

Ce: Consumo de energía

CKH: Costo de Kilowatts – Hora

NoH: Número de horas utilizadas al mes

Costo de consumo de energía por computadora

$$CCe = 0.3 \frac{Kw}{Computadora} * 4.5404 \frac{C\$}{Kw * H} * 160 \frac{H}{Mes}$$

$$CCe = 217.9392 \frac{C\$}{Mes * Computadora}$$

Una vez obtenido el costo del consumo de energía por computadora mensual, se puede calcular el costo del consumo de energía por computadora por etapa, con la siguiente fórmula:

$$CUMT = CCe * NoE * TDES$$

Donde:

CMUT: Costo de utilización de medios técnicos

CCe: Costo de consumo de energía

NoE: Cantidad de equipos de computación

TDES: Tiempo de desarrollo expresado en meses

Consideración: La cantidad de PC será igual a la cantidad de Analistas-Programador por etapa

- **ESTUDIO PRELIMINAR**

$$CMUT = 217.9392 \frac{C\$}{Mes * PC} * 3 PC * 4.001 Meses$$

$$CMUT = C\$ 2,615.9242$$

- **ANÁLISIS**

$$CMUT = 217.9392 \frac{C\$}{Mes * PC} * 6 PC * 5.0407 Meses$$

$$CMUT = C\$ 6,591.3967$$

- **DISEÑO Y DESARROLLO**

$$CMUT = 217.9392 \frac{C\$}{Mes * PC} * 11 PC * 8.7168 Meses$$

$$CMUT = C\$ 20,897.0566$$

- **PRUEBA E IMPLEMENTACIÓN**

$$CMUT = 217.9392 \frac{C\$}{Mes * PC} * 8 PC * 5.2423 Meses$$

$$CMUT = C\$ 9,140.0213$$

De los cálculos anteriores, se obtiene la siguiente tabla:

ETAPA	CCE	NOE	TDES	CMUT
Estudio preliminar	C\$ 217.9392	3	4.001	C\$2,615.9242
Análisis	C\$ 217.9392	6	5.0407	C\$6,591.3967
Diseño y Desarrollo	C\$ 217.9392	11	8.7168	C\$20,897.0566
Prueba e Implementación	C\$ 217.9392	8	5.2423	C\$9,140.0213
TOTAL				C\$ 11,227.36

Tabla 2: Costo de consumo de energía de computadores por etapa

COSTO DE CONSUMO DE ENERGÍA POR COMPUTADORA

Durante el tiempo de desarrollo del sistema, se necesitará un router para brindar acceso a internet a todos los equipos de cómputo.

En promedio, un router consume 0.008 KW, según algunos entes reguladores de energía de países de Latino América.

Para saber el costo de consumo de energía, se utiliza la misma fórmula utilizada para el costo de consumo de energía por computadora:

$$CCe = 0.008 \frac{Kw}{Router} * 4.5404 \frac{C\$}{Kw * H} * 160 \frac{H}{Mes}$$

$$CCe = 5.8117 \frac{C\$}{Mes * Router}$$

Una vez obtenido el costo del consumo de energía por router mensual, se puede calcular el costo del consumo de energía por router por etapa, con la fórmula del Costo de utilización de medios técnicos. Como sólo se utilizará un router por cada etapa, la fórmula queda de la siguiente manera:

$$CUMT = CCe * TDES$$

- **ESTUDIO PRELIMINAR**

$$CMU = 5.8117 \frac{C\$}{Mes} * 4.001 Meses = C\$23.2526$$

- **ANÁLISIS**

$$CMU = 5.8117 \frac{C\$}{Mes} * 5.0407 Meses = C\$29.2950$$

- **DISEÑO Y DESARROLLO**

$$CMU = 5.8117 \frac{C\$}{Mes} * 8.7168 Meses = C\$50.6594$$

- **PRUEBA E IMPLEMENTACIÓN**

$$CMUT = 5.8117 \frac{C\$}{Mes} * 5.2423 Meses = C\$30.4666$$

De los cálculos anteriores, se obtiene la siguiente tabla:

ETAPA	CCE	TDES	CMUT
Estudio preliminar	C\$5.8117	4.001	C\$23.2526

Análisis	C\$5.8117	5.0407	C\$29.2950
Diseño y Desarrollo	C\$5.8117	8.7168	C\$50.6594
Prueba e Implementación	C\$5.8117	5.2423	C\$30.4666
TOTAL			C\$ 133.6736

Tabla 3: Costo de consumo de energía por router por etapa

COSTO DE CONSUMO DE ENERGÍA GENERAL

EQUIPO	CUMT
Computadoras	C\$ 11,227.36
Router	C\$ 133.6736
TOTAL, CMUT	C\$ 11,361.0336

Tabla 4: Costo total CMUT

Durante las 4 etapas de desarrollo del sistema, se deberá realizar una inversión de **C\$ 11,361.0336**

CÁLCULO DE COSTO DE ABASTECIMIENTO TÉCNICO DE MATERIALES

A continuación, se detalla una proyección de materiales que serán utilizados durante el desarrollo del software:

- ESTUDIO PRELIMINAR**

CANTIDAD	ARTÍCULO	PRECIO UNITARIO⁴⁵ (C\$)	TOTAL (C\$)
-----------------	-----------------	-----------------------------------------------	--------------------

⁴⁵ Fuente: www.gonperlibrerias.com consultado el 01/05/2017

4	Resma papel bond 40 8.5 X 11 RIPAX 97%	596.4	2385.6
3	Lapiceros PENTEL BK-66	5.41	16.23
3	Lápiz portaminas retráctil M-301 ULTRA 0.5 MM	41.47	124.41
3	Borrador de goma Pelikan WS-30	2.09	6.27
1	Caja de folder AMPO MANILA T/C CAJA DE 100 UD T/C	113.91	113.91
3	Corrector DACATI T/BOLIGRAFO YM-305	13.69	41.07
1	Cajas de grapas DACATI STANDARD SP26/6	22.53	22.53
9	Minas PAPERMATE 0.5 HB	11.63	104.67
3	Reglas	15.00	45
3	Cartucho de color para impresora	771.2231 ⁴⁶	2313.6693
1	Engrapadora ACME de escritorio TOP 125	52.16	52.12
3	Cuadernos universitarios #3 LISO 60 HJS	12.64	37.92
3	Marcador permanente Artline 107 punta redonda	8.28	24.84
3	Marcador acrílico Artline 157	11.47	34.41
3	Memoria MAXELL USB 8GB	596.4	1789.2
1	Splinder de 100 CD	252	252
1	Splinder de 50 DVD	184.03	184,03
2	Caja de Discos MAXELL para DVD SL5	10.23	20,46
SUB-TOTAL (C\$)			7568.3393
IVA (15%)			1135.2509

⁴⁶ Fuente: Comtech consultado el 01/05/2017

TOTAL (C\$)	8,703.5902
--------------------	-------------------

Figura 2.38: CMAT de la etapa de estudio preliminar
Fuente: Elaboración propia

- ANÁLISIS**

CANTIDAD	ARTÍCULO	PRECIO UNITARIO (C\$)	TOTAL (C\$)
3	Lapiceros PENTEL BK-66	5.41	16.23
3	Borrador de goma Pelikan WS-30	2.09	6.27
9	Minas PAPERMATE 0.5 HB	11.63	104.67
3	Cuadernos universitarios #3 LISO 60 HJS	12.64	37.92
SUB-TOTAL (C\$)			165.09
IVA (15%)			24.7635
TOTAL (C\$)			189.8535

Figura 2.39: CMAT de la etapa de análisis
Fuente: Elaboración propia

- DISEÑO Y DESARROLLO**

CANTIDAD	ARTÍCULO	PRECIO UNITARIO (C\$)	TOTAL (C\$)
3	Lapiceros PENTEL BK-66	5.41	16.23
3	Borrador de goma Pelikan WS-30	2.09	6.27
9	Minas PAPERMATE 0.5 HB	11.63	104.67
3	Cuadernos universitarios #3 LISO 60 HJS	12.64	37.92

SUB-TOTAL (C\$)	165.09
IVA (15%)	24.7635
TOTAL (C\$)	189.8535

Figura 2.40: CMAT de la etapa de diseño y desarrollo

Fuente: Elaboración propia

• **PRUEBA E IMPLEMENTACIÓN**

CANTIDAD	ARTÍCULO	PRECIO UNITARIO (C\$)	TOTAL (C\$)
3	Lapiceros PENTEL BK-66	5.41	16.23
3	Borrador de goma Pelikan WS-30	2.09	6.27
9	Minas PAPERMATE 0.5 HB	11.63	104.67
3	Cuadernos universitarios #3 LISO 60 HJS	12.64	37.92
SUB-TOTAL (C\$)			165.09
IVA (15%)			24.7635
TOTAL (C\$)			189.8535

Figura 2.41: CMAT de la etapa de prueba e implementación

Fuente: Elaboración propia

CMAT= Total Fase Estudio Preliminar + Total Fase Análisis + Total Fase Diseño y Desarrollo + Total Fase Prueba e implementación.

CMAT = C\$ 8,703.5902 + C\$ 189.8535+ C\$ 189.8535 + C\$ 189.8535

CMAT = C\$ 9,273.1507

CMAT = \$ 438.8743031

SUSTITUYENDO EN LA FÓRMULA:

$$CD = CFT + CCe + CMAT$$

$$CD = C\$7,018,608 + C\$ 11, 361. 0336 + C\$ 9,273.1507$$

$$CD = C\$7,039,242.1843$$

Se puede calcular los Costos Indirectos:

$$CI = CD * 15\%$$

$$CI = C\$7,039,242.1843 * 0.15$$

$$CI = C\$1,055,886.3276$$

Y también los Costos Totales del Proyecto:

$$CTP = CD + CI$$

$$CTP = C\$7,039,242.1843 + C\$1,055,886.3276$$

$$CTP = C\$ 8, 095, 128. 5119$$

$$CTP= \$269,837.6170$$

9.7. ANEXO 7: Adquisición de Equipos

En el marco del desarrollo CORE-SIL se propone fortalecer la infraestructura tecnológica del MITRAB.

Para ello se propone la adquisición de los siguientes equipos y licencias:

DESCRIPCIÓN	CANT	PRECIO	TOTAL
EQUIPOS DE SEGURIDAD			
SEGURIDAD PERIMETRAL: Palo Alto Networks® PA-3020 next-generation firewall appliances, 3 años de Garantía y Soporte	1	\$42,000.00	\$42,000.00
Seguridad WAF F5 Silverline	1	\$40,000.00	\$40,000.00
Control de Acceso con Huella Digital, Marca BIOTRACK modelo BIOHOME LR + Instalación	1	\$1,000.00	\$1,000.00
EQUIPOS DE PROCESAMIENTO Y ALMACENAMIENTO			
Servers 2 x Intel Xeon E5-2697 v4 2.3ghz 18 core 45mb cache, 512gb ram ddr4ll, tarjeta de fibra canal 10Ge, tarjeta c/2 puertos 10Ge Ethernet, sin discos	2	\$25,000.00	\$50,000.00
Almacenamiento unificado NetAPP con tarjeta de cache 1TB, con 2 controladoras de 4 puertos c/u, c/puertos UTA corra fibra canal 4, 8, 16GB, CIFS, FCoE, NFS, Ethernet 1, 10GE y ISCSI 1, 10GE, Closterisable y virtualizado, soportar fallo de arreglo a 2 discos, escalamiento vertical y horizontal, almacenamiento inicial 9TB Utilizables	1	\$35,000.00	\$35,000.00

Licencia VMware vSphere Essential Plus Kit	1	\$6,000.00	\$6,000.00
COMUNICACIÓN			
EQUIPOS CORE: Cisco Nexus 5672 UP 48 Port 16 Port unificados ejecutar protocolos fibra canal a 8 y 16Gbps, FCoE, ISCSI a 1 y 10GE y Ethernet a 1 y 10GE, 32 port SFP+ ejecutables a 1 y 10GE Ethernet FCoE y ISCSI, 3 años de garantía y soporte	2	\$25,000.00	\$50,000.00
SWITCHES DE ACCESO: WS-C2960X-48TD-L Catalyst 2960-X 48 GigE, 2 x 10G SFP+, LAN Base 3 años de garantía y soporte	2	\$5,000.00	\$10,000.00
RESPALDO DE ENERGÍA			
SU10000RT3U UPS de Doble Conversión SmartOnline 208/240V 10kVA 9kW, 6U, Autonomía Extendida, ranura para Tarjeta de Red, USB, DB9, Switch de Derivación, NEMA	1	\$3,000.00	\$3,000.00
Gabinete de módulo de baterías externas para rack/torre 240V 3U más cableado de CD para sistemas UPS seleccionados (BP240V10RT3U)	1	\$14,000.00	\$14,000.00
TOTAL(USD)			\$251,000.00
TOTAL(USD)			C\$7,530,000

9.8. ANEXO 8: COSTOS DE EQUIPOS PARA VIRTUALIZACIÓN

Comparación de costos de inversión PC convencional versus costos de equipos para virtualización de escritorios.

CONCEPTO	PC CONVENCIONAL	VIARTUALIZACION DE ESCRITORIOS
COMPRA DE EQUIPOS	C\$4,980,035.42	C\$6,927,895.69
MANTENIMIENTO	C\$752,146.58	C\$95,931.11
CONSUMO DE ENERGIA	C\$7,151,510.48	C\$79,461.23
REINVERSIÓN	C\$4,118,112.47	C\$1,827,000.00
TOTAL	C\$17,001,804.96	C\$8,930,288.02

Fuente: Elaboración propia con dato proporcionados por la División de Informática del MITRAB

9.9. ANEXO 9: Plantillas de Coleman

CASO DE USO: GESTIONAR SEGURIDAD			
Definición	En este caso de uso se manipula la información referente a los conductores		
Prioridad	<input checked="" type="radio"/> (1) Vital	<input type="radio"/> (2) Importante	<input type="radio"/> (3) Conveniente
Urgencia	<input checked="" type="radio"/> (1) Inmediata	<input type="radio"/> (2) Necesario	<input type="radio"/> (3) Puede esperar
ACTORES			
Administrador de Sistema	Es una representación del elemento humano que interactuará intensivamente con el sistema en casi todas sus funciones.		
Desarrollador de Sistema	Es una representación del elemento humano que administra el esquema de seguridad de una aplicación en concreto.		
ESCENARIOS			

Nombre	Registrar nuevo usuario
Pre-Condiciones	El registro de usuario debe de ser único.
Iniciado por	Administrador del Sistema / Desarrollador de Sistema
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "Crear usuario"
Operaciones	Realizar consumo del WebService "Crear usuario" El servidor validar datos del nuevo usuario. El servidor retorna respuesta del WebService "Crear usuario"
Excepciones	Si los datos enviados son correctos se procede a crear el nuevo usuario. Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Editar información del usuario
Pre-Condiciones	El registro de usuario a actualizar debe de existir.
Iniciado por	Administrador de Sistema / Desarrollador de Sistema
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "Actualizar usuario"
Operaciones	Realizar consumo del WebService "Actualizar usuario" El servidor validar datos del usuario a actualizar. El servidor Retorna respuesta del WebService "Actualizar usuario"
Excepciones	Si los datos enviados son correctos se procede actualizar usuario Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Eliminar información del usuario
Pre-Condiciones	El registro de usuario a eliminar debe de existir.
Iniciado por	Administrador del Sistema / Desarrollador de Sistema
Finalizado por	Sistema.

Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "Eliminar usuario"
Operaciones	Realizar consumo del WebService "Eliminar usuario" El servidor validar datos de usuario a eliminar. El servidor retorna respuesta del WebService "Eliminar usuario"
Excepciones	Si los datos enviados son correctos se procede eliminar usuario Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Listar información de un usuario.
Pre-Condiciones	El registro de los usuarios a buscar debe de existir.
Iniciado por	Administrador de Sistema / Desarrollador de Sistema
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "listar usuario"
Operaciones	Realizar consumo del WebService "listar usuarios" El servidor validar datos solicitados. El servidor retorna respuesta del WebService "listar usuarios"
Excepciones	Si los datos enviados son correctos se procede listar datos de usuarios Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Ver información de un usuario.
Pre-Condiciones	El registro de usuario debe de existir.
Iniciado por	1. Administrador de Sistema / Desarrollador de Sistema
Finalizado por	2. Sistema.
Post-Condiciones	3. El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "ver usuario"
Operaciones	1. Realizar consumo del WebService "ver usuario" 2. El servidor validar datos del usuario. 4. El servidor retorna respuesta del WebService "ver usuario"
Excepciones	5. Si los datos enviados son correctos se procede retornar datos de usuario. 3. Si los datos enviados son erróneos se retornará "answer 0"

Nombre	Registrar nuevo Rol
Pre-Condiciones	El registro de rol de usuario debe de ser único
Iniciado por	Administrador del Sistema / Desarrollador de Sistema
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo Webservice "Crear usuario"
Operaciones	Realizar consumo del Webservice "Crear rol" El servidor validar datos del nuevo rol. El servidor retorna respuesta del Webservice "Crear rol"
Excepciones	Si los datos enviados son correctos se procede a crear el nuevo rol. Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Editar Rol
Pre-Condiciones	El registro de rol a actualizar debe de existir.
Iniciado por	Administrador del Sistema / Desarrollador de Sistema
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo Webservice "Actualizar rol"
Operaciones	Realizar consumo del Webservice "Actualizar rol" El servidor validar datos del rol a actualizar. El servidor Retorna respuesta del Webservice "Actualizar rol"
Excepciones	Si los datos enviados son correctos se procede actualizar rol Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Eliminar Rol
Pre-Condiciones	El registro del rol a eliminar debe de existir.
Iniciado por	Administrador del Sistema / Desarrollador de Sistema
Finalizado por	Sistema.

Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "Eliminar rol"
Operaciones	Realizar consumo del WebService "Eliminar rol" El servidor validar datos del rol a eliminar. El servidor retorna respuesta del WebService "Eliminar rol"
Excepciones	Si los datos enviados son correctos se procede eliminar rol Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Listar Roles
Pre-Condiciones	El registro de los roles debe de existir.
Iniciado por	Administrador de Sistema / Desarrollador de Sistema
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "listar rol"
Operaciones	Realizar consumo del WebService "listar rol" El servidor validar datos solicitados. El servidor retorna respuesta del WebService "listar rol"
Excepciones	Si los datos enviados son correctos se procede listar datos de usuarios Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Ver detalles de un Rol
Pre-Condiciones	El registro del rol debe de existir.
Iniciado por	Administrador de Sistema / Desarrollador de Sistema
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "ver rol"
Operaciones	Realizar consumo del WebService "ver rol" El servidor validar datos del usuario. El servidor retorna respuesta del WebService "ver rol"
Excepciones	Si los datos enviados son correctos se procede retornar datos de rol.

	Si los datos enviados son erróneos se retornará "answer 0"
--	------------------------------------------------------------

Nombre	Registrar nueva Aplicación
Pre-Condiciones	El registro de la aplicación de usuario debe de ser único
Iniciado por	Administrador del Sistema / Desarrollador de Sistema
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "Crear aplicación"
Operaciones	Realizar consumo del WebService "Crear aplicación" El servidor validar datos de la nueva aplicación. El servidor retorna respuesta del WebService "Crear aplicación"
Excepciones	Si los datos enviados son correctos se procede a crear la nueva aplicación Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Editar información de la Aplicación
Pre-Condiciones	El registro de la aplicación a actualizar debe de existir
Iniciado por	Administrador de Sistema / Desarrollador de Sistema
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "Actualizar Aplicación"
Operaciones	Realizar consumo del WebService "Actualizar aplicación" El servidor validar datos de la aplicación a actualizar. El servidor Retorna respuesta del WebService "Actualizar aplicación"
Excepciones	Si los datos enviados son correctos se procede actualizar aplicación Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Eliminar información de la Aplicación
Pre-Condiciones	El registro de la aplicación a eliminar debe de existir
Iniciado por	Administrador del Sistema / Desarrollador de Sistema

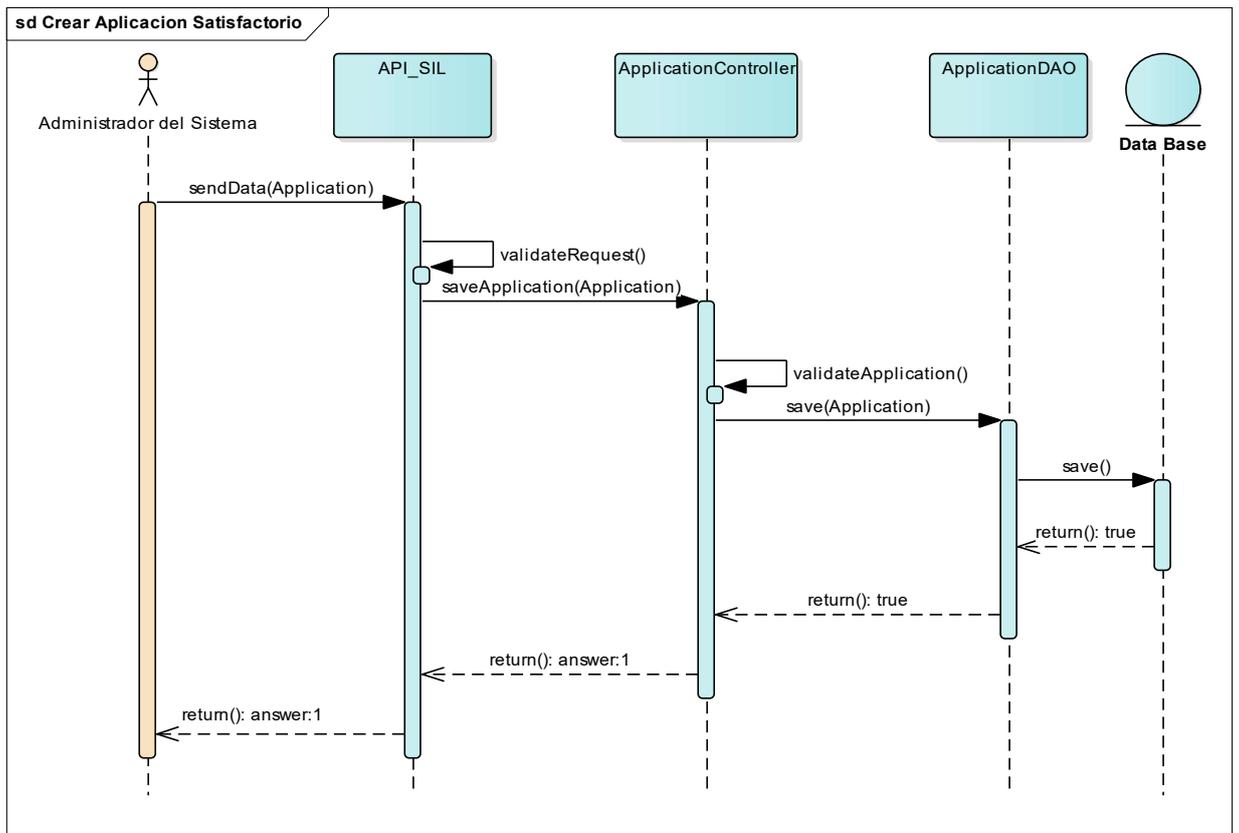
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "Eliminar aplicación"
Operaciones	Realizar consumo del WebService "Eliminar aplicación" El servidor validar datos de aplicación a eliminar. El servidor retorna respuesta del WebService "Eliminar aplicación"
Excepciones	Si los datos enviados son correctos se procede eliminar aplicación Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Listar Aplicaciones
Pre-Condiciones	El registro de las aplicaciones a listar debe de existir
Iniciado por	Administrador de Sistema / Desarrollador de Sistema
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "listar aplicación"
Operaciones	Realizar consumo del WebService "listar aplicación" El servidor validar datos solicitados. El servidor retorna respuesta del WebService "listar aplicación"
Excepciones	Si los datos enviados son correctos se procede listar datos de la aplicación Si los datos enviados son erróneos se retornará "answer 0"
Nombre	Ver detalles de una Aplicación
Pre-Condiciones	El registro de la aplicación debe de existir
Iniciado por	Administrador de Sistema / Desarrollador de Sistema
Finalizado por	Sistema.
Post-Condiciones	El caso de uso finaliza cuando el servidor retorna respuesta del consumo WebService "ver aplicación"
Operaciones	Realizar consumo del WebService "ver aplicación" El servidor validar datos de la aplicación. El servidor retorna respuesta del WebService "ver aplicación"

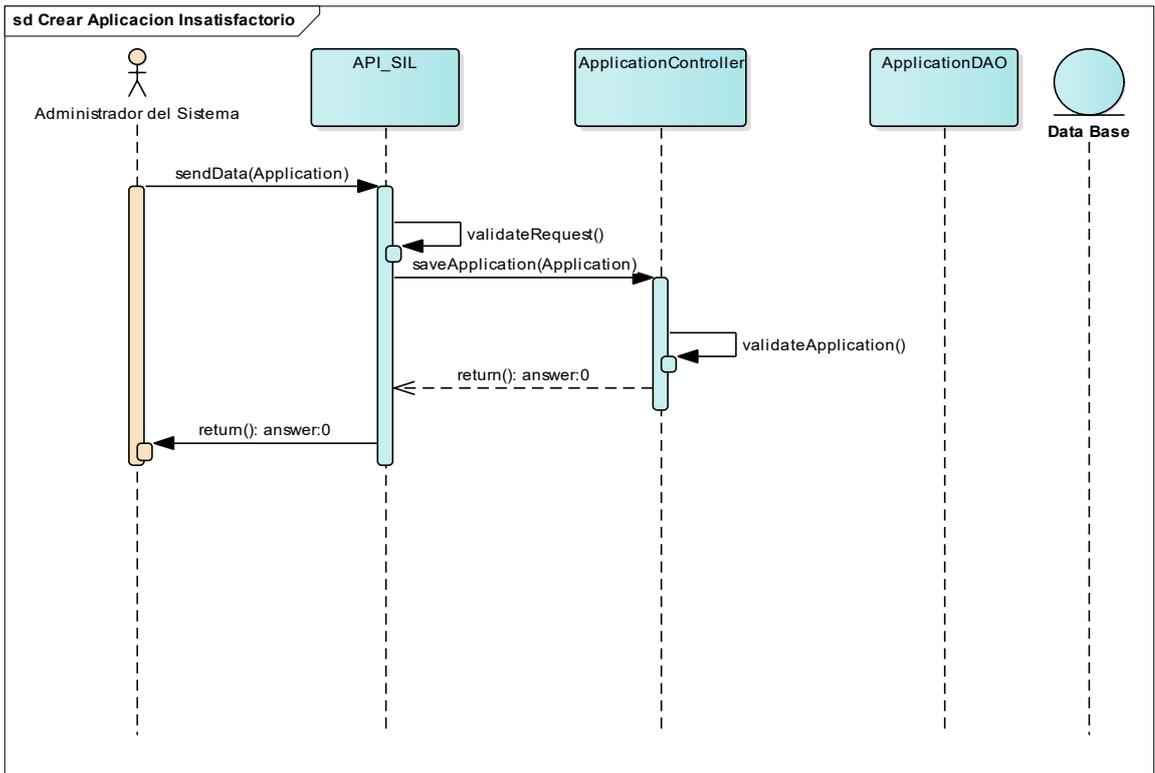
Excepciones	<p>Si los datos enviados son correctos se procede retornar datos de la aplicación.</p> <p>Si los datos enviados son erróneos se retornará "answer 0"</p>
--------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------

Continúa en DVD: Plantillas de Coleman.

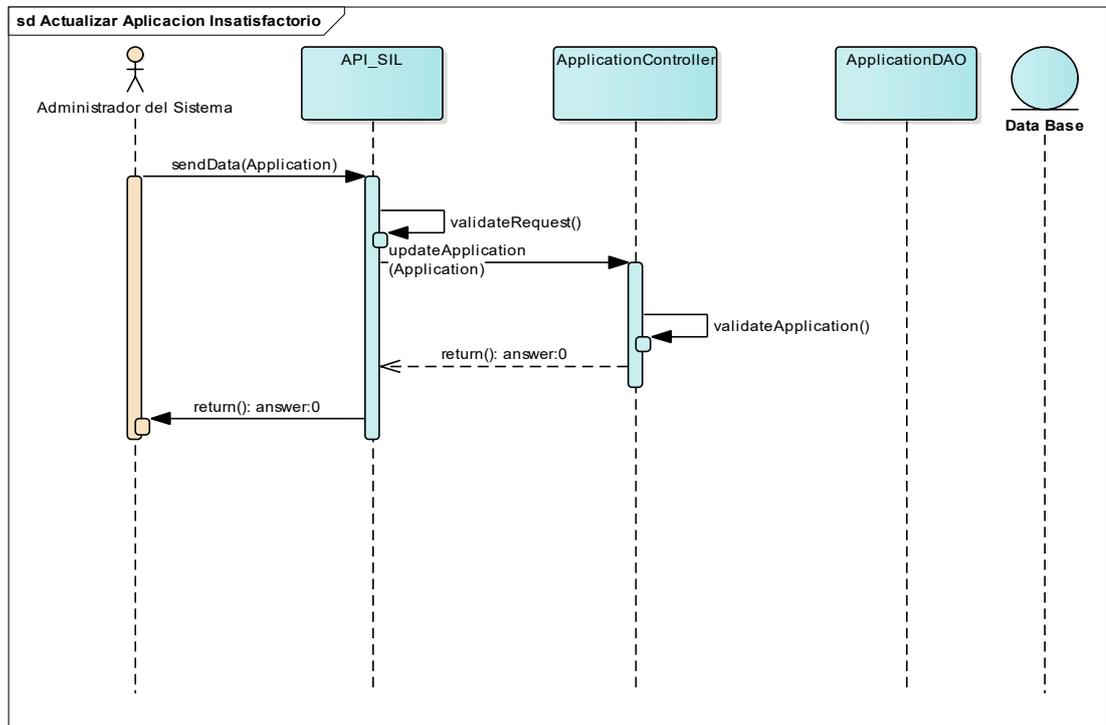
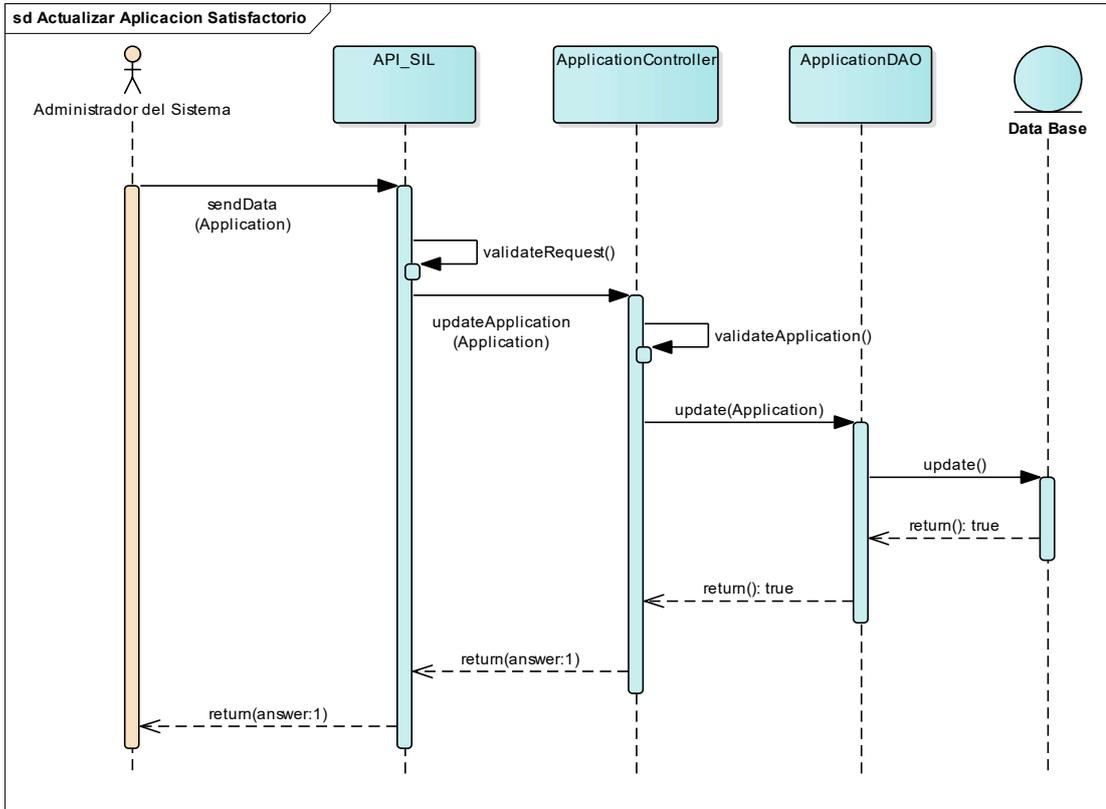
9.10. ANEXO 10: Diagramas de Secuencia

9.10.1. Crear Aplicación

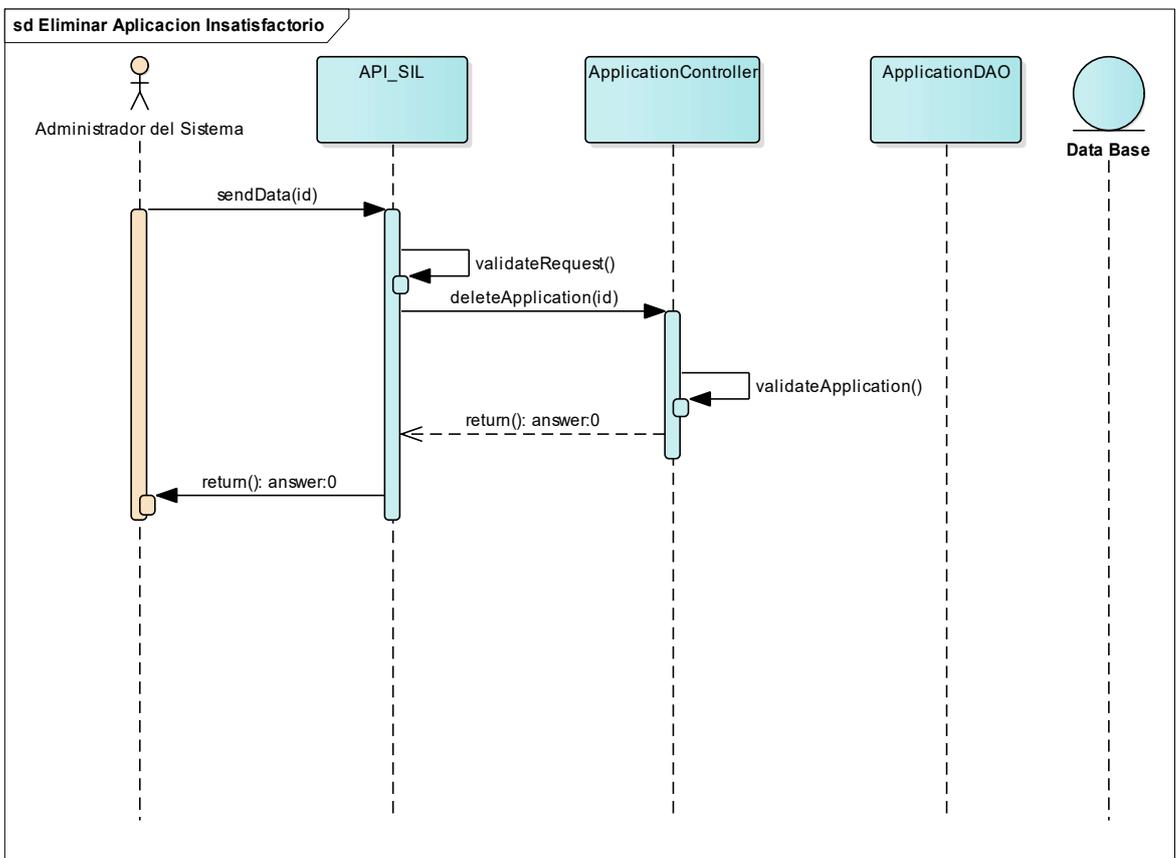
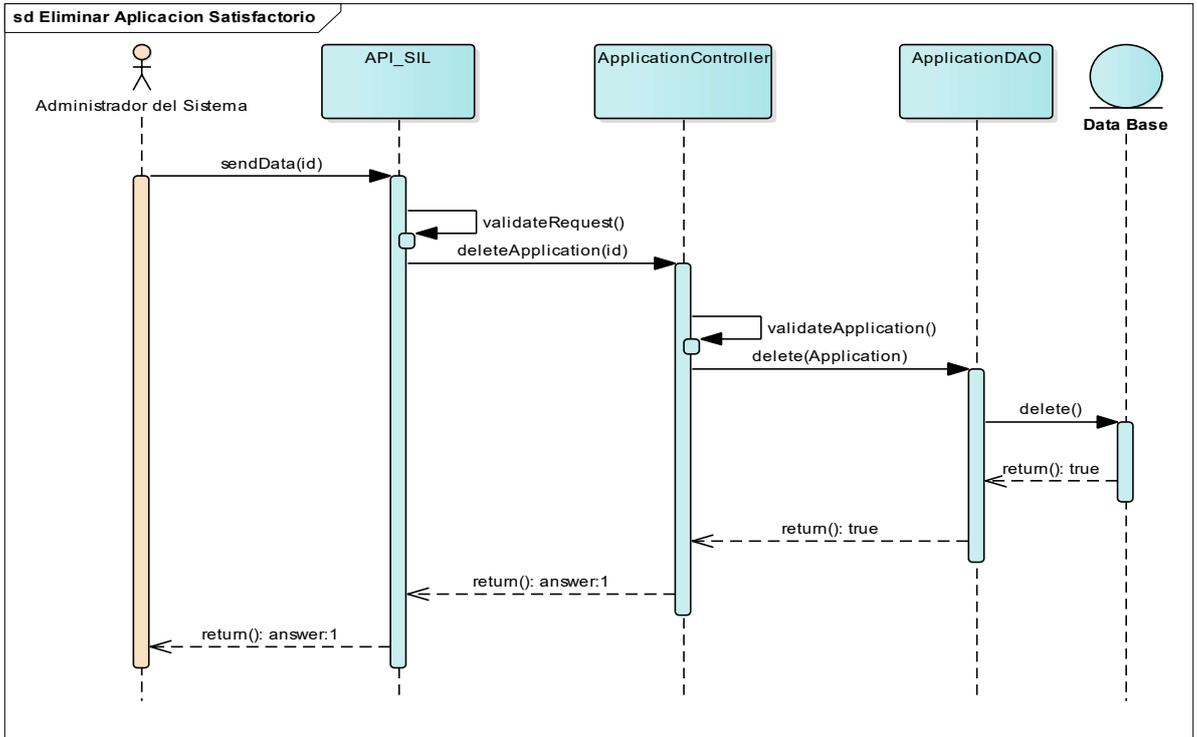




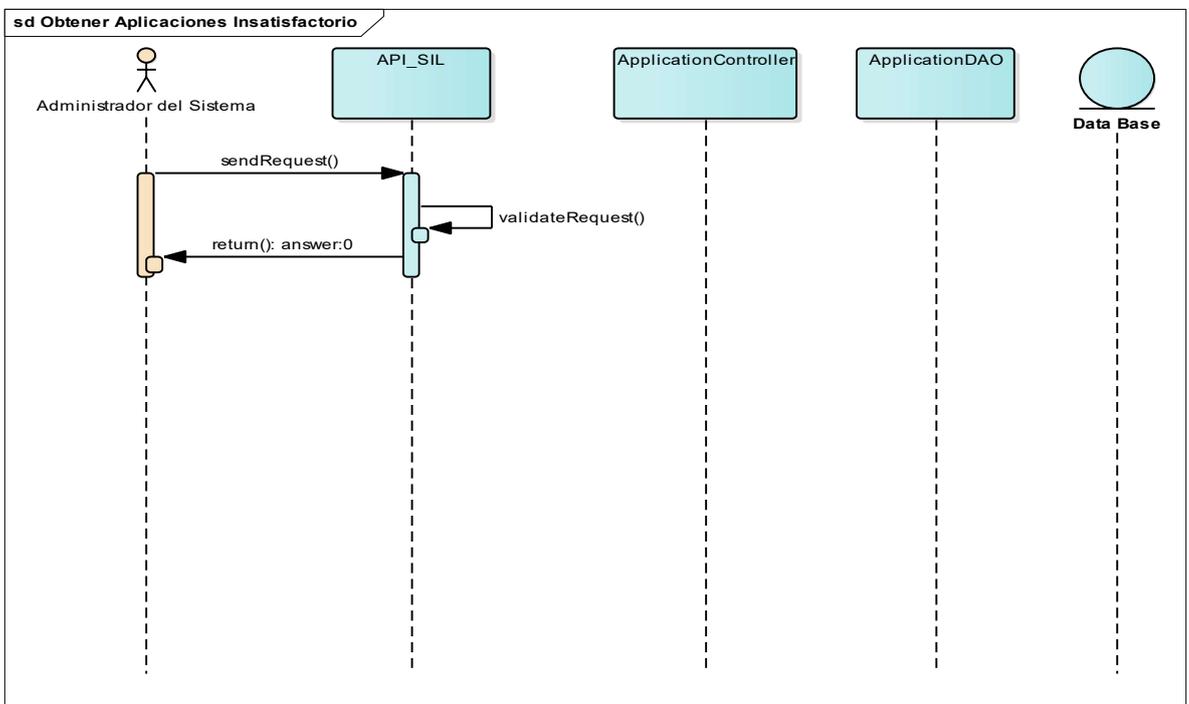
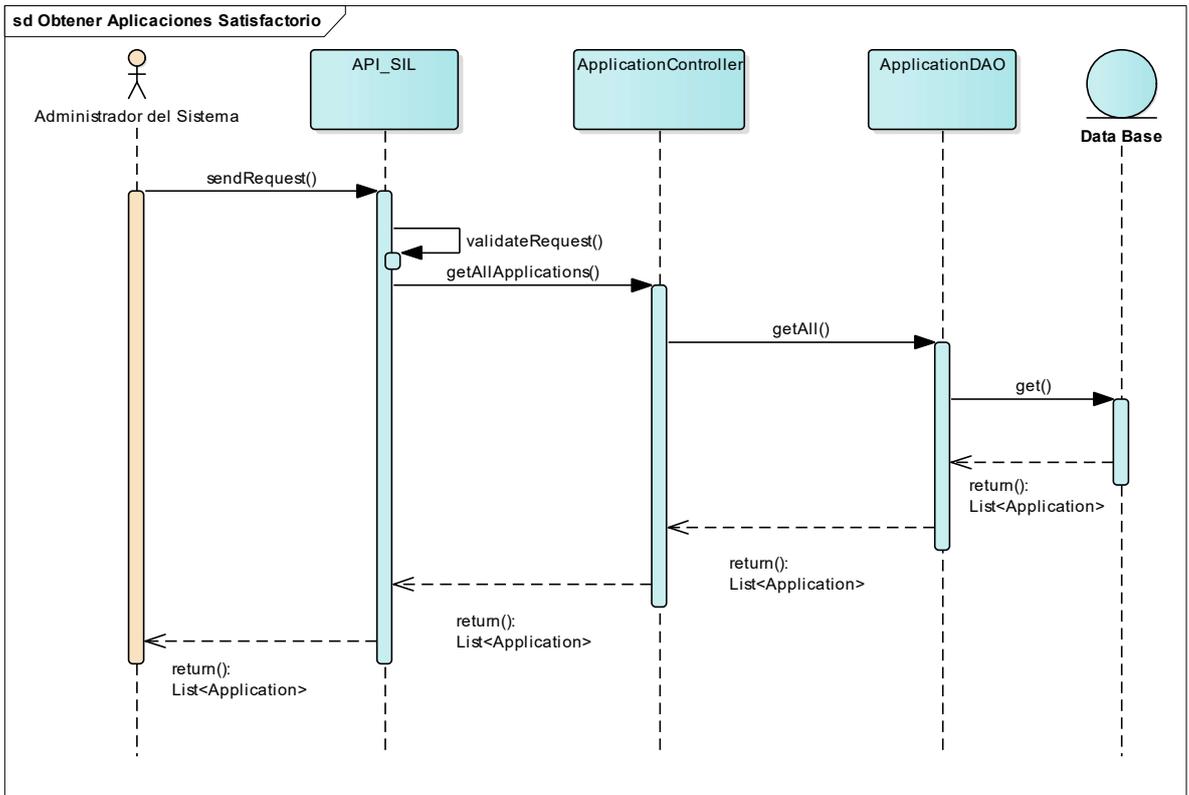
9.10.2. Actualizar Aplicación



9.10.3. Eliminar Aplicación



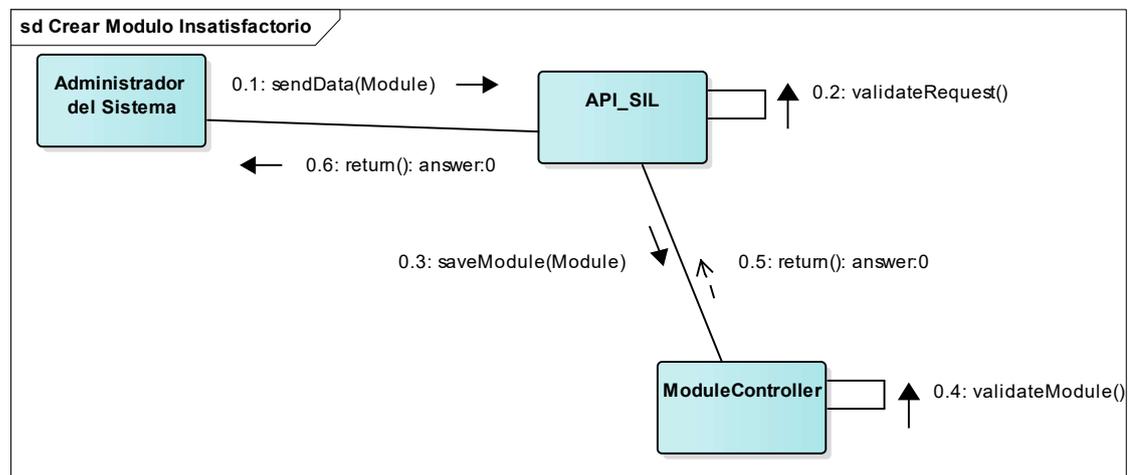
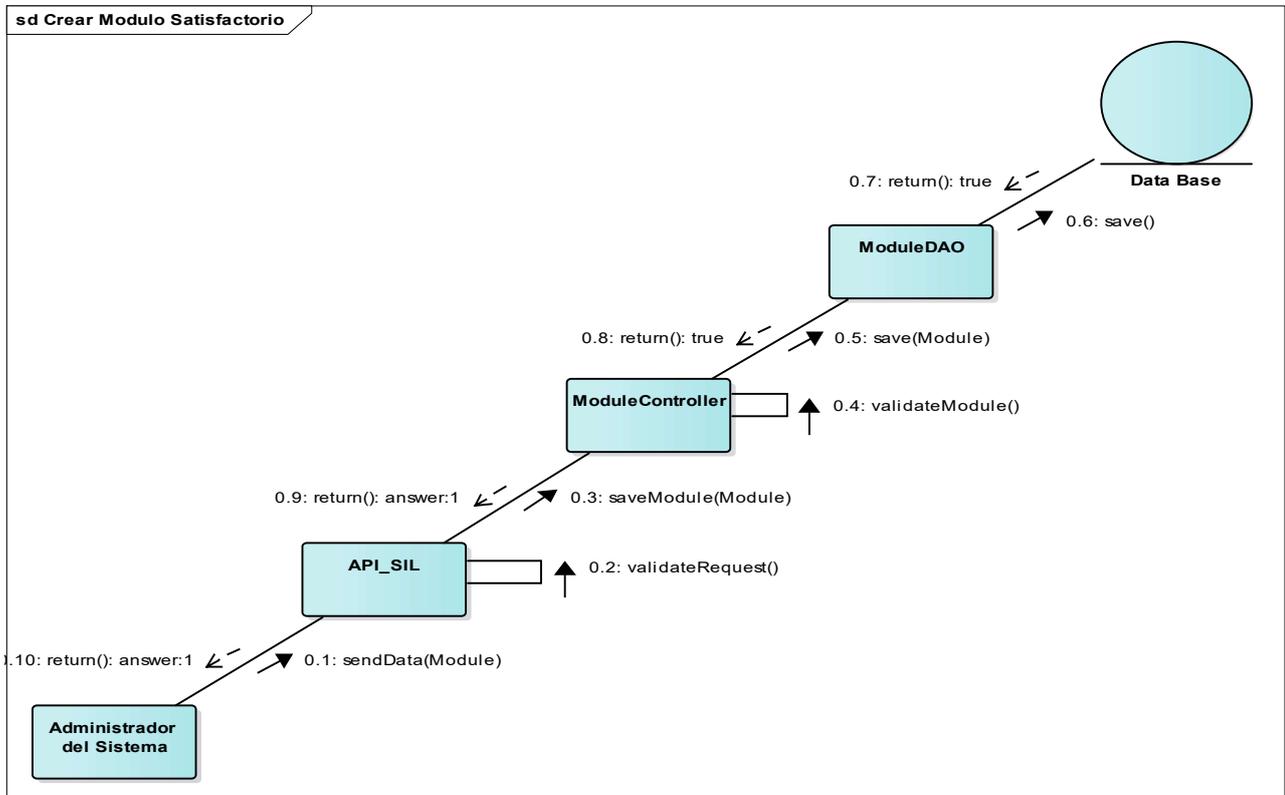
9.10.4. Obtener Aplicación.



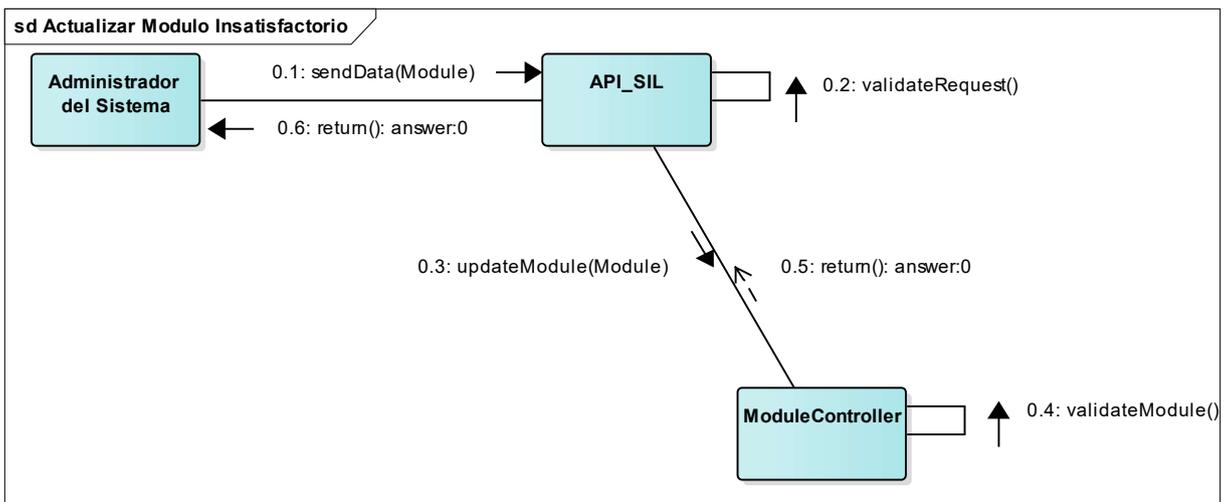
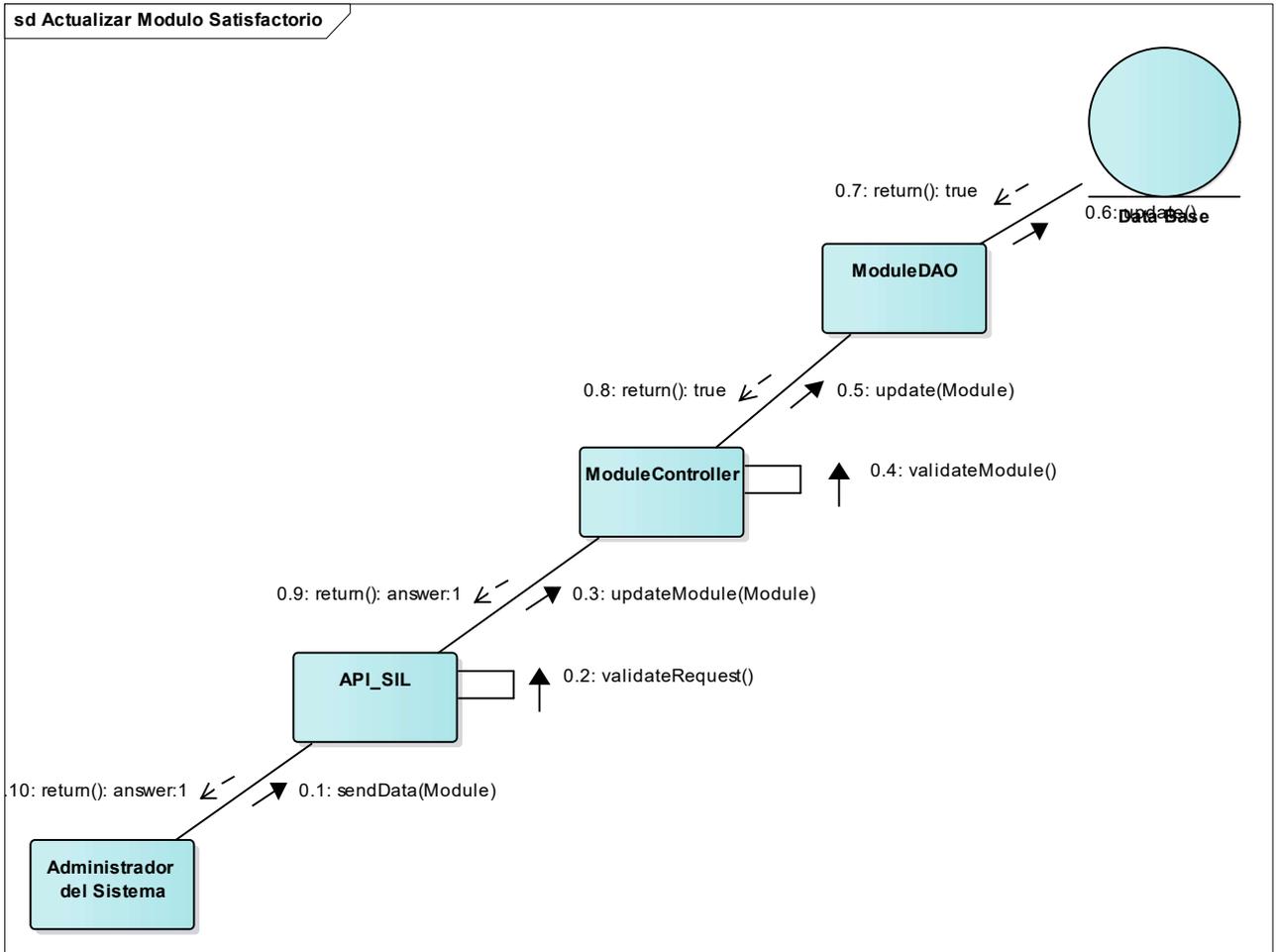
Continua en DVD: Diagramas de Secuencia.

9.11. ANEXO 11: Diagramas de Comunicación

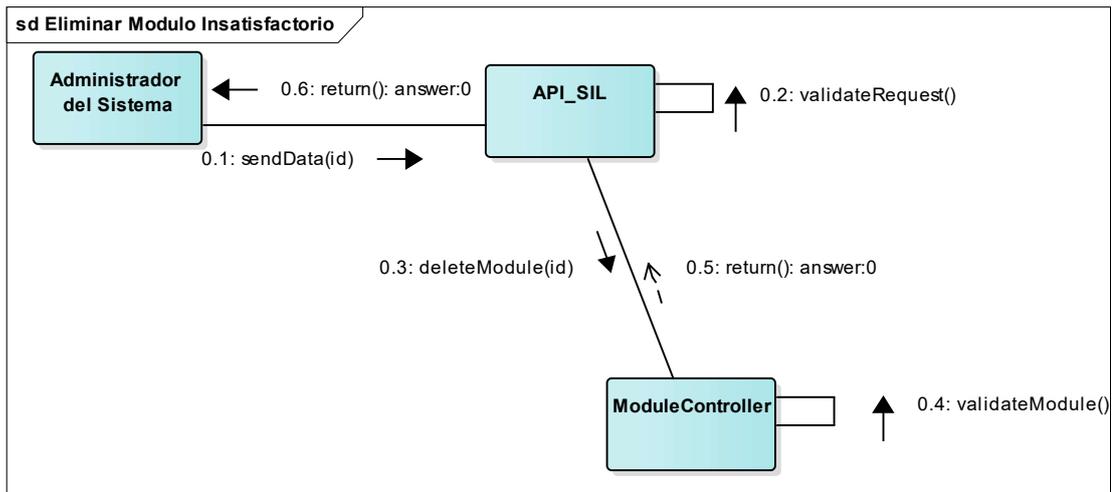
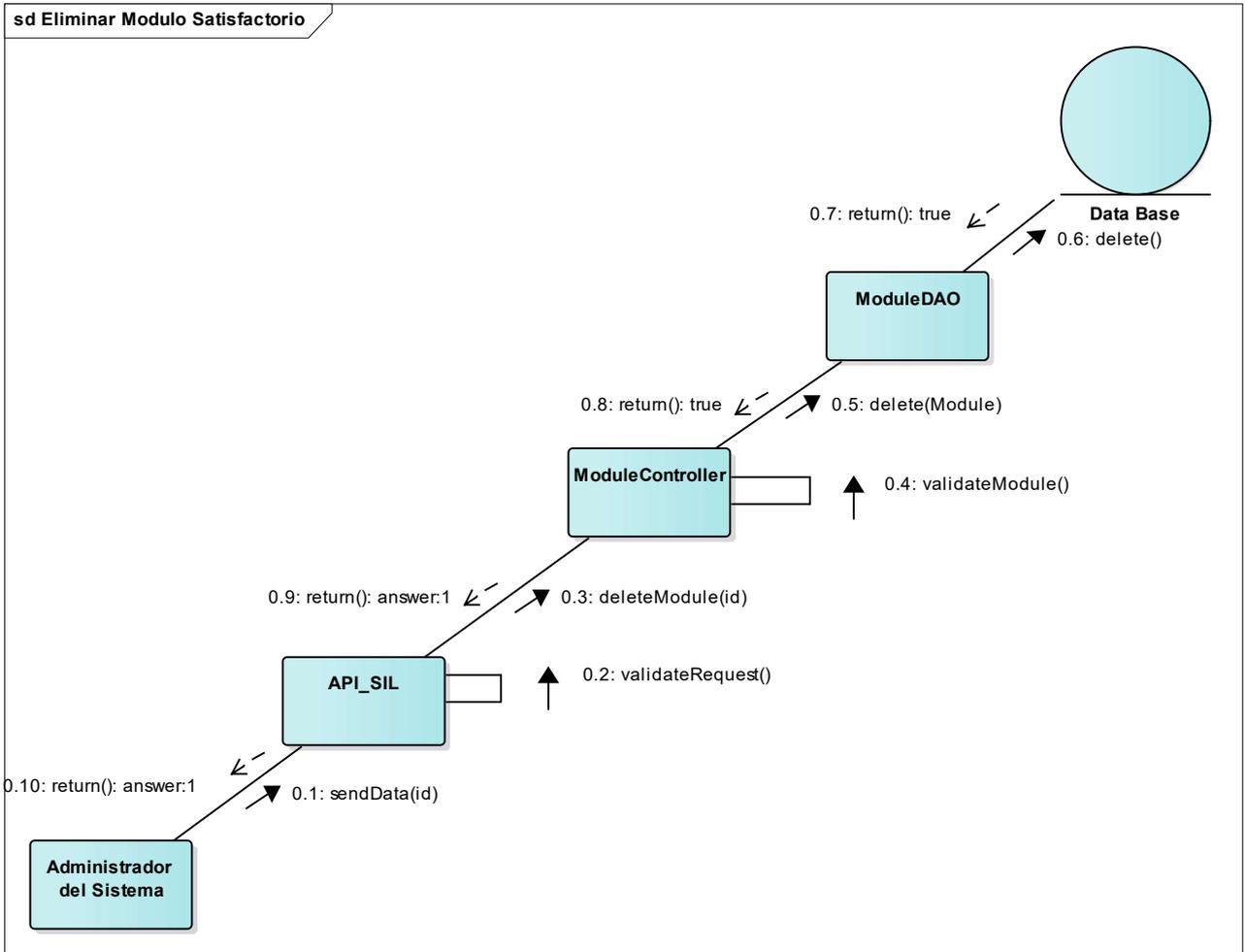
9.11.1. Crear Modulo de Aplicación.



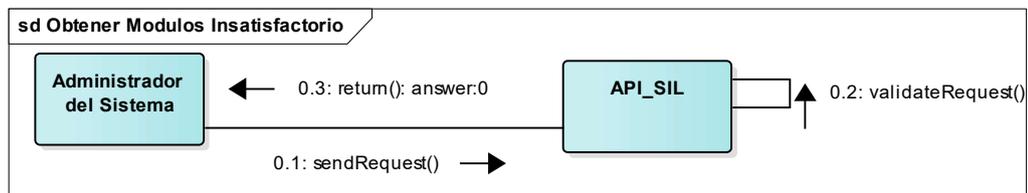
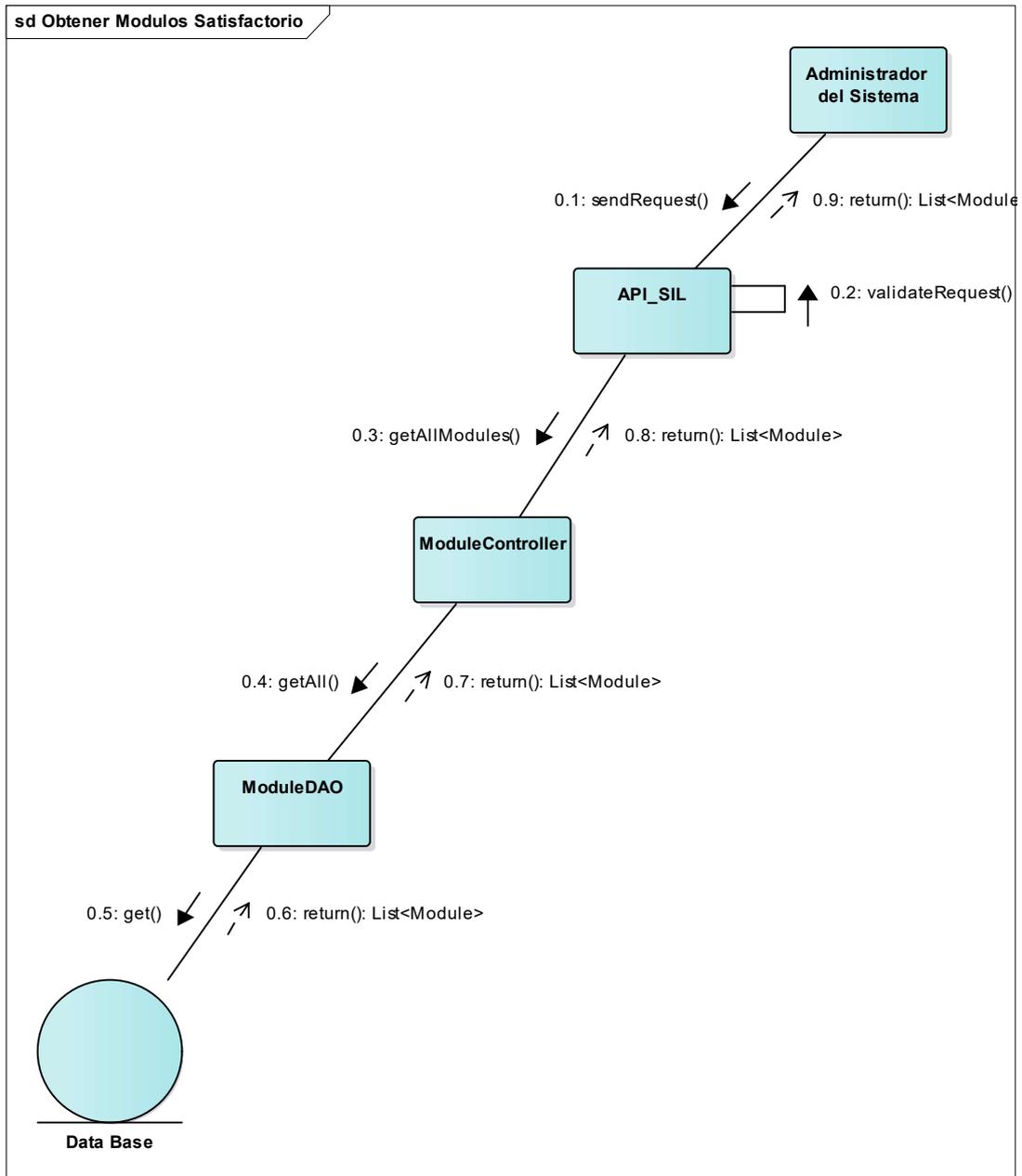
9.11.2. Actualizar Módulo de Aplicación.



9.11.3. Eliminar Módulo de Aplicación.



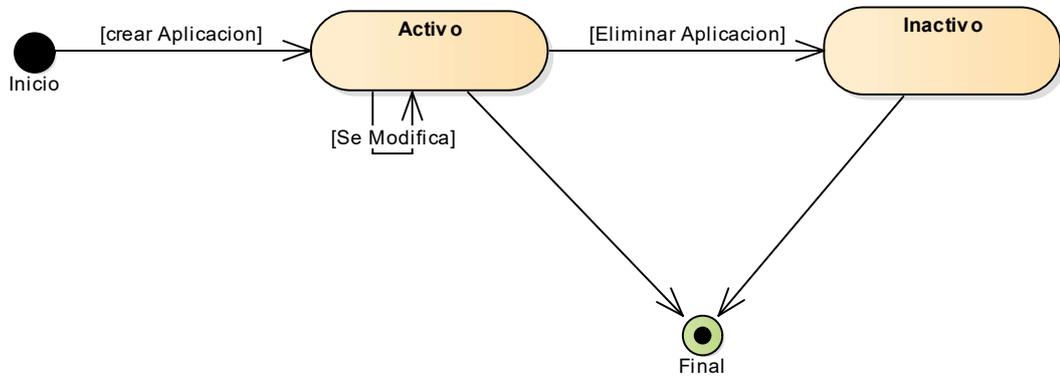
9.11.4. Obtener Modulo de Aplicación.



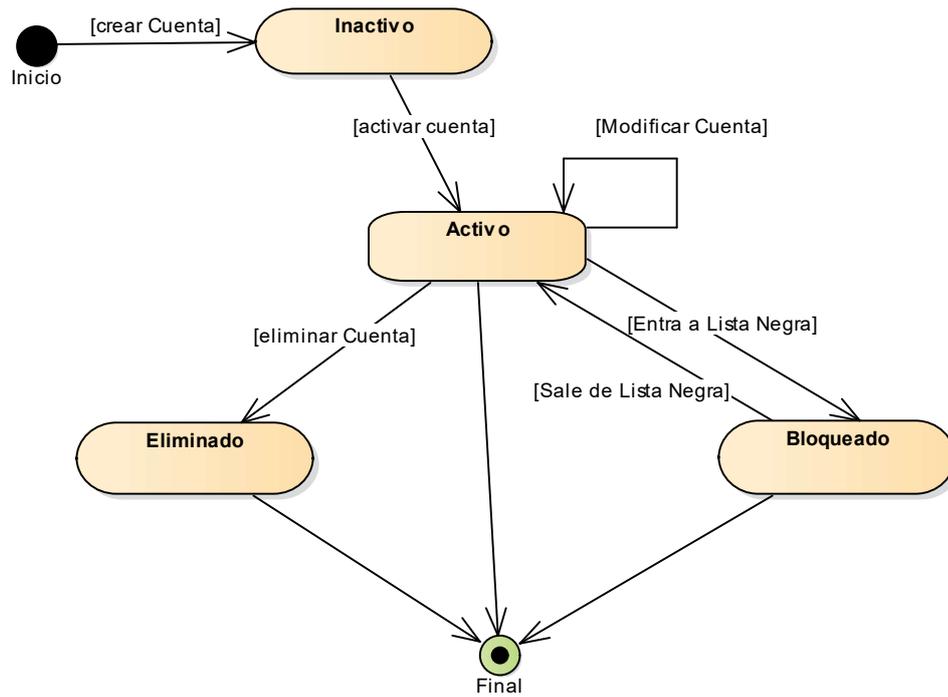
Continua en DVD: Diagramas de Comunicación.

9.12. ANEXO 12: Diagramas de Estado

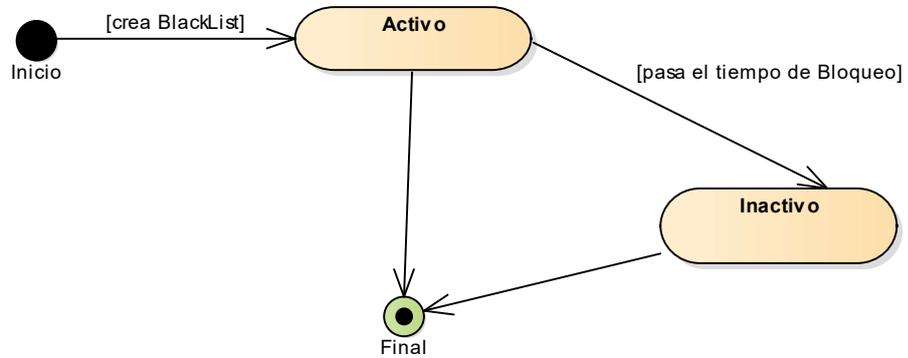
9.12.1 Diagrama de estado Aplicación



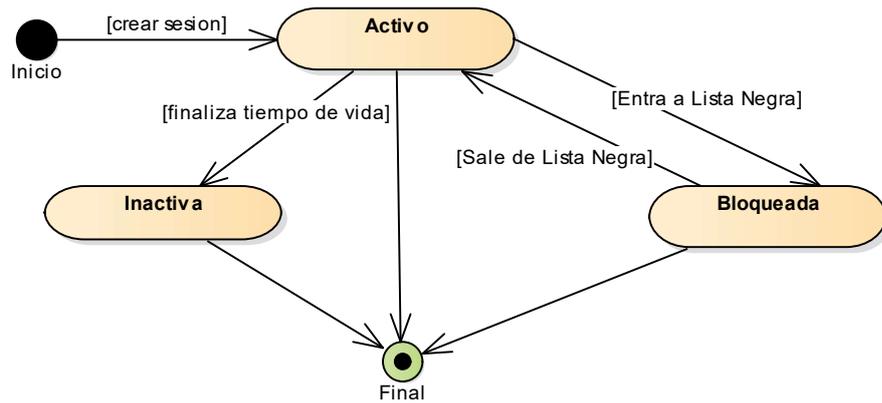
9.12.2 Diagrama de estado Cuenta



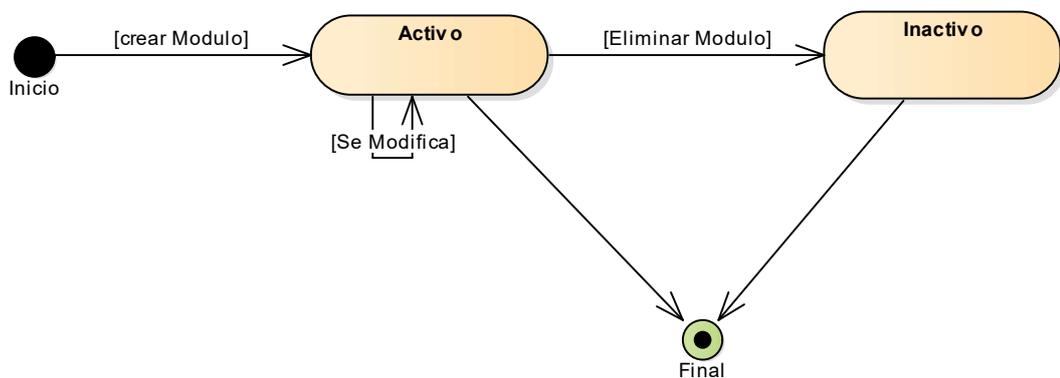
9.12.3 Diagrama de estado Lista Negra



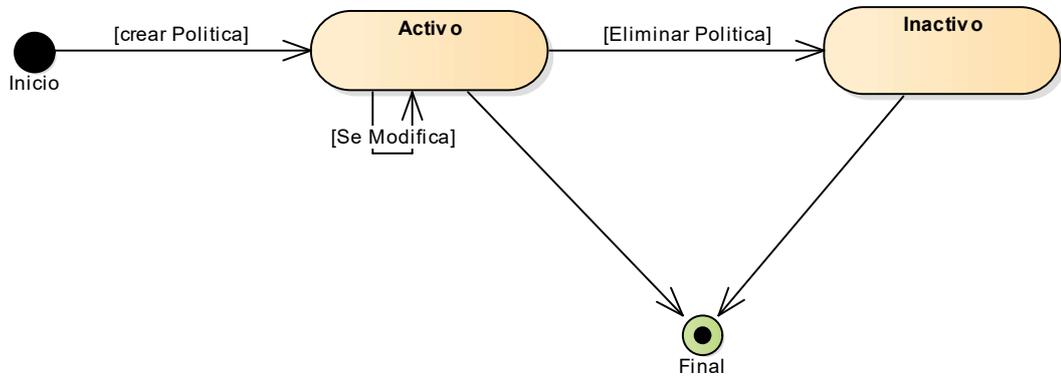
9.12.4 Diagrama de estado Sesión



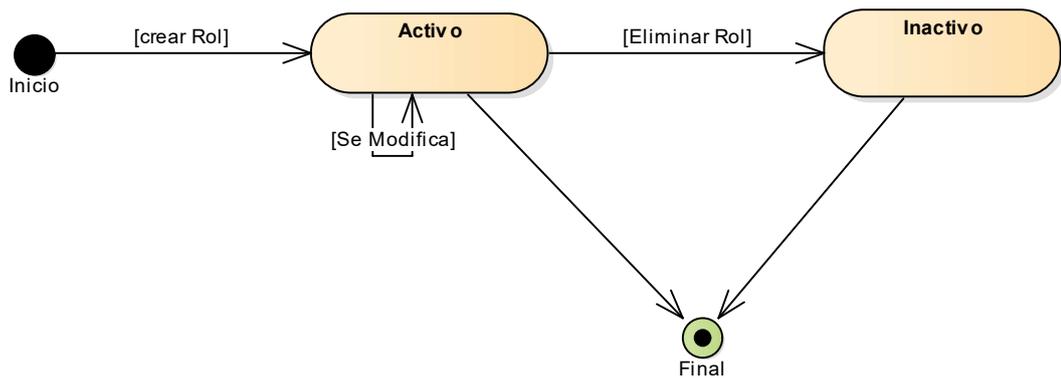
9.12.5 Diagrama de estado Modulo



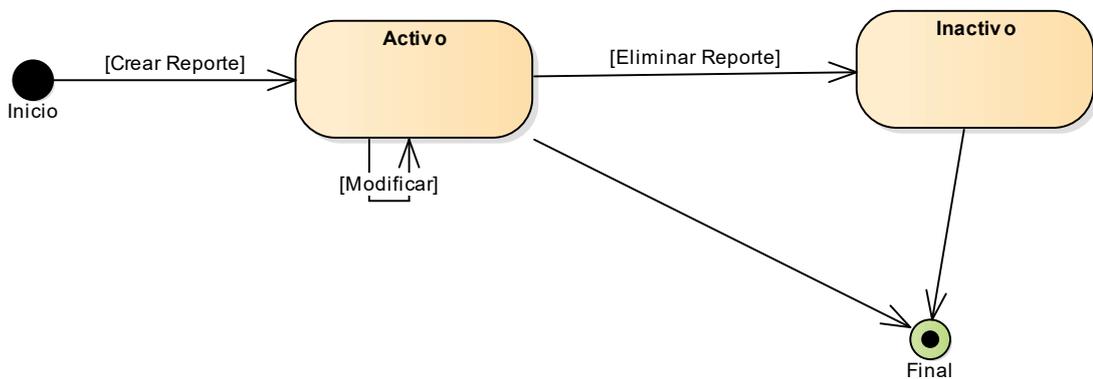
9.12.6 Diagrama de estado Política



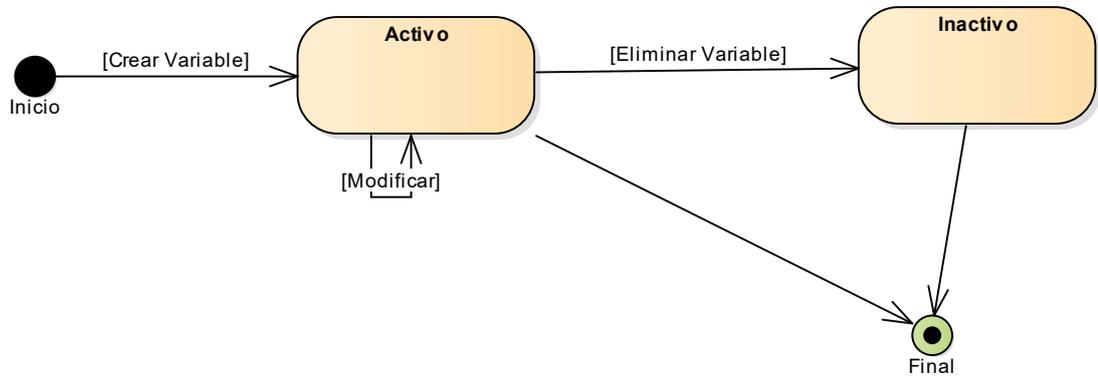
9.12.7 Diagrama de estado Rol



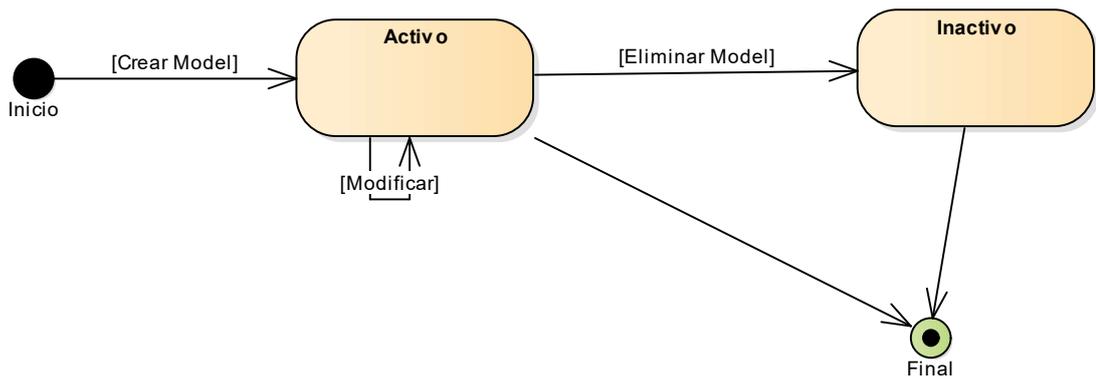
9.12.8 Diagrama de estado Reporte



9.12.9 Diagrama de estado Variable



9.12.10 Diagrama de estado Model



9.13. ANEXO 13: Manual de uso del API

CORE

API and SDK Documentation

Version: 1.0

Core Gerencial del Sistema de informacion Laboral

Aplicaciones

actualizar Aplicacion

Actualizar Aplicacion

Servicio web para actualizar una entidad de tipo aplicacion

POST

```
/updateApplication
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/updateApplication"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

crear Aplicacion

Crear Aplicacion

Servicio web para crear una entidad de tipo aplicacion

POST

```
/saveApplication
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X POST "http://192.168.2.80:9090/core/saveApplication"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

eliminar Aplicacion

Eliminar Aplicacion

Servicio web para eliminar una entidad de tipo aplicacion

DELETE

```
/deleteApplication/{id}
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X DELETE "http://192.168.2.80:9090/core/deleteApplication/{id}"
```

Parameters

Path parameters

Name	Description
id*	String <i>identificador de la entidad</i> Required

Responses

Status: 200 -

obtener Aplicaciones

Obtener Aplicaciones

Servicio web para obtener aplicaciones

GET

```
/getAllApplications
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getAllApplications"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Responses

Status: 200 -

Bitacora

obtener Bitacoras

Obtener Bitacoras

Servicio web para obtener la bitacora del sistema

GET

```
/getAllLogbooks
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getAllLogbooks"
```

Parameters

Header parameters

Name	Description
Accept*	String Required

Content-Type*	String Required
api-key*	String Required

Responses

Status: 200 -

Cuenta

login

Servicio web para Iniciar sesion en el sistema

POST

/login

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/login"
```

Parameters

Header parameters

Name	Description
api-key*	String Required

Body parameters

Name	Description
body *	▼ Login de Usuario { [] Required: email,pass email: string pass: string

Responses

Status: 200 -

new Account

Servicio web para crear una nueva cuenta de usuario

POST

/newAccount

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/newAccount"
```

Parameters

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

recover Password

Servicio web para cambiar la contraseña luego de una recuperacion

POST

```
/recoverPassword
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/recoverPassword"
```

Parameters

Header parameters

Name	Description
api-key*	String Required
Cookie*	String Required
token*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

resend Security Email

Servicio web para re-enviar correo de seguridad

POST

```
/resendSecurityEmail/{email}
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/resendSecurityEmail/{email}"
```

Parameters

Path parameters

Name	Description
email*	String Required

Header parameters

Name	Description
api-key*	String Required
Cookie*	String Required
token*	String Required

Responses

Status: 200 -

InstanciasDeModelos

delete Form Record

Servicio web para eliminar un Instancias de Modelos dinamico en especifico

POST

```
/deleteRecord/{coleccion}/{idRecord}
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/deleteRecord/{coleccion}/{idRecord}"
```

Parameters

Path parameters

Name	Description
coleccion*	String <i>Coleccion de la base de datos de la Instancias de Modelos dinamico</i> Required
idRecord*	String <i>Identificador de la Instancias de Modelos dinamico</i> Required

Header parameters

Name	Description
api-key*	String Required

Responses

Status: 200 -

get AutoGenerateFormUpdate

Servicio web para obtener un Modelo creado a partir de un Instancias de Modelos en especifico

GET

```
/autoGenerateFormHtmlUpdate/{coleccion}/{idRecord}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/autoGenerateFormHtmlUpdate/{coleccion}/{idRecord}"
```

Parameters

Path parameters

Name	Description
coleccion*	String <i>Coleccion de la base de datos de las Instancias de Modelos dinamico</i> Required
idRecord*	String <i>Identificador de la Instancias de Modelos dinamico</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 -

get File

TODO: Add Description

GET

```
/getFile/{nombreColeccion}/{idFile}
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X GET "http://192.168.2.80:9090/core/getFile/{nombreColeccion}/{idFile}"
```

Parameters

Path parameters

Name	Description
nombreColeccion*	String <i>nombre de la coleccion dinamica de la cual se obtendra el archivo</i> Required
idFile*	String <i>id del archivo que se quiere su imagen de preview</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 - archivo en bytes del Modelo en pdf

get File Preview

Servicio Web para obtener una imagen con el preview de un archivo de tipo documento en pdf

GET

```
/getFilePreview/{idFile}
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X GET "http://192.168.2.80:9090/core/getFilePreview/{idFile}"
```

Parameters

Path parameters

Name	Description
idFile*	String <i>id del archivo que se quiere su imagen de preview</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 - archivo tipo imagen en bytes

get Form Records

Servicio web para obtener las Instancias de Modelos de una determinada coleccion de la base de datos

GET

```
/getRecords/{coleccion}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getRecords/{coleccion}"
```

Parameters

Path parameters

Name	Description
coleccion*	String <i>Coleccion de la base de datos de la Instancias de Modelos dinamico</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 -

get Form Records HTML

Servicio web para obtener los Instancias de Modelos en html de una determinada coleccion , de manera paginada

GET

```
/getRecordsHtml/{coleccion}/{page}/{size}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getRecordsHtml/{coleccion}/{page}/{size}"
```

Parameters

Path parameters

Name	Description
coleccion*	String <i>Coleccion de la base de datos de la Instancias de Modelos dinamico</i> Required
page*	Integer <i>Numero de pagina a solicitar</i> Required
size*	Integer <i>Cantidad de elementos solicitados</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 -

get Reference

Servicio web para obtener los campos relacionados con otro Instancias de Modelos dinamico de la base de datos

GET

```
/getReference/{idVariable}/{variableValue}
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X GET "http://192.168.2.80:9090/core/getReference/{idVariable}/{variableValue}"
```

Parameters

Path parameters

Name	Description
idVariable*	String <i>identificador de la variable de referencia</i> Required
variableValue*	String <i>valor de la variable de referrenncia que sera utilizado para hacer referencia al registro dinamico deseado</i> Required

Header parameters

Name	Description
Cookie*	String Required

api-key*	String Required
----------	--------------------

Responses

Status: 200 -

new Form Record

Servicio web para agregar un nuevo Instancias de Modelos dinamico a la coleccion de la base de datos especificada

POST

```
/newRecord/{coleccion}
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X POST "http://192.168.2.80:9090/core/newRecord/{coleccion}"
```

Parameters

Path parameters

Name	Description
coleccion*	String <i>Coleccion de la base de datos de la Instancias de Modelos dinamico</i> Required

Header parameters

Name	Description
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

search Form Records HTML

Servicio web para buscar y obtener los registros en html y de manera paginada

GET

```
/searchRecordsHtml/{coleccion}/{search}/{page}/{size}
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X GET "http://192.168.2.80:9090/core/searchRecordsHtml/{coleccion}/{search}/{page}/{size}"
```

Parameters

Path parameters

Name	Description
coleccion*	String <i>Coleccion de la base de datos de la Instancias de Modelos dinamico</i> Required
search*	String <i>Termino buscado</i> Required
page*	Integer <i>Numero de pagina a solicitar</i> Required
size*	Integer <i>Cantidad de elementos solicitados</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 -

update Form Record

Servicio web para actualizar un Instancias de Modelos dinamico en especifico

POST

```
/updateRecord/{coleccion}/{idRecord}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/updateRecord/{coleccion}/{idRecord}"
```

Parameters

Path parameters

Name	Description
coleccion*	String <i>Coleccion de la base de datos de la Instancias de Modelos dinamico</i> Required
idRecord*	String <i>Identificador de la Instancias de Modelos dinamico</i> Required

Header parameters

Name	Description
------	-------------

api-key*	String Required
----------	--------------------

Body parameters

Name	Description
body *	

Responses

Status: 200 -

upload File

Servicio web para subir un determinado documento o imagen al sistema para su almacenamiento y disponibilidad

POST

```
/uploadFile/{fileName}
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X POST "http://192.168.2.80:9090/core/uploadFile/{fileName}"
```

Parameters

Path parameters

Name	Description
fileName*	String <i>nombre del archivo que se subira al sistema</i> Required

Header parameters

Name	Description
api-key*	String Required
Cookie*	String Required
token*	String Required

Form parameters

Name	Description
File*	File (file) Required

Responses

Status: 200 -

obtener Lista Negra

Obtener Lista Negra

Servicio web para obtener la lista negra del sistema

GET

```
/getAllBlacklists
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X GET "http://192.168.2.80:9090/core/getAllBlacklists"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Responses

Status: 200 -

Modelo

actualizar Modelo

Actualizar Modelo

Servicio web para actualizar una entidad de tipo Modelo

POST

```
/updateForm
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X POST "http://192.168.2.80:9090/core/updateForm"
```

Parameters

Header parameters

Name	Description
------	-------------

Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

crear Modelo

Crear Modelo

Servicio web para crear una entidad de tipo Modelo

POST

/saveForm

Usage and SDK Samples
[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/saveForm"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

eliminar Modelo

Eliminar Modelo

Servicio web para eliminar un Modelo

DELETE

```
/deleteForm/{id}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X DELETE "http://192.168.2.80:9090/core/deleteForm/{id}"
```

Parameters

Path parameters

Name	Description
id*	String <i>identificador de la entidad</i> Required

Responses

Status: 200 -

get AutoGenerateFormSave

Servicio web para generar la vista con los datos del Modelo y así utilizarlo para guardar el Instancias de Modelos

GET

```
/autoGenerateFormHtmlSave/{coleccion}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/autoGenerateFormHtmlSave/{coleccion}"
```

Parameters

Path parameters

Name	Description
coleccion*	String <i>Coleccion de la base de datos de la Instancias de Modelos dinamico</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 -

get Caso

Servicio web para obtener los datos generales de un caso

GET

```
/getCaso/{idCase}
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X GET "http://192.168.2.80:9090/core/getCaso/{idCase}"
```

Parameters

Path parameters

Name	Description
idCase*	String <i>id del caso que se quiere obtener la informacion</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 - { "_id": "06f3e689ab434e748b4d830305bb3d72", "idForm": "58939936b8efea1e34bfa802", "idArea": "6050082ba02d4670aa1538839afdb79f", "nombreArea": "Dirección General de Higiene y Seguridad del Trabajo", "idProcess": "71b80613f8c8463784076679f2f27070", "nombreProceso": "Tramite de Licencia en Materia de Higiene y Seguridad del Trabajo", "idEmpresa": "37e7bc68255c4fae9535763669f9390d", "razonSocial": "Dasoltex S.A", "idEmpleado": "", "nombreEmpleado": "", "estadoCaso": "Licencia aprobada (Falta entrega)", "fecha": 1479246360000, "dateAdd": 1487022448936, "dateUpdate": 1487648384963, "idUser": "394e6f6aae7844d58a200cac28768f00", "userName": "fregdytorres@hotmail.com", "recordsCount": 3, "records": "<div class='row m-0'> <section id='cd-timeline' class='cd-container'><div class='cd-timeline-block'> <div class='cd-timeline-img cd-picture'><i class='fa fa-file-o fa-lg'></i></div><div class='cd-timeline-content'> <h5 class='text-bold'>Formato de solicitud para el Tramite Licencia</h5><h6>fregdytorres@hotmail.com</h6>Ver 15-11-2016 03:59 pm </div></div><div class='cd-timeline-block'> <div class='cd-timeline-img cd-picture'><i class='fa fa-file-o fa-lg'></i></div><div class='cd-timeline-content'> <h5 class='text-bold'>Documentacion requerida para Tramite de Licencia</h5><h6>fregdytorres@hotmail.com</h6>Ver 15-11-2016 03:59 pm </div></div><div class='cd-timeline-block'> <div class='cd-timeline-img cd-picture'><i class='fa fa-file-o fa-lg'></i></div><div class='cd-timeline-content'> <h5 class='text-bold'>Acta de Inspeccion de Higiene y Seguridad del Trabajo. Sector Industria</h5><h6>fregdytorres@hotmail.com</h6>Ver 01-12-2016 09:42 am </div></div></section></div> }"

get Lista de Forms

Servicio web para obtener la lista de Modelos que el usuario tiene acceso segun sus roles

GET

```
/getListForms/{search}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getListForms/{search}"
```

Parameters

Path parameters

Name	Description
search*	String <i>retorna todos los Modelos si se envia el simbolo '\$', caso contrario se busca segun el texto introducido</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 - Tabla html con los datos cargados de la lista de Modelos obtenidos segun el parametro de busqueda

getForms

Servicio web para obtener los Modelos de un tipo en especifico

GET

```
/getForms/{formType}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getForms/{formType}"
```

Parameters

Path parameters

Name	Description
formType*	String <i>tipo de Modelo, 0 para Modelos estandar, 1 para Modelos maestros y 2 para los casos</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 -

obtener Modelo

Obtener Modelo

Servicio web para obtener todos los Modelos

GET

```
/getAllForms
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X GET "http://192.168.2.80:9090/core/getAllForms"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Responses

Status: 200 -

Modulos

actualizar Modulo

Actualizar Modulo

Servicio web para actualizar una entidad de tipo modulo

POST

```
/updateModule
```

Usage and SDK Samples

Curl Java Android Obj-C JavaScript C# PHP Perl Python

```
curl -X POST "http://192.168.2.80:9090/core/updateModule"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

crear Modulo

Crear Modulo

Servicio web para crear una entidad de tipo modulo

POST

/saveModule

Usage and SDK Samples
[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/saveModule"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

obtener Modulo

Obtener Modulo

Servicio web para obtener todos los módulos

GET

```
/getAllModules
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getAllModules"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Responses

Status: 200 -

Políticas

actualizar Política

Actualizar Política

Servicio web para actualizar una entidad de tipo Política

POST

```
/updatePolicy
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/updatePolicy"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required

api-key*	String Required
----------	--------------------

Body parameters

Name	Description
body *	

Responses

Status: 200 -

crear Politica

Crear Politica

Servicio web para crear una entidad de tipo Politica

POST

/savePolicy

Usage and SDK Samples
[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/savePolicy"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

eliminar Politica

Eliminar Politica

Servicio web para eliminar una politica

DELETE

```
/deletePolicy/{id}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X DELETE "http://192.168.2.80:9090/core/deletePolicy/{id}"
```

Parameters

Path parameters

Name	Description
id*	String <i>identificador de la entidad</i> Required

Responses

Status: 200 -

obtener Politica

Obtener Politica

Obtener todas las politicas

GET

```
/getAllPolicies
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getAllPolicies"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Responses

Status: 200 -

Reportes

actualizar Reporte

Actualizar Reporte

Servicio web para actualizar una entidad de tipo Reporte

POST

```
/updateReport
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/updateReport"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

crear Reporte

Crear Reporte

Servicio web para crear una entidad de tipo Reporte

POST

```
/saveReport
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/saveReport"
```

Parameters

Header parameters

Name	Description
Accept*	String Required

Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

eliminar Reporte

Eliminar Reporte

Servicio web para eliminar un Reporte

DELETE

/deleteReport/{id}

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X DELETE "http://192.168.2.80:9090/core/deleteReport/{id}"
```

Parameters

Path parameters

Name	Description
id*	String <i>identificador de la entidad</i> Required

Responses

Status: 200 -

generate Report Multi Operation

Servicio web para generar reportes de distintos tipos y partir de diferentes Instancias de Modelos y cruce de Variables

POST

/generateReport/{coleccion}

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/generateReport/{coleccion}"
```

Parameters

Path parameters

Name	Description
coleccion*	String <i>Coleccion de la base de datos de la Instancias de Modelos dinamico</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

get Dynamic Dashboard

Servicio web para obtener un determinado dashboard,es decir, un conjunto de reportes relacionados

GET

```
/getDashboard/{coleccion}
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getDashboard/{coleccion}"
```

Parameters

Path parameters

Name	Description
coleccion*	String <i>Coleccion de la base de datos de la Instancias de Modelos dinamico</i> Required

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 -

get Executive Dashboard

Metodo para obtener los valores de todos los indicadores de un dashboard

GET

```
/getExecutiveDashboard
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getExecutiveDashboard"
```

Parameters

Header parameters

Name	Description
Cookie*	String Required
api-key*	String Required

Responses

Status: 200 - [{"name": "Nombre del indicador", "type": 2, "results": [{"_id": "Managua", "result": 2}, {"_id": "Jinotega", "result": 1}]}] type:1 = indicador con un solo valor type:2 = indicador con muchos valores

obtener Reporte

Obtener Reporte

Servicio web para obtener los reportes del sistema

GET

```
/getAllReports
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getAllReports"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Responses

Status: 200 -

Roles

actualizar Rol

Actualizar Rol

Servicio web para actualizar una entidad de tipo Rol

POST

```
/updateRole
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/updateRole"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

crear Rol

Crear Rol

Servicio web para crear una entidad de tipo Rol

POST

```
/saveRole
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/saveRole"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

eliminar Rol

Eliminar Rol

Servicio web para eliminar un modulo

DELETE

```
/deleteModule/{id}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X DELETE "http://192,168,2,80:9090/core/deleteModule/{id}"
```

Parameters

Path parameters

Name	Description
id*	String <i>identificador de la entidad</i> Required

Responses

Status: 200 -

obtener Roles

Obtener Roles

Servicio web para obtener los roles del sistema

GET

```
/getAllRoles
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getAllRoles"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Responses

Status: 200 -

Sesiones

obtener Sesiones

Obtener Sesiones

Servicio web para obtener las sesiones del sistema

GET

```
/getAllSessions
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getAllSessions"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Responses

Status: 200 -

Variables

actualizar Variable

Actualizar Variable

Servicio web para actualizar una entidad de tipo Variable

POST

```
/updateVariable
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/updateVariable"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

crear Variable

Crear Variable

Servicio web para crear una entidad de tipo Variable

POST

```
/saveVariable
```

Usage and SDK Samples

[Curl](#) [Java](#) [Android](#) [Obj-C](#) [JavaScript](#) [C#](#) [PHP](#) [Perl](#) [Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/saveVariable"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

eliminar Variable

Eliminar Variable

Servicio web para eliminar una Variable

DELETE

```
/deleteVariable/{id}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X DELETE "http://192.168.2.80:9090/core/deleteVariable/{id}"
```

Parameters

Path parameters

Name	Description
id*	String <i>identificador de la entidad</i> Required

Responses

Status: 200 -

obtener Variable

Obtener Variable

Servicio web para obtener las Variables del sistema

GET

```
/getAllVariables
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getAllVariables"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Responses

Status: 200 -

Webservices

actualizar Servicio Web

Actualizar Servicio Web

Servicio web para actualizar una entidad de tipo Webservice

POST

```
/updateWebservice
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/updateWebservice"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
------	-------------

body *	
--------	--

Responses

Status: 200 -

crear Servicio Web

Crear Servicio Web

Servicio web para crear una entidad de tipo Webservice

POST

```
/saveWebservice
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X POST "http://192.168.2.80:9090/core/saveWebservice"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Body parameters

Name	Description
body *	

Responses

Status: 200 -

eliminar Servicio Web

Eliminar Servicio Web

Servicio web para eliminar un webservice del sistema

DELETE

```
/deleteWebservice/{id}
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X DELETE "http://192.168.2.80:9090/core/deleteWebservice/{id}"
```

Parameters

Path parameters

Name	Description
id*	String <i>identificador de la entidad</i> Required

Responses

Status: 200 -

obtener Servicio Web

Obtener Servicio Web

Servicio web para obtener los webservice del sistema

GET

```
/getAllWebservices
```

Usage and SDK Samples

[Curl](#)
[Java](#)
[Android](#)
[Obj-C](#)
[JavaScript](#)
[C#](#)
[PHP](#)
[Perl](#)
[Python](#)

```
curl -X GET "http://192.168.2.80:9090/core/getAllWebservices"
```

Parameters

Header parameters

Name	Description
Accept*	String Required
Content-Type*	String Required
api-key*	String Required

Responses

Status: 200 -

Suggestions, contact, support and error reporting;

Information URL: <https://helloreverb.com> (<https://helloreverb.com>)

Contact Info: hello@helloreverb.com (hello@helloreverb.com)

All rights reserved

<http://apache.org/licenses/LICENSE-2.0.html>

Generated 2017-10-16T09:40:55.790Z