



INSTITUTO TECNOLÓGICO
"CORDILLERA"

CARRERA DE SISTEMAS

**"CONTROL DE INDICADORES DE GESTIÓN DE CALIDAD MEDIANTE
UNA APLICACIÓN WEB PARA LA DIRECCIÓN RIPS DEL MINISTERIO
COORDINADOR DE DESARROLLO SOCIAL"**

**Proyecto de investigación previo a la obtención del título de Tecnólogo en
Análisis de Sistemas**

Autor: Fabián Ernesto Ruiz Buitrón

Tutor: Ing. Hugo Heredia

Quito, Octubre 2014

ACTA DE APROBACIÓN DEL PROYECTO DE GRADO

En mi calidad de tutor del trabajo sobre el tema: **“CONTROL DE INDICADORES DE GESTIÓN DE CALIDAD MEDIANTE APLICACIÓN WEB PARA LA DIRECCIÓN RIPS DEL MINISTERIO COORDINADOR DE DESARROLLO SOCIAL”**, presentado por el ciudadano: **Fabián Ernesto Ruiz Buitrón**, estudiante de la Escuela de Sistemas, considero que dicho informe reúne los requisitos y méritos suficientes para ser sometido a la evaluación por parte del Tribunal de Grado, que el Honorable Consejo de Escuela designe, para su correspondiente estudio y calificación.

Quito, Octubre del 2014

Ingeniero Hugo Heredia

TUTOR

Ingeniera Diana Terán

LECTOR

DECLARACIÓN DE AUTENTICIDAD

El abajo firmante, declara que los contenidos y los resultados obtenidos en el presente proyecto, como requerimiento previo para la obtención del Título de Tecnólogo Analista de Sistemas, son absolutamente originales, auténticos y personales y de exclusiva responsabilidad legal y académica de los autores.

Fabián Ernesto Ruiz Buitrón

C.C. 1704725769

APROBACIÓN DEL TRIBUNAL DEL TRABAJO DE GRADUACIÓN

Los miembros del Tribunal de Grado designado por el Honorable Concejo de la Carrera de Sistemas, aprueban el trabajo de investigación de acuerdo con las disposiciones reglamentarias emitidas por el Centro de Investigaciones Tecnológicas y Proyectos del Instituto Tecnológico Superior Cordillera” para proyectos de grado de Tecnólogos en Analistas de Sistemas: al Sr.: Fabián Ernesto Ruiz Buitrón.

Quito, noviembre de 2014

Para constancia firman:

PRESIDENTE

VOCAL1

VOCAL2

AGRADECIMIENTO

Mi agradecimiento a todos los profesores del Instituto Cordillera que, con su paciencia y dedicación, supieron ayudarme a superar todas las dificultades que conlleva retomar los estudios después de muchos años de haberlos abandonado e impulsarme a continuar hasta terminar esta etapa en mi formación profesional

Gracias

DEDICATORIA

A mis hijos por ser el motor que me
impulsa a seguir cumpliendo metas, por
su comprensión cuando tuve que
sacrificar su tiempo para asistir a clases
para lograr terminar esta etapa que hoy
culmina.

A mis padres por su apoyo incondicional
para demostrarme que para iniciar el
camino hacia cumplir con un sueño, no
se necesita nada que más que tomar la
decisión y tener la voluntad de continuar
sin importar las dificultades que se
encuentre.

Fabián Ernesto Ruiz Buitrón

CONTRATO DE CESIÓN SOBRE DERECHOS PROPIEDAD INTELECTUAL

Comparecen a la celebración del presente contrato de cesión y transferencia de derechos de propiedad intelectual, por una parte, el estudiante Fabián Ernesto Ruiz Buitrón, por sus propios y personales derechos, a quien en lo posterior se le denominará el “CEDENTE”; y, por otra parte, el INSTITUTO SUPERIOR TECNOLÓGICO CORDILLERA, representado por su Rector el Ingeniero Ernesto Flores Córdova, a quien en lo posterior se lo denominará el “CESIONARIO”. Los comparecientes son mayores de edad, domiciliados en esta ciudad de Quito Distrito Metropolitano, hábiles y capaces para contraer derechos y obligaciones, quienes acuerdan al tenor de las siguientes cláusulas:

PRIMERA: ANTECEDENTE.- a) El Cedente dentro del pensum de estudio en la carrera de análisis de sistemas que imparte el Instituto Superior Tecnológico Cordillera, y con el objeto de obtener el título de Tecnólogo en Análisis de Sistemas, el estudiante participa en el proyecto de grado denominado “CONTROL DE INDICADORES DE GESTIÓN DE CALIDAD MEDIANTE UNA APLICACIÓN WEB PARA LA DIRECCIÓN RIPS DEL MINISTERIO COORDINADOR DE DESARROLLO SOCIAL”, el cual incluye la creación y desarrollo del programa de ordenador o software, para lo cual ha implementado los conocimientos adquiridos en su calidad de alumno. b) Por iniciativa y responsabilidad del Instituto Superior Tecnológico Cordillera se desarrolla la creación del programa de ordenador, motivo por el cual se regula de forma clara la cesión de los derechos de autor que genera la obra literaria y que es producto del proyecto de grado, el mismo que culminado es de plena aplicación técnica, administrativa y de reproducción.

SEGUNDA: CESIÓN Y TRANSFERENCIA.- Con el antecedente indicado, el Cedente libre y voluntariamente cede y transfiere de manera perpetua y gratuita

todos los derechos patrimoniales del programa de ordenador descrito en la cláusula anterior a favor del Cesionario, sin reservarse para sí ningún privilegio especial (código fuente, código objeto, diagramas de flujo, planos, manuales de uso, etc.). El Cesionario podrá explotar el programa de ordenador por cualquier medio o procedimiento tal cual lo establece el Artículo 20 de la Ley de Propiedad Intelectual, esto es, realizar, autorizar o prohibir, entre otros: a) La reproducción del programa de ordenador por cualquier forma o procedimiento; b) La comunicación pública del software; c) La distribución pública de ejemplares o copias, la comercialización, arrendamiento o alquiler del programa de ordenador; d) Cualquier transformación o modificación del programa de ordenador; e) La protección y registro en el IEPI el programa de ordenador a nombre del Cesionario; f) Ejercer la protección jurídica del programa de ordenador; g) Los demás derechos establecidos en la Ley de Propiedad Intelectual y otros cuerpos legales que normen sobre la cesión de derechos de autor y derechos patrimoniales.

TERCERA: OBLIGACIÓN DEL CEDENTE.- El cedente no podrá transferir a ningún tercero los derechos que conforman la estructura, secuencia y organización del programa de ordenador que es objeto del presente contrato, como tampoco emplearlo o utilizarlo a título personal, ya que siempre se deberá guardar la exclusividad del programa de ordenador a favor del Cesionario.

CUARTA: CUANTÍA.- La cesión objeto del presente contrato, se realiza a título gratuito y por ende el Cesionario ni sus administradores deben cancelar valor alguno o regalías por este contrato y por los derechos que se derivan del mismo.

QUINTA: PLAZO.- La vigencia del presente contrato es indefinida.

SEXTA: DOMICILIO, JURISDICCIÓN Y COMPETENCIA.- Las partes fijan como su domicilio la ciudad de Quito. Toda controversia o diferencia derivada de éste, será resuelta directamente entre las partes y, si esto no fuere factible, se

solicitará la asistencia de un Mediador del Centro de Arbitraje y Mediación de la Cámara de Comercio de Quito. En el evento que el conflicto no fuere resuelto mediante este procedimiento, en el plazo de diez días calendario desde su inicio, pudiendo prorrogarse por mutuo acuerdo este plazo, las partes someterán sus controversias a la resolución de un árbitro, que se sujetará a lo dispuesto en la Ley de Arbitraje y Mediación, al Reglamento del Centro de Arbitraje y Mediación de la Cámara de comercio de Quito, y a las siguientes normas: a) El árbitro será seleccionado conforme a lo establecido en la Ley de Arbitraje y Mediación; b) Las partes renuncian a la jurisdicción ordinaria, se obligan a acatar el laudo arbitral y se comprometen a no interponer ningún tipo de recurso en contra del laudo arbitral; c) Para la ejecución de medidas cautelares, el árbitro está facultado para solicitar el auxilio de los funcionarios públicos, judiciales, policiales y administrativos, sin que sea necesario recurrir a juez ordinario alguno; d) El procedimiento será confidencial y en derecho; e) El lugar de arbitraje serán las instalaciones del centro de arbitraje y mediación de la Cámara de Comercio de Quito; f) El idioma del arbitraje será el español; y, g) La reconvencción, caso de haberla, seguirá los mismos procedimientos antes indicados para el juicio principal.

SÉPTIMA: ACEPTACIÓN.- Las partes contratantes aceptan el contenido del presente contrato, por ser hecho en seguridad de sus respectivos intereses.

En aceptación firman a los 27 días del mes de Octubre del dos mil catorce.

f) _____

C.C. N° 1704725769

CEDENTE

f) _____

Instituto Superior Tecnológico Cordillera

CESIONARIO

ÍNDICE GENERAL

Contenido	Página
Acta de Aprobación del Proyecto de Grado.....	ii
Declaración de Autenticidad	iii
Aprobación del Tribunal del Trabajo de Graduación	iv
Agradecimiento	v
Dedicatoria	vi
Contrato de Cesión Sobre Derechos Propiedad Intelectual	vii
Índice General	x
Índice de Tablas	xiii
Índice de Figuras	xv
Índice de Anexos.....	xvii
Capítulo I: Antecedentes	1
1.01 Contexto	1
1.02 Justificación.....	1
1.03 Definición del Problema Central (Matriz T)	2
Capítulo II: Análisis de Involucrados	3
2.01 Requerimientos.....	3
2.01.01 Descripción del Sistema Actual	3
2.01.02 Visión y Alcance	4
2.01.03 Entrevistas	6
2.01.04 Matriz de Requerimientos	7
2.01.05 Descripción Detallada	7
2.02 Mapeo de Involucrados	17
2.03 Matriz de Involucrados.....	18

Capítulo III: Problemas y Objetivos	19
3.01 Árbol de Problemas	19
3.02 Árbol de Objetivos	20
3.03 Diagramas de Casos de Uso	21
3.04 Casos de Uso de Realización	21
3.05 Diagrama de Secuencias del Sistema.	26
3.06 Especificación de Casos de Uso	27
3.06.01 Actualizar Formatos	27
3.06.02 Elaborar Informes	28
3.06.03 Consultar Indicadores	29
3.06.04 Revisar Indicadores.....	30
Capítulo IV Análisis de Alternativas	32
4.01 Matriz de Análisis de Alternativas.	32
4.02 Matriz de Impactos de Objetivos	33
4.03 Estándares para el Diseño de Clases	33
4.04 Diagrama de Clases	33
4.05 Modelo Lógico - Físico	35
4.06 Diagrama de Componentes	35
4.07 Diagramas de Estrategias.	36
4.08 Matriz de Marco Lógico.....	37
4.09 Vistas Arquitectónicas.....	37
Capítulo V: Propuesta	45
5.01 Especificación de Estándares de Programación	45
5.02 Diseño de Interfaces de Usuario.....	45
5.03 Especificación de Pruebas de Unidad.	47
5.04 Especificación de Pruebas de Aceptación.....	52

5.05	Especificación de Pruebas de Carga.....	54
5.06	Configuración del Ambiente Mínimo/Ideal.	56
Capítulo VI: Aspectos Administrativos		64
6.01	Recursos.	64
6.01.01	Recursos Humanos.....	64
6.01.02	Recursos Materiales	65
6.01.03	Recursos Tecnológicos.....	65
6.02	Presupuesto.....	66
6.03	Cronograma.....	66
Capítulo VII: Conclusiones y Recomendaciones.....		67
7.01	Conclusiones.	67
7.02	Recomendaciones.....	68
Bibliografía		69
Anexos		71

Índice de Tablas

Tabla 1 Entrevista EN-002.....	7
Tabla 2 Descripción de Requerimiento RQF001	8
Tabla 3 Descripción de Requerimiento RQF002	9
Tabla 4 Descripción de Requerimiento RQF003	10
Tabla 5 Descripción de Requerimiento RQF004	11
Tabla 6 Descripción de Requerimiento RQF005	12
Tabla 7 Descripción de Requerimiento RQF006	13
Tabla 8 Descripción de Requerimiento RQF007	14
Tabla 9 Descripción de Requerimiento RQF008	15
Tabla 10 Descripción de Requerimiento RQF009	16
Tabla 11 Descripción de Requerimiento RQF010	17
Tabla 12 Caso de Realización DCR001. Actualizar Formatos	22
Tabla 13 Caso de Realización DCR002 - Elaborar Informes	24
Tabla 14 Caso de Realización DCR003 - Consultar Informes	25
Tabla 15 Caso de Realización DCR004 - Revisar Indicadores.....	26
Tabla 16 Especificación Actualizar Formatos CU-001	27
Tabla 17 Flujo Básico Actualizar Formatos CU-001.....	28
Tabla 18 Especificación Elaborar Informes CU-002	28
Tabla 19 Flujo Básico Elaborar Informes CU-002	29
Tabla 20 Especificación Consultar Indicadores CU-003	29
Tabla 21 Flujo Básico Consultar Indicadores CU-003	30
Tabla 22 Especificación Revisar Indicadores CU-004	30
Tabla 23 Flujo Básico Revisar Indicadores CU-004	31
Tabla 24 Matriz de Análisis de Alternativas.....	32

Tabla 25 Componentes Diseño Interfaces.....	46
Tabla 26 Prueba de Unidad Ingreso Sistema	47
Tabla 27 Prueba de Unidad Mensajes Error	48
Tabla 28 Prueba de Unidad Módulo Principal.....	48
Tabla 29 Prueba de Unidad Mantenimiento Estado.....	48
Tabla 30 Prueba de Unidad Responsables de Procesos	49
Tabla 31 Prueba de Unidad Personal Rips.....	49
Tabla 32 Prueba de Unidad Lista Maestra Registros.....	50
Tabla 33 Prueba de Unidad Ingreso Indicadores	50
Tabla 34 Prueba de Unidad Reportes Personal Rips.....	51
Tabla 35 Prueba de Unidad Consulta Indicadores	51
Tabla 36 Prueba Aceptación PA001 Actualizar Formatos	52
Tabla 37 Prueba Aceptación PA002 Elaborar Informes.....	53
Tabla 38 Prueba Aceptación PA003 Consultar Informes	54
Tabla 39 Prueba de Carga PC001	55
Tabla 40 Prueba de Carga PC002	55
Tabla 41 Prueba de Carga PC003	56
Tabla 42 Variables de Ambiente Linux	60
Tabla 43 Variables de Ambiente Windows	61
Tabla 44 Recursos Humanos.....	64
Tabla 45 Recursos Materiales	65
Tabla 46 Recursos Tecnológicos	65
Tabla 47 Presupuesto	66

Índice de Figuras

<i>Figura 1</i> Mapa de Procesos	3
<i>Figura 2</i> Árbol de Problemas.....	19
<i>Figura 3</i> Árbol de Objetivos	20
<i>Figura 4</i> Diagrama de Caso de Uso.....	21
<i>Figura 5</i> Diagrama Realización - Actualizar Formatos	23
<i>Figura 6</i> Diagrama Realización - Elaborar Informes	23
<i>Figura 7</i> Diagrama Realización - Consultar Informes.....	24
<i>Figura 8</i> Diagrama Realización - Revisar Indicadores.....	25
<i>Figura 9</i> Actualizar Formatos - Diagrama Secuencia.....	26
<i>Figura 10</i> Diagrama de Clases.....	34
<i>Figura 11</i> Diagrama de Componentes	35
<i>Figura 12</i> Diagrama de Estrategias.....	36
<i>Figura 13</i> Arquitectura - Vista Lógica.....	38
<i>Figura 14</i> Arquitectura por Capas	38
<i>Figura 15</i> Arquitectura - Vista Física	39
<i>Figura 16</i> Vista de Desarrollo Ripssgcenterprice	39
<i>Figura 17</i> Modelo Vista de Desarrollo Ripssgcenterprice	40
<i>Figura 18</i> Vista de Desarrollo Ripssgcejb	40
<i>Figura 19</i> Modelo Vista de Desarrollo Ripssgcejb.....	41
<i>Figura 20</i> Vista de Desarrollo Ripssgweb	41
<i>Figura 21</i> Modelo Vista de Desarrollo Ripssgweb.....	42
<i>Figura 22</i> Vista de Procesos	44
<i>Figura 23</i> Diseño de Interfaces.....	46
<i>Figura 24</i> Diseño Interfaz Pantalla Ingreso	47
<i>Figura 25</i> Variable de Ambiente Sistema.....	62

Figura 26 Interfaz Administrador Aplicaciones	62
---	----

Índice de Anexos

A.01 Matriz de Marco Lógico	72
A.02 Entrevista EN-001	73
A.03 Java Code Conventions	74
A.04 Matriz T.....	74
A.05 Mapeo de Involucrados	74
A.06 Matriz de Involucrados.....	74
A.07 Matriz de Requerimientos	74
A.08 Matriz de Impacto de Objetivos	74
A.09 Diagrama de Secuencia - Elaborar Informe	74
A.10 Modelo Lógico de Base de Datos	74
A.11 Modelo Físico de Base de Datos	74
A.12 Diagrama de Objetos Orientado a Modelo de Base de Datos	74
A.13 Estado Actual de Servidores	74
A.14 Cronograma.....	74
A.15 Manual de Instalación.....	116
A.16 Manual Técnico	135
A.17 Manual de Usuario.....	150

Capítulo I: Antecedentes

1.01 Contexto

La Dirección del Registro Interconectado de Programas Sociales (RIPS), parte del Ministerio de Coordinación de Desarrollo Social, es la encargada de dirigir, articular y coordinar el Registro Interconectado de Programas Sociales mediante el diseño de sistemas, procesos y herramientas tecnológicas que permitan crear, actualizar y mantener bases de datos interconectadas de beneficiarios de programas sociales a nivel interinstitucional e individual para direccionar la planificación de los diferentes proyectos y programas sociales.

Para cumplir con su objetivo tiene definidas metas que tienen que ser alcanzadas en los tiempos planificados, por lo que se vuelve crítico determinar estándares que enmarquen sus acciones, comprobar los resultados obtenidos con respecto a los planteados, tomar decisiones acertadas cuando los resultados no son los esperados, poner en marcha sistemas de control que emitan alertas tempranas para implementar acciones correctivas.

1.02 Justificación

La Dirección del Registro Integrado de Programas Sociales, una de las direcciones técnicas del Ministerio Coordinador de Desarrollo Social, ante la necesidad de definir y documentar sus procesos, determinar y controlar indicadores adecuados que permitan el monitoreo de lo planificado y realizar una mejora continua de todos sus campos de acción, implementó un Sistema de Gestión de Calidad (SGC) bajo la Norma ISO 9001:2008 a través de una empresa consultora.

En cumplimiento de esta implementación se definieron y establecieron todas las áreas, actividades y responsabilidades y se documentaron todos los procesos que se ejecutan para su actividad diaria. Un punto débil del sistema es que los sistemas de registro, control y monitoreo de documentos e indicadores se realiza de forma manual, lo que no está de acuerdo a la naturaleza misma de una Dirección eminentemente técnica, con estrecha relación con la tecnología y que cuenta con un área de desarrollo, pero por el amplio campo de acción que tienen que satisfacer, no está entre sus prioridades el desarrollo de ninguna aplicación que fortalezca el Sistema de Gestión de Calidad (SGC).

El sistema de Gestión de la Calidad debe ser mantenido y mejorado como es su naturaleza y, ante esta circunstancia, se crea la necesidad de que este vacío de control existente y que tiene enorme importancia para el seguimiento y control de la eficiencia de la gestión de la Dirección deba ser satisfecho buscando facilitar el manejo del ingreso, monitoreo y planificación de todas las actividades de la Dirección a través de los indicadores de gestión y ese es el objetivo del proyecto.

1.03 Definición del Problema Central (Matriz T)

El Sistema de Gestión de Calidad se lleva de forma manual para elaboración de informes, recopilación de indicadores, subida de indicadores a los formatos establecidos en el SGC, reportes de indicadores, realizar depósito de archivos de los informes en los repositorios determinados en un servidor, lo que implica demanda de tiempo y dificultad en el control de versionamiento de formatos establecidos y utilizados, y el riesgo de pérdida de información principalmente de los repositorios que están disponibles para los usuarios. (**Ver Anexo A.04. Matriz T**)

Capítulo II: Análisis de Involucrados

2.01 Requerimientos

2.01.01 Descripción del Sistema Actual

Los Procesos establecidos y que inciden directamente en el desarrollo del proyecto constan en el Mapa de Procesos de la Dirección RIPS.

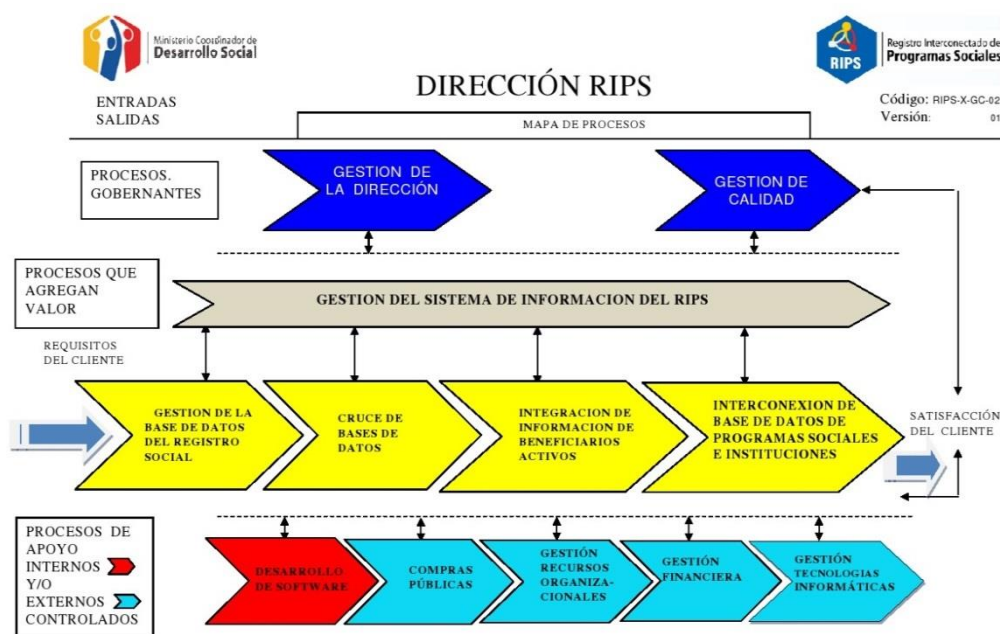


Figura 1

Mapa de Procesos

Nota: Describe la estructura actual del sistema establecida para el funcionamiento de la Dirección RIPS enmarcada en un Sistema de Gestión de Calidad.

El sistema utilizado actualmente para el manejo de reportes e indicadores es Manual. Los formatos incluidos en la Lista Maestra de Registros del Sistema de Gestión de calidad están diseñados por los responsables de procesos en hojas de cálculo, son aprobados por el Comité de Gestión de Calidad, se ingresan para control de uso y

versionamiento en la Lista Maestra de Registros y luego que son utilizados por las personas responsables para llenarlos con los indicadores y pasar a la persona que realiza la función de Administrador del Sistema de Gestión de Calidad, quien lleva el registro de indicadores y emite reportes para la Dirección.

Los formatos están a disposición o en poder de los usuarios lo que dificulta demasiado el control de que no se realicen cambios no registrados a los formatos aprobados y por consiguiente tampoco son incluidos en el SGC, existiendo alto riesgo inclusive de que se estén utilizando versiones obsoletas.

Luego de que estos formatos se llenan con los indicadores son depositados en repositorios que están en un servidor a disposición de los usuarios con el consiguiente riesgo de manipulación posterior o pérdida de la información.

2.01.02 Visión y Alcance

El sistema debe facilitar el manejo de los documentos controlados por el Sistema de Gestión de Calidad, evitando el uso de versiones obsoletas o no autorizadas de los formatos de registros, simplificando el ingreso y consulta de los indicadores de gestión por los responsables establecidos y eliminando el proceso manual que está definido.

Debe contener las versiones actualizadas de los formatos válidos que están registrados en la Lista Maestra de Registros del Sistema de Gestión de Calidad y que son requeridos para el ingreso de los informes de indicadores por los responsables de

los procesos de la Dirección, debiendo mantenerse actualizados los parámetros tanto para seguridades como para completar los formatos establecidos.

Los indicadores ingresados deben ser de fácil acceso cuando el usuario lo requiera pero evitando el riesgo de manipulación o pérdida de información.

El Avance deberá ser medido en base a los entregables definidos en el cronograma aprobado.

Entre los involucrados están:

Procesos de Apoyo:

Área Financiera

Área Administrativa

Área Tecnológica

Todos los responsables directos de los procesos de la Dirección RIPS:

Director(a)

Asistente de Dirección

Administrador Interconexión de Bases de Datos

Técnicos Administración Bases de Datos

Administrador del Sistema de Información.

Encargado Infraestructura

Equipo de Desarrollo.

También están involucradas todas aquellos usuarios (personas e instituciones) que de una u otra manera perciben la calidad de la información que la Dirección RIPS procesa y que sirve de insumo para la toma de decisiones sobre políticas de alcance social y también para la selección de posibles beneficiarios de los Programas Sociales del Gobierno.

Dirección del Registro Social.

Dirección del SIISE.

Dirección de Talento Humano

Ministerio de Inclusión Económica y Social.

Ministerio de Salud Pública.

Secretaría Técnica de Discapacidades.

Corporación Nacional de Finanzas Populares y Solidarias.

Ministerio de Educación.

Público en General

2.01.03 Entrevistas

Entrevista con identificador EN-001 realizada para levantamiento de requerimientos del proyecto. **(Ver Anexo A.02. Entrevista EN-001)**

La siguiente tabla recoge la segunda entrevista para levantamiento de requerimientos.

Tabla 1

Entrevista EN-002

ENTREVISTA		
Identificador: EN-002		
Preguntas	Objetivos	Análisis Posterior
De dónde se obtienen los datos fuentes para completar los campos de los formatos de informes, principalmente de los formatos que serán utilizados para ingresar los indicadores de varios procesos de la Dirección RIPS	Determinar una sola fuente que se encargue del mantenimiento de los parámetros necesarios para completar los formatos.	Se determina que la persona que Administra del Sistema de Gestión de Calidad será la responsable de mantener actualizados los parámetros necesarios para el funcionamiento del sistema.
La forma de presentación del sistema en su Menú Principal, será estándar para todos los usuarios o dependerá de los permisos permitidos para cada uno.	Definir si al existir diferentes roles en los usuarios es necesario tomar medidas para modificar a discreción de la persona autorizada, la presentación del Menú Principal del sistema.	Queda claro que es necesario un menú dinámico que permita asignar los módulos del sistema libremente según los roles creados, y que estas modificaciones debe poder realizarlo un usuario autorizado sin recurrir al programador.

2.01.04 Matriz de Requerimientos

Matriz que detalla todos los requerimientos realizados por el Usuario.(**Ver Anexo A.07. Matriz de Requerimientos**).

2.01.05 Descripción Detallada

Descripción detallada levantamiento de requerimientos RQF001 – Datos de Persona

Tabla 2 Descripción de Requerimiento RQF001

Descripción de Requerimiento RQF001

Descripción del requerimiento.	Estado	Diseño	
El sistema debe permitir la creación y mantenimiento de los usuarios.			
Creado por	Administrador del Sistema de Gestión de Calidad	Actualizado por	Administrador del Sistema de Gestión de Calidad
Fecha Creación	02/05/2014	Fecha de Actualización	12/06/2014
identificador		RQF001	
Tipo de Requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada		Datos de Persona (Cédula, nombres, apellidos, dirección, teléfono convencional, teléfono fijo, email)	
Descripción		De la lista de personas se selecciona a los usuarios del sistema, generando nombre de usuario y contraseña para ingreso y asignación de permisos.	
Datos de salida		Notificación que ya se ejecutó el ingreso o cambio de los datos de personas registradas y de los usuarios del sistema creados.	
Resultados Esperados		Ingreso de datos iniciales para el control de acceso de usuarios al sistema, reforzando la seguridad del mismo.	
Origen		Requerimiento del Usuario (Administrador del Sistema de Gestión de Calidad)	
Dirigido a		Dirección del RIPS. Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuarios del sistema autorizados.	
Prioridad		5	
Requerimientos Asociados		RQF002 RQF009	
ESPECIFICACIÓN			
Precondiciones		N/A	
Pos condiciones		No se debe poder realizar el ingreso duplicado de datos de la misma persona y se debe validar el ingreso solo de cédulas válidas.	
Criterios de Aceptación		Los datos del requerimiento son básicos para el funcionamiento y control de seguridad de acceso a la información por parte de los usuarios del sistema.	

Descripción detallada de levantamiento de requerimientos RQF002 – Permiso de

Acceso a Módulos por usuario

Tabla 3 Descripción de Requerimiento RQF002

Descripción de Requerimiento RQF002

Descripción del requerimiento.		Estado	Diseño
El Sistema debe permitir otorgar o retirar permisos de acceso para realizar actividades específicas según sea autorizado a cada usuario del sistema.			
Creado por	Administrador del Sistema de Gestión de Calidad	Actualizado por	Administrador del Sistema de Gestión de Calidad
Fecha Creación	02/05/2014	Fecha de Actualización	13/06/2014
identificador		RQF002	
Tipo de Requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada		Módulos que puede acceder un usuario dependiendo de los permisos que se le conceden para el cumplimiento de sus actividades.	
Descripción		Conceder permisos a los módulos que se autoriza acceder a cada usuario del sistema. Estos módulos son los únicos que debe poder acceder el usuario a través de la aplicación	
Datos de salida		Notificación de ingreso o cambio de los permisos concedidos.	
Resultados Esperados		Los módulos que un usuario pueda acceder a través de la aplicación serán exclusivamente los que se autorizaron permisos.	
Origen		Requerimiento del Usuario (Administrador del Sistema de Gestión de Calidad)	
Dirigido a		Dirección del RIPS. Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuarios del sistema autorizados.	
Prioridad		5	
Requerimientos Asociados		RQF001 RQF009	
ESPECIFICACIÓN			
Precondiciones		N/A	
Pos condiciones		El usuario del sistema no se debe poder acceder a ningún módulo que no se le haya autorizado.	
Criterios de Aceptación		Los datos del requerimiento son básicos para el funcionamiento y control de seguridad de acceso a la información por parte de los usuarios del sistema.	

Nota: Descripción detallada de levantamiento de requerimientos RQF003
Presentación de Módulos

Tabla 4 Descripción de Requerimiento RQF003

Descripción de Requerimiento RQF003

Descripción del requerimiento.		Estado	Diseño
El sistema debe permitir al usuario designado, modificar el nombre y el orden de cada uno de los módulos que lo componen.			
Creado por	Administrador del Sistema de Gestión de Calidad	Actualizado por	Administrador del Sistema de Gestión de Calidad
Fecha Creación	02/05/2014	Fecha de Actualización	13/06/2014
identificador		RQF003	
Tipo de Requerimiento	No crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	Nombre y orden como se deben presentar los módulos en la aplicación.		
Descripción	Renombrar y ordenar la presentación de los módulos en la aplicación.		
Datos de salida	Notificación de los cambios realizados.		
Resultados Esperados	Los módulos que un usuario pueda acceder a través de la aplicación se presentarán con los nombres y en el orden establecido.		
Origen	Requerimiento del Usuario (Administrador del Sistema de Gestión de Calidad)		
Dirigido a	Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuarios del sistema autorizados.		
Prioridad	3		
Requerimientos Asociados	N/A		
ESPECIFICACIÓN			
Precondiciones	N/A		
Pos condiciones	El usuario del sistema accederá a la aplicación con los nombres de los módulos y en el orden determinado.		
Criterios de Aceptación	Los datos del requerimiento facilitan el uso de la aplicación por parte del usuario.		

Nota: Descripción detallada de levantamiento de requerimientos RQF004 – Lista Maestra de Registros.

Tabla 5 Descripción de Requerimiento RQF004

Descripción de Requerimiento RQF004

Descripción del requerimiento.		Estado	Diseño
El sistema debe contemplar el ingreso y mantenimiento de la Lista Maestra de Registros del SGC.			
Creado por	Administrador del Sistema de Gestión de Calidad	Actualizado por	Administrador del Sistema de Gestión de Calidad
Fecha Creación	02/05/2014	Fecha de Actualización	13/06/2014
identificador		RQF004	
Tipo de Requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	Datos que constan en la Lista Maestra de Registros del Sistema de Gestión de Calidad.		
Descripción	Asegurar uso de formatos controlados para ingreso de indicadores del Sistema de Gestión de Calidad.		
Datos de salida	Confirmación de ingreso o modificación de datos en la Lista Maestra de Registros del SGC.		
Resultados Esperados	Control de formatos de registros en el SGC.		
Origen	Requerimiento del Usuario (Administrador del Sistema de Gestión de Calidad)		
Dirigido a	Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuarios del sistema autorizados.		
Prioridad	5		
Requerimientos Asociados	RQF005		
ESPECIFICACIÓN			
Precondiciones	N/A		
Pos condiciones	El usuario del sistema accederá únicamente a los formatos y versiones válidas del Sistema de Gestión de Calidad.		
Criterios de Aceptación	Los datos del requerimiento son básicos para el funcionamiento del sistema.		

Descripción detallada de levantamiento de requerimientos RQF005 – Datos de

Entrada de Formato

Tabla 6 Descripción de Requerimiento RQF005

Descripción de Requerimiento RQF005

Descripción del requerimiento.	Estado	Diseño	
El sistema debe contar con los formatos de indicadores establecidos en el Sistema de Gestión de Calidad del RIPS contenidos en la Lista Maestra de Registros.			
Creado por	Administrador del Sistema de Gestión de Calidad	Actualizado por	Administrador del Sistema de Gestión de Calidad
Fecha Creación	02/05/2014	Fecha de Actualización	13/06/2014
identificador		RQF005	
Tipo de Requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada		Los datos de entrada dependerán del formato de indicador utilizado (Código y versión del formato, tipo de indicador responsable del proceso, fecha, componente, valor y periodo del indicador, observaciones, función, nombre y firma de la persona que la realiza).	
Descripción		Ingreso de indicadores de acuerdo al formato y versión seleccionado.	
Datos de salida		Confirmación de ingreso o modificación de datos del indicador.	
Resultados Esperados		Control de versionamiento de los formatos de indicadores del Sistema de Gestión de Calidad.	
Origen		Requerimiento del Usuario (Administrador del Sistema de Gestión de Calidad)	
Dirigido a		Administrador del Sistema de Gestión de Calidad.	
		Área de desarrollo.	
		Usuarios del sistema autorizados.	
Prioridad		5	
Requerimientos Asociados		RQF004 RQF006 RQF007	
ESPECIFICACIÓN			
Precondiciones		N/A	
Pos condiciones		El usuario del sistema accederá únicamente a los formatos y versiones válidas del Sistema de Gestión de Calidad. Los indicadores ingresados se deberán poder recuperar manteniendo el código, versión y todos los datos ingresados al reporte.	
Criterios de Aceptación		Los datos del requerimiento son básicos para el funcionamiento del sistema.	

Descripción detallada de levantamiento de requerimientos RQF006 – Datos de Responsables de Procesos

CONTROL DE INDICADORES DE GESTIÓN DE CALIDAD MEDIANTE APLICACIÓN WEB PARA LA DIRECCIÓN RIPS DEL MINISTERIO COORDINADOR DE DESARROLLO SOCIAL

Tabla 7 Descripción de Requerimiento RQF006

Descripción de Requerimiento RQF006

Descripción del requerimiento.	Estado	Diseño	
El sistema debe contemplar el ingreso y mantenimiento de los responsables de los procesos.			
Creado por	Administrador del Sistema de Gestión de Calidad	Actualizado por	Administrador del Sistema de Gestión de Calidad
Fecha Creación	02/05/2014	Fecha de Actualización	13/06/2014
identificador		RQF006	
Tipo de Requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	Datos de Responsables de los Procesos de la Dirección RIPS (Nombre, Formato de Indicadores, orden en la lista de responsables de procesos)		
Descripción	Disponibilidad de lista de responsables de procesos para seleccionar según el formato de indicadores que se seleccione.		
Datos de salida	Confirmación de ingreso o modificación de responsables de procesos.		
Resultados Esperados	Control de ingreso de responsables de procesos..		
Origen	Requerimiento del Usuario (Administrador del Sistema de Gestión de Calidad)		
Dirigido a	Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuarios del sistema autorizados.		
Prioridad	5		
Requerimientos Asociados	RQF005		
ESPECIFICACIÓN			
Precondiciones	N/A		
Pos condiciones	Evitar que el usuario del Sistema ingrese responsables de procesos no autorizados..		
Criterios de Aceptación	Los datos del requerimiento son básicos para el funcionamiento del sistema.		

Descripción detallada de levantamiento de requerimientos RQF007 – Datos de Personal RIPS.

Tabla 8 Descripción de Requerimiento RQF007

Descripción de Requerimiento RQF007

Descripción del requerimiento.		Estado	Diseño
El sistema debe contemplar el ingreso y mantenimiento de las personas, firmas y función de los responsables de elaborar, revisar y aprobar cada uno de los informes de indicadores.			
Creado por	Administrador del Sistema de Gestión de Calidad	Actualizado por	Administrador del Sistema de Gestión de Calidad
Fecha Creación	02/05/2014	Fecha de Actualización	13/06/2014
identificador		RQF007	
Tipo de Requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	Datos de las personas, firmas y función que cumple en los formatos controlados de indicadores del SGC, (cédula, formato, apellidos, nombres, función, orden e imagen de rúbrica).		
Descripción	Disponer en cada formato controlado del SGC, de lista personas y funciones que desempeñan en cada formato de registro		
Datos de salida	Confirmación de ingreso o modificación de personas, firma y función.		
Resultados Esperados	Selección adecuada de personas y funciones que desempeñan, así como de la rúbrica según la persona seleccionada.		
Origen	Requerimiento del Usuario (Administrador del Sistema de Gestión de Calidad)		
Dirigido a	Administrador del Sistema de Gestión de Calidad.		
	Área de desarrollo.		
	Usuarios del sistema autorizados.		
Prioridad	5		
Requerimientos Asociados	RQF005		
ESPECIFICACIÓN			
Precondiciones	N/A		
Pos condiciones	La rúbrica debe depender de la persona seleccionada.		
Criterios de Aceptación	Los datos del requerimiento son básicos para el funcionamiento del sistema.		

Descripción detallada de levantamiento de requerimientos RQF008 – Recuperación de Reportes.

Tabla 9 Descripción de Requerimiento RQF008

Descripción de Requerimiento RQF008

Descripción del requerimiento.		Estado	Diseño
El sistema debe contemplar la posterior recuperación manteniendo la misma versión de los informes de indicadores ingresados por los responsables.			
Creado por	Administrador del Sistema de Gestión de Calidad	Actualizado por	Administrador del Sistema de Gestión de Calidad
Fecha Creación	02/05/2014	Fecha de Actualización	13/06/2014
identificador		RQF008	
Tipo de Requerimiento	No Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada Descripción	Código de formato y fecha de reporte. Recuperación de reporte de ingreso de indicadores ya realizados.		
Datos de salida	Datos de reporte de indicadores ingresados anteriormente.		
Resultados Esperados	Recuperar cualquier reporte de indicadores ingresados anteriormente, en base a código de formato y fecha de reporte.		
Origen	Requerimiento del Usuario (Administrador del Sistema de Gestión de Calidad)		
Dirigido a	Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuarios del sistema autorizados.		
Prioridad	3		
Requerimientos Asociados	RQF005		
ESPECIFICACIÓN			
Precondiciones	N/A		
Pos condiciones	La información del reporte recuperado debe ser la misma que cuando fue ingresado.		
Criterios de Aceptación	Es necesario la recuperación de los datos ingresados .		

Descripción detallada de levantamiento de requerimientos RQF009 – Ingreso de Usuario

Tabla 10 Descripción de Requerimiento RQF009

Descripción de Requerimiento RQF009

Descripción del requerimiento.	Estado	Diseño	
Para el ingreso al sistema debe ser validado el usuario, la contraseña y los permisos concedidos, de los que dependerán las actividades que el sistema le permita acceder.			
Creado por	Administrador del Sistema de Gestión de Calidad	Actualizado por	Administrador del Sistema de Gestión de Calidad
Fecha Creación	02/05/2014	Fecha de Actualización	13/06/2014
identificador		RQF009	
Tipo de Requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada		Usuario y contraseña.	
Descripción		Ingreso al sistema de usuarios.	
Datos de salida		Presentación del menú principal o notificación de error l ingresar al sistema.	
		Acceso al menú principal del sistema.	
Resultados Esperados		Requerimiento del Usuario (Administrador del Sistema de Gestión de Calidad)	
Origen			
Dirigido a		Administrador del Sistema de Gestión de Calidad.	
		Área de desarrollo.	
		Usuarios del sistema autorizados.	
Prioridad		5	
Requerimientos Asociados		RQF010	
ESPECIFICACIÓN			
Precondiciones		N/A	
Pos condiciones		Visualización del menú principal.	
Criterios de Aceptación		Es imprescindible para el sistema.	

Descripción detallada de levantamiento de requerimientos RQF010 – Presentación de Módulos en aplicativo.

Tabla 11 Descripción de Requerimiento RQF010

Descripción de Requerimiento RQF010

Descripción del requerimiento.		Estado	Diseño
El sistema debe tener Menú Principal.			
Creado por	Administrador del Sistema de Gestión de Calidad	Actualizado por	Administrador del Sistema de Gestión de Calidad
Fecha Creación	02/05/2014	Fecha de Actualización	13/06/2014
identificador		RQF010	
Tipo de Requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	Nombres de módulos y orden establecido en los permisos de acuerdo a las actividades definidas para el usuario ingresado.		
Descripción	Módulo principal con menú de acuerdo a los permisos establecidos.		
Datos de salida	Visualización de Módulo Principal y Menú.		
Resultados Esperados	Acceso al menú principal del sistema para navegación entre los módulos.		
Origen	Requerimiento del Usuario (Administrador del Sistema de Gestión de Calidad)		
Dirigido a	Administrador del Sistema de Gestión de Calidad.		
Prioridad	Área de desarrollo.		
Requerimientos Asociados	Usuarios del sistema autorizados.		
	5		
	RQF009		
ESPECIFICACIÓN			
Precondiciones	N/A		
Pos condiciones	Visualización del menú principal y navegación entre módulos.		
Criterios de Aceptación	Es imprescindible para el sistema.		

2.02 Mapeo de Involucrados

Descripción de todas las personas directamente involucradas como los Responsables de Procesos y los usuarios Internos y Externos que utilizan las Bases de Datos que son procesadas por la Dirección. **(Ver Anexo A.05. Mapeo de Involucrados).**

2.03 Matriz de Involucrados

Se encuentran detallados tanto los involucrados directos como indirectos. El interés de los involucrados varía en el interés que tengan en el proceso. Las Áreas Financiera, Administrativa, Tecnológica y de Talento Humano son procesos de apoyo que tienen como función proporcionar los recursos necesarios para cumplir con la implementación y cumplimiento de la mejora continua. Los involucrados que intervienen directamente en el proceso son Administrador de Base de Datos, Técnico de Base de Datos, Equipo de Validación y Depuración, Administrador del Sistema de Información, Encargado Infraestructura, Equipo de Desarrollo, Administrador Interconexión de Bases de Datos, Técnicos Administración Bases de Datos. El interés de las Direcciones del Registro Social, SIISE y Talento Humano es de usuarios internos que utilizan la información de la Dirección. Como usuarios internos están el Ministerio de Inclusión Económica y Social, Ministerio de Salud Pública, Secretaría Técnica de Discapacidades, Corporación Nacional de Finanzas Populares y Solidarias, Ministerio de Educación, Público en General que utilizan la información de las bases de datos como insumos para sus procesos internos o acceden para verificación de información que es de interés público. **(Ver Anexo A.06 Matriz de Involucrados).**

Capítulo III: Problemas y Objetivos

3.01 Árbol de Problemas

Se toma como base el problema central y las causas y efectos directos y secundarios percibidos por los involucrados.

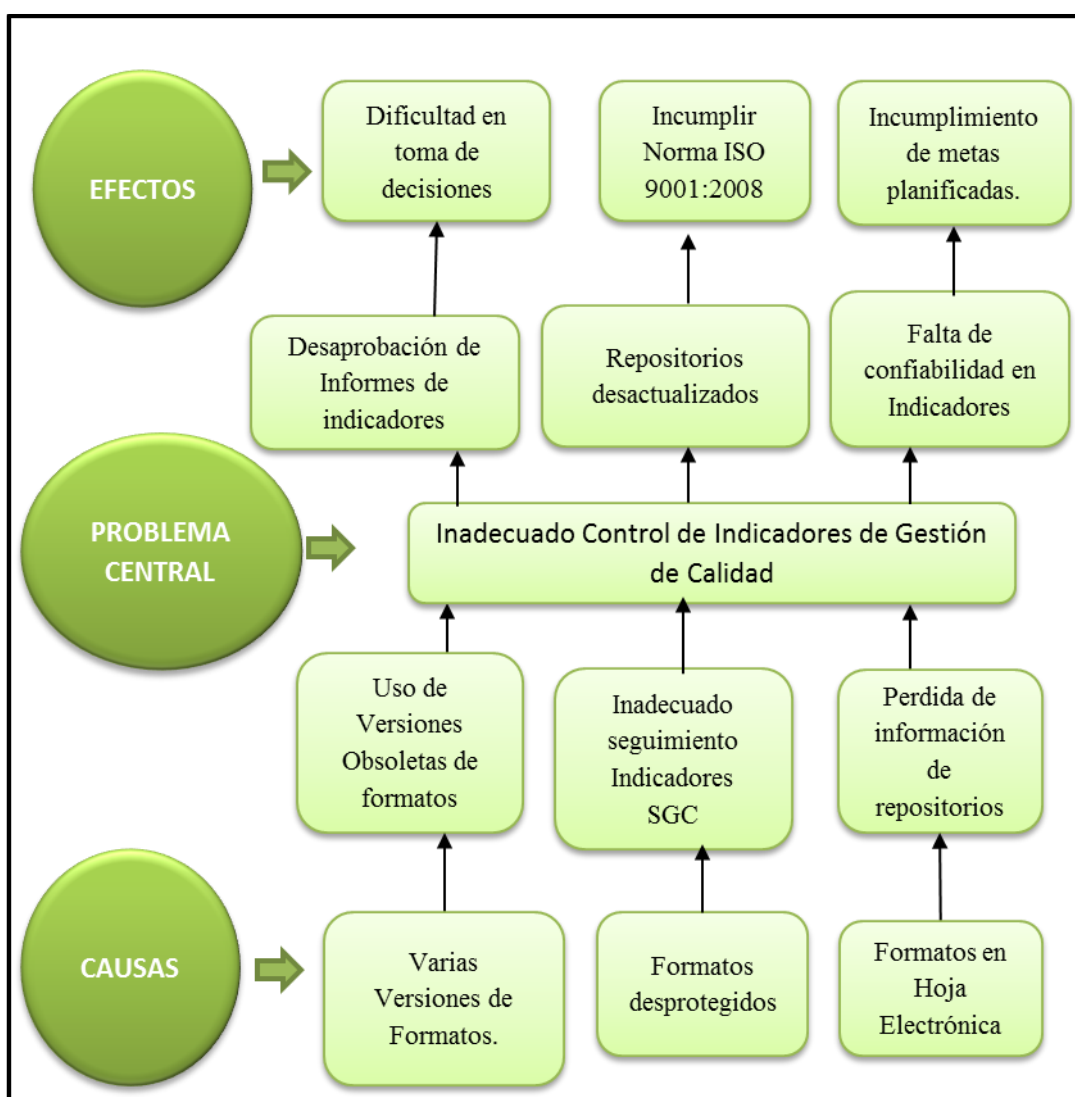


Figura 2 Árbol de Problemas

Árbol de Problemas

El primer paso para alcanzar la solución es identificar las deficiencias y sus causas.

3.02 Árbol de Objetivos

Se cambian las condiciones negativas a positivas que son deseadas y posibles de alcanzar, transformándose en medios para alcanzar fines que son el propósito del proyecto.

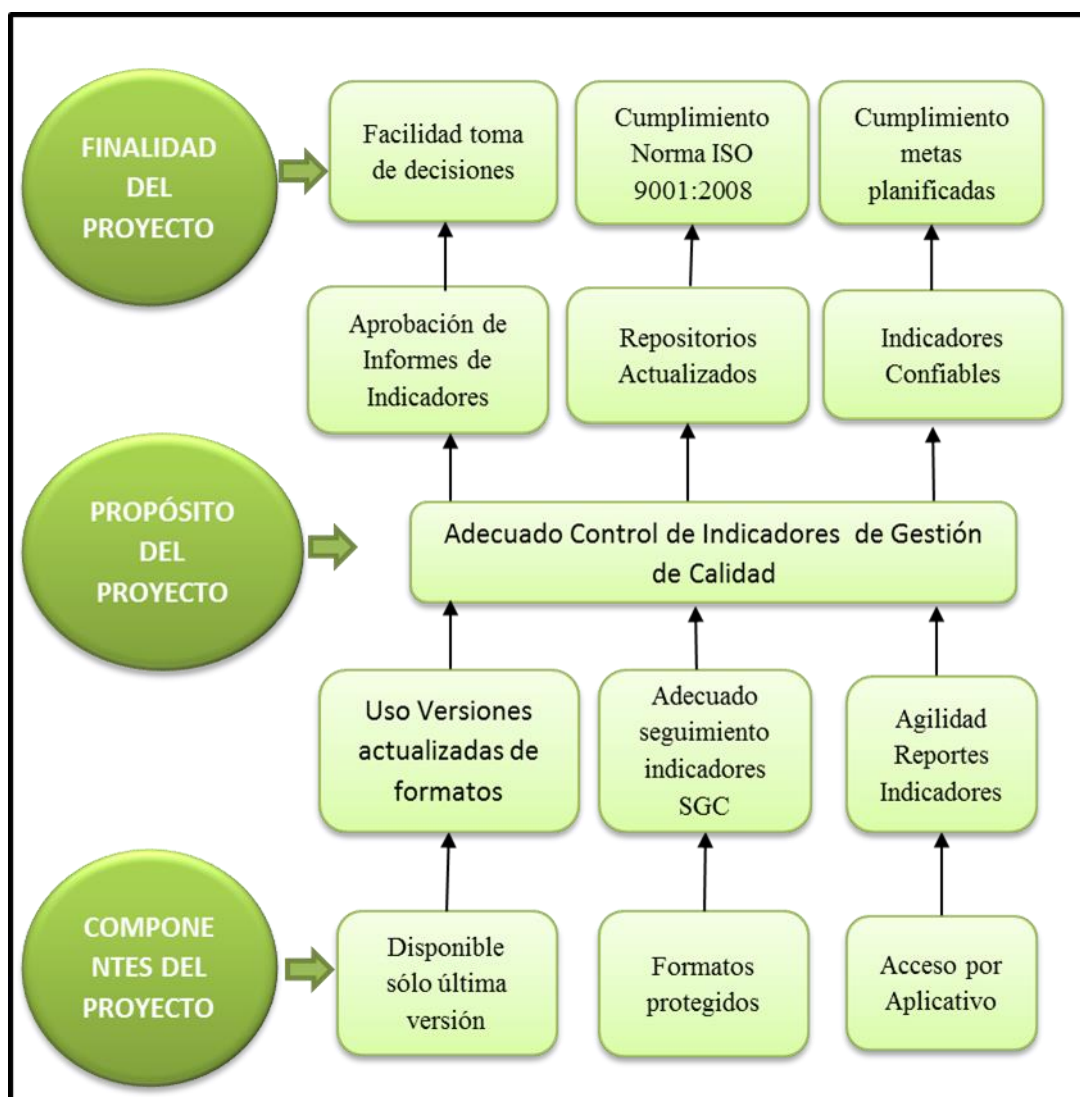


Figura 3 Árbol de Objetivos

Árbol de Objetivos

El aprovechar las deficiencias del sistema actual para convertirlas en medios para encontrar la solución a cada una, implementándolas de forma adecuada para mejorar su eficiencia es el objeto del proyecto.

3.03 Diagramas de Casos de Uso

Incluye las principales acciones que se deben realizar para alcanzar los objetivos.

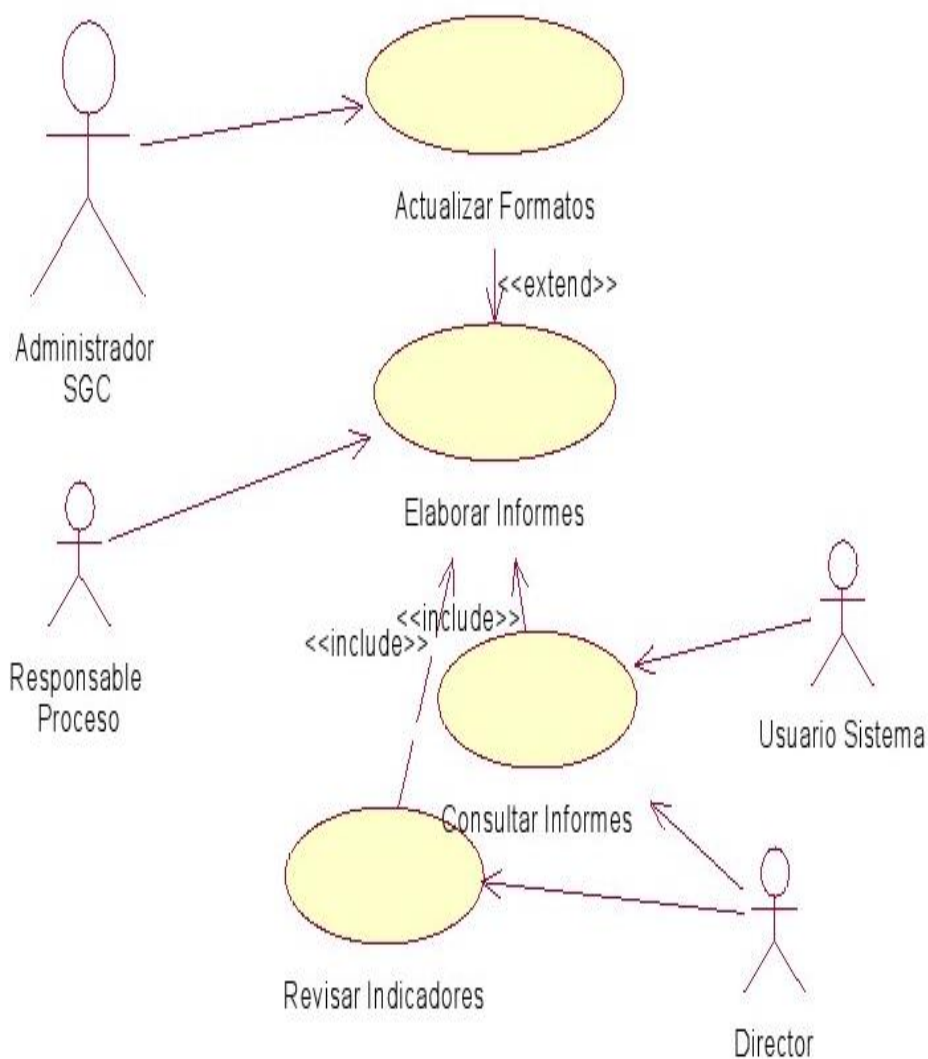


Figura 4 Diagrama de Caso de Uso

Diagrama de Caso de Uso

3.04 Casos de Uso de Realización

Actualizar Formatos – Casos de Realización

Tabla 12 Caso de Realización DCR001. Actualizar Formatos

Caso de Realización DCR001. Actualizar Formatos

Nombre	DR_Actualizar Formatos
Identificador	DCR001
Responsabilidades	Administrador SGC
Tipo	Sistema
Referencias Casos de Uso	CU003
Referencias Requisitos	RQF001 RQF002 RQF003 RQF004 RQF005 RQF006 RQF007 RQF009

PRECONDICIONES

De Instancia

Debe estar creado el usuario del Administrador del SGC para que pueda ingresar al sistema.
El usuario autorizado debe ingresar con usuario y contraseña válidos y con rol con permisos para acceder a módulos de administración.
Los formatos deben estar creados para ingresar los datos necesarios para su funcionamiento.
Debe crear los usuarios y contraseñas de los responsables de procesos y usuarios autorizados
Debe Actualizar datos de estado de registros.
Deba Actualizar datos de Lista Maestra de Registros.
Debe Actualizar datos de campos de indicadores para cada formato.
Debe Actualizar datos de responsables de procesos.
Debe Actualizar datos de personal RIPS que realiza alguna acción en los informes.

De Relación

Deben relacionar los módulos del aplicativo con los roles y el orden en que van a aparecer en el menú principal.

POS CONDICIONES

De Instancia

Deben quedar creados los usuarios, contraseñas y roles de las personas que van a utilizar el sistema.
Deben ser ingresados todos los parámetros necesarios para que los formatos se visualicen.

De Relación

Deben estar relacionados los módulos del aplicativo con los roles y el orden en que van a aparecer en el menú principal.

SALIDAS PANTALLA

Pantalla login.
Mensajes de error si es del caso.
Menú Principal (Selecciona opción menú).
Pantallas de Ingreso de Parámetros.
Notificación de error o de ingreso correcto realizado.

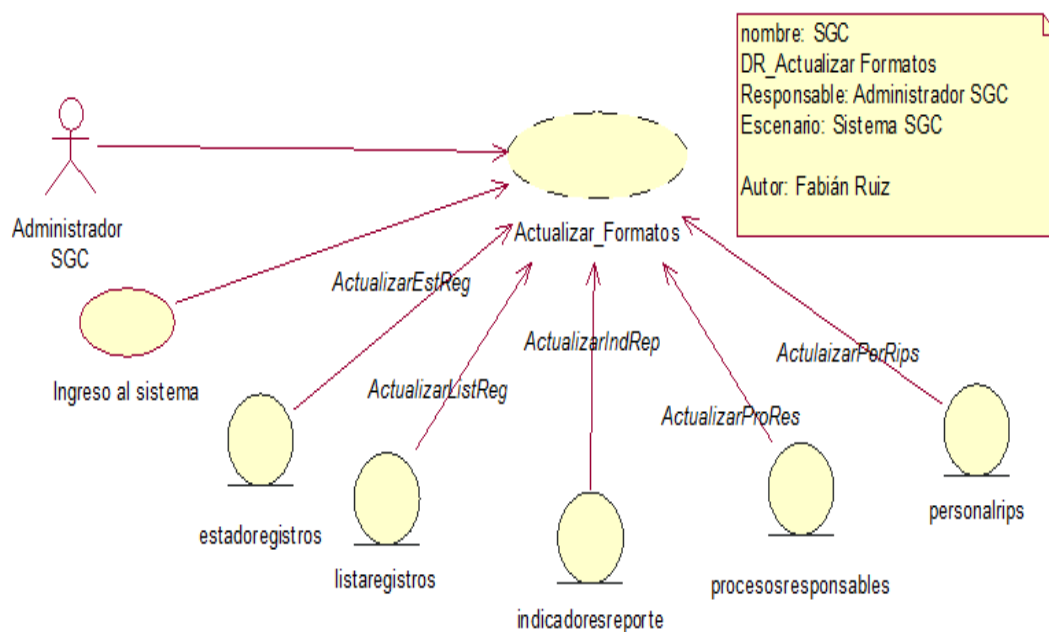


Figura 5 Diagrama Realización - Actualizar Formatos

Diagrama Realización - Actualizar Formatos

Elaborar Informe – Casos de Realización

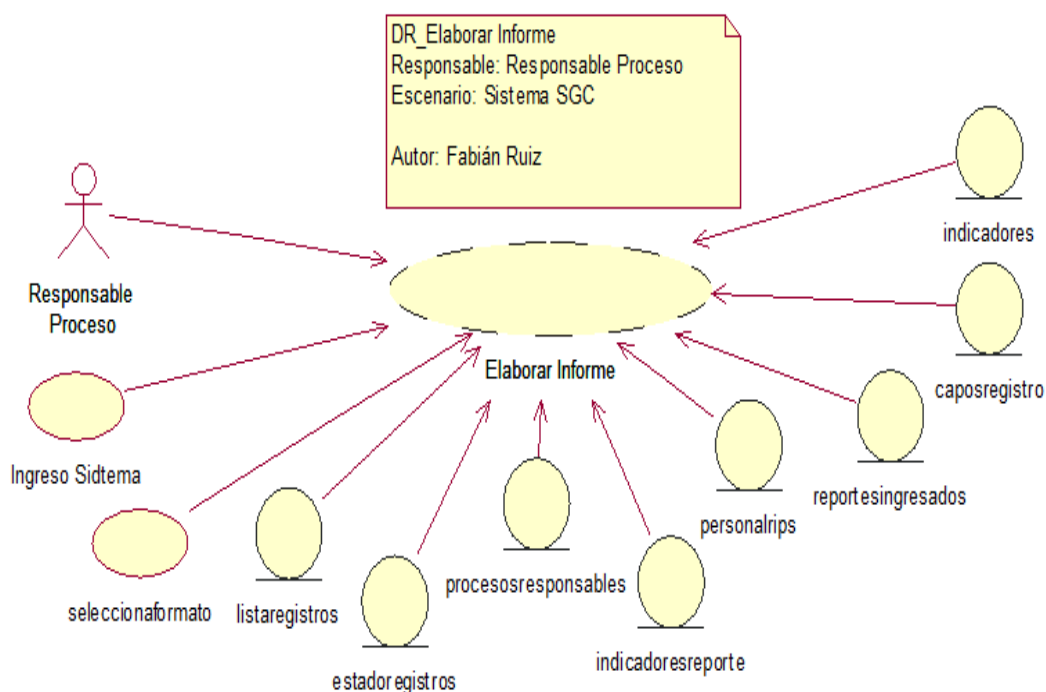


Figura 6 Diagrama Realización - Elaborar Informes

Diagrama Realización - Elaborar Informes

Tabla 13 Caso de Realización DCR002 - Elaborar Informes

Caso de Realización DCR002 - Elaborar Informes

Nombre	DR_Elaborar Informes
Identificador	DCR002
Responsabilidades	Responsable Proceso
Tipo	Sistema
Referencias Casos de Uso	CU002
Referencias Requisitos	RQF005 RQF009

PRECONDICIONES

De Instancia
El usuario debe ingresar con usuario y contraseña válidos para ingresar al sistema con el rol con permisos para acceder a ingreso de informes.
Los formatos deben estar creados.
Los parámetros necesarios deben haber sido ingresados por el Administrador del SGC.
Se venen llenar todos los campos obligatorios de los formatos.
Se deben seleccionar las personas que intervienen en el proceso.

De Relación

POS CONDICIONES

De Instancia

De Relación

SALIDAS PANTALLA
Pantalla login.
Mensajes de error si es del caso.
Menú Principal (Selecciona opción menú).
Pantalla de Ingreso de Informe.
Notificación de error o de ingreso correcto realizado.

Consultar Informes – Casos de Realización

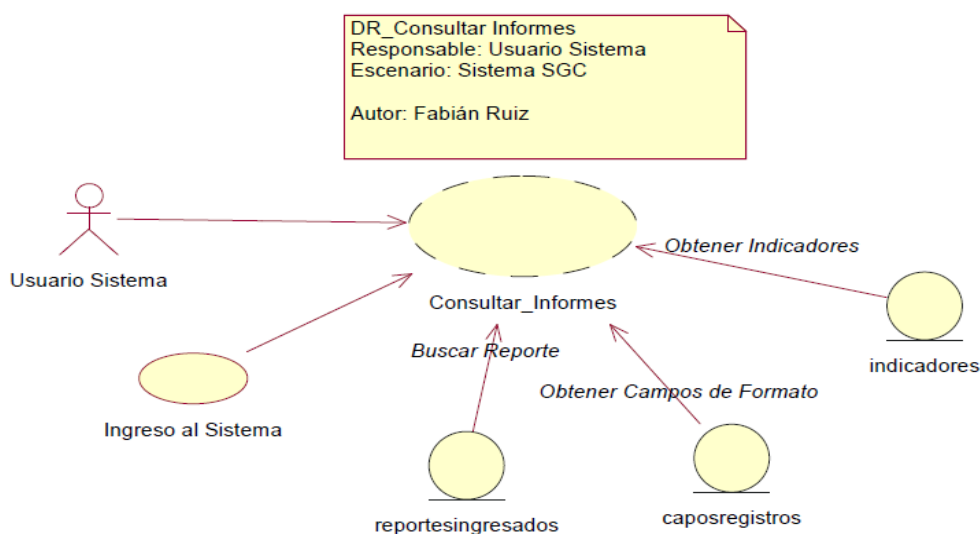


Figura 7 Diagrama Realización - Consultar Informes

Diagrama Realización - Consultar Informes

Tabla 14 Caso de Realización DCR003 - Consultar Informes

Caso de Realización DCR003 - Consultar Informes

Nombre	DR_Consultar Informes
Identificador	DCR003
Responsabilidades	Director
Tipo	Sistema
Referencias Casos de Uso	CU001
Referencias Requisitos	RQF008 RQF012

PRECONDICIONES

De Instancia
El usuario debe ingresar con usuario y contraseña válidos y con rol con permisos para acceder a los reportes de indicadores.
Los informes deben haber sido ingresados.

De Relación
El reporte de indicadores debe contener parámetros definidos para la búsqueda.

POS CONDICIONES

De Instancia

De Relación

SALIDAS PANTALLA
Pantalla login.
Mensajes de error si es del caso.
Menú Principal (Seleccionar opción de menú).
Pantalla de reporte.

Revisar Indicadores – Casos de Realización

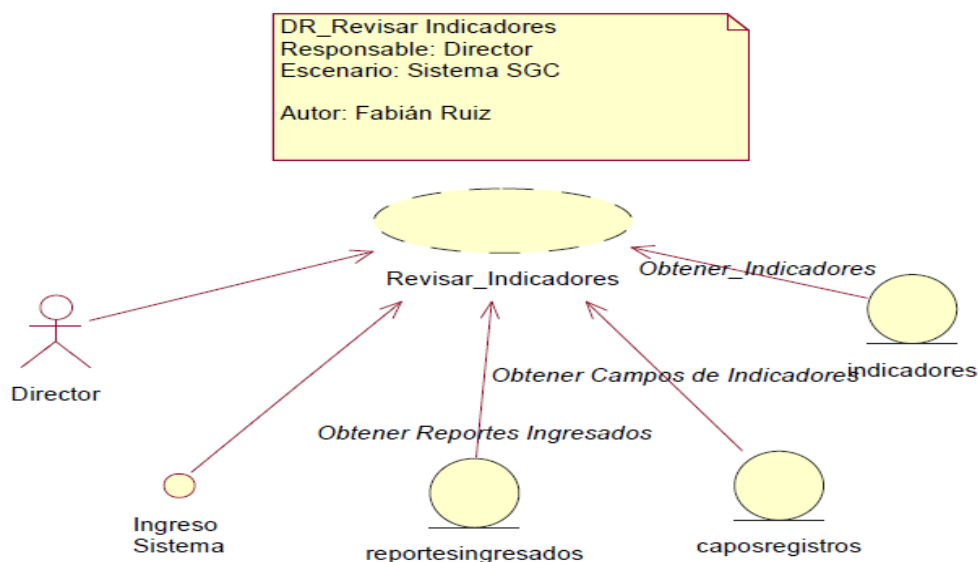


Figura 8 Diagrama Realización - Revisar Indicadores

Diagrama Realización - Revisar Indicadores

Tabla 15 Caso de Realización DCR004 - Revisar Indicadores

Caso de Realización DCR004 - Revisar Indicadores

Nombre	DR_Revisar Indicadores
Identificador	DCR004
Responsabilidades	Administrador SGC – Responsable Proceso – Usuarios Sistema
Tipo	Sistema
Referencias Casos de Uso	CU004
Referencias Requisitos	RQF_011

PRECONDICIONES

De Instancia

El usuario autorizado debe ingresar con usuario y contraseña válidos e ingresará con rol con permisos para acceder a consulta de informes.

Los informes deben estar ingresados.

Debe buscar los campos que debe reportar.

Debe buscar los indicadores a reportar.

De Relación

Se deberá seleccionar en base a código de formado de la lista maestra y fecha específica.

POS CONDICIONES

De Instancia

El reporte seleccionado no podrá der modificado, solo se podrá visualizar.

De Relación

SALIDAS PANTALLA

Pantalla login.

Mensajes de error si es del caso.

Menú Principal (Selecciona opción menú).

Pantallas de selección de informe específico.

Notificación de error o Pantalla de visualización de informe solicitado.

3.05 Diagrama de Secuencias del Sistema.

Actualizar Formatos – Diagrama de Secuencia

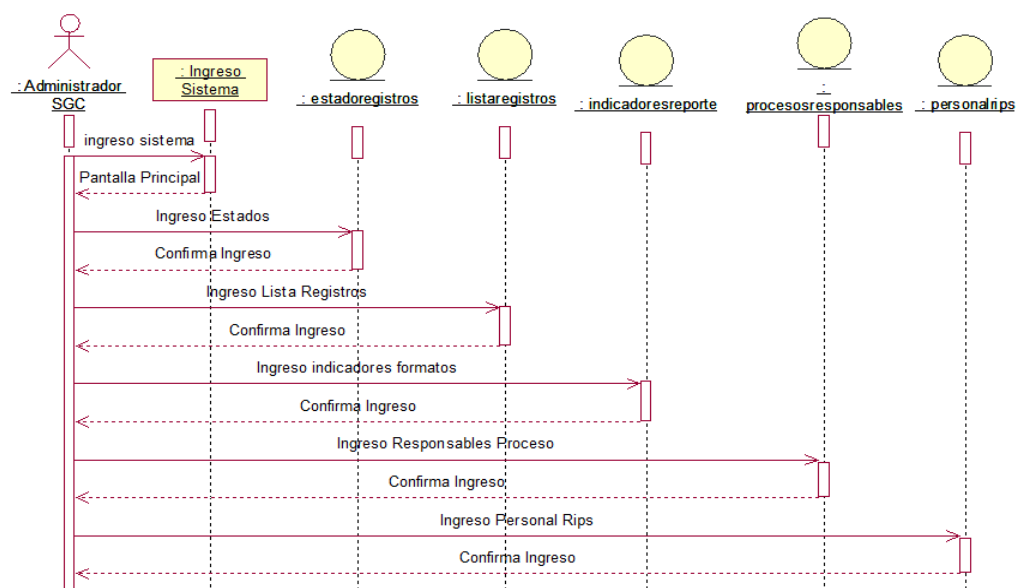


Figura 9 Actualizar Formatos - Diagrama Secuencia

Actualizar Formatos - Diagrama Secuencia

Elaborar Informe – Diagrama de Secuencia

El Anexo referenciado a continuación detalla el proceso de realizar el ingreso de los datos de un informe de cualquiera de los Formatos de la Lista Maestra de Registros. (Ver Anexo A.09 Diagrama de Secuencia Elaborar Informe).

3.06 Especificación de Casos de Uso

3.06.01 Actualizar Formatos

Tabla 16 Especificación Actualizar Formatos CU-001

Especificación Actualizar Formatos CU-001

Nombre	Actualizar Formatos
Identificador	CU-001
Objetivo Final	Ingresar parámetros necesarios para completar los formatos de informes y mantener actualizada la información con las últimas modificaciones realizadas.
Tipo	Sistema
Usuario/Actor	Administrador SGC
Desencadenador	Administrador SGC
PRECONDICIONES	
De Instancia	Iniciar Sesión.
	Registrar las modificaciones en las tablas correspondientes.
De Relación	El sistema debe validar que los datos ingresados cumplan con la especificación necesaria para ser válidos.
POS CONDICIONES	
De Instancia	
De Relación	
SALIDAS PANTALLA	
De Instancia	Pantalla Login para acceso al sistema.
	Módulos de mantenimiento de los registros de las tablas relacionadas con los formatos de informes.
De Relación	

Tabla 17 Flujo Básico Actualizar Formatos CU-001

Flujo Básico Actualizar Formatos CU-001

FLUJO BÁSICO		
Paso	Acciones del usuario	Acciones del Sistema
1	El Administrador del Sistema Selecciona el Módulo de Ingreso, Modificación o Eliminación de la información que necesita procesar entre las opciones que tiene en el Menú Principal.	El sistema despliega la pantalla solicitada.
2	El Administrador del Sistema Ingresa o Modifica la información en los campos correspondientes según la acción que desea realizar y hace click en la opción Guardar (Ingresa) o Actualizar.	El sistema valida los datos que tengan los formatos y condiciones especificadas e ingresa o actualiza la información.

3.06.02 Elaborar Informes

Tabla 18 Especificación Elaborar Informes CU-002

Especificación Elaborar Informes CU-002

Nombre	Elaborar Informes
Identificador	CU-002
Objetivo Final	Ingresa los datos de Indicadores de los procesos al sistema para facilitar el seguimiento, control y toma de decisiones.
Tipo	Sistema
Usuario/Actor	Responsable de Proceso
Desencadenador	Administrador SGC
PRECONDICIONES	
De Instancia	Iniciar Sesión.
	Seleccionar el módulo correspondiente el Menú Principal del sistema.
	Deben estar actualizados los parámetros necesarios para completar los formatos antes de poder ingresar los informes.
De Relación	
	El sistema debe validar que los datos ingresados cumplan con la especificación necesaria para ser válidos.
POS CONDICIONES	
De Instancia	
De Relación	
SALIDAS PANTALLA	
De Instancia	Pantalla Login para acceso al sistema.
	Módulos de Ingreso de indicadores.
De Relación	

Tabla 19 Flujo Básico Elaborar Informes CU-002

Flujo Básico Elaborar Informes CU-002

FLUJO BÁSICO		
Paso	Acciones del usuario	Acciones del Sistema
1	El Responsable de Proceso debe seleccionar el módulo con el formato de informe que le corresponde a su sección del Menú Principal del sistema.	El sistema despliega la pantalla solicitada.
2	El Responsable de Proceso debe seleccionar los indicadores que le corresponden, seleccionar el nombre del responsable que está ingresando el informe, el periodo al que corresponde el informe, debe llenar todos los campos que correspondan a los indicadores, debe seleccionar quién elabora, quién revisa y quién aprueba el informe, deben constar las firmas de estas personas en el formato. Debe hacer click en la opción Guardar (Ingresar) y confirmar que desea realizar el ingreso o cancelar si no está seguro..	El sistema valida los datos que tengan los formatos y condiciones especificadas e ingresa o actualiza la información.

3.06.03 Consultar Indicadores

Tabla 20 Especificación Consultar Indicadores CU-003

Especificación Consultar Indicadores CU-003

Nombre	Consultar Informes
Identificador	CU-003
Objetivo Final	Consultar Informes de los procesos en el sistema para facilitar el seguimiento, control y toma de decisiones.
Tipo	Sistema
Usuario/Actor	Usuario del sistema
Desencadenador	Responsable de Proceso
PRECONDICIONES	
De Instancia	
Iniciar Sesión.	
Seleccionar el módulo correspondiente el Menú Principal del sistema.	
Deben estar ingresados al sistema los informes de indicadores que se desea consultar.	
De Relación	
POS CONDICIONES	
De Instancia	
De Relación	
SALIDAS PANTALLA	
De Instancia	
Pantalla Login para acceso al sistema.	
Módulos de Consulta de Informes.	

Tabla 21 Flujo Básico Consultar Indicadores CU-003

Flujo Básico Consultar Indicadores CU-003

FLUJO BÁSICO		
Paso	Acciones del usuario	Acciones del Sistema
1	El Responsable de Proceso debe seleccionar el módulo de consulta de informe que le corresponde a la información que desea revisar del Menú Principal del sistema.	El sistema despliega la pantalla solicitada.
2	Debe ingresar al sistema el código del formato a revisar y la fecha del informe	El sistema valida los parámetros ingresados y visualiza el informe solicitado o notifica que no hay informe para esos parámetros

3.06.04 Revisar Indicadores

Tabla 22 Especificación Revisar Indicadores CU-004

Especificación Revisar Indicadores CU-004

Nombre	Revisar Indicadores
Identificador	CU-004
Objetivo Final	Consultar Indicadores subidos al sistema para seguimiento, control y toma de decisiones e ingreso de indicadores a GPR.
Tipo	Sistema
Usuario/Actor	Director
Desencadenador	Responsable de Proceso
PRECONDICIONES	
De Instancia	Iniciar Sesión con rol adecuado. Seleccionar el módulo correspondiente el Menú Principal del sistema. Deben estar ingresados al sistema los informes de indicadores que se desea consultar.
De Relación	
POS CONDICIONES	
De Instancia	
De Relación	
SALIDAS PANTALLA	
De Instancia	Pantalla Login para acceso al sistema. Módulos de Consulta de Informes.
De Relación	

Tabla 23 Flujo Básico Revisar Indicadores CU-004

Flujo Básico Revisar Indicadores CU-004

FLUJO BÁSICO		
Paso	Acciones del usuario	Acciones del Sistema
1	El Director debe seleccionar el módulo de consulta de indicadores del Menú Principal del Sistema.	El sistema despliega la pantalla solicitada.
2	Debe ingresar al sistema el código del Proceso que a revisar y la fecha de corte.	El sistema valida los parámetros ingresados y visualiza los reportes o notifica que no hay reportes para esos parámetros

Capítulo IV Análisis de Alternativas

4.01 Matriz de Análisis de Alternativas.

Para elaborar la Matriz de Análisis de Alternativas se toma en consideración los parámetros que permitan implantar la solución con efectividad: identificar las soluciones alternativas, evaluar la factibilidad técnica, financiera, social y política, y la estrategia general diseñada.

Tabla 24 Matriz de Análisis de Alternativas

Matriz de Análisis de Alternativas

MATRIZ DE ANÁLISIS DE ALTERNATIVAS							
OBJETIVOS	Impacto sobre el Propósito	Factibilidad de Técnica	Factibilidad Financiera	Factibilidad Social	Factibilidad Política	Total	Categorías
Indicadores actualizados	5	4	4	3	5	21	Medio Alto - Alto
Control eficiente de uso de formatos	5	4	4	4	5	22	Medio Alto - Alto
Facilidad en toma de decisiones.	5	4	4	4	4	21	Medio Alto - Alto
Uso solo de Formatos Controlados	5	4	4	4	5	22	Medio Alto - Alto
Facilidad de monitoreo de cumplimiento de indicadores.	5	4	4	4	4	21	Medio Alto - Alto
TOTAL	25	20	20	19	23	107	

De la Matriz de Análisis de Alternativas se puede deducir que las soluciones propuestas son adecuadas y tienen un alto porcentaje de posibilidad de alcanzar los

objetivos que se está buscando para dar solución a las deficiencias del Sistema de Gestión de la Calidad, al estar todos los objetivos en categoría Medio Alto - Alto.

4.02 Matriz de Impactos de Objetivos

La Matriz de Análisis de Objetivos se realiza para determinar factibilidad o no de lograr implementar la solución midiendo su impacto en diferentes áreas como en el género y el ambiental, así como determinar el grado de relevancia que tenga y su sostenibilidad en el tiempo, evitando trabajar en una solución inaplicable o que no vaya a proporcionar la solución que se está buscando o no colme las expectativas del cliente. (Ver Tabla 08. Matiz de Impacto de Objetivos)

4.03 Estándares para el Diseño de Clases

Es importante la utilización de convenciones de código para facilitar la lectura del software y entender rápido código nuevo, enmarcando la distribución del código fuente en el estándar de cualquier otro producto.

Para el diseño de clases se utilizan los estándares de codificación del lenguaje Java, JAVA CODE CONVENTIONS – ORACLE, presentados en Java Language Specification, de Sun Microsystems Inc.

(<http://www.oracle.com/technetwork/java/codeconventions-150003.pdf>). (Ver Anexo

A.03 Java Code Conventions).

4.04 Diagrama de Clases

Refleja las entidades de la estructura de la base de datos.

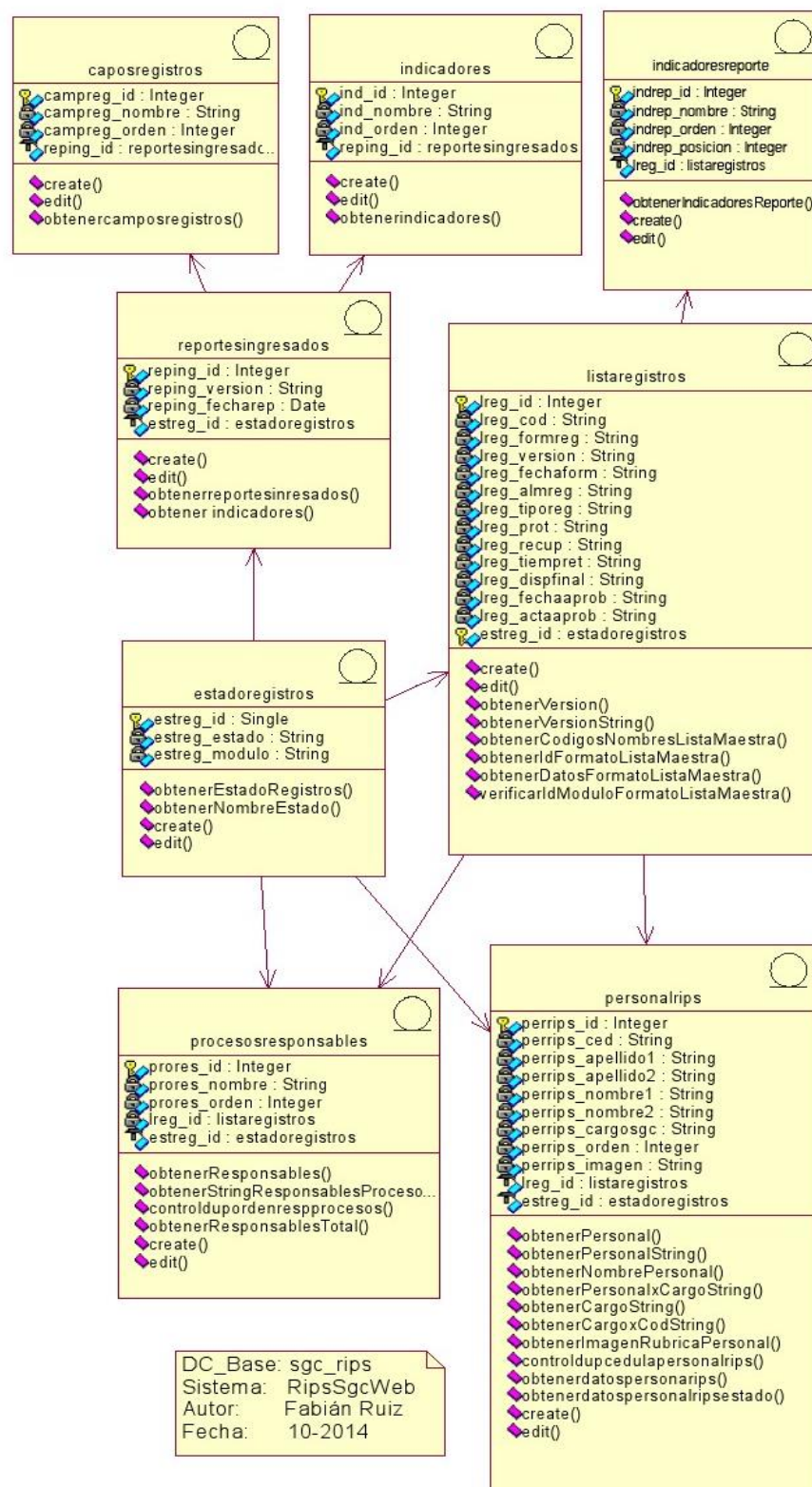


Figura 10 Diagrama de Clases

Diagrama de Clases

4.05 Modelo Lógico - Físico

Los modelos describen la estructura de la base de datos.

El Modelo Lógico visualiza la forma en que se van a relacionar las entidades para cumplir con lo que es requerido por el usuario final. **(Ver Anexo A.10 Modelo Lógico de Base de Datos).**

El Modelo Físico evidencia la estructura de datos a bajo nivel que se está implementando. **(Ver Anexo A.11 Modelo Físico de Base de Datos)**

El Diagrama de Objetos muestra la estructura de datos y restricciones de seguridad. **(Ver Anexo A.12 Diagrama de Objeto Orientada a Modelo de Base de Datos).**

4.06 Diagrama de Componentes

El diagrama representa los componentes del sistema. La base de datos será en Postgres, el servidor de aplicaciones Glassfish que controlará las conexiones a la base de datos y los servicios y la capa Web. El sistema estará protegido por un firewall.

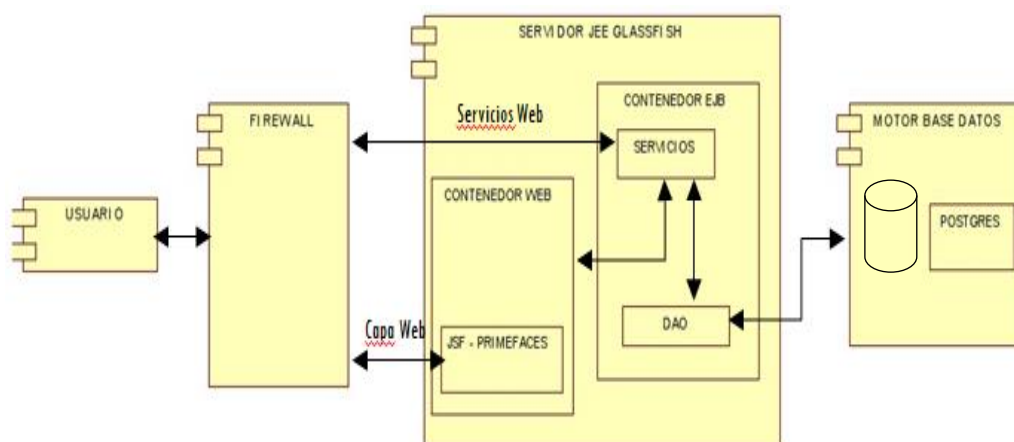


Figura 11 Diagrama de Componentes

Diagrama de Componentes

4.07 Diagramas de Estrategias.

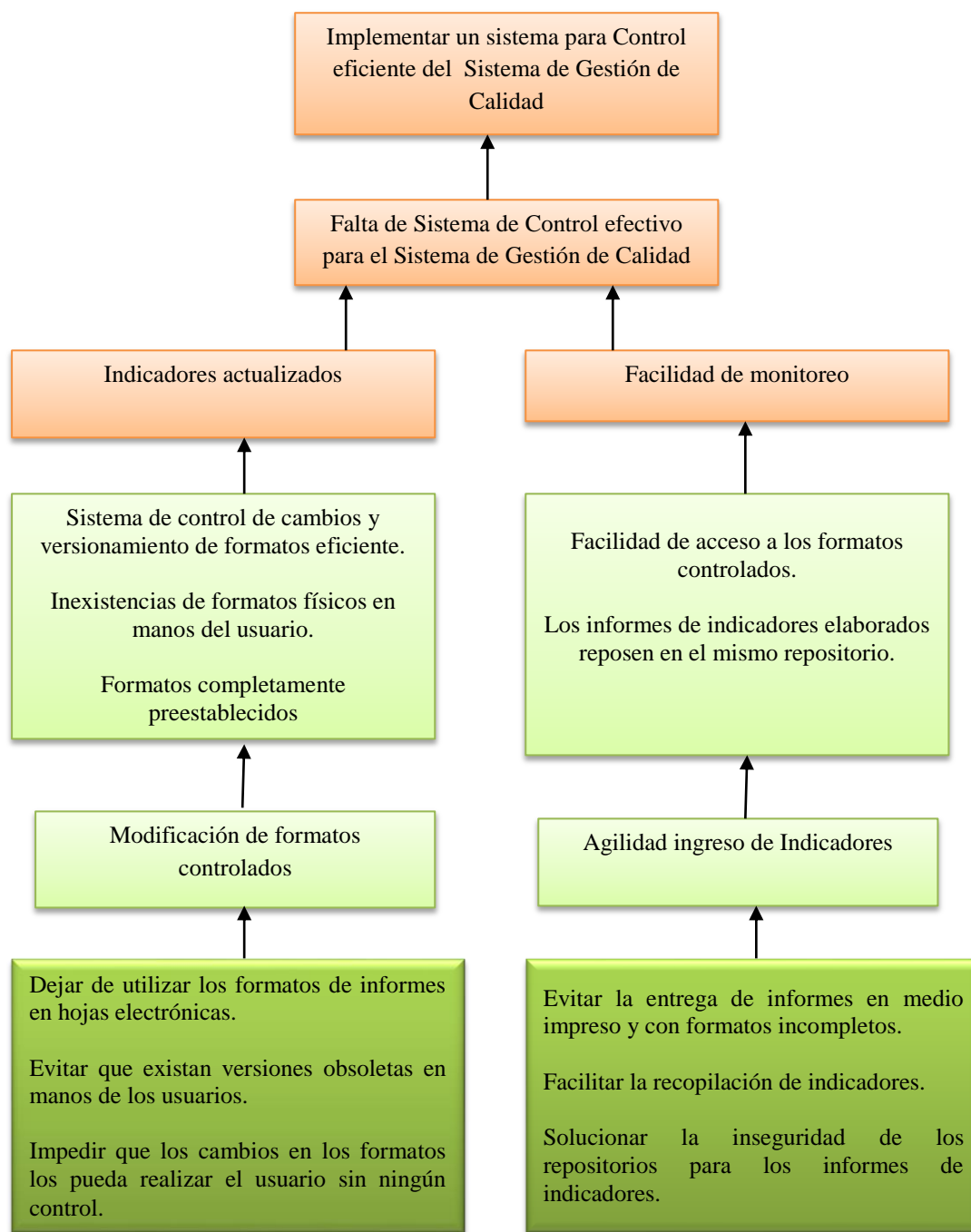


Figura 12 Diagrama de Estrategias

Diagrama de Estrategias

Para elaborar el Diagrama de Estrategias se debe establecer la estructura y alcance, se vincula a cada estrategia todos los objetivos que pueden ser trabajados de forma preferente, se debe determinar los límites del proyecto y determinar los objetivos específicos.

El diagrama de estrategias evidencia las decisiones tomadas durante la elaboración de la estrategia, las consecuencias sobre los objetivos seleccionados y los impactos esperados. Es un complemento a los diagramas de objetivos y efectos.

4.08 Matriz de Marco Lógico.

En esta Matriz queda consignada la manera en que se desea cumplir con el requerimiento para dar solución al problema planteado.

Por medio de las actividades se obtendrán todos los insumos necesarios para la construcción de los componentes, los que nos llevan a cumplir con el propósito del proyecto hasta alcanza la finalidad prevista.

Los factores externos son indispensables para el éxito del proyecto y con todos estos elementos articulados se podrá desarrollar la solución.

Los avances deben ser medibles por medio de los indicadores a través de los medios de verificación. (**Ver Anexo A.01 Matriz de Marco Lógico**).

4.09 Vistas Arquitectónicas

4.09.01 Vista Lógica

La arquitectura lógica apoya principalmente los requisitos funcionales lo que el sistema debe brindar en términos de servicios a sus usuarios. El sistema se descompone en una serie de abstracciones clave, tomadas (principalmente) del

dominio del problema en la forma de objetos o clases de objetos. Aquí se aplican los principios de abstracción, encapsulamiento y herencia. Esta descomposición no solo se hace para potenciar el análisis funcional, sino también sirve para idéntica mecanismos y elementos de diseños comunes a diversas partes del sistema.

Distribución de la Arquitectura utilizada para la aplicación.

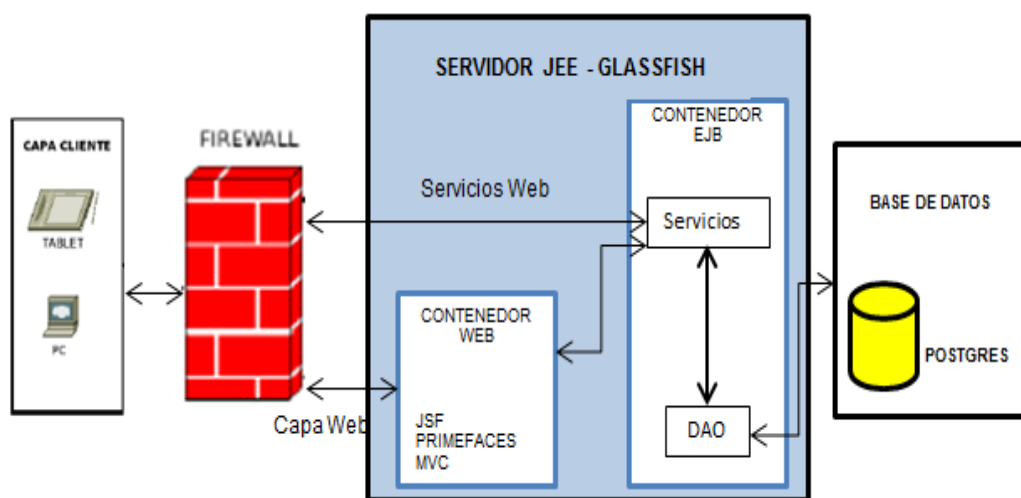


Figura 13 Arquitectura - Vista Lógica

Arquitectura - Vista Lógica

La Arquitectura por Capas muestra la estructura por capas utilizada para desarrollar la aplicación

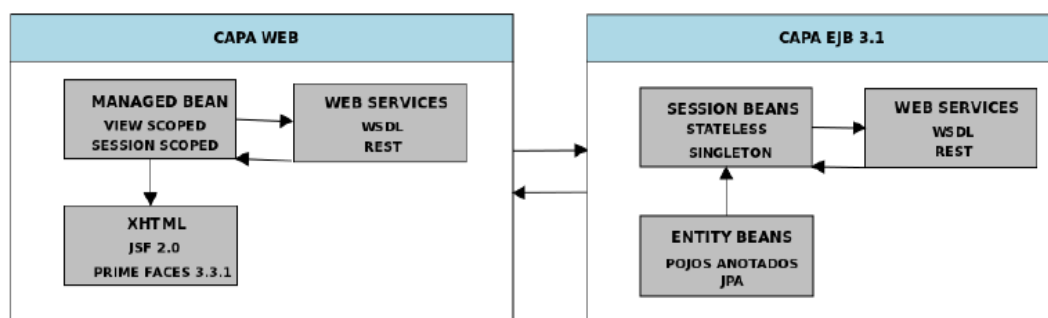


Figura 14 Arquitectura por Capas

Arquitectura por Capas

4.09.02 Vista Física

Vista del sistema con la distribución de los componentes en los servidores.

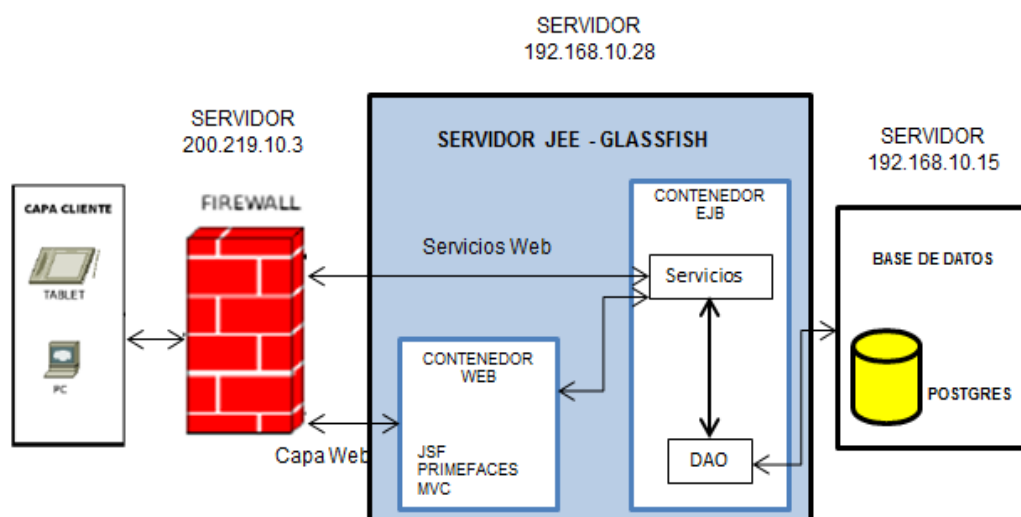


Figura 15 Arquitectura - Vista Física

Arquitectura - Vista Física

4.09.03 Vista de Desarrollo

Proyecto RipsSgcEnterprise: Solución al problema planteado por el usuario.

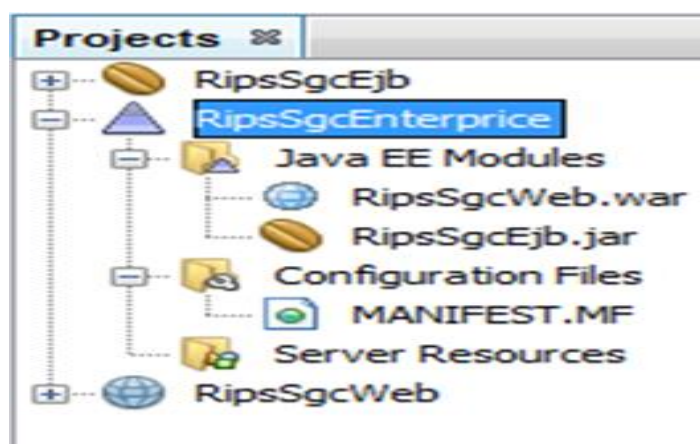


Figura 16 Vista de Desarrollo RipsSgcEnterprise

Vista de Desarrollo RipsSgcEnterprise

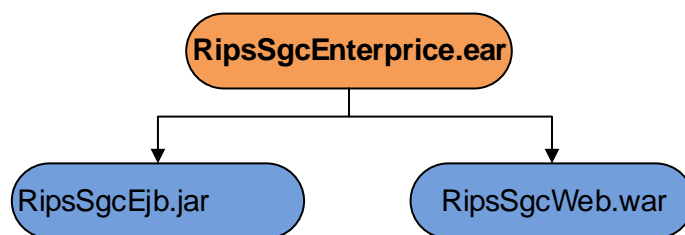


Figura 17 Modelo Vista de Desarrollo RipsSgcEnterprise

Modelo Vista de Desarrollo RipsSgcEnterprise

El sistema ha sido empaquetado en un archivo EAR llamado **RipsSgcEnterprise.ear** el cual agrupa a un módulo EJB y un módulo Web.

Módulo RipsSgcEjb: Este módulo contiene clases Java que representan a Beans Empresariales (Session Beans) y Beans de Persistencia (Entity Beans) los cuales han sido empaquetados en un archivo llamado **RipsSgcEjb.jar**

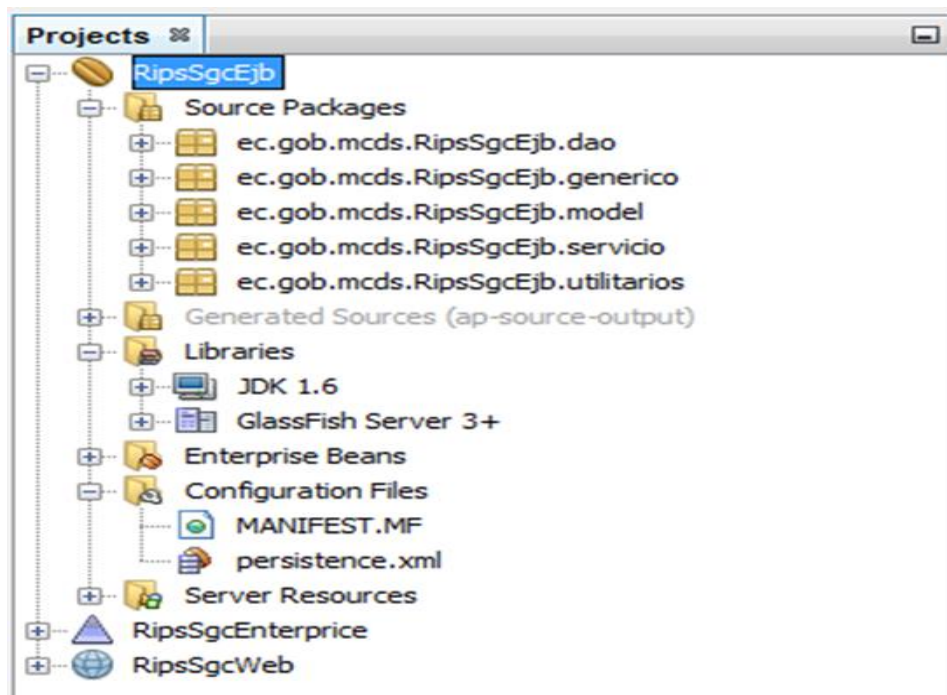


Figura 18 Vista de Desarrollo RipsSgcEjb

Vista de Desarrollo RipsSgcEjb

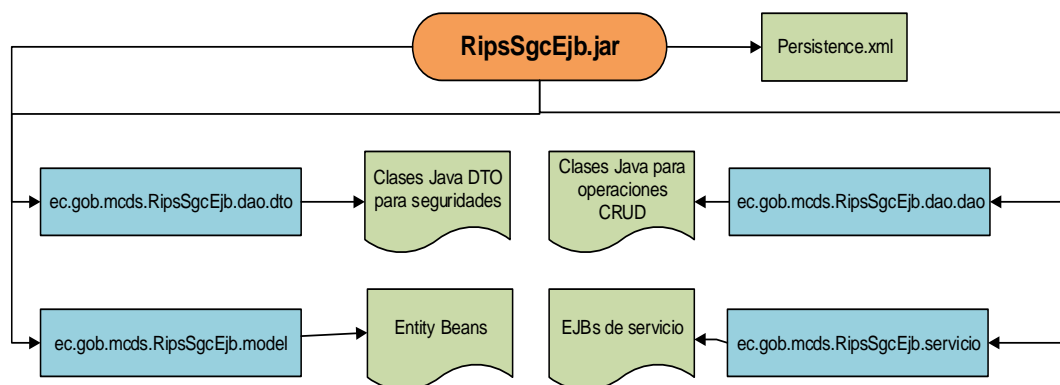


Figura 19 Modelo Vista de Desarrollo RipsSgcEjb

Modelo Vista de Desarrollo RipsSgcEjb

Módulo RipsSgcWeb.

Este módulo contiene clases Java que representan a Beans de Control (Managed Beans), archivos XHTML, imágenes, hojas de estilo y otras librerías que han sido empaquetados en un archivo llamado **RipsSgcWeb.war**

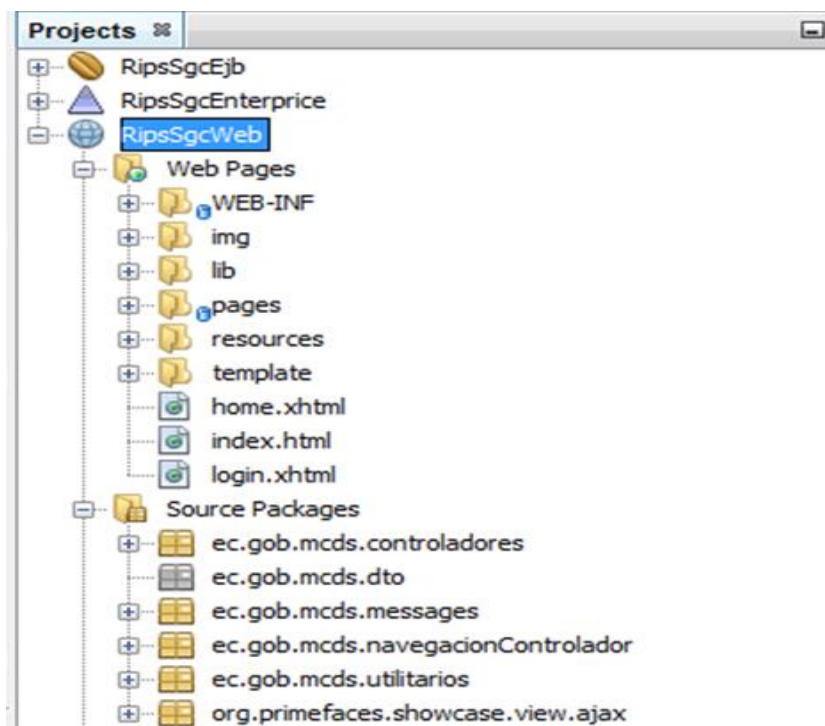


Figura 20 Vista de Desarrollo RipsSgcWeb

Vista de Desarrollo RipsSgcWeb

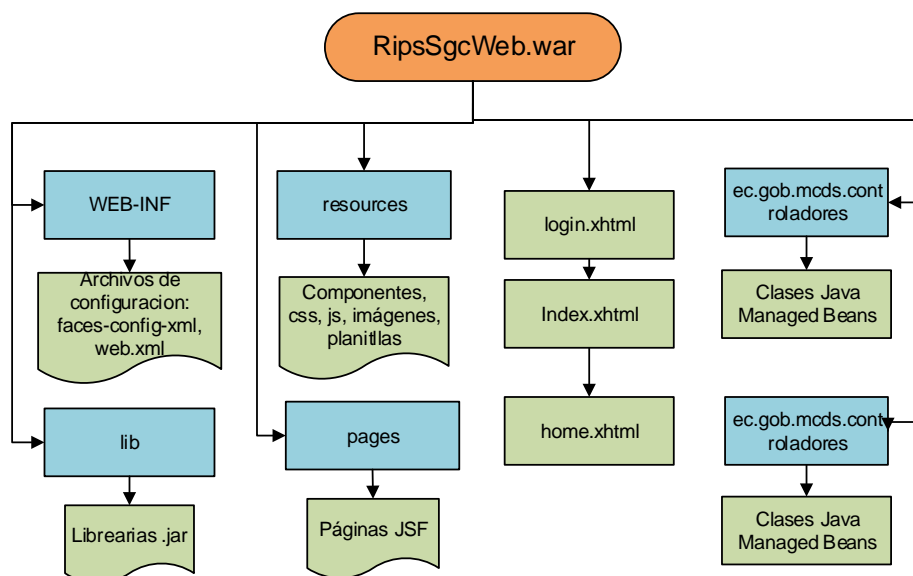


Figura 21 Modelo Vista de Desarrollo RipsSgcWeb

Modelo Vista de Desarrollo RipsSgcWeb

La estructura de paquetes y clases que debe respetarse para la construcción de las aplicaciones se define de la siguiente manera:

Capa EJB: Capa encargada de manejar lógica del negocio y administrar conexiones a la base de datos, la paquetería debe definirse como:

- **ec.gob.mcds.nombreAplicacion.model:** Dentro de este paquete deben estar todas las clases mapeadas a la base de datos el nombre de estas clases deben coincidir con el de las tablas de bases de datos.
- **ec.gob.mcds.nombreAplicacion.generico:** Dentro de este paquete debe estar 1 sola clase encargada de manejar las transacciones de Create Update Delete (Crud) la misma que recibe como parámetro cualquier objeto de tipo model esta debe ser heredada a todos los Ejbs Daos. Además esta clase no es un EJB.

- **ec.gob.mcds.nombreAplicacion.dao:** Dentro de este paquete deben estar todas las clases de acceso a la base de datos especificando el nombre de la clase y con la terminación Dao, por ejemplo si vamos a procesar la tabla rol de una bd el nombre de la clase sería RolDao. Además estas clases son EJBS del tipo Stateless.
- **ec.gob.mcds.nombreAplicacion.servicio:** Dentro de este paquete tiene que estar todas las clases de servicio las mismas que serán accedidas por la capa Web estas son EJBs de tipo Stateless las cuales heredarán las clases puente además el nombre de las clases deben terminar con el acrónimo Servicio, por ejemplo si vamos a hacer un servicio de la tabla rol sería RolServicio.
- **ec.gob.mcds.nombreAplicacion.dto:** En este paquete debe ir los pojos que servirán para almacenamiento de información a través de llenado de registros por parte de los DAOS y SERVICIOS.
- **ec.gob.mcds.nombreAplicacion.utilitarios:** En este paquete deben ir todas las clases de utilitarios de lógica de negocio como pueden ser ayudas de fechas, conexiones a otras bases de datos, etc.

Adicionalmente si se tiene que realizar tareas programadas estas deben ser programadas solo en la capa ejb y no deben ser usadas en la capa web estas deben ser Clases de tipo Singleton y el nombre del paquete debe ser **ec.gob.mcds.nombreAplicacion.tareasProgramadas**

(Santillán, Chrístian. DOCUMENTO DE ARQUITECTURA JEE PARA EL DESARROLLO DE SOLUCIONES INFORMÁTICAS PARA EL MINISTERIO COORDINADOR DE DESARROLLO SOCIAL, 2013-07-25. 10-11 p.)

4.09.04 Vista de Procesos

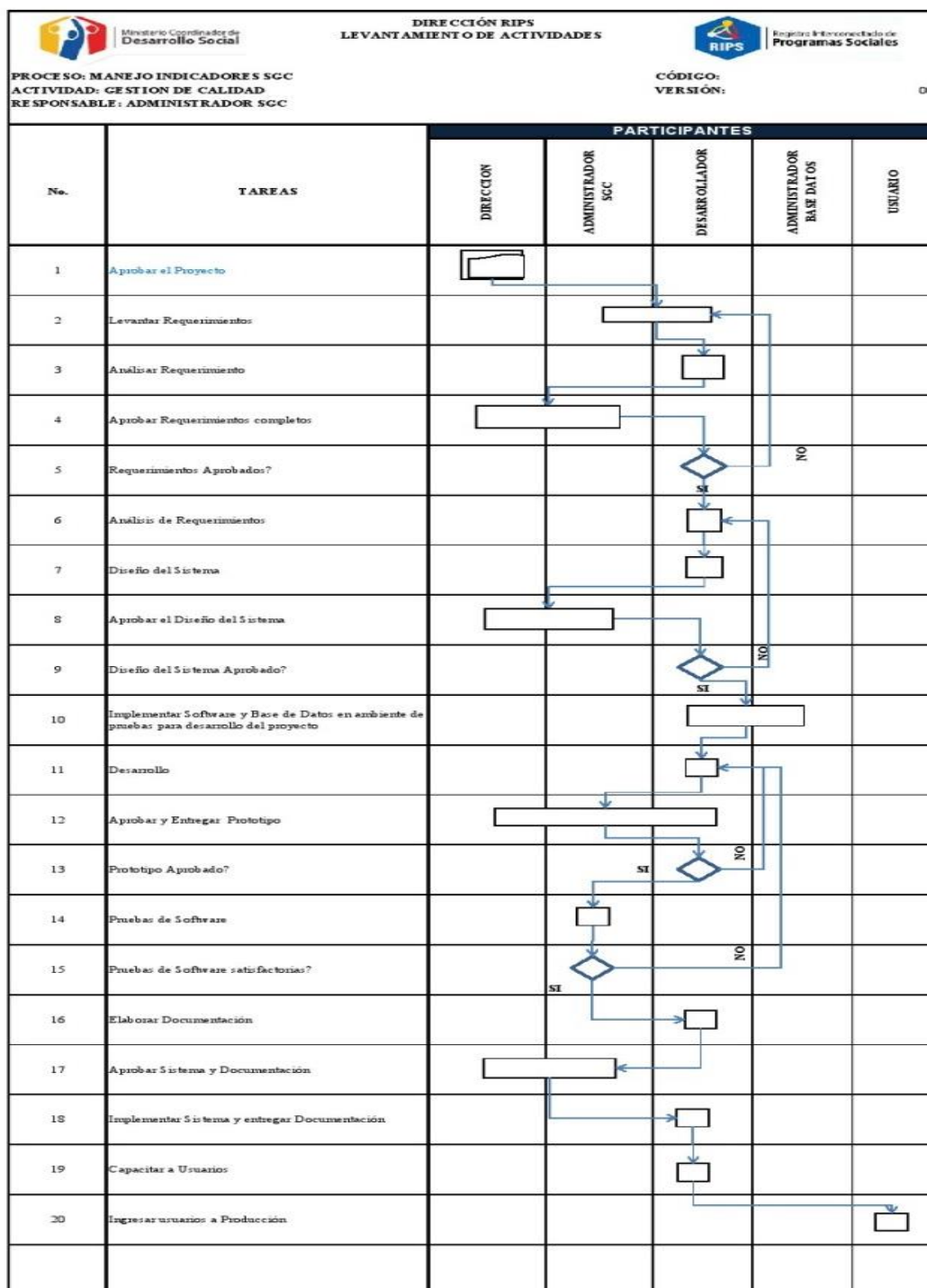


Figura 22 Vista de Procesos

Vista de Procesos

Contiene detalle de todas las actividades que se van a realizar en el desarrollo del diseño de la solución.

Capítulo V: Propuesta

5.01 Especificación de Estándares de Programación

Es importante la utilización de convenciones de código para facilitar la lectura del software y entender código nuevo mucho más rápido, enmarcando la distribución del código fuente en el estándar de cualquier otro producto.

Como estándares de Programación se utilizan los estándares de codificación del lenguaje Java, JAVA CODE CONVENTIONS – ORACLE, presentados en Java Language Specification, de Sun Microsystems Inc. (Java Code Conventions[en línea]. 2550 Garcia Avenue, Mountain View, California 94043-1100 U.S.A.: □ 1997, Sun Microsystems, Inc. All rights reserved. Disponible en Internet: <http://www.oracle.com/technetwork/java/codeconventions-150003.pdf>) (**Ver Anexo A.03 Java Code Conventions**)

5.02 Diseño de Interfaces de Usuario

Para el diseño de las interfaces que utilizará el usuario para interactuar con el sistema se utilizarán los siguientes componentes:

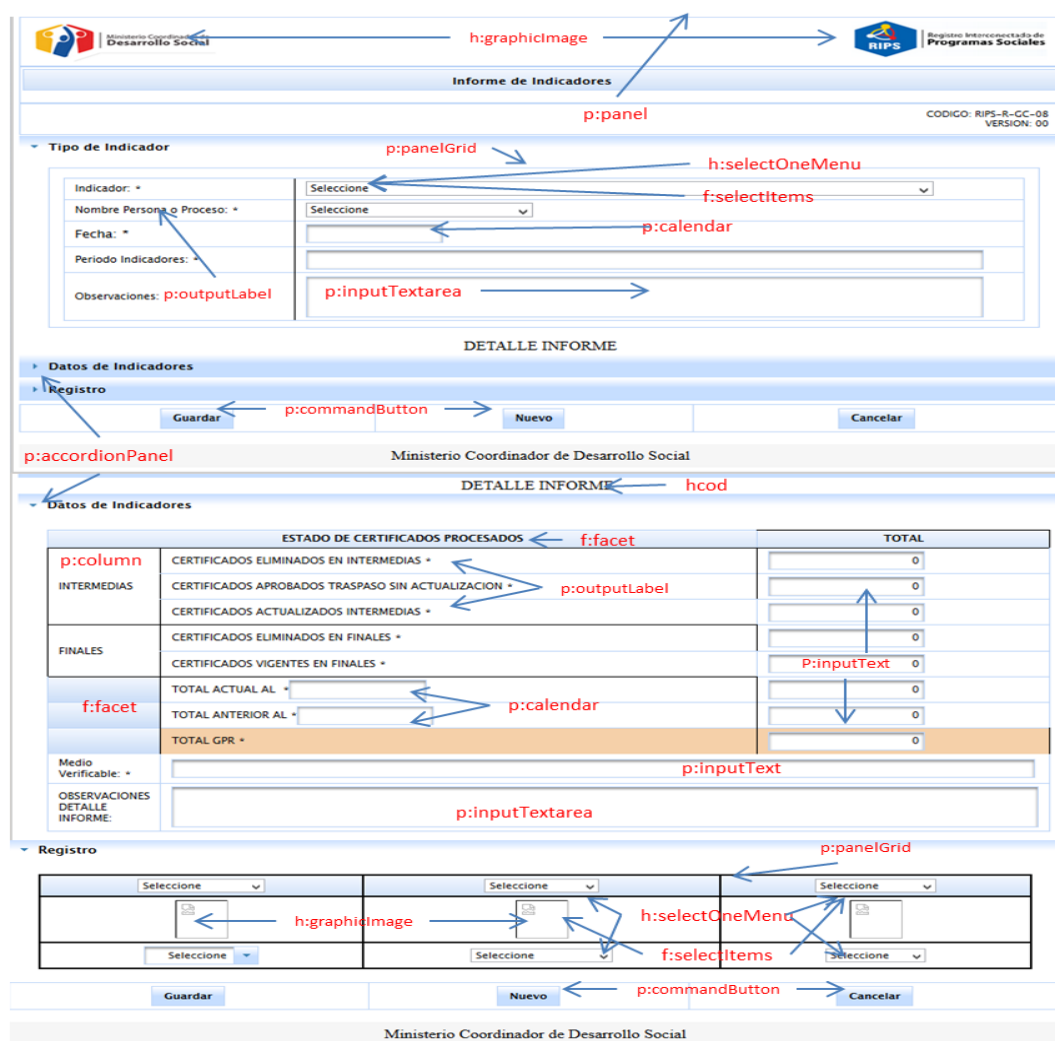
Todas las interfaces de usuario tendrán un estándar en cuanto a los colores, logotipos y la forma de navegación.

En la tabla adjunta se especifica las herramientas que se utilizaron para el diseño de las interfaces del usuario de la aplicación

Tabla 25 Componentes Diseño Interfaces

Componentes Diseño Interfaces

Componentes a utilizar para diseño de Interfaces	
Nomenclatura	Componente
h: =	"html"
f: =	"core"
ui: =	"facelets"
p: =	"primefaces"



The screenshot shows a web application interface for 'Informe de Indicadores' (Indicator Report) from the 'Ministerio Coordinador de Desarrollo Social' (Ministry of Social Development). The interface is annotated with PrimeFaces component names and their locations:

- h:graphicImage**: Points to the logos of the Ministry and RIPS at the top.
- p:panel**: Points to the main container area.
- p:panelGrid**: Points to the grid layout containing the form fields.
- h:selectOneMenu**: Points to the 'Indicador' dropdown menu.
- f:selectItems**: Points to the 'Nombre Persona o Proceso' dropdown menu.
- p:calendar**: Points to the 'Fecha' date picker.
- p:inputTextarea**: Points to the 'Observaciones' text area.
- p:outputLabel**: Points to the 'Periodo Indicadores' label.
- p:commandButton**: Points to the 'Guardar' (Save) button.
- p:accordionPanel**: Points to the 'Registro' (Record) section.
- h:cod**: Points to the 'DETALLE INFORME' (Report Detail) section.
- f:facet**: Points to the table headers for 'ESTADO DE CERTIFICADOS PROCESADOS'.
- p:column**: Points to the table columns.
- p:inputText**: Points to the input fields for 'TOTAL ACTUAL AL' and 'TOTAL ANTERIOR AL'.
- p:inputTextarea**: Points to the 'OBSERVACIONES DETALLE INFORME' text area.
- h:graphicImage**: Points to the document icons in the bottom table.
- h:selectOneMenu**: Points to the dropdown menus in the bottom table.
- f:selectItems**: Points to the dropdown menus in the bottom table.
- p:commandButton**: Points to the 'Nuevo' (New) button in the bottom table.

Figura 23 Diseño de Interfaces

Diseño de Interfaces

Los componentes registrados en las figuras y detallados en la tabla anterior son todos los utilizados para construir todas las interfaces del sistema.

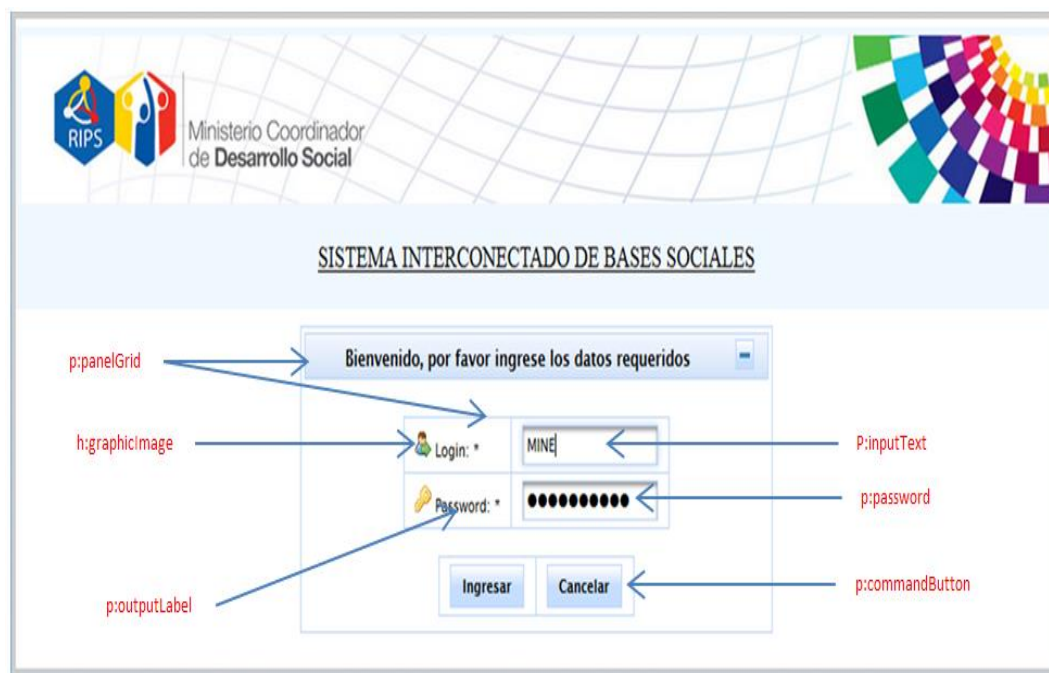


Figura 24

Diseño Interfaz Pantalla Ingreso

5.03 Especificación de Pruebas de Unidad.

5.3.1 Acceso al Sistema.

Tabla 26

Prueba de Unidad Ingreso Sistema

Caso de Prueba1: Ingresar al Sistema		
Entrada	Resultado Esperado	Resultado Obtenido
Ingreso al sistema mediante Login	Visualización de Módulo de Ingreso al Sistema. Validación de usuario y contraseña. Visualización de Menú Principal.	Visualización de Pantalla de inicio se adecua a lo solicitado. La validación de datos del usuario es correcta. El Menú Principal no se adecuaba al ancho de la pantalla pero fue corregido Prueba Correcta.

Tabla 27

Prueba de Unidad Mensajes error

Caso de Prueba2: Desplegar mensajes de error		
Entrada	Resultado Esperado	Resultado Obtenido
Ingresar usuario y/o contraseña errados	Desplegar mensajes de error al validar los datos ingresados.	El sistema de Ingreso no visualiza mensajes de error, simplemente no responde. Se corrige y se visualiza mensaje de error al ingresar al sistema. Prueba Correcta.

5.3.2 Módulo Menú Principal.

Tabla 28

Prueba de Unidad Módulo Principal

Caso de Prueba3: Acceder a los Módulos del Sistema desde el Menú Principal		
Entrada	Resultado Esperado	Resultado Obtenido
Selección de Módulos que son componentes del sistema	Visualización de Módulos al llamarlos desde las opciones del Menú Principal. Todos los Módulos deben mostrarse como componentes del Módulo Principal.	No se despliega el Menú Principal. Se soluciona llamada a página principal y se visualiza el Menú Principal. Prueba terminada de forma correcta.

5.3.3 Módulos Mantenimiento de Tabla Estado.

Tabla 29

Prueba de Unidad Mantenimiento Estado

Caso de Prueba4: Ingresar, Modificar o Eliminar registros de Tabla Estado		
Entrada	Resultado Esperado	Resultado Obtenido
Ingreso de Nuevo Estado.	Ingreso de Estado nuevo a la Base de Datos.	No notifica cuando se ingresa estado repetido, visualiza mensaje error.
Cambio de Datos de Estado.	Registro de las modificaciones realizadas.	No registra las modificaciones.
Eliminación de Estado.	Eliminación Lógica de los registros.	Solucionados los errores, la prueba Correcta.
Propiciar errores	Desplegar mensajes error.	

5.3.4 Módulos Mantenimiento de Responsables de Procesos.

Tabla 30

Prueba de Unidad Responsables de Procesos

Caso de Prueba5: Ingresar, Modificar o Eliminar registros de Tabla Responsables de Procesos		
Entrada	Resultado Esperado	Resultado Obtenido
Ingreso de Responsables. Cambio de Datos de Responsables de Procesos. Eliminación de Lógica de Responsables. Propiciar errores	Ingreso de nuevos Responsables de Procesos a la Base de Datos. Registro de las modificaciones realizadas. Eliminación Lógica de los registros. Desplegar mensajes de error.	Debe poder ingresarse más de una vez a una persona como responsable porque puede estar a carga de varios procesos al mismo tiempo. Incluir combo para seleccionar estado. Se completa lo solicitado y la prueba es Correcta.

5.3.5 Módulos Mantenimiento de Ingreso de Personal del Rips.

Tabla 31

Prueba de Unidad Personal Rips

Caso de Prueba6: Ingresar, Modificar o Eliminar registros de Tabla Personal del Rips		
Entrada	Resultado Esperado	Resultado Obtenido
Ingreso de datos de personas con roles de usuarios del Rips. Cambio de Datos de Personal Rips. Eliminación de Lógica de Personal Rips. Propiciar errores	Ingreso de datos de nuevos registros de personal del Rips a la Base de Datos. Registro de las modificaciones realizadas. Eliminación Lógica de los registros. Desplegar mensajes de error.	Al seleccionar el código del formato debe visualizarse el nombre y módulo que le corresponde. Al seleccionar estado debe visualizarse el nombre del estado. Debe indicarse cuales campos son obligatorios para llenar. Solucionado lo solicitado la prueba es correcta.

5.3.6 Módulos Mantenimiento de Lista Maestra de Registros.

Tabla 32

Prueba de Unidad Lista Maestra Registros

Caso de Prueba7: Ingresar, Modificar o Eliminar Lista Maestra de Registros		
Entrada	Resultado Esperado	Resultado Obtenido
Ingreso de datos de Lista Maestra de Registros.	Ingreso de datos de nuevos formatos de Lista Maestra de Registros a la Base de Datos.	Permite ingresar código de formato duplicado.
Modificación de registros de la Lista Maestra de Registros.	Registro de las modificaciones realizadas.	No tiene campo de estado para el formato.
Eliminación de Lógica de Lista Maestra de Registros.	Eliminación Lógica de los registros.	La fecha no está en el formato "yyyy-MM-dd"
Propiciar errores	Desplegar mensajes error.	Se corrige lo solicitado y la prueba es Correcta.

5.3.7 Módulos de Ingreso de Indicadores.

Tabla 33

Prueba de Unidad Ingreso Indicadores

Caso de Prueba8: Ingresar, Modificar, Eliminar Registros de Indicadores		
Entrada	Resultado Esperado	Resultado Obtenido
Ingreso de datos de Indicadores del SGC.	Ingreso de datos de nuevos Informes de indicadores a la Base de Datos.	El combo de indicadores debe estar ordenado alfabéticamente.
Modificación de registros de Indicadores.	Se debe poder seleccionar los indicadores de acuerdo al proceso.	Los responsables de proceso se deben visualizar solo los vigentes.
Eliminación de Lógica de Informes de Indicadores.	Se deben poder seleccionar los responsables de procesos.	La rúbrica del personal que elabora, revisa y aprueba el informe no son claras y son muy pequeñas.
Propiciar errores	Según la actividad que realiza una de las personas del Rips, se debe visualizar la rúbrica que le corresponda.	Los subtotales deben actualizarse cada vez que se ingresa un valor.
	Registro de las modificaciones realizadas.	No guardó primer intento.
	Eliminación Lógica de los registros.	Se solucionan todos los inconvenientes y la última prueba es Correcta.
	Desplegar mensajes error.	

5.3.8 Módulos de Reportes de Personal del Rips.

Tabla 34

Prueba de Unidad Reportes Personal Rips

Caso de Prueba9: Obtener reporte de registros de Tabla Personal del Rips		
Entrada	Resultado Esperado	Resultado Obtenido
Solicitar emisión de Reporte de Personal Rips.	Presentación del Reporte de Personal Rips solicitado. Desplegar mensajes de error.	El reporte es completo, es necesario poner un campo de selección. Se visualiza el reporte pero no salen los apellidos y nombres completos. Es necesario agregar campo orden en que aparecen en el combo. Se realizan las modificaciones solicitadas y la prueba es Correcta.

5.3.9 Módulos de Consulta de Informe de Indicadores.

Tabla 35

Prueba de Unidad Consulta Indicadores

Caso de Prueba10: Obtener Informe de Indicadores ingresados al sistema.		
Entrada	Resultado Esperado	Resultado Obtenido
Seleccionar Código de Formato y fecha de Reporte.	Formato de Informe de la fecha solicitada con la información que originalmente fue ingresado. Desplegar mensajes de error.	En campo de selección de código de formato está correcto pero fechas salen duplicadas. Los datos no corresponden al Informe solicitado El Código de formato corresponde pero no la fecha solicitada. Solucionados los inconvenientes surgidos el reporte solicitado se visualiza de acuerdo a los datos originalmente ingresados. Prueba Completa.

5.04 Especificación de Pruebas de Aceptación.

Las pruebas de aceptación de los módulos del sistema se realizan para determinar si cumplen con los requerimientos realizados o realizar los ajustes necesarios.

Tabla 36

Prueba Aceptación PA001 Actualizar Formatos

Identificador de la Prueba	PA001
Caso de Uso	Actualizar Formatos
Tipo de Usuario	Administrador del SGC
Objetivo de la Prueba	Probar todos los módulos que tiene habilitados para el rol del usuario y detectar las posibles fallas o sugerencias al sistema que pueda existir.
Secuencia de Eventos	
Verificar que los módulos de ingreso de indicadores estén de acuerdo a los formatos establecidos. Verificar que tanto los responsables de procesos como los indicadores estén actualizados y disponibles. Verificar que las rúbricas que deben aparecer en los formatos de indicadores estén completas y correspondan a la función que desempeñan en el proceso.	
Resultados Esperados	
Los módulos de ingreso de indicadores estén de acuerdo a los formatos establecidos y que constan en la Lista Maestra de Registros. El sistema permitió el ingreso y actualización de los responsables de procesos así como de todos los parámetros que se necesitan para cumplir con lo establecido. Las rúbricas que deben aparecer en los formatos de indicadores estén completas y correspondan a la función que desempeñan en el proceso.	
Comentarios	
El ingreso de los parámetros necesarios, para que funcione la aplicación, a la base de datos debe realizarse sin ningún contratiempo y por aplicación.	

Con las modificaciones realizadas en las Pruebas de Unidad y efectuadas las recomendaciones efectuadas, los Módulos para Actualizar Formatos cumplen con los requerimientos realizados.

Tabla 37

Prueba Aceptación PA002 Elaborar Informes

Identificador de la Prueba	PA002
Caso de Uso	Elaborar Informes
Tipo de Usuario	Administrador del SGC Responsable de Proceso
Objetivo de la Prueba	Probar todos los módulos que tiene habilitados para el rol del usuario y detectar las posibles fallas o sugerencias al sistema que pueda existir.
Secuencia de Eventos	
<p>Acceso a la Aplicación por medio de un navegador a la URL designada.</p> <p>Verificar que los colores, logotipos y demás estándares solicitados en los requerimientos han sido cumplidos.</p> <p>Verificar que consten todos los módulos que deben ser parte del aplicativo.</p> <p>Verificar la navegabilidad de la aplicación de forma segura.</p> <p>Verificar la seguridad al momento de ingresar al sistema con usuario y contraseña.</p> <p>Verificar que los cálculos solicitados en los formatos específicos se estén realizando correctamente.</p> <p>Verificar que los tiempos de respuesta del sistema estén dentro de lo que es considerado normal.</p> <p>Ingresar informes para medir si el sistema es amigable, sencillo de utilizar y efectivo.</p>	
Resultados Esperados	
<p>Ingreso sin inconvenientes a través de cualquier navegador.</p> <p>Se respetaron los estándares, han sido cumplidos los colores, logotipos y demás estándares solicitados en los requerimientos han sido fuente en la presentación visual del aplicativo..</p> <p>Constan todos los módulos que deben ser parte del aplicativo.</p> <p>Navegabilidad de la aplicación de forma segura y sin ningún inconveniente.</p> <p>La seguridad al momento de ingresar al sistema con usuario y contraseña está comprobada y los usuarios ingresan con el rol específico que están autorizados.</p> <p>Los cálculos que se tienen que realizar al llenar algunos formatos se están realizando correctamente.</p> <p>Los tiempos de respuesta del sistema estén dentro de lo que es considerado normal.</p> <p>Ingreso de indicadores en los formatos establecidos y que son parte del sistema.</p>	
Comentarios	
<p>Todas las pruebas deben ser realizadas buscando generar errores en el sistema, verificando todos los detalles de acuerdo a lo consignado en el levantamiento de requerimientos.</p> <p>Las pruebas deben cubrir la forma (presentación de la aplicación) como la parte técnica con enfoque a cumplir con lo solicitado en los requerimientos.</p>	

Tabla 38

Prueba Aceptación PA003 Consultar Informes

Identificador de la Prueba	PA003
Caso de Uso	Consulta Informes e Indicadores
Tipo de Usuario	Director Usuario
Objetivo de la Prueba	Probar todos los módulos que tiene habilitados para el rol del usuario y detectar las posibles fallas o sugerencias al sistema que pueda existir.
Secuencia de Eventos	
<p>Ingresar al sistema con usuario y contraseña.</p> <p>Verificar que el ingreso se realice con el rol asignado a cada usuario y pueda visualizar solo los módulos que le están asignados.</p> <p>Realizar Consulta de informes de indicadores.</p> <p>Verificar los datos de los informes con los originales para comprobar fiabilidad.</p> <p>Comprobar navegabilidad del sistema</p>	
Resultados Esperados	
<p>Ingreso al sistema con usuario y contraseña validada.</p> <p>Los módulos que el usuario pueda visualizar al ingresar al sistema serán exclusivamente los que le estén autorizados de acuerdo al rol asignado a cada usuario.</p> <p>Visualizar y confirmar los informes de indicadores consultados.</p> <p>Los datos de los informes son iguales a los que originalmente se ingresaron permitiendo comprobar la fiabilidad del sistema.</p> <p>Navegabilidad del sistema fácil y amigable.</p>	
Comentarios	
<p>Realizar las pruebas tratando de presionar al sistema en el proceso para medir las respuestas solicitadas.</p>	

La consulta de indicadores ingresados al sistema y los reportes de indicadores del sistema fueron revisados y se determina que cumplen con los requerimientos realizados y las sugerencias de las pruebas fueron acogidas.

5.05 Especificación de Pruebas de Carga.

Pruebas para determinar y validar la respuesta de la aplicación cuando es sometida a una carga de usuarios y/o transacciones que se espera en el ambiente de producción.

Tabla 39**Prueba de Carga PC001**

Identificador de la Prueba	PC001
Tipo de prueba	Ingresar información en un módulo específico con datos quemados en el código en los campos obligatorios de los módulos.
Objetivo de la Prueba	Verificar rendimiento del sistema midiendo el número de transacciones por minuto al ingresar información.
Descripción El usuario deberá realizar el ingreso de información de personas, usuarios, roles e informes por medio del aplicativo. Los datos de los campos obligatorios de los módulos deben estar cargados por código para evitar que la destreza en la transcripción de información sea un factor que incida en la prueba.	
Resultados Esperados Se espera un promedio mínimo de 50 transacciones por minuto	
Comentarios El promedio de transacciones por minuto fue de 52 por lo que se alcanzó el resultado esperado en esta prueba por lo que se considera Prueba Superada.	

Tabla 40**Prueba de Carga PC002**

Identificador de la Prueba	PC002
Tipo de prueba	Ingresar y consultar informes
Objetivo de la Prueba	Medir tiempo de respuesta del sistema al momentos de consultar y guardar información por medio del aplicativo
Descripción El usuario deberá ingresar información en los formatos de reportes y luego consultarlos por medio de la aplicación y medir el tiempo de respuesta que tiene el sistema a las acciones realizadas.	
Resultados Esperados Ingreso al Sistema: un segundo. Acceso al Menú Principal: un segundo y medio Acceso a los otros módulos: Máximo dos segundos Ingreso de datos de reportes: Máximo dos segundos. Consulta de Reportes: Máximo dos segundos.	
Comentarios Al probar el aplicativo por el lapso de 30 minutos, los tiempos de respuesta se mantuvieron por debajo del máximo esperado, por lo que la Prueba está Superada.	

Tabla 41

Prueba de Carga PC003

Identificador de la Prueba	PC003
Tipo de prueba	Ingreso al Sistema de 60 Usuarios de forma simultáneamente.
Objetivo de la Prueba	Verificar que no baje el rendimiento del sistema al ingresar el número máximo de usuarios que tiene la Dirección.
Descripción	
Realizar el ingreso al sistema para realizar pruebas funcionales e ingreso de información por medio de la aplicación de un total de 60 usuarios (capacidad máxima de usuarios que tiene la Dirección=	
Resultados Esperados	
Los tiempos de respuesta del sistema y el rendimiento sean similares a los que se obtuvieron en las pruebas con Identificador PC001 y PC002	
Comentarios	
Después del ingreso al sistema de 60 usuarios de forma simultánea y probar la aplicación por el lapso de una hora, no existieron ni disminución de los tiempos de respuesta a las peticiones realizadas ni baja de rendimiento ni interrupciones del sistema. Se alcanzaron los resultados esperados. Prueba Superada.	

5.06 Configuración del Ambiente Mínimo/Ideal.

Características de Servidores.

Servidor de BDD es el 10.34 en el cual está instalado Postgres 9.2, las características de espacio de disco y memoria RAM constan en el Anexo “Estado Actual de Servidores” que detalla las características y configuración de los servidores necesarios para su funcionamiento.

El ambiente utilizado es un entorno de Virtualización: RedHatEnterprise Virtualization. Cada cuchilla que forma parte del arreglo de host para el cluster de virtualización del chasis Blade es un servidor Blade HS23. El servidor 10.34 cuenta con un microprocesador de 8 cores (4 sockets de dos núcleos cada uno), esto dentro del ambiente de virtualización. **(Ver Anexo A.13 Estado Actual de Servidores).**

En el servidor 36 que tiene sistema operativo 5.1 está instalado Glasshish 3.0.1, servidor de aplicaciones que maneja todas las peticiones del aplicativo.

Características de Equipo para usuario.

El usuario debe tener un equipo con las características detalladas o superior.

Procesador Pentium III, 600 MHz o superior, Memoria 512 Mb o superior, Disco Duro, 300 Mb disponibles, Video 8 Mb resolución 800 x 600

El software que debe tener el equipo del usuario adicional al básico de cualquier sistema (sistema operativo) debe tener instalado un navegador (Google Chrome o Mozilla Firefox), no debe utilizarse Internet Explorer porque existen problemas con cierta funcionalidad de algunos de los componentes.

Estructura de Archivos

Es la representación de datos en archivos que asegura que los usuarios y programas pueden acceder y escribir a los archivos.

Un buen diseño de estructuras de archivos brindará acceso a grandes capacidades de información, sin gastar tiempo de espera

La definición de la estructura de archivos del proyecto es muy importante porque es la fuente de la organización en la etapa de desarrollo, implementación y mantenimiento del proyecto y la manera que esa estructura se presenta el momento del desarrollo.

Estructura de Archivos en un Programa en Java

La estructura de un programa en Java tiene como objetivo describir la estructura del código fuente de una aplicación Java y presentar los conceptos de comentario y de identificador dentro del código fuente de un programa, principalmente en los comentarios y los identificadores. La estructura de un programa de Java permite a los programadores de cualquier otro lenguaje leer código en Java sin mucha dificultad.

Java emplea siempre la Programación Orientada a Objetos por lo que todo el código se incluye dentro de las clases (combinaciones de datos, sean constantes o variables, y métodos).

Las sentencias se separan entre sí por punto y coma.

El código debe estar comentado para facilitar la interpretación de cualquier persona que lo analice.

```
/**
```

```
* Ejemplo de formato de comentario a utilizar
```

```
* en nuestros archivos de código
```

```
*/
```

Los identificadores son nombres que se les asignan a variables, métodos, clases, etc, en el código fuente de un programa. Los identificadores sólo existen en el código del programa y se utiliza los siguientes parámetros:

Un identificador comienza por una letra, un carácter de subrayado (_) o un carácter de dólar (\$). Aunque no se recomienda emplear el carácter \$, ya que el compilador suele utilizarlos de forma interna para crear identificadores propios.

Los siguientes caracteres pueden ser también dígitos. Pero no pueden emplearse espacios en blanco u otros caracteres como el signo de interrogación (?) o el signo del tanto por ciento (%).

No hay límite máximo de caracteres.

En los identificadores del código fuente de un programa en Java se distinguen las mayúsculas de las minúsculas. Por ejemplo, casa, CASA y Casa son tres identificadores diferentes.

Pueden incluir caracteres Unicode, con lo que se pueden emplear secuencias de escape

/uxxxx para representar estos caracteres.

No puede emplearse el identificador de una variable o cualquier otro elemento del código fuente del programa para otro ya existente en el mismo bloque. Excepción: variable miembro y local con el mismo identificador.

Existe una serie de palabras reservadas que no pueden emplearse como identificadores.

Variables de Ambiente

Las variables son utilizadas para almacenar números, secuencias de caracteres e informaciones en general. Una Variable de Ambiente es una variable establecida por el sistema o por un usuario, en la línea de comando del sistema, y puede ser usada por cualquier programa. Los diferentes programas (sistemas operativos, software de

desarrollo, aplicaciones, etc) tienen diversas variables predefinidas tales como, nombre de usuario, directorio y el nombre del sistema.

Las variables de sistema más comunes son:

Tabla 42

Variables de Ambiente Linux

Variable	LINUX
\$EDITOR	Contiene la ruta al editor de texto, usualmente un editor liviano
\$PAGER	Contiene la ruta al programa usado para listar el contenido de archivos
%DATE%	Se expande a la fecha actual usando el mismo formato que el comando DATE.
%TIME%	Se expande a la hora actual usando el mismo formato que el comando TIME.
%RANDOM%	Se expande a un número decimal aleatorio entre 0 y 32767.
%ERRORLEVEL%	Se expande al valor de NIVEL DE ERROR actual (usualmente, el código de retorno del último comando externo ejecutado).
\$PATH	Contiene una lista separada por dos puntos de directorios en los cuales el intérprete de comandos buscará los archivos ejecutables que no se invocan con una ruta.
\$HOME	Contiene la ubicación del directorio de usuario en Linux.
\$HOSTNAME	Contiene el nombre de host del sistema.
\$USER	Nombre de usuario actual logueado en el sistema.

Nota: Detalle de variables de ambiente que son utilizadas cuando se trabaja en ambiente Linux.

La mayoría de las variables son usadas para nombres de caminos de bibliotecas e informaciones como tipo y plataforma del sistema operacional. No todas las variables están presentes en el sistema del usuario.

Tabla 43

Variables de Ambiente Windows

Variable	Windows Vista/7
%ALLUSERSPROFILE% (%PROGRAMDATA%)	C:\ProgramData
%APPDATA%	C:\Users\{username}\AppData\Roaming
%COMPUTERNAME%	{nombre de ordenador}
%COMMONPROGRAMFILES%	C:\Archivos de programa\Archivos Comunes
%COMMONPROGRAMFILES(x86)%	C:\Archivos de programa (x86)\Archivos Comunes
%COMSPEC%	C:\Windows\System32\cmd.exe
%HOMEDRIVE%	C:\
%HOMEPATH%	\Usuarios\{username}
%LOCALAPPDATA%	C:\Usuarios\{username}\Application Data\Local
%LOGONSERVER%	\\{domain_logon_server}
%PATH%	C:\Windows\system32, C:\Windows, C:\Windows\System32\Wbem
%PATHEXT%	.com, .exe, .bat, .cmd, .vbs, .vbe, .js, .jse, .wsf, .wsh, .msc
%PROGRAMFILES%	C:\Archivos de programa\
%PROGRAMFILES(X86)%	C:\Archivos de programa (x86) (solo en versiones 64-bit)
%PROMPT%	Código del formato de Cmd.exe. El código es normalmente \$P\$G
%SystemDrive%	C:\
%SystemRoot%	C:\Windows
%TEMP% y %TMP%	C:\Users\{username}\AppData\Local\Temp
%USERDOMAIN%	{userdomain}
%USERNAME%	{username}
%USERPROFILE%	C:\Users\{username}
%WINDIR%	C:\Windows
%PUBLIC%	C:\Users\Public
%PSModulePath%	%SystemRoot%\system32\WindowsPowerShell\v1.0\Modules\

(Otros: Variables de entorno en Windows 7/8/2008 y XP/2003 [en línea]:

<[http://nksistemas.com/variables-de-entorno-en-windows-782008-y-](http://nksistemas.com/variables-de-entorno-en-windows-782008-y-xp2003/)

xp2003/>[Consulta:2 de Octubre 2014])

Detalle de variables de ambiente que son utilizadas cuando se trabaja en ambiente Windows.

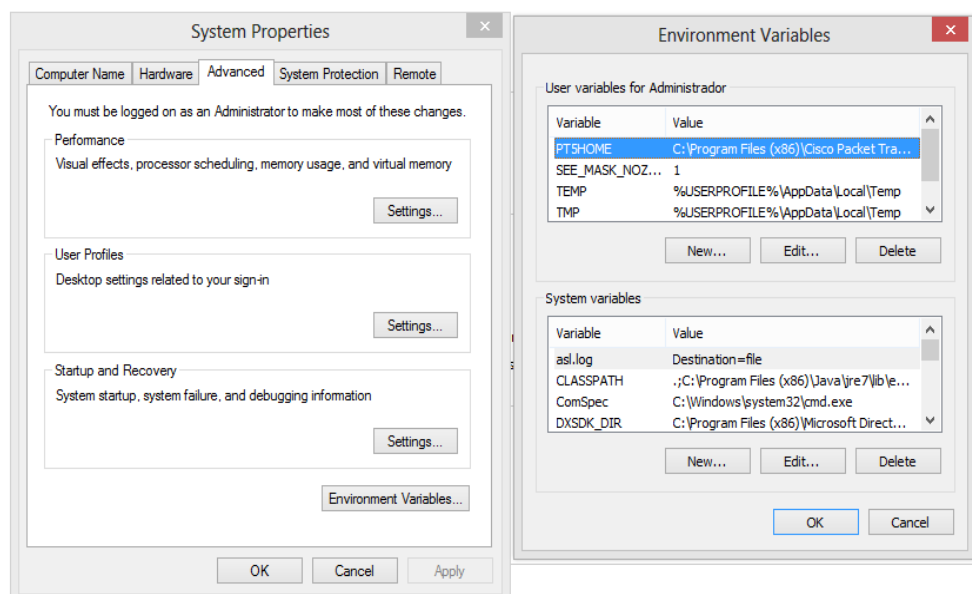


Figura 25

Variable de Ambiente Sistema

Variables de ambiente utilizados por el sistema según la configuración del usuario logueado.

La interfaz del servidor de aplicaciones utilizada es la de Glassfish.

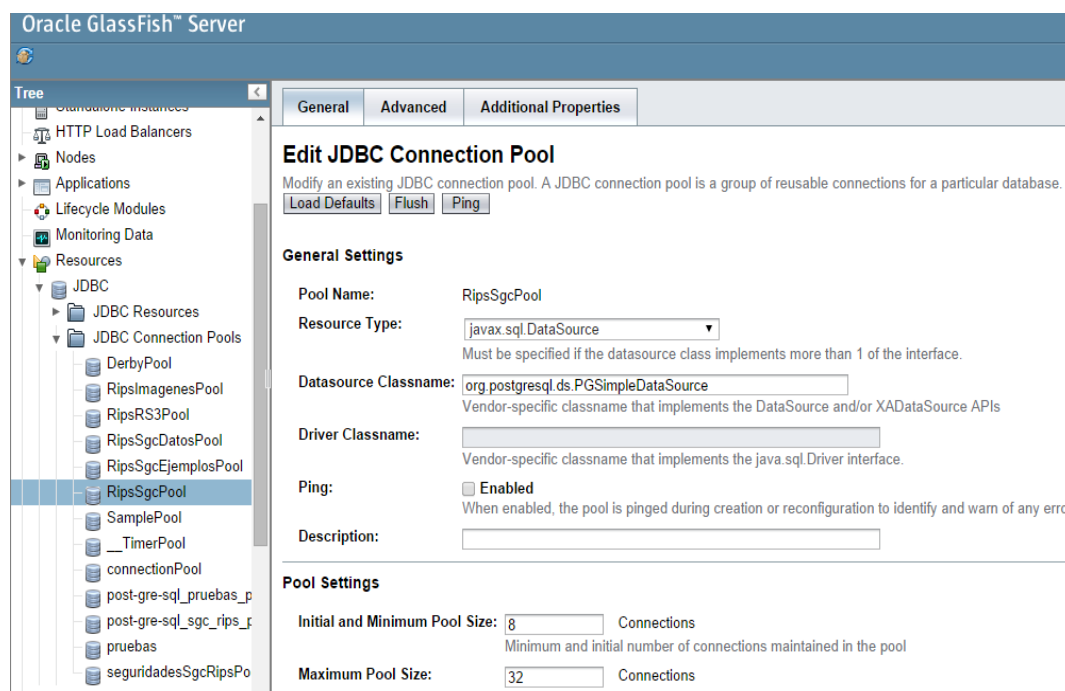


Figura 26

Interfaz Administrador Aplicaciones

Para interactuar con el administrador de aplicaciones se utiliza GlassFish Server Administration Console, versión Oracle GlassFish Server 3.1.2.2 (build 5).

Es el encargado de administrar las conexiones a las bases de datos y su rendimiento de acuerdo a la carga de peticiones que tenga.

Capítulo VI: Aspectos Administrativos

6.01 Recursos.

6.01.01 Recursos Humanos.

Tabla 44

Recursos Humanos

RECURSOS HUMANOS		
Nombre	Actividad	Responsabilidad
Fabián Ruiz Buitrón	Desarrollo del proyecto.	Investigar, diseñar y desarrollar el sistema de control de indicadores para la Dirección del Registro Interconectado de Programas Sociales.
Ingeniero Hugo Heredia	Guiar el desarrollo del proyecto.	Revisar el progreso de la elaboración de la documentación del proyecto y desarrollo del sistema
Ingeniero	Revisar el desarrollo del proyecto.	Asegurar que el desarrollo del proyecto cumpla con las normas establecidas.
Economista Guadalupe Muñoz	Acompañar el desarrollo del proyecto.	Verificar que el proyecto cumpla con los requerimientos del usuario

Detalla el recurso humano que acompañó la construcción y desarrollo del proyecto, iniciando con el tutor que trazó el camino a recorrer, el revisor que asegura la calidad en el cumplimiento con las normas establecidas, la personas responsable de acompañar el proceso por parte del cliente verificando el cumplimiento de los requerimientos y el desarrollador que se encargó de superar los inconvenientes surgidos en el avance del proyecto.

6.01.02 Recursos Materiales

Tabla 45

Recursos Materiales

RECURSOS MATERIALES			
Descripción	Cantidad	Valor Unitario	Valor Total
Programa Académico de Profesionalización y Transferencia de Tecnología	1	722	722
Resmas de papel 75g (A4)	8	3.8	30.4
Impresiones	2500	0.05	125
Empastado	1	40	40
Anillados	2	10	20
DVD	2	1.5	3
TOTAL RECURSOS			940.4

Detalle de los recursos materiales y recursos financieros que fueron necesarios para el desarrollo del proyecto.

6.01.03 Recursos Tecnológicos.

Tabla 46

Recursos Tecnológicos

RECURSOS TECNOLÓGICOS		
HARDWARE	SOFTWARE	MANUALES Y GUÍAS
Servidores (Detalle Anexo)	Sistemas Operativos (Windows 8, Centos 5.9, Ubuntu 14.2)	Norma ISO 9001:20089
Computadora personal	Protección Antivirus	Normativa y Documentación del Sistema Gestión de Calidad
Router	PostgresQL 9.2 (Administrador de Base de Datos)	
Línea Telefónica	Glassfish 3.1 (Servidor de Aplicaciones)	
Conexión a Internet	NetBeans 7.2	
	Power Designer (Modelado de Base de Datos)	
	Rational Rouse	

Detalle de hardware, software y documentación técnica que fue necesaria para el proyecto.

6.02 Presupuesto.

Tabla 47

Presupuesto

PRESUPUESTO		
	Fuente	Valor Total
Programa Académico de Profesionalización y Transferencia de Tecnología	Recursos propios	722
Resmas de papel 75g (A4)	Recursos propios	30.4
Impresiones	Recursos propios	125
Empastados	Recursos propios	40
Anillados	Recursos propios	20
DVD	Recursos propios	3
Total Recursos Propios		940.4
Total Presupuesto		940.4

Nota: El detalle del presupuesto refleja que el valor de la inversión realizada en el proyecto se financió con recursos propios.

6.03 Cronograma.

Especifica los tiempos e hitos a cumplir según lo programado. (**Ver Anexo A.14 Cronograma**).

Capítulo VII: Conclusiones y Recomendaciones

7.01 Conclusiones.

Con la información obtenida a través de las diferentes etapas de avance en el desarrollo del proyecto, se ha evidenciado que no es suficiente tener implementados procesos bien definidos y documentados, inclusive bajo parámetros de normas internacionales como la ISO 9001:2008 que cubija la gestión de la calidad, si no se cuenta con apoyo de la tecnología que permita cumplir con agilidad en el cumplimiento de los requerimientos, tanto de lo establecido en los manuales de los procesos como en las demandas de cada uno de los usuarios del sistema.

Sin los controles que deben garantizar la calidad de la gestión realizada, el cumplimiento de lo ya definido y el poder continuar de manera eficiente con la mejora continua, se hace muy complicado mantener los altos estándares de calidad exigidos, demandando mucho tiempo de las personas responsables de esos controles para evitar afectar de manera importante a la disponibilidad de la información necesaria para la toma de decisiones.

Por eso se hace necesario pasar esas labores que se realizan manualmente, a que sean ejecutadas de forma sistematizada, con las debidas seguridades, mejorando el control, invirtiendo mucho menos tiempo de revisiones, dando agilidad a los procesos establecidos para orientarlos en busca de mejorar la integridad, confidencialidad y disponibilidad de la información que se maneja.

7.02 Recomendaciones.

Implementar, documentar y apuntalar a los procesos establecidos con todas las facilidades que los recursos y la tecnología disponible permitan implementar, incentivando al personal a la detección temprana de dificultades en el cumplimiento de las metas planteadas y realizar sugerencias para controlar o disminuir el problema y evitar o al menos atenuar las consecuencias.

El personal debe tener facilidad de acceso a la normativa, información y documentación necesarias para cumplir con sus labores, con la seguridad y tranquilidad de que lo que está a su alcance es lo más actualizado en versionamiento y que siempre estará disponible cuando lo requiera y de esa manera estandarizar el conocimiento del todo el personal con respecto reglas, manuales y procedimientos internos.

Bibliografía

BATANI, Cery Navathe (2012) Database design using entity-relationship diagrams.

Tylor & Francis. Disponible en Internet en: <http://books.google.com.ec/books>.

Dirección RIPS. Política y Objetivos de la Calidad. *Política de la Calidad*, 3.

Java Code Conventions[en línea]. 2550 Garcia Avenue, Mountain View, California 94043-1100 U.S.A.: 1997, Sun Microsystems, Inc. All rights reserved. Disponible en Internet: <http://www.oracle.com/technetwork/java/codeconventions-150003.pdf>

Melgarejo, A.(2014) Algorítmica 2. Rational Rose – Diagrama de clases.

Recuperado de

<https://docs.google.com/file/d/0Bypkey8lnLjfemNYZEd4YktnVG8/edit?pli=1>.

Otros: Variables de entorno en Windows 7/8/2008 y XP/2003 [en línea]:

<[http://nksistemas.com/variables-de-entorno-en-windows-782008-y-](http://nksistemas.com/variables-de-entorno-en-windows-782008-y-xp2003/)

[xp2003/](http://nksistemas.com/variables-de-entorno-en-windows-782008-y-xp2003/)>[Consulta:2 de Octubre 2014]

PRESSMAN, Roger S. (1998) Ingeniería del software un enfoque práctico (Edic:

1998). Tomo 1 Caracas – Venezuela: Edit Quinta

Reducir Código [en línea] Javahispano.org. mayo 27, 2014

:<<http://www.javahispano.org/java-se/post/2355755>>

Santillán, Chrístian. (2013). Documento de Arquitectura Jee para el desarrollo de

soluciones informáticas para el Ministerio Coordinador de Desarrollo Social, 4-7.

CONTROL DE INDICADORES DE GESTIÓN DE CALIDAD MEDIANTE APLICACIÓN WEB
PARA LA DIRECCIÓN RIPS DEL MINISTERIO COORDINADOR DE DESARROLLO SOCIAL

Santillán, Christian. (2013). Documento de Arquitectura Jee para el desarrollo de soluciones informáticas para el Ministerio Coordinador de Desarrollo Social, 7-8.

Santillán, Christian. (2013). Documento de Arquitectura Jee para el desarrollo de soluciones informáticas para el Ministerio Coordinador de Desarrollo Social, 10-11.

ANEXOS

A.01

Matriz de Marco Lógico

Matriz de Marco Lógico

	Indicadores	Medios de Verificación	Supuestos
Finalidad del Proyecto Facilidad toma de decisiones, cumplimiento Norma ISO 9001:2008 y cumplimiento metas planificadas.	El tiempo de consolidación de indicadores para la toma de decisiones se reducirá substancialmente	Los informes de indicadores subidos en el sistema	Posibilidad de tomar accione preventivas o correctivas para el cumplimiento de las metas planificadas.
Propósito del Proyecto Adecuado Control de Indicadores de Gestión de Calidad.	Manejo seguro de versionamiento de formatos controlados solamente a través de aplicativo	Verificación de uso de formatos controlados a través de la Lista Maestra de Registros	Cumplir con lo establecido por la Norma ISO 9001:2008 en Gestión de Calidad.
Componentes del Proyecto Disponibilidad únicamente de última versión de formatos. Formatos Protegidos Acceso por aplicativo a formatos e indicadores	Requerimientos levantados y datos de la Lista Maestra del Sistema de Gestión de Calidad	Informes de Prueba de software	El cumplimiento se evidencia al poner a disposición de los usuarios el sistema.
Actividades Levantamiento y consolidación de todos los formatos de registros. Lista de todos los usuarios que se van a habilitar al Sistema. Lista de parámetros necesarios para que el sistema requerido funcione. Determinar el soporte para las pruebas a realizar durante el proceso.	Parámetros necesarios para el desarrollo e implementación del sistema	Listas de cada uno de los datos necesarios para que inicie el sistema satisfactoriamente	Se cuenta con el apoyo de todo el equipo de la Dirección para el desarrollo de la solución

A.02

Entrevista EN-001

Entrevista		
Identificador: EN-001		
Preguntas	Objetivos	Análisis Posterior
Forma de Acceso al Sistema	Identificar si el sistema va a estar restringido solo a usuarios autorizados o es público.	El sistema debe controlar que puedan tener acceso solo los usuarios autorizados en base a un código y contraseña.
Los Usuarios según las actividades que realizan tendrán privilegios específicos o todos tendrán el mismo nivel de responsabilidad.	Definir si los usuarios tendrán diferentes roles y la forma de definir y administrar el alcance de los mismos.	Los usuarios tendrán diferentes roles definidos por una persona responsable de establecerlos, y es necesario un sencillo sistema de administración de roles y privilegios que deberán ser administrados por un usuario autorizado.
Cuál debe ser la finalidad que debe tener el sistema a desarrollar.	Conocer qué motivo la necesidad de desarrollar el sistema.	Tener todos los formatos de informes de indicadores del Sistema de Gestión de Calidad que sean válidos y que siempre conste la última versión registrada en la Lista Maestra de Registros. El ingreso de los indicadores deberá realizarse simplemente completando los campos que cada formato requiera y debe garantizar la recuperación de los reportes cuando el usuario lo requiera, manteniendo la misma versión con el que fue ingresado y con los mismos valores originales.
Quiénes son las personas encargadas del ingreso de informes de indicadores	Dar claridad a la utilidad y las personas que van a utilizar cada uno de los formatos requeridos por el sistema.	Los formatos serán utilizados por los responsables de los procesos, pero los formatos aprobados en el SGC pueden ser utilizados por más de un proceso, con ciertas modificaciones establecidas que deberán ser tomadas en cuenta.
Cuál es el documento fuente que servirá de guía para elaborar los formatos de informes y cuál es el alcance del sistema.	Definir un parámetro en el que se sustente la planificación del sistema y hasta dónde debe llegar su funcionalidad.	Queda establecido que el documento guía será la Lista Maestra de Registros que contiene la totalidad de los formatos aprobados para el SGC y que el sistema se enfoca en permitir el ingreso de informes de indicadores del SGC, controlar el uso de versiones actualizadas y vigentes y recuperar los informes cuando sea requerido por el usuario. Queda también establecido que el diseño de los formatos

deberán ser los mismos, en todos su campos, que los que actualmente están vigentes y en uso en el SGC,

A.03

Java Code Conventions

Java Code Conventions

September 12, 1997

Copyright Information

© 1997, Sun Microsystems, Inc. All rights reserved.
2550 Garcia Avenue, Mountain View, California 94043-1100 U.S.A.

This document is protected by copyright. No part of this document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any.

The information described in this document may be protected by one or more U.S. patents, foreign patents, or pending applications.

TRADEMARKS

Sun, Sun Microsystems, Sun Microelectronics, the Sun Logo, SunXTL, JavaSoft, JavaOS, the JavaSoft Logo, Java, HotJava Views, HotJavaChips, picoJava, microJava, UltraJava, JDBC, the Java Cup and Steam Logo, "Write Once, Run Anywhere" and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

UNIX® is a registered trademark in the United States and other countries, exclusively licensed through X/Open Company, Ltd.

Adobe® is a registered trademark of Adobe Systems, Inc.

Netscape Navigator™ is a trademark of Netscape Communications Corporation.

All other product names mentioned herein are the trademarks of their respective owners.

THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THE DOCUMENT. SUN MICROSYSTEMS, INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

June 2, 1997

1	Introduction	1
1.1	Why Have Code Conventions	1
1.2	Acknowledgments	1
2	File Names	1
2.1	File Suffixes	2
2.2	Common File Names	2
3	File Organization	2
3.1	Java Source Files	2
3.1.1	Beginning Comments	3
3.1.2	Package and Import Statements	3
3.1.3	Class and Interface Declarations	3
4	Indentation	4
4.1	Line Length	4
4.2	Wrapping Lines	4
5	Comments	6
5.1	Implementation Comment Formats	6
5.1.1	Block Comments	6
5.1.2	Single-Line Comments	7
5.1.3	Trailing Comments	7
5.1.4	End-Of-Line Comments	7
5.2	Documentation Comments	8
6	Declarations	9
6.1	Number Per Line	9
6.2	Placement	9
6.3	Initialization	10
6.4	Class and Interface Declarations	10
7	Statements	10
7.1	Simple Statements	10
7.2	Compound Statements	11
7.3	return Statements	11
7.4	if, if-else, if-else-if-else Statements	11
7.5	for Statements	12
7.6	while Statements	12
7.7	do-while Statements	12
7.8	switch Statements	12
7.9	try-catch Statements	13
8	White Space	13
8.1	Blank Lines	13
8.2	Blank Spaces	14
9	Naming Conventions	14
10	Programming Practices	15
10.1	Providing Access to Instance and Class Variables	15

June 2, 1997

10.2	Referring to Class Variables and Methods	16
10.3	Constants	16
10.4	Variable Assignments	16
10.5	Miscellaneous Practices	17
10.5.1	Parentheses	17
10.5.2	Returning Values	17
10.5.3	Expressions before '?' in the Conditional Operator	17
10.5.4	Special Comments	17
11	Code Examples	18
11.1	Java Source File Example	18

Java Code Conventions

1 - Introduction

1.1 Why Have Code Conventions

Code conventions are important to programmers for a number of reasons:

- 80% of the lifetime cost of a piece of software goes to maintenance.
- Hardly any software is maintained for its whole life by the original author.
- Code conventions improve the readability of the software, allowing engineers to understand new code more quickly and thoroughly.
- If you ship your source code as a product, you need to make sure it is as well packaged and clean as any other product you create.

1.2 Acknowledgments

This document reflects the Java language coding standards presented in the *Java Language Specification*, from Sun Microsystems. Major contributions are from Peter King, Patrick Naughton, Mike DeMoney, Jonni Kanerva, Kathy Walrath, and Scott Hommel.

For questions concerning adaptation, modification, or redistribution of this document, please read our copyright notice at <http://java.sun.com/docs/codeconv/html/Copyright.doc.html>.

Comments on this document should be submitted to our feedback form at <http://java.sun.com/docs/forms/sendusmail.html>.

2 - File Names

This section lists commonly used file suffixes and names.

2.1 File Suffixes

JavaSoft uses the following file suffixes:

File Type	Suffix
Java source	.java
Java bytecode	.class

2.2 Common File Names

Frequently used file names include:

File Name	Use
GNUmakefile	The preferred name for makefiles. We use gnumake to build our software.
README	The preferred name for the file that summarizes the contents of a particular directory.

3 - File Organization

A file consists of sections that should be separated by blank lines and an optional comment identifying each section.

Files longer than 2000 lines are cumbersome and should be avoided.

For an example of a Java program properly formatted, see "Java Source File Example" on page 19.

3.1 Java Source Files

Each Java source file contains a single public class or interface. When private classes and interfaces are associated with a public class, you can put them in the same source file as the public class. The public class should be the first class or interface in the file.

Java source files have the following ordering:

- Beginning comments (see "Beginning Comments" on page 4)
- Package and Import statements; for example:

```
import java.applet.Applet;
import java.awt.*;
import java.net.*;
```
- Class and interface declarations (see "Class and Interface Declarations" on page 4)

3 - File Organization

3.1.1 Beginning Comments

All source files should begin with a c-style comment that lists the programmer(s), the date, a copyright notice, and also a brief description of the purpose of the program. For example:

```
/*
 * Classname
 *
 * Version info
 *
 * Copyright notice
 */
```

3.1.2 Package and Import Statements

The first non-comment line of most Java source files is a package statement. After that, import statements can follow. For example:

```
package java.awt;

import java.awt.peer.CanvasPeer;
```

3.1.3 Class and Interface Declarations

The following table describes the parts of a class or interface declaration, in the order that they should appear. See "Java Source File Example" on page 19 for an example that includes comments.

	Part of Class/Interface Declaration	Notes
1	Class/interface documentation comment (<code>/**...*/</code>)	See "Documentation Comments" on page 9 for information on what should be in this comment.
2	class or interface statement	
3	Class/interface implementation comment (<code>/*...*/</code>), if necessary	This comment should contain any class-wide or interface-wide information that wasn't appropriate for the class/interface documentation comment.
4	Class (static) variables	First the public class variables, then the protected, and then the private.
5	Instance variables	First public, then protected, and then private.
6	Constructors	

4 - Indentation

	Part of Class/Interface Declaration	Notes
7	Methods	These methods should be grouped by functionality rather than by scope or accessibility. For example, a private class method can be in between two public instance methods. The goal is to make reading and understanding the code easier.

4 - Indentation

Four spaces should be used as the unit of indentation. The exact construction of the indentation (spaces vs. tabs) is unspecified. Tabs must be set exactly every 8 spaces (not 4).

4.1 Line Length

Avoid lines longer than 80 characters, since they're not handled well by many terminals and tools.

Note: Examples for use in documentation should have a shorter line length—generally no more than 70 characters.

4.2 Wrapping Lines

When an expression will not fit on a single line, break it according to these general principles:

- Break after a comma.
- Break before an operator.
- Prefer higher-level breaks to lower-level breaks.
- Align the new line with the beginning of the expression at the same level on the previous line.
- If the above rules lead to confusing code or to code that's squished up against the right margin, just indent 8 spaces instead.

Here are some examples of breaking method calls:

```
function(longExpression1, longExpression2, longExpression3,
        longExpression4, longExpression5);

var = function1(longExpression1,
                function2(longExpression2,
                          longExpression3));
```

4 - Indentation

Following are two examples of breaking an arithmetic expression. The first is preferred, since the break occurs outside the parenthesized expression, which is at a higher level.

```
longName1 = longName2 * (longName3 + longName4 - longName5)
                + 4 * longName6; // PREFER

longName1 = longName2 * (longName3 + longName4
                - longName5) + 4 * longName6; // AVOID
```

Following are two examples of indenting method declarations. The first is the conventional case. The second would shift the second and third lines to the far right if it used conventional indentation, so instead it indents only 8 spaces.

```
//CONVENTIONAL INDENTATION
someMethod(int anArg, Object anotherArg, String yetAnotherArg,
            Object andStillAnother) {
    ...
}

//INDENT 8 SPACES TO AVOID VERY DEEP INDENTS
private static synchronized horkingLongMethodName(int anArg,
            Object anotherArg, String yetAnotherArg,
            Object andStillAnother) {
    ...
}
```

Line wrapping for if statements should generally use the 8-space rule, since conventional (4 space) indentation makes seeing the body difficult. For example:

```
//DON'T USE THIS INDENTATION
if ((condition1 && condition2)
    || (condition3 && condition4)
    ||!(condition5 && condition6)) { //BAD WRAPS
    doSomethingAboutIt();           //MAKE THIS LINE EASY TO MISS
}

//USE THIS INDENTATION INSTEAD
if ((condition1 && condition2)
    || (condition3 && condition4)
    ||!(condition5 && condition6)) {
    doSomethingAboutIt();
}

//OR USE THIS
if ((condition1 && condition2) || (condition3 && condition4)
    ||!(condition5 && condition6)) {
    doSomethingAboutIt();
}
```

Here are three acceptable ways to format ternary expressions:

```
alpha = (aLongBooleanExpression) ? beta : gamma;

alpha = (aLongBooleanExpression) ? beta
                                : gamma;

alpha = (aLongBooleanExpression)
        ? beta
        : gamma;
```

5 - Comments

Java programs can have two kinds of comments: implementation comments and documentation comments. Implementation comments are those found in C++, which are delimited by `/*...*/`, and `//`. Documentation comments (known as "doc comments") are Java-only, and are delimited by `/**...*/`. Doc comments can be extracted to HTML files using the javadoc tool.

Implementation comments are mean for commenting out code or for comments about the particular implementation. Doc comments are meant to describe the specification of the code, from an implementation-free perspective. to be read by developers who might not necessarily have the source code at hand.

Comments should be used to give overviews of code and provide additional information that is not readily available in the code itself. Comments should contain only information that is relevant to reading and understanding the program. For example, information about how the corresponding package is built or in what directory it resides should not be included as a comment.

Discussion of nontrivial or nonobvious design decisions is appropriate, but avoid duplicating information that is present in (and clear from) the code. It is too easy for redundant comments to get out of date. In general, avoid any comments that are likely to get out of date as the code evolves.

Note: The frequency of comments sometimes reflects poor quality of code. When you feel compelled to add a comment, consider rewriting the code to make it clearer.

Comments should not be enclosed in large boxes drawn with asterisks or other characters. Comments should never include special characters such as form-feed and backspace.

5.1 Implementation Comment Formats

Programs can have four styles of implementation comments: block, single-line, trailing and end-of-line.

5.1.1 Block Comments

Block comments are used to provide descriptions of files, methods, data structures and algorithms. Block comments should be used at the beginning of each file and before each method. They can also be used in other places, such as within methods. Block comments inside a function or method should be indented to the same level as the code they describe.

A block comment should be preceded by a blank line to set it apart from the rest of the code. Block comments have an asterisk "*" at the beginning of each line except the first.

```
/*  
 * Here is a block comment.  
 */
```

5 - Comments

Block comments can start with `/*-`, which is recognized by `indent(1)` as the beginning of a block comment that should not be reformatted. Example:

```
/*
 * Here is a block comment with some very special
 * formatting that I want indent(1) to ignore.
 *
 *   one
 *     two
 *       three
 */
```

Note: If you don't use `indent(1)`, you don't have to use `/*-` in your code or make any other concessions to the possibility that someone else might run `indent(1)` on your code.

See also "Documentation Comments" on page 9.

5.1.2 Single-Line Comments

Short comments can appear on a single line indented to the level of the code that follows. If a comment can't be written in a single line, it should follow the block comment format (see section 5.1.1). A single-line comment should be preceded by a blank line. Here's an example of a single-line comment in Java code (also see "Documentation Comments" on page 9):

```
if (condition) {
    /* Handle the condition. */
    ...
}
```

5.1.3 Trailing Comments

Very short comments can appear on the same line as the code they describe, but should be shifted far enough to separate them from the statements. If more than one short comment appears in a chunk of code, they should all be indented to the same tab setting. Avoid the assembly language style of commenting every line of executable code with a trailing comment.

Here's an example of a trailing comment in Java code (also see "Documentation Comments" on page 9):

```
if (a == 2) {
    return TRUE;           /* special case */
} else {
    return isprime(a);     /* works only for odd a */
}
```

5.1.4 End-Of-Line Comments

The `//` comment delimiter begins a comment that continues to the newline. It can comment out a complete line or only a partial line. It shouldn't be used on consecutive multiple lines for text comments; however, it can be used in consecutive multiple lines for commenting out sections of code. Examples of all three styles follow:

```
if (foo > 1) {  
    // Do a double-flip.  
    ...  
}  
else  
    return false;           // Explain why here.  
  
//if (bar > 1) {  
//  
//    // Do a triple-flip.  
//    ...  
//}  
//else  
//    return false;
```

5.2 Documentation Comments

Note: See "Java Source File Example" on page 19 for examples of the comment formats described here.

For further details, see "How to Write Doc Comments for Javadoc" which includes information on the doc comment tags (@return, @param, @see):

<http://java.sun.com/products/jdk/javadoc/writingdoccomments.html>

For further details about doc comments and javadoc, see the javadoc home page at:

<http://java.sun.com/products/jdk/javadoc/>

Doc comments describe Java classes, interfaces, constructors, methods, and fields. Each doc comment is set inside the comment delimiters `/**...*/`, with one comment per API. This comment should appear just before the declaration:

```
/**  
 * The Example class provides ...  
 */  
class Example { ...
```

Notice that classes and interfaces are not indented, while their members are. The first line of doc comment (`/**`) for classes and interfaces is not indented; subsequent doc comment lines each have 1 space of indentation (to vertically align the asterisks). Members, including constructors, have 4 spaces for the first doc comment line and 5 spaces thereafter.

If you need to give information about a class, interface, variable, or method that isn't appropriate for documentation, use an implementation block comment (see section 5.1.1) or single-line (see section 5.1.2) comment immediately *after* the declaration. For example, details about the implementation of a class should go in in such an implementation block comment *following* the class statement, not in the class doc comment.

Doc comments should not be positioned inside a method or constructor definition block, because Java associates documentation comments with the first declaration *after* the comment.

6 - Declarations

6.1 Number Per Line

One declaration per line is recommended since it encourages commenting. In other words,

```
int level; // indentation level
int size; // size of table
```

is preferred over

```
int level, size;
```

In absolutely no case should variables and functions be declared on the same line. Example:

```
long dbaddr, getDbaddr(); // WRONG!
```

Do not put different types on the same line. Example:

```
int foo, foobar[]; //WRONG!
```

Note: The examples above use one space between the type and the identifier. Another acceptable alternative is to use tabs, e.g.:

```
int      level;      // indentation level
int      size;       // size of table
Object   currentEntry; // currently selected table entry
```

6.2 Placement

Put declarations only at the beginning of blocks. (A block is any code surrounded by curly braces "{" and "}"). Don't wait to declare variables until their first use; it can confuse the unwary programmer and hamper code portability within the scope.

```
void MyMethod() {
    int int1;      // beginning of method block

    if (condition) {
        int int2;  // beginning of "if" block
        ...
    }
}
```

The one exception to the rule is indexes of for loops, which in Java can be declared in the for statement:

```
for (int i = 0; i < maxLoops; i++) { ...
```

Avoid local declarations that hide declarations at higher levels. For example, do not declare the same variable name in an inner block:

7 - Statements

```
int count;
...
func() {
    if (condition) {
        int count;    // AVOID!
        ...
    }
    ...
}
```

6.3 Initialization

Try to initialize local variables where they're declared. The only reason not to initialize a variable where it's declared is if the initial value depends on some computation occurring first.

6.4 Class and Interface Declarations

When coding Java classes and interfaces, the following formatting rules should be followed:

- No space between a method name and the parenthesis "(" starting its parameter list
- Open brace "{" appears at the end of the same line as the declaration statement
- Closing brace "}" starts a line by itself indented to match its corresponding opening statement, except when it is a null statement the "}" should appear immediately after the "{"

```
class Sample extends Object {
    int ivar1;
    int ivar2;

    Sample(int i, int j) {
        ivar1 = i;
        ivar2 = j;
    }

    int emptyMethod() {}

    ...
}
```

- Methods are separated by a blank line

7 - Statements

7.1 Simple Statements

Each line should contain at most one statement. Example:

```
argv++; argc--;    // AVOID!
```


7 - Statements

Do not use the comma operator to group multiple statements unless it is for an obvious reason.
Example:

```
if (err) {  
    Format.print(System.out, "error"), exit(1); //VERY WRONG!  
}
```

7.2 Compound Statements

Compound statements are statements that contain lists of statements enclosed in braces
" { statements } ". See the following sections for examples.

- The enclosed statements should be indented one more level than the compound statement.
- The opening brace should be at the end of the line that begins the compound statement; the closing brace should begin a line and be indented to the beginning of the compound statement.
- Braces are used around all statements, even singletons, when they are part of a control structure, such as a if-else or for statement. This makes it easier to add statements without accidentally introducing bugs due to forgetting to add braces.

7.3 return Statements

A return statement with a value should not use parentheses unless they make the return value more obvious in some way. Example:

```
return;  
  
return myDisk.size();  
  
return (size ? size : defaultSize);
```

7.4 if, if-else, if-else-if-else Statements

The if-else class of statements should have the following form:

```
if (condition) {  
    statements;  
}  
  
if (condition) {  
    statements;  
} else {  
    statements;  
}  
  
if (condition) {  
    statements;  
} else if (condition) {  
    statements;  
} else if (condition) {  
    statements;  
}
```

Note: if statements always use braces {}. Avoid the following error-prone form:

```
if (condition) //AVOID! THIS OMITTS THE BRACES {}!  
    statement;
```

7.5 for Statements

A for statement should have the following form:

```
for (initialization; condition; update) {  
    statements;  
}
```

An empty for statement (one in which all the work is done in the initialization, condition, and update clauses) should have the following form:

```
for (initialization; condition; update);
```

When using the comma operator in the initialization or update clause of a for statement, avoid the complexity of using more than three variables. If needed, use separate statements before the for loop (for the initialization clause) or at the end of the loop (for the update clause).

7.6 while Statements

A while statement should have the following form:

```
while (condition) {  
    statements;  
}
```

An empty while statement should have the following form:

```
while (condition);
```

7.7 do-while Statements

A do-while statement should have the following form:

```
do {  
    statements;  
} while (condition);
```

7.8 switch Statements

A switch statement should have the following form:

8 - White Space

```
switch (condition) {  
  case ABC:  
    statements;  
    /* falls through */  
  case DEF:  
    statements;  
    break;  
  
  case XYZ:  
    statements;  
    break;  
  
  default:  
    statements;  
    break;  
}
```

Every time a case falls through (doesn't include a break statement), add a comment where the break statement would normally be. This is shown in the preceding code example with the `/* falls through */` comment.

Every switch statement should include a default case. The break in the default case is redundant, but it prevents a fall-through error if later another case is added.

7.9 try-catch Statements

A try-catch statement should have the following format:

```
try {  
  statements;  
} catch (ExceptionClass e) {  
  statements;  
}
```

8 - White Space

8.1 Blank Lines

Blank lines improve readability by setting off sections of code that are logically related.

Two blank lines should always be used in the following circumstances:

- Between sections of a source file
- Between class and interface definitions

One blank line should always be used in the following circumstances:

- Between methods
- Between the local variables in a method and its first statement
- Before a block (see section 5.1.1) or single-line (see section 5.1.2) comment

- Between logical sections inside a method to improve readability

8.2 Blank Spaces

Blank spaces should be used in the following circumstances:

- A keyword followed by a parenthesis should be separated by a space. Example:

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

Note that a blank space should not be used between a method name and its opening parenthesis. This helps to distinguish keywords from method calls.

- A blank space should appear after commas in argument lists.
- All binary operators except `.` should be separated from their operands by spaces. Blank spaces should never separate unary operators such as unary minus, increment ("`++`"), and decrement ("`--`") from their operands. Example:

```
a += c + d;  
a = (a + b) / (c * d);  
  
while (d++ = s++) {  
    n++;  
}  
prints("size is " + foo + "\n");
```

- The expressions in a `for` statement should be separated by blank spaces. Example:

```
for (expr1; expr2; expr3)
```

- Casts should be followed by a blank. Examples:

```
myMethod((byte) aNum, (Object) x);  
myFunc((int) (cp + 5), ((int) (i + 3))  
        + 1);
```

9 - Naming Conventions

Naming conventions make programs more understandable by making them easier to read. They can also give information about the function of the identifier—for example, whether it's a constant, package, or class—which can be helpful in understanding the code.

The conventions given in this section are high level. Further conventions are given at *(to be determined)*.

10 - Programming Practices

Identifier Type	Rules for Naming	Examples
Classes	Class names should be nouns, in mixed case with the first letter of each internal word capitalized. Try to keep your class names simple and descriptive. Use whole words—avoid acronyms and abbreviations (unless the abbreviation is much more widely used than the long form, such as URL or HTML).	<pre>class Raster; class ImageSprite;</pre>
Interfaces	Interface names should be capitalized like class names.	<pre>interface RasterDelegate; interface Storing;</pre>
Methods	Methods should be verbs, in mixed case with the first letter lowercase, with the first letter of each internal word capitalized.	<pre>run(); runFast(); getBackground();</pre>
Variables	Except for variables, all instance, class, and class constants are in mixed case with a lowercase first letter. Internal words start with capital letters. Variable names should be short yet meaningful. The choice of a variable name should be mnemonic—that is, designed to indicate to the casual observer the intent of its use. One-character variable names should be avoided except for temporary “throwaway” variables. Common names for temporary variables are i, j, k, m, and n for integers; c, d, and e for characters.	<pre>int i; char *cp; float myWidth;</pre>
Constants	The names of variables declared class constants and of ANSI constants should be all uppercase with words separated by underscores (“_”). (ANSI constants should be avoided, for ease of debugging.)	<pre>int MIN_WIDTH = 4; int MAX_WIDTH = 999; int GET_THE_CPU = 1;</pre>

10 - Programming Practices

10.1 Providing Access to Instance and Class Variables

Don't make any instance or class variable public without good reason. Often, instance variables don't need to be explicitly set or gotten—often that happens as a side effect of method calls.

One example of appropriate public instance variables is the case where the class is essentially a data structure, with no behavior. In other words, if you would have used a `struct` instead of a class (if Java supported `struct`), then it's appropriate to make the class's instance variables public.

10.2 Referring to Class Variables and Methods

Avoid using an object to access a class (static) variable or method. Use a class name instead. For example:

```
classMethod();           //OK
AClass.classMethod();    //OK

anObject.classMethod();  //AVOID!
```

10.3 Constants

Numerical constants (literals) should not be coded directly, except for -1, 0, and 1, which can appear in a `for` loop as counter values.

10.4 Variable Assignments

Avoid assigning several variables to the same value in a single statement. It is hard to read. Example:

```
fooBar.fChar = barFoo.lchar = 'c'; // AVOID!
```

Do not use the assignment operator in a place where it can be easily confused with the equality operator. Example:

```
if (c++ = d++) {           // AVOID! Java disallows
    ...
}
```

should be written as

```
if ((c++ = d++) != 0) {
    ...
}
```

Do not use embedded assignments in an attempt to improve run-time performance. This is the job of the compiler, and besides, it rarely actually helps. Example:

```
d = (a = b + c) + r;        // AVOID!
```

should be written as

```
a = b + c;
d = a + r;
```

10.5 Miscellaneous Practices

10.5.1 Parentheses

It is generally a good idea to use parentheses liberally in expressions involving mixed operators to avoid operator precedence problems. Even if the operator precedence seems clear to you, it might not be to others—you shouldn't assume that other programmers know precedence as well as you do.

```
if (a == b && c == d)    // AVOID!  
  
if ((a == b) && (c == d)) // RIGHT
```

10.5.2 Returning Values

Try to make the structure of your program match the intent. Example:

```
if (booleanExpression) {  
    return TRUE;  
} else {  
    return FALSE;  
}
```

should instead be written as

```
return booleanExpression;
```

Similarly,

```
if (condition) {  
    return x;  
}  
return y;
```

should be written as

```
return (condition ? x : y);
```

10.5.3 Expressions before '?' in the Conditional Operator

If an expression containing a binary operator appears before the ? in the ternary ?: operator, it should be parenthesized. Example:

```
(x >= 0) ? x : -x
```

10.5.4 Special Comments

Use XXX in a comment to flag something that is bogus but works. Use FIXME to flag something that is bogus and broken.

11 - Code Examples

11.1 Java Source File Example

The following example shows how to format a Java source file containing a single public class. Interfaces are formatted similarly. For more information, see "Class and Interface Declarations" on page 4 and "Documentation Comments" on page 9

```
/*
 * %W% %E% Firstname Lastname
 *
 * Copyright (c) 1993-1996 Sun Microsystems, Inc. All Rights Reserved.
 *
 * This software is the confidential and proprietary information of Sun
 * Microsystems, Inc. ("Confidential Information"). You shall not
 * disclose such Confidential Information and shall use it only in
 * accordance with the terms of the license agreement you entered into
 * with Sun.
 *
 * SUN MAKES NO REPRESENTATIONS OR WARRANTIES ABOUT THE SUITABILITY OF
 * THE SOFTWARE, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED
 * TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
 * PARTICULAR PURPOSE, OR NON-INFRINGEMENT. SUN SHALL NOT BE LIABLE FOR
 * ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR
 * DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES.
 */
package java.blah;

import java.blah.blahdy.BlahBlah;

/**
 * Class description goes here.
 *
 * @version      1.10 04 Oct 1996
 * @author       Firstname Lastname
 */
public class Blah extends SomeClass {
    /* A class implementation comment can go here. */

    /** classVar1 documentation comment */
    public static int classVar1;

    /**
     * classVar2 documentation comment that happens to be
     * more than one line long
     */
    private static Object classVar2;

    /** instanceVar1 documentation comment */
    public Object instanceVar1;

    /** instanceVar2 documentation comment */
    protected int instanceVar2;

    /** instanceVar3 documentation comment */
    private Object[] instanceVar3;
```


11 - Code Examples

```
/**
 * ...method Blah documentation comment...
 */
public Blah() {
    // ...implementation goes here...
}

/**
 * ...method doSomething documentation comment...
 */
public void doSomething() {
    // ...implementation goes here...
}

/**
 * ...method doSomethingElse documentation comment...
 * @param someParam description
 */
public void doSomethingElse(Object someParam) {
    // ...implementation goes here...
}
}
```

A.04

Matriz T

ANÁLISIS DE FUERZAS T						
Situación Empeorada	Situación Actual				Situación Mejorada	
	Eliminación del Sistema de Gestión de Calidad implementado	Inadecuado Control de Calidad	Control de Indicadores de Gestión de Calidad	Mejorar el control de los indicadores de Gestión de Calidad de la Dirección RIPS		
Fuerzas Impulsadoras	I	PC	I	PC	Fuerzas Bloqueadoras	
Registro de Indicadores según lo planificado	1	5	5	4	Uso de versiones obsoletas de formatos.	
Uso de formatos y documentación actualizada.	3	4	5	3	Inadecuado seguimiento de indicadores de gestión	
Facilidad de Acceso a la Documentación y a Consulta de Indicadores	3	4	5	3	Pérdida de información de repositorios digitales	
Registro de errores detectados para ejecución de Acciones Preventivas o Correctivas.	2	4	5	4	Falta de confiabilidad de indicadores	

NOMENCLATURA

El Nivel de Impacto de la fuerza sobre las condiciones de la problemática actual se representa con I y Cuánto se puede modificar o aprovechar la fuerza para llegar a la situación deseada se representa con PC.

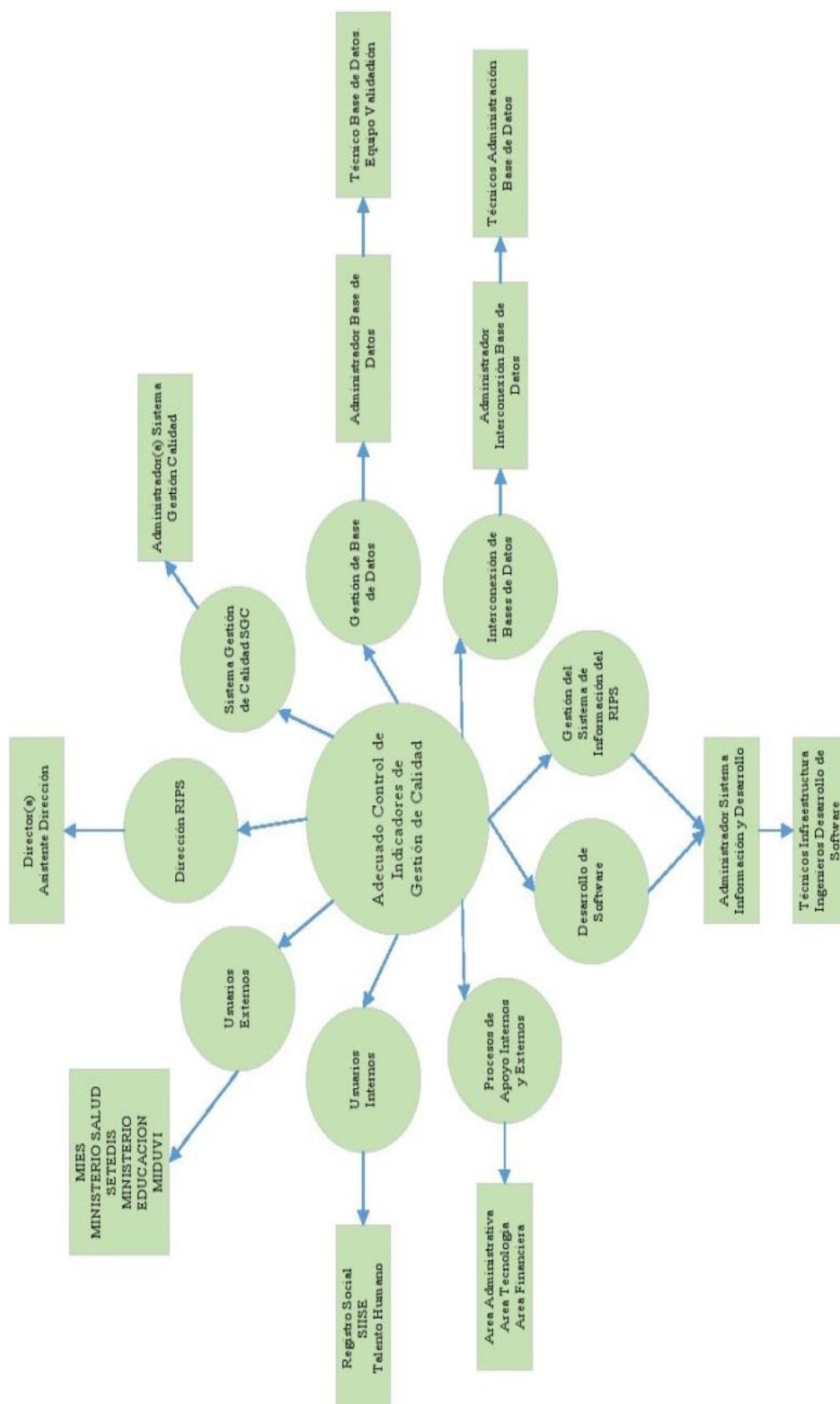
ESCALA - SEMAFORIZACIÓN

BAJO	1
MEDIO BAJO	2
MEDIO	3
MEDIO ALTO	4
ALTO	5

Se hace imperativo implementar acciones que lleven a la implementación de las fuerzas impulsadoras lo que permitirá asegurar el control y seguimiento del cumplimiento de la planificación en base al monitoreo de los indicadores de gestión, lo que facilitará la toma de decisiones.

A.05

Mapeo de Involucrados



A.06

Matriz de Involucrados

<i>Actores Involucrados</i>	<i>Intereses sobre el problema central</i>	<i>Problemas Percibidos</i>	<i>Recursos, Mandatos y Capacidades</i>	<i>Intereses sobre el Proyecto</i>	<i>Conflictos Potenciales</i>
Área Financiera Área Administrativa Área Tecnológica	Evitar el dejar el Sistema de Gestión de Calidad	Detectar errores en la información proporcionada por los sistemas puestos a disposición por la Dirección RIPS	R.1. Humanos R.2. Financieros	Revalorizar la inversión realizada al implementar el Sistema de Gestión de Calidad en la Dirección	Eliminación del Sistema de Gestión de Calidad
Dirección del Registro Social. Dirección del SIISE. Dirección de Talento Humano	Aumentar la confiabilidad de la información proporcionada por los sistemas puestos a disposición por la Dirección RIPS	Detectar errores en la información proporcionada por los sistemas puestos a disposición por la Dirección RIPS	R.1. Humanos	Mejoramiento continuo en la calidad de la información generada por la Dirección RIPS	Eliminación del Sistema de Gestión de Calidad
Ministerio de Inclusión Económica y Social. Ministerio de Salud Pública. Secretaría Técnica de Discapacidades. Corporación Nacional de Finanzas Populares y Solidarias. Ministerio de Educación. Público en General	Aumentar la confiabilidad de la información proporcionada por los sistemas puestos a disposición por la Dirección RIPS y que es utilizada para la toma de decisiones por las instituciones.	Detectar errores en la información proporcionada por los sistemas puestos a disposición por la Dirección RIPS	R.1. Humanos	Mejoramiento continuo en la calidad de la información generada por la Dirección RIPS	Eliminación del Sistema de Gestión de Calidad

Director(a) Asistente de Dirección	Facilidad en el monitoreo de indicadores para toma de decisiones.	Indicadores desactualizados.	R1.Humanos	Dar solución a las deficiencias del Sistema de Gestión de Calidad en lo relacionado al manejo de documentos y registro de indicadores.	Procesos que no se enmarquen en el Sistema de Gestión de Calidad
	Consulta fácil de informes de las actividades realizadas.	Dificultad para acceder a Formatos de Registros controlados.	R2.Técnicos R3.Financiero		
			M1.Manual de Usuario M2.Manual Técnico M3.Norma ISO 9001:2008		
Administrador del Sistema de Gestión de Calidad	Fácil ingreso de indicadores por cada responsable de procesos.	Invertir demasiado tiempo en verificar que los documentos utilizados corresponden a la última versión y constan como controlados en el sistema.	R1.Humanos	Facilidad en el ingreso, verificación y control del cumplimiento de las metas propuestas.	Eliminación del Sistema de Gestión de Calidad
	Control uso documentos controlados.		M1.Manual de Usuario M2.Manual Técnico M3.Norma ISO 9001:2008		
	Eliminar la recolección de registros para depositarlos en los repositorios.	Dificultad en obtener y centralizar los indicadores de gestión.	M4.Formatos y Documentos Controlados	Documentación actualizada de los procesos para consulta o seguimiento.	
				Seguridad en el manejo de la información sin riesgo de pérdida.	

Administrador de Base de Datos. Técnico de Base de Datos. Equipo de Validación y Depuración Administrador del Sistema de Información. Encargado Infraestructura. Equipo de Desarrollo. Administrador Interconexión de Bases de Datos. Técnicos Administración Bases de Datos.	Evitar dificultades en el ingreso de indicadores. Complicaciones en el acceso a las últimas versiones de documentos controlados. Eliminar la dificultad de consulta de informes y documentos en los repositorios.	Inversión de tiempo en preparar los informes en los formatos controlados. Errores al utilizar formatos de documentos controlados con versiones obsoletas. Dificultad para ubicar los registros que se necesita consultar en los repositorios. Omisión en registro de errores detectados por dificultad de ubicar el formato y realizar el seguimiento y/o complicación de levantar las Acciones Preventivas o Correctivas	R.1 Humanos M1.Manual de Usuario M2.Manual Técnico M3 Norma ISO 9001:2008 M4.Formatos y Documentos Controlados	Facilidad en el control, ingreso, verificación y entrega de reportes sobre el resultado en el cumplimiento de las metas propuestas a través de los indicadores de gestión. Documentación actualizada de los procesos de la dirección al alcance de todos los usuarios para consulta o seguimiento. Seguridad en el manejo de la información sin riesgo de pérdida de información. Facilidad de acceso a la documentación para consulta. Simplicidad en el ingreso de indicadores de gestión. Formatos para llenar los informes accesibles de forma fácil. Registro de errores detectados y levantamiento de Acciones Preventivas o Correctivas de forma fácil.	Eliminación del Sistema de Gestión de Calidad
--	---	--	--	--	---

A.07

Matriz de Requerimientos

MATRIZ DE REQUERIMIENTOS						
Identificador	Descripción	Fuente	Prioridad	Tipo	Estado	Usuarios Involucrados
REQUERIMIENTOS FUNCIONALES						
RQF001	El sistema debe permitir la creación y mantenimiento de los usuarios del sistema.	Dirección del RIPS y/o Administrador del Sistema de Gestión de Calidad.	Alta	Funcional	Requerido	Dirección del RIPS. Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuarios del sistema autorizados.
RQF002	El Sistema debe permitir otorgar o retirar permisos de acceso para realizar actividades específicas según sea autorizado a cada usuario del sistema.	Dirección del RIPS y/o Administrador del Sistema de Gestión de Calidad.	Alta	Funcional	Requerido	Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuario autorizado.
RQF003	El sistema debe permitir al usuario designado, modificar el nombre y el orden de cada uno de los módulos que lo	Dirección del RIPS y/o Administrador del Sistema de Gestión de Calidad.	Alta	Funcional	Requerido	Administrador del Sistema de Gestión de Calidad. Área de

componen.	Administrador del Sistema de Gestión de Calidad (SGC)	Alta	Funcional	Requerido	Válido	desarrollo. Usuario autorizado. Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuario autorizado.
RQF004	El sistema debe contemplar el ingreso y mantenimiento de la Lista Maestra de Registros del SGC.					
RQF005	El sistema debe contar con los formatos de indicadores establecidos en el Sistema de Gestión de Calidad del RIPS contenidos en la Lista Maestra de Registros.	Alta	Funcional	Requerido	Válido	Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuario autorizado.
RQF006	El sistema debe contemplar el ingreso y mantenimiento de los responsables de los procesos.	Alta	Funcional	Requerido	Válido	Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuario autorizado.
RQF007	El sistema debe contemplar el ingreso y mantenimiento de las personas, firmas y función de los responsables de	Alta	Funcional	Requerido	Válido	Administrador del Sistema de Gestión de Calidad. Área de desarrollo.

	elaborar, revisar y aprobar cada uno de los informes de indicadores.	Administrador del Sistema de Gestión de Calidad (SGC)	Alta	Funcional	Requerido	Válido	Usuario autorizado.
RQF008	El sistema debe contemplar la posterior recuperación manteniendo la misma versión de los informes de indicadores ingresados por los responsables. Para el ingreso al sistema debe ser validado el usuario, la contraseña y los permisos concedidos, de los que dependerán las actividades que el sistema le permita acceder.	Administrador del Sistema de Gestión de Calidad (SGC)	Alta	Funcional	Requerido	Válido	Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuario autorizado.
RQF009	El sistema debe tener Menú Principal.	Administrador del Sistema de Gestión de Calidad (SGC)	Alta	Funcional	Requerido	Válido	Administrador del Sistema de Gestión de Calidad. Área de desarrollo. Usuario autorizado.
RQF010	El sistema debe permitir ingresar los informes de indicadores	Responsable de Procesos Administrador del Sistema de Gestión de Calidad (SGC)	Alta	Funcional	Requerido	Válido	Administrador del Sistema de Gestión de Calidad. Responsable

		Director	Alta	Funcional	Requerido	Válido	Proceso Usuario autorizado.
RQF012	El sistema debe permitir visualizar los indicadores						Administrador del Sistema de Gestión de Calidad. Usuario autorizado.
REQUERIMIENTOS NO FUNCIONALES							
RQNF001	El sistema debe ser de fácil acceso y navegabilidad.	Área de desarrollo. Administrador del Sistema de Gestión de Calidad (SGC)	Alta	No funcional	Requerido	N/A	N/A
RQNF002	El sistema debe ser seguro y mantener la integridad de la información.	Área de desarrollo.	Alta	No funcional	Requerido	N/A	N/A
RQNF003	El sistema debe contar con mecanismos adecuados de autenticación y validación de los usuarios del sistema.	Área de desarrollo. Administrador del Sistema de Gestión de Calidad (SGC)	Alta	No funcional	Requerido	N/A	N/A
RQNF004	El sistema debe controlar el acceso a los diferentes módulos.	Área de desarrollo.	Media	No funcional	Requerido	N/A	N/A
RQNF005	El sistema debe adaptarse al hardware presente en la empresa.	Área de desarrollo.	Media	No funcional	Requerido	N/A	N/A

RQNF006	El sistema debe facilitar el trabajo de las personas que lo usan.	Alta	No funcional	Deseado	N/A	N/A
---------	---	------	--------------	---------	-----	-----

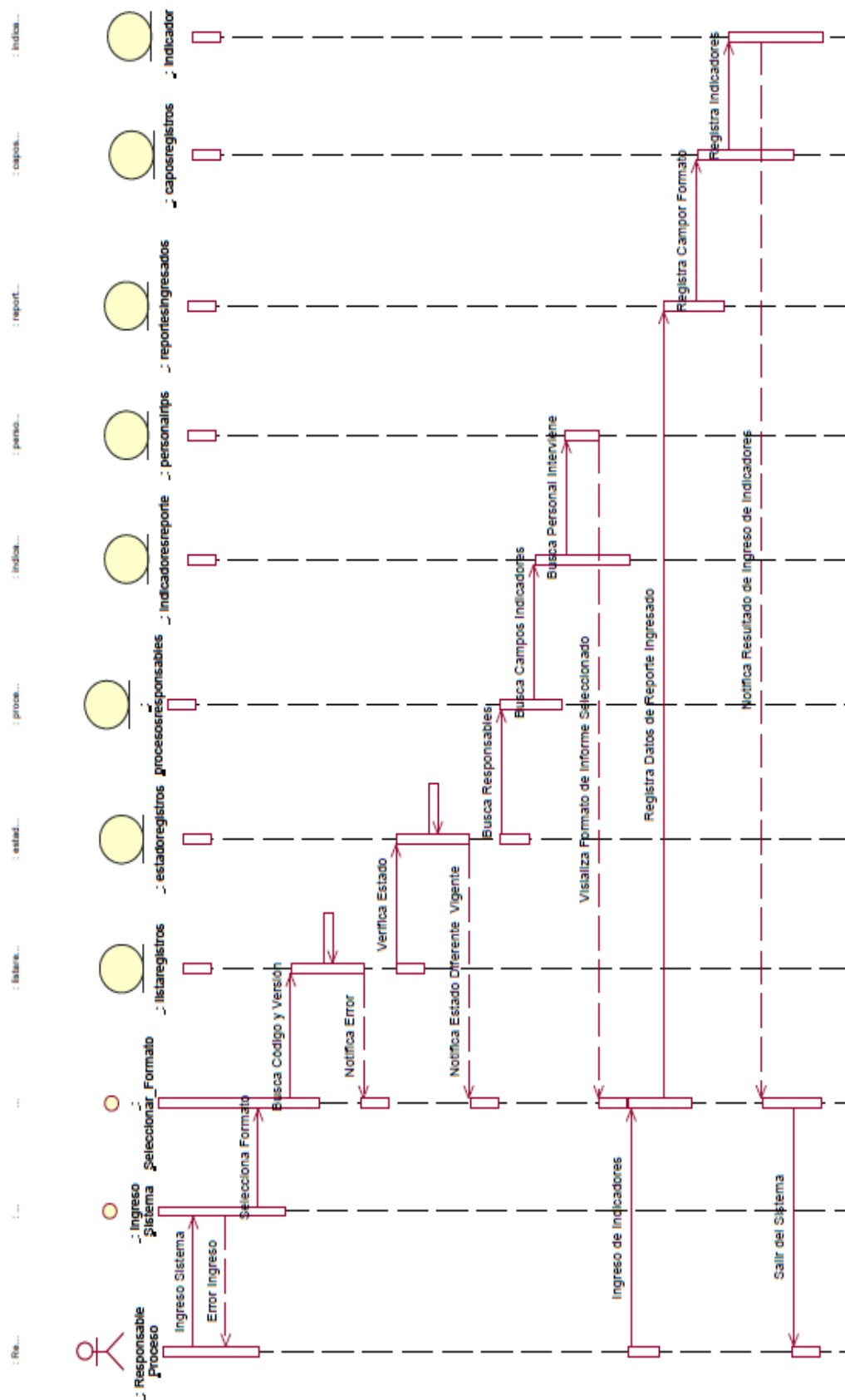
A.08

Matriz de Impacto de Objetivos

MATRIZ DE IMPACTO DE OBJETIVOS					
Factibilidad de Lograse (Alta-Media-Baja) (4 - 2 - 1)	Impacto en Género (Alta-Media-Baja) (4 - 2 - 1)	Impacto Ambiental (Alta-Media-Baja) (4 - 2 - 1)	Relevancia (Alta-Media-Baja) (4 - 2 - 1)	Sostenibilidad (Alta-Media-Baja) (4 - 2 - 1)	Total
OBJETIVOS	Cuenta con financiamiento	No excluye a nadie por el género	Reduce el uso material físico (papel, medios magnéticos).	Beneficia a usuarios internos y externos al reducir tiempos de proceso y mejorar la calidad de la información.	Se mantiene fortalecida el área al facilitar los procesos.
	Existe tecnología apropiada para su realización	Fácil adaptabilidad para cualquier persona.	Propicia entrega de información en medio magnético para colaborar con la cultura ambiental	Disminuye tiempos de ejecución y monitoreo	Mejora seguridad en manejo de información se hace sostenible en el tiempo.
	Existe la decisión política para ejecutarse	Propende a igualdad de género frente a los derechos y responsabilidades	Reduce el consumo de energía al no tener que imprimir la documentación.	Confiabilidad al procesar información bajo parámetros de normas internacionales eficientes.	Existe la decisión política para sostenerse.
	Tiene alto impacto en el propósito que se busca.	Apoya a tener igualdad de oportunidades para el manejo de la información	Alarga la vida útil de impresora y fotocopadoras, lo que alarga su vida útil.	Promueve el trabajo en equipo.	Fortalece la calidad de la información que le da valor agregado
	Existe predisposición del personal involucrado para adelantar el proyecto		Soporta la estandarización de manejo de información entre las diferentes Direcciones		
TOTAL	20 puntos	16 puntos	12 puntos	18 puntos	82 Puntos
					22 a 32 BAJA
					33 a 44 MEDIA BAJA
					45 a 66 MEDIA ALTA
					67 a 88 ALTA

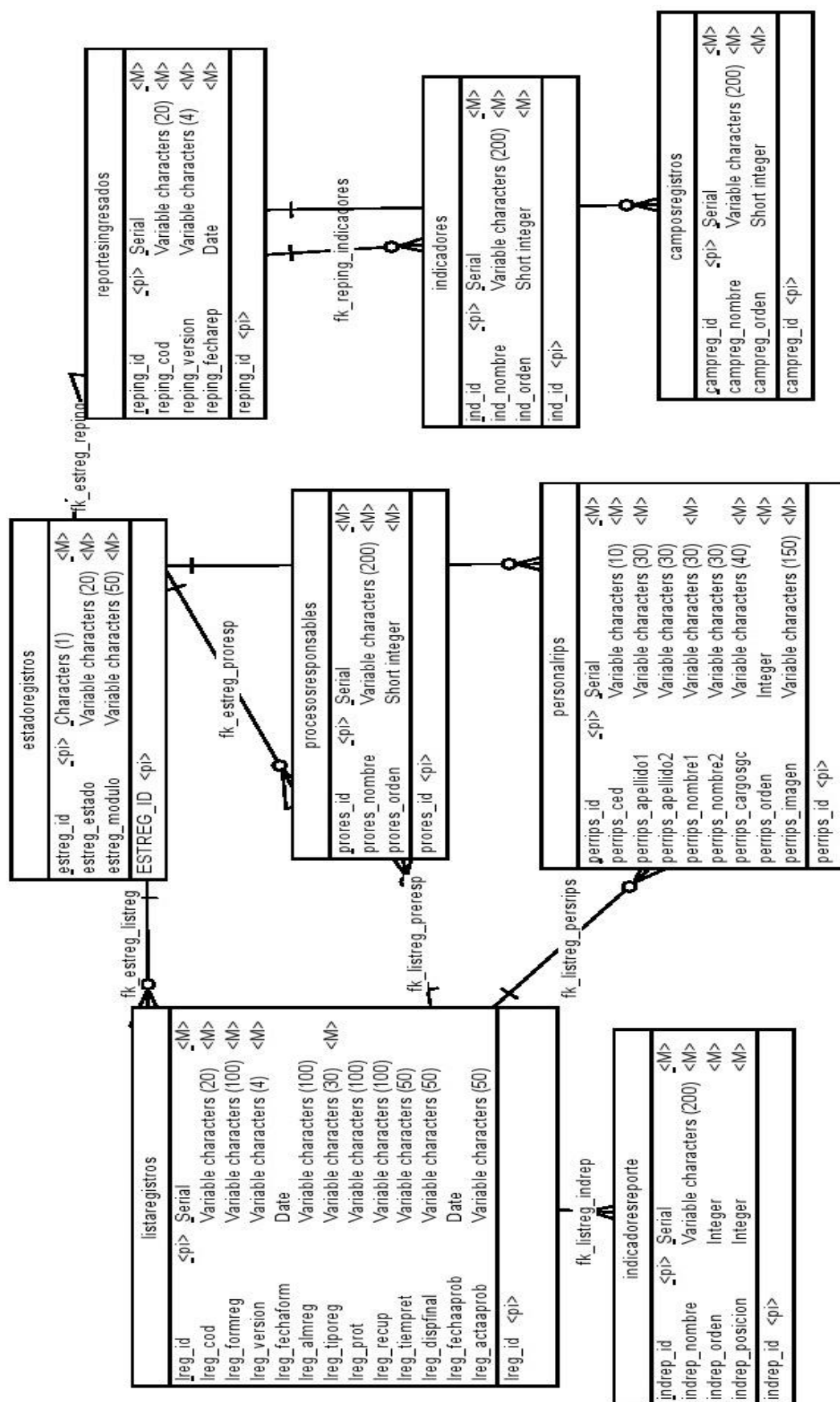
El análisis de la Matriz de Impacto de los Objetivos deja en evidencia que la factibilidad de implementar la solución es muy alta al contar con todos los elementos necesarios para ello, su impacto en género es favorable porque propicia igualdad de derechos y responsabilidades sin distingo de género, tienen un buen impacto ambiental y un alto grado de relevancia y sostenibilidad en el tiempo, factores que indican la favorabilidad del proyecto.

Diagrama de Secuencia - Elaborar Informe



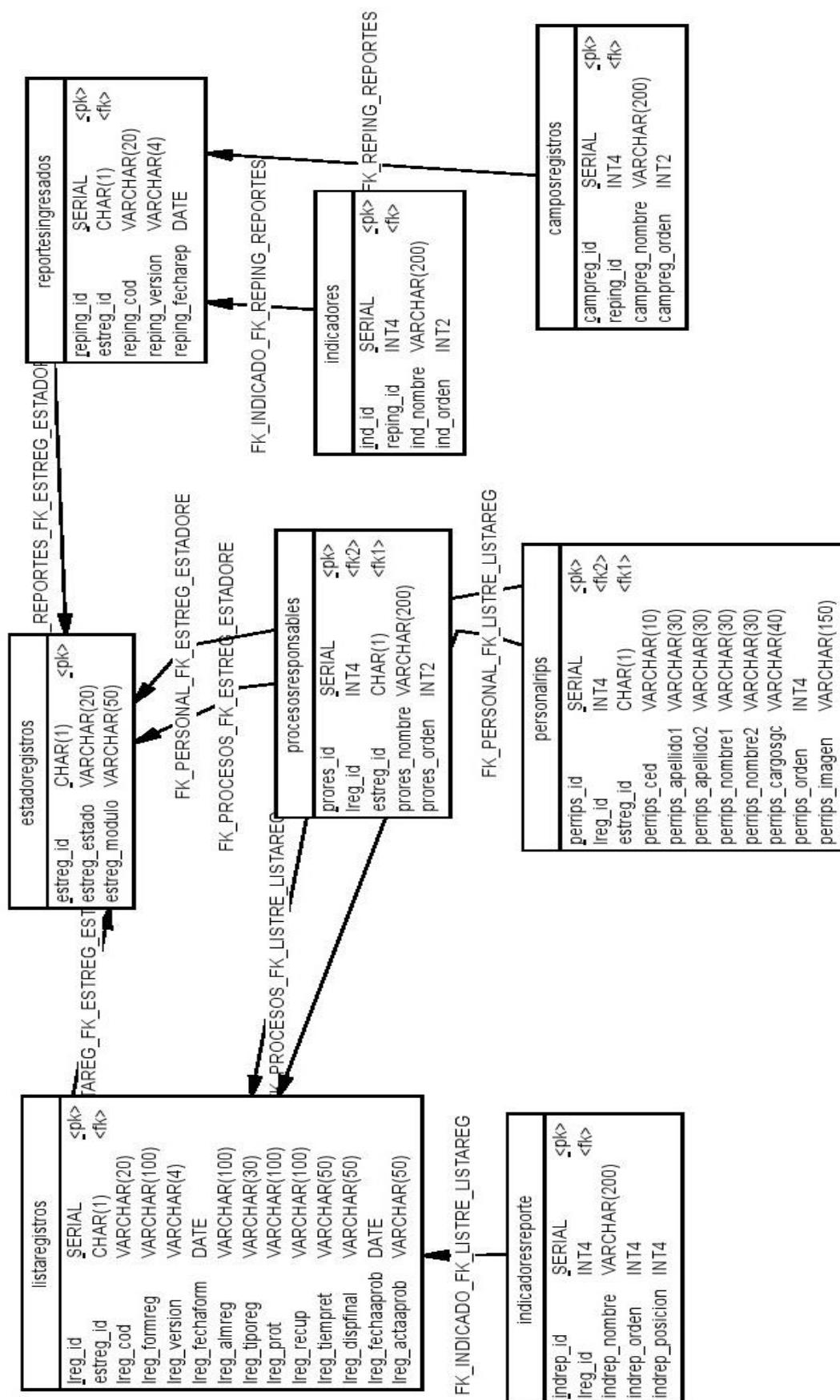
A.10

Modelo Lógico de Base de Datos



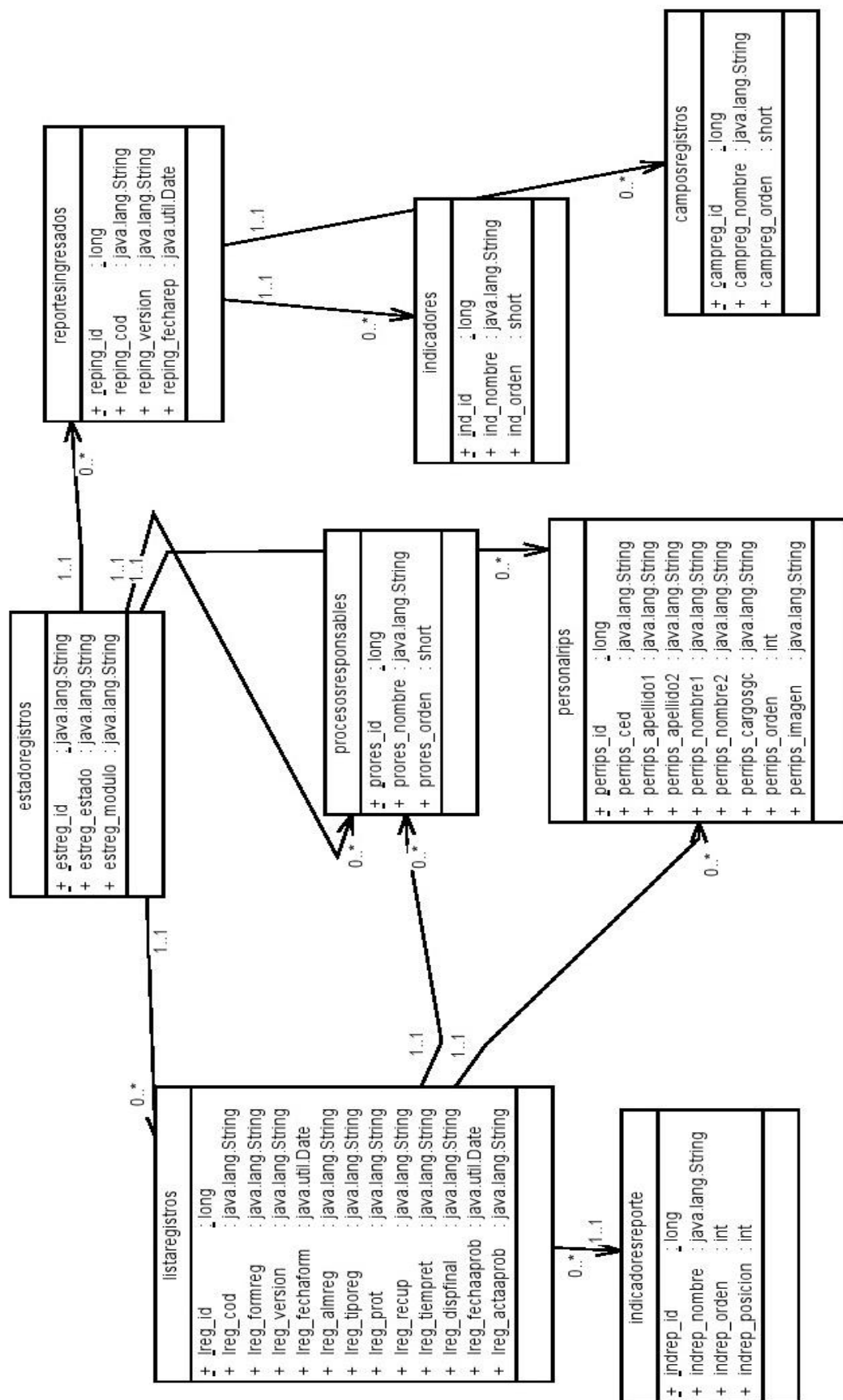
A.11

Modelo Físico de Base de Datos



A.12

Diagrama de Objetos Orientado a Modelo de Base de Datos



A.13

Estado Actual de Servidores

ESTADO ACTUAL DE SERVIDORES

DESCRIPCION	SISTEMAS INSTALADOS	BASES DE DATOS / APLICACIONES / SERVICIOS	DIRECCION IP		DISCO DURO				MEMORIA RAM				Nodo en ejecución
			INTRANET	TAMANO VIRTUAL	TAMANO REAL	TOTAL GB	USADO GB	DISP GB	USO	INSTALADA GB	OPTIMIZADO GB	UTILIZABLE GB	
BDD Producción RS	- Pagos		192.168.10.34	10	14	5.7	2.6	2.9	47%	32	32	11	nodo 3
	- RS2												
	- RS3												
	- RS Fotos												
	- RegistroSocialFCC												
	- Siirs												
	- Siirs-test												
	- imágenes												
	- justicia_jovenas												
	- monitoreo_rips												
- organizaciones													
- postgres													
- seguridades_rips													
Sibas Web	- ServicioImágenesCron		192.168.10.36	20	19	18	5.5	11	34%	8	8	4	cualquier host
	- antropometriaEnterprise												
	- SeguridadesMcdsEnterprise												
	- Sibas Enterprise												
	- SibasEnterprise												

Servidor de BDD 10.34 en el cual está instalado PostgreSQL. Ambiente utilizado es un entorno de Virtualización: RedHatEnterprise Virtualization. Cada cuchilla que forma parte del arreglo de host para el cluster de virtualización del chasis Blade es un servidor Blade HS23. El servidor 10.34 cuenta con un microprocesador de 8 cores (4 sockets de dos núcleos cada uno), esto dentro del ambiente de virtualización.

A.14

Cronograma

