

**SISTEMA DE INFORMACIÓN PARA EL FONDO DE EMPLEADOS DEL
INSTITUTO COLOMBIANO DE BIENESTAR FAMILIAR REGIONAL CALDAS**

**Hilda Lorena Echeverry López
Catalina Aranzazu Mejía**



**UNIVERSIDAD DE MANIZALES
FACULTAD DE INGENIERÍA
PROGRAMA INGENIERIA DE SISTEMAS Y TELECOMUNICACIONES
MANIZALES
2008**

**SISTEMA DE INFORMACION PARA EL FONDO DE EMPLEADOS DEL
INSTITUTO COLOMBIANO DE BIENESTAR FAMILIAR REGIONAL CALDAS**

**Hilda Lorena Echeverry López
Catalina Aranzazu Mejía**

Trabajo de Grado presentado como opción parcial para optar
al título de Ingeniero de Sistemas y Telecomunicaciones

Presidente
Carlos Alberto Gutierrez Rodas
Ingeniero de Sistemas

**UNIVERSIDAD DE MANIZALES
FACULTAD DE INGENIERÍA
PROGRAMA INGENIERIA DE SISTEMAS Y TELECOMUNICACIONES
MANIZALES
2008**

PÁGINA DE ACEPTACIÓN

**JAIME ARIEL RIOS OROZCO
JURADO**

**DIANA PATRICIA CHALA
JURADO**

**NESTOR ALZATE
JURADO**

Manizales, 11 de Julio de 2008

CONTENIDO

INTRODUCCIÓN.....	1
1. ÁREA PROBLEMÁTICA.....	3
2. OBJETIVOS.....	5
OBJETIVOS ESPECÍFICOS	5
3. JUSTIFICACIÓN.....	6
4. MARCO TEÓRICO	8
4.1 SISTEMAS DE INFORMACIÓN	8
4.1.1 ELEMENTOS DEL SISTEMA DE INFORMACIÓN.....	8
4.1.2 TIPOS DE SISTEMAS DE INFORMACIÓN.....	9
4.2 FONDOS DE EMPLEADOS	9
4.2.1 DEFINICIÓN.....	9
4.2.2 VÍNCULO DE ASOCIACIÓN	10
4.2.3 HISTORIA DE LOS FONDO DE EMPLEADOS.....	10
4.2.4 OBJETIVOS.....	10
4.2.5 CARACTERÍSTICAS	10
4.3 ICBF	11
4.4 ANTECEDENTES	11
4.4.1 FONDO FROSST.....	11
4.4.2 SICCOOPWEB.....	11
4.4.3 LINUX.....	11
4.4.4 SAFIX y SX.....	12
4.4.5 GESTION COOPERATIVO CAFÉ.....	12
4.5 UML.....	12
4.6 JAVA.....	13
4.7 JSP	14
4.8 BASES DE DATOS.....	15
4.9 JDBC	16
4.10 DREAMWEAVER	16
4.11 POSTGRES.....	17
4.12 SQL	17
4.13 TOMCAT	18
4.14 RATIONAL MODELER	19
5. METODOLOGÍA	20
5.1 TIPO DE TRABAJO	20
5.2 PROCEDIMIENTO	20
5.2.1 FASE 1	20
5.2.2 FASE 2	20

5.2.3 FASE 3	21
5.2.4 FASE 4	21
5.2.5 FASE 5	21
5.2.6 FASE 6	22
5.2.7 FASE 7	22
5.2.8 FASE 8	22
6. RESULTADOS.....	23
6.1 DESCRIPCIÓN DE RESULTADOS	23
6.2 DISCUSIÓN DE RESULTADOS	24
7. CONCLUSIONES	25
8. RECOMENDACIONES	26
BIBLIOGRAFÍA	27
LISTADE FIGURAS	
LISTA DE ANEXOS	

LISTA DE FIGURAS

	Pág.
FIGURA 1: CASOS DE USO (ACTOR ASOCIADO)	31
FIGURA 2 CASO DE USO (CONTEXTO: ACTOR ADMINISTRADOR)	32
FIGURA 3 DIAGRAMA DE ACTIVIDAD (ADMINISTRAR)	34
FIGURA 4 DIAGRAMA DE ACTIVIDAD (PROCESAR CONSULTAS)	35
FIGURA 5 DIAGRAMA DE ACTIVIDAD (PROCESAR CRÉDITOS)	36
FIGURA 6 DIAGRAMA DE ACTIVIDAD (PROCESAR NÓMINA)	37
FIGURA 7 DIAGRAMA DE ACTIVIDAD: PÁGINA PRINCIPAL	39
FIGURA 8 DIAGRAMA DE CLASES ANÁLISIS: PÁGINA PRINCIPAL	40
FIGURA 9 DIAGRAMA DE ACTIVIDAD: PÁGINA ADMINISTRATIVA	41
FIGURA 10 DIAGRAMA DE ACTIVIDAD: PÁGINA DE PROCESOS	42
FIGURA 11 DIAGRAMA DE CLASES ANÁLISIS: PROCESOS DE AHORRO	43
FIGURA 12 DIAGRAMA DE ACTIVIDAD: PROCESO DE CRÉDITO	44
FIGURA 13 DIAGRAMA DE CLASES ANÁLISIS: PROCESO DE CRÉDITO	45
FIGURA 14 PROCESO DE REPORTE PARA NÓMINA (AHORRO)	46
FIGURA 15 DIAGRAMA DE CLASES ANÁLISIS: PROCESO DE REPORTE PARA NÓMINA (AHORRO)	47
FIGURA 16 PROCESO DE REPORTE PARA NÓMINA (CRÉDITO)	48
FIGURA 17 PROCESO DE REPORTE PARA NÓMINA (CRÉDITO)	49
FIGURA 18 DIAGRAMA DE CLASES NÚCLEO: ACTIVOS FIJOS	50
FIGURA 19 DIAGRAMA DE CLASES NÚCLEO: ADMINISTRATIVAS Y PROCEDIMENTALES	51
FIGURA 20 DIAGRAMA DE COMPONENTES: DISTRIBUCIÓN DE RECURSOS	53
FIGURA 21 DIAGRAMA DE INTERFAZ: PÁGINA PRINCIPAL	54
FIGURA 22 DIAGRAMA DE INTERFAZ: MENÚ ADMINISTRADOR	55
FIGURA 23 DIAGRAMA DE COMPONENTES: MENÚ ADMINISTRADOR	56
FIGURA 24 DIAGRAMA DE INTERFAZ: INGRESAR CLASE DE ACTIVOS	57
FIGURA 25 DIAGRAMA DE COMPONENTES: INGRESAR CLASE DE ACTIVOS	58
FIGURA 26 DIAGRAMA DE INTERFAZ: CONSULTA CLASE DE ACTIVOS	60
FIGURA 27 DIAGRAMA DE COMPONENTES: CONSULTA CLASE DE ACTIVOS	60
FIGURA 28 DIAGRAMA DE INTERFAZ: MODIFICA CLASE DE ACTIVOS	62
FIGURA 29 DIAGRAMA DE COMPONENTES: MODIFICA CLASE DE ACTIVOS	63
FIGURA 30 DIAGRAMA DE INTERFAZ: ELIMINA CLASE DE ACTIVOS	65
FIGURA 31 DIAGRAMA DE COMPONENTES: ELIMINA CLASE DE ACTIVOS	65
FIGURA 32 DIAGRAMA DE INTERFAZ: INGRESAR ESTADOS	67
FIGURA 33 DIAGRAMA DE COMPONENTES: INGRESAR ESTADOS	68
FIGURA 34 DIAGRAMA DE INTERFAZ: CONSULTA ESTADOS	70
FIGURA 35 DIAGRAMA DE COMPONENTES: CONSULTA ESTADOS	70
FIGURA 36 DIAGRAMA DE INTERFAZ: MODIFICA ESTADOS	72
FIGURA 37 DIAGRAMA DE COMPONENTES: MODIFICA ESTADOS	73
FIGURA 38 DIAGRAMA DE INTERFAZ: ELIMINA ESTADOS	75
FIGURA 39 DIAGRAMA DE COMPONENTES: ELIMINA ESTADOS	75
FIGURA 40 DIAGRAMA DE INTERFAZ: INGRESAR PROVEEDORES	77
FIGURA 41 DIAGRAMA DE COMPONENTES: INGRESAR PROVEEDORES	78
FIGURA 42 DIAGRAMA DE INTERFAZ: CONSULTA PROVEEDOR	80
FIGURA 43 DIAGRAMA DE COMPONENTES: CONSULTA PROVEEDOR	80
FIGURA 44 DIAGRAMA DE INTERFAZ: MODIFICA PROVEEDOR	83

FIGURA 45 DIAGRAMA DE COMPONENTES: MODIFICA PROVEEDOR	84
FIGURA 46 DIAGRAMA DE INTERFAZ: ELIMINA PROVEEDOR	88
FIGURA 47 DIAGRAMA DE COMPONENTES: ELIMINA PROVEEDOR	88
FIGURA 48 DIAGRAMA DE INTERFAZ: INGRESAR RESPONSABLE.....	90
FIGURA 49 DIAGRAMA DE COMPONENTES: INGRESAR RESPONSABLE.....	91
FIGURA 50 DIAGRAMA DE INTERFAZ: CONSULTA RESPONSABLE	93
FIGURA 51 DIAGRAMA DE COMPONENTES: CONSULTA RESPONSABLE	93
FIGURA 52 DIAGRAMA DE INTERFAZ: MODIFICA RESPONSABLE.....	96
FIGURA 53 DIAGRAMA DE COMPONENTES: MODIFICA RESPONSABLE	97
FIGURA 54 DIAGRAMA DE INTERFAZ: ELIMINA RESPONSABLE	100
FIGURA 55 DIAGRAMA DE COMPONENTES: ELIMINA RESPONSABLE	100
FIGURA 56 DIAGRAMA DE INTERFAZ: INGRESAR UBICACIÓN	102
FIGURA 57 DIAGRAMA DE COMPONENTES: INGRESAR UBICACIÓN	102
FIGURA 58 DIAGRAMA DE INTERFAZ: CONSULTA UBICACIÓN.....	104
FIGURA 59 DIAGRAMA DE COMPONENTES: CONSULTA UBICACIÓN.....	104
FIGURA 60 DIAGRAMA DE INTERFAZ: MODIFICA UBICACIÓN	106
FIGURA 61 DIAGRAMA DE COMPONENTES: MODIFICA UBICACIÓN.....	107
FIGURA 62 DIAGRAMA DE INTERFAZ: ELIMINA UBICACIÓN.....	109
FIGURA 63 DIAGRAMA DE COMPONENTES: ELIMINA UBICACIÓN.....	109
FIGURA 64 DIAGRAMA DE COMPONENTES: INGRESAR ASOCIADOS	111
FIGURA 65 DIAGRAMA DE COMPONENTES: INGRESAR ASOCIADOS	112
FIGURA 66 DIAGRAMA DE INTERFAZ: CONSULTA ASOCIADO	115
FIGURA 67 DIAGRAMA DE COMPONENTES: CONSULTA ASOCIADO	116
FIGURA 68 DIAGRAMA DE COMPONENTES: MODIFICAR ASOCIADO	119
FIGURA 69 DIAGRAMA DE COMPONENTES: MODIFICA ASOCIADO	120
FIGURA 70 DIAGRAMA DE INTERFAZ: ELIMINA ASOCIADO	124
FIGURA 71 DIAGRAMA DE COMPONENTES: ELIMINA ASOCIADO	124
FIGURA 72 DIAGRAMA DE INTERFAZ: INGRESAR PLAZOS	126
FIGURA 73 DIAGRAMA DE COMPONENTES: MENÚ PROCESADOR.....	127
FIGURA 74 DIAGRAMA DE INTERFAZ: CONSULTAR PLAZOS	129
FIGURA 75 DIAGRAMA DE COMPONENTES: CONSULTAR PLAZOS	129
FIGURA 76 DIAGRAMA DE INTERFAZ: MODIFICAR PLAZOS.....	131
FIGURA 77 DIAGRAMA DE COMPONENTES: MODIFICAR PLAZOS.....	132
FIGURA 78 DIAGRAMA DE INTERFAZ: ELIMINAR PLAZOS	135
FIGURA 79 DIAGRAMA DE INTERFAZ: ELIMINAR PLAZOS	135
FIGURA 80 DIAGRAMA DE INTERFAZ: INGRESAR CLASES CREDITOS.....	137
FIGURA 81 DIAGRAMA DE COMPONENTES: INGRESAR CLASES CREDITOS	138
FIGURA 82 DIAGRAMA DE INTERFAZ: CONSULTAR CLASES CREDITOS	140
FIGURA 83 DIAGRAMA DE COMPONENTES: CONSULTAR CLASES CREDITOS	141
FIGURA 84 DIAGRAMA DE INTERFAZ: MODIFICAR CLASES CRÉDITO.....	143
FIGURA 85 DIAGRAMA DE COMPONENTES: MODIFICAR CLASES CRÉDITO.....	144
FIGURA 86 DIAGRAMA DE INTERFAZ: ELIMINAR CLASES CRÉDITO.....	146
FIGURA 87 DIAGRAMA DE COMPONENTES: ELIMINAR CLASES CRÉDITO	147
FIGURA 88 DIAGRAMA DE INTERFAZ: INGRESAR ACTIVIDADES	149
FIGURA 89 DIAGRAMA DE COMPONENTES: INGRESAR ACTIVIDADES.....	149
FIGURA 90 DIAGRAMA DE COMPONENTES: CONSULTAR ACTIVIDADES	151
FIGURA 91 DIAGRAMA DE COMPONENTES: CONSULTAR ACTIVIDADES	152
FIGURA 92 DIAGRAMA DE INTERFAZ: MODIFICAR ACTIVIDADES.....	154

FIGURA 93 DIAGRAMA DE COMPONENTES: MODIFICAR ACTIVIDADES.....	155
FIGURA 94 DIAGRAMA DE INTERFAZ: ELIMINAR ACTIVIDADES.....	157
FIGURA 95 DIAGRAMA DE COMPONENTES: CONSULTAR ACTIVIDADES.....	158
FIGURA 96 DIAGRAMA DE INTERFAZ: INGRESAR ANALISTA.....	160
FIGURA 97 DIAGRAMA DE COMPONENTES: INGRESAR ANALISTA.....	161
FIGURA 98 DIAGRAMA DE INTERFAZ: CONSULTAR ANALISTAS.....	163
FIGURA 99 DIAGRAMA DE COMPONENTES: CONSULTAR ANALISTAS.....	164
FIGURA 100 DIAGRAMA DE INTERFAZ: MODIFICAR ANALISTAS.....	167
FIGURA 101 DIAGRAMA DE COMPONENTES: MODIFICAR ANALISTAS.....	168
FIGURA 102 DIAGRAMA DE INTERFAZ: ELIMINAR ANALISTAS.....	172
FIGURA 103 DIAGRAMA DE COMPONENTES: ELIMINAR ANALISTAS.....	173
FIGURA 104 DIAGRAMA DE INTERFAZ: MODULO PROCESADOR.....	175
FIGURA 105 DIAGRAMA DE COMPONENTES: MODULO PROCESADOR.....	176
FIGURA 106 DIAGRAMA DE INTERFAZ: REGISTRO DE CRÉDITOS.....	177
FIGURA 107 DIAGRAMA DE COMPONENTES: REGISTRO DE CRÉDITOS.....	178
FIGURA 108 DIAGRAMA DE INTERFAZ: CONSULTA DE CRÉDITOS.....	181
FIGURA 109 DIAGRAMA DE COMPONENTES: CONSULTA DE CRÉDITOS.....	182
FIGURA 110 109 DIAGRAMA DE INTERFAZ: CANCELAR CRÉDITOS.....	185
FIGURA 111 DIAGRAMA DE COMPONENTES: CANCELAR CRÉDITOS.....	186
FIGURA 112 DIAGRAMA DE INTERFAZ: REGISTRO DE CUENTAS.....	188
FIGURA 114 DIAGRAMA DE INTERFAZ: CONSULTA DE CUENTAS.....	191
FIGURA 115 DIAGRAMA DE COMPONENTES: CONSULTA DE CUENTAS.....	192
FIGURA 116 DIAGRAMA DE INTERFAZ: RETIRO DE CUENTAS.....	195
FIGURA 117 DIAGRAMA DE COMPONENTES: RETIRO DE CUENTAS.....	196
FIGURA 118 DIAGRAMA DE INTERFAZ: GENERACIÓN DE REPORTE MENSUAL.....	198
FIGURA 119 DIAGRAMA DE COMPONENTES: GENERACIÓN DE REPORTE MENSUAL.....	199
FIGURA 120 DIAGRAMA DE INTERFAZ: DEPURACION MOVIMIENTO.....	202
FIGURA 121 DIAGRAMA DE COMPONENTES: DEPURACIÓN MOVIMIENTO.....	203
FIGURA 122 DIAGRAMA DE INTERFAZ: GENERACIÓN REPORTE NÓMINA.....	205
FIGURA 123 DIAGRAMA DE COMPONENTES: GENERACIÓN REPORTE NÓMINA.....	206
FIGURA 124 DIAGRAMA DE INTERFAZ: MENÚ ASOCIADO.....	209
FIGURA 125 DIAGRAMA DE COMPONENTES: MENÚ ASOCIADO.....	210
FIGURA 126 DIAGRAMA DE INTERFAZ: CONSULTA SALDO AHORROS.....	211
FIGURA 127 DIAGRAMA DE COMPONENTES: CONSULTA SALDO AHORROS.....	212
FIGURA 128 DIAGRAMA DE INTERFAZ: CONSULTA SALDO EN CRÉDITO.....	214
FIGURA 129 DIAGRAMA DE COMPONENTES: CONSULTA SALDO EN CREDITO.....	215

LISTA DE ANEXOS

	Pág.
ANEXO A. MODELO DE REQUISITOS.....	31
ANEXO B. MODELO DE DISEÑO.....	53
ANEXO C. MODELO DE PRUEBAS.....	218
ANEXO D. CREACIÓN DE LA BASE DE DATOS.....	222
ANEXO E. MANUAL TECNICO	218

RESUMEN

El Fondo de Empleados del Instituto Colombiano de Bienestar Familiar es una entidad que presta los servicios de ahorro y sus derivados, como la liquidación de intereses y prestamos.

El proyecto es planteado inicialmente desde un punto de vista para el desarrollo tecnológico, en el área de la gestión para los procesos de ahorro y préstamo en línea.

La interacción vía Internet permitirá agilizar el proceso de consulta por parte de los asociados, los cuales podrán ver el estado de sus cuentas por medio de consultas diseñadas para ser vistas por la Web.

Actualmente se presenta una problemática, la cual radica en demoras en los procedimientos, debido a que todo el proceso de verificación y generación de informes mensuales para nómina se hacen en forma manual, aumentando estadísticamente el índice de errores, lo que genera la necesidad de reproceso para su validación.

Las quejas y reclamos hacen del Fondo de Empleados del Instituto Colombiano de Bienestar Familiar sede Manizales una entidad saturada en procesos de verificación y validación de la información; donde uno de los factores que influyen negativamente es que solo se cuenta con un funcionario para todos los menesteres, hecho que demanda la necesidad sentida de un sistema de gestión de descuentos y consultas en tiempo real.

PALABRAS CLAVES:

Fondo de Empleados
Procesos
Web
Sistema de información

ABSTRACT

The Fund for Employees of the Colombian Institute of Family Welfare is an entity that provides savings services and its derivatives, such as interest and liquidation of loans.

The project was initially raised terms for the technological development in the area of management processes for savings and loans online.

The interaction via the Internet will streamline the consultation process by partners, who may view the status of their accounts through consultations designed to be viewed on the Web.

Today is a problem, which lies in delays in proceedings due to the fact that the entire process of verification and reporting for monthly payroll made by hand, increasing the statistical error rate, which generates the need for reprocessing for validation.

The complaints made Fund Employees of the Colombian Institute of Family Welfare headquarters Manizales an entity saturated process of verification and validation of information, where one of the factors that adversely affect is that there is only one officer for every actions , which demand the felt need of a management system of discounts and consultations in real time.

KEY WORDS:

Found of Employees
Process
Web

RESUMEN ANALÍTICO

Título del Proyecto	Sistema de información para el Fondo de Empleados de bienestar Familiar – Regional Caldas
Autor(es)	ECHEVERRY LOPEZ, Hilda Lorena hiloelo@hotmail.com ARANZAZU MEJIA, Catalina cata.aranzazu@gmail.com
Presidente	GUTIRREZ RODAS, Carlos Alberto carlos.gutierrez@icbf.gov.co Ingeniero de Sistemas, Jefe de Sistemas del ICBF, ICBF – Regional Caldas
Tipo de documento	Trabajo de grado
Referencia documento	Hilda Lorena Echeverry L, Catalina Aranzazu M. Título Sistema de información para el Fondo de Empleados de bienestar Familiar – Regional Caldas. Manizales, 2008, 230 pags. Universidad de Manizales, Facultad de Ingeniería.
Institución	Ingeniería de Sistemas y Telecomunicaciones, Facultad de Ingeniería, Universidad de Manizales.
Palabras claves	Fondo de Empleados Sistema de información Web
Descripción	<p>El Fondo de Empleados del Instituto Colombiano de Bienestar Familiar es una entidad que presta los servicios de ahorro y sus derivados, como la liquidación de intereses y préstamos.</p> <p>El proyecto es planteado inicialmente desde un punto de vista para el desarrollo tecnológico, en el área de la gestión para los procesos de ahorro y préstamo en línea.</p> <p>La interacción vía Internet permitirá agilizar el proceso de consulta por parte de los asociados, los cuales podrán ver el estado de sus cuentas por medio de consultas diseñadas para ser vistas por la WEB.</p>

Actualmente se presenta una problemática, la cual radica en demoras en los procedimientos, debido a que todo el proceso de verificación y generación de informes mensuales para nómina se hacen en forma manual, aumentando estadísticamente el índice de errores, lo que genera la necesidad de reproceso para su validación.

Las quejas y reclamos hacen del Fondo de Empleados del Instituto Colombiano de Bienestar Familiar sede Manizales una entidad saturada en procesos de verificación y validación de la información; donde uno de los factores que influyen negativamente es que solo se cuenta con un funcionario para todos los menesteres, hecho que demanda la necesidad sentida de un sistema de gestión de descuentos y consultas en tiempo real.

Metodología

Para el análisis, el diseño y la implementación de este sistema se utilizó una metodología que permitiera llegar a el desarrollo completo y satisfactorio del sistemas de información, teniendo en cuenta que esto depende de las metodologías utilizadas, de la interacción de los usuarios responsables del desarrollo y de las herramientas utilizadas para el proceso completo de un proyecto de tipo informático como lo es el Sistema de Información Integrado para el Fondo de Empleados del Instituto Colombiano de Bienestar Familiar.

Para cumplir con este proceso se basó toda la metodología en el ciclo de vida del software, permitiendo hacerle un completo seguimiento al sistema en cada una de sus fases.

Con el desarrollo de este Sistema de Información realizamos una gestión administrativa del Fondo de Empleados, buscando que las operaciones administrativas se ejecuten con fluidez y eficacia.

Conclusiones

El desarrollo del proyecto aportó experiencias que enriquecen la vida profesional de los actores del mismo, la investigación aplicada necesaria para el desarrollo tecnológico es punto de partida para una dinámica profesional, debido a que la naturaleza de las temáticas manejadas por los ingenieros de sistemas y

telecomunicaciones son muy cambiantes, es necesario obtener una cultura de actualización y adaptación de técnicas y tecnologías no cursadas durante el transcurso de la carrera.

Como aporte altamente significativo se incluyó un almacén de datos para los activos fijos manejados por el Fondo, pese a la no inclusión como parte inicial de los objetivos específicos, dicho módulo se establece como novedad.

El uso de tecnologías WEB en el desarrollo del proyecto fue un factor clave de éxito para su aceptación y aplicación, debido a que es necesario y ponderante hacer uso de los recurso con los que cuenta el Instituto Colombiano de Bienestar Familiar regional Caldas. El uso de plataformas existentes en las empresas hace de los proyectos de desarrollo efectivos en el uso y explotación de recursos, convirtiéndose en una fortaleza importante de éxito.

El uso de una herramienta de gestión de procesos como es el WorkFlow, llegó a ser de una gran ayuda para todo el proceso de sistematización de solicitudes de crédito, además estamos recurriendo a la utilización de herramientas que permiten no solo realizar la actividad propia de un fondo como son la manipulación de créditos y ahorros si no también interactuar de forma clara y eficiente con el usuario final en este caso el asociado.

Anexos

Modelo de Requisitos
Modelo de Análisis
Modelo de Diseño
Modelo de Pruebas
Creación de La Base de datos

INTRODUCCIÓN

El Fondo de Empleados del Instituto Colombiano de Bienestar Familiar es una entidad que presta los servicios de ahorro y sus derivados, como la liquidación de intereses y prestamos. Su gestión básicamente es administrativa, cumpliendo los estándares establecidos por la misma entidad y las reglas indicadas en los estatutos de dicha entidad.

El proyecto se plantea inicialmente desde un punto de vista para el desarrollo tecnológico, en el área de la gestión para los procesos de ahorro y préstamo en línea, es decir, lograr realizar los procedimientos que actualmente se hacen en el Fondo de Empleados por medio de un sistema (software) que permita interactuar vía WEB con sus clientes.

El proyecto pretende cubrir las necesidades de automatización para los procesos de cartera y servicios en línea, es decir, tomando como base el hallazgo inicial, el cual vislumbra procesos llevados manualmente, se pretende crear un sistema que además de llevar los procesos en forma automática también permita la consulta y administración vía Internet, permitiendo a los usuarios interactuar con el Fondo de Empleados en forma más ágil y oportuna.

La interacción vía Internet permitirá agilizar el proceso de consulta por parte de los asociados, los cuales podrán ver el estado de sus cuentas por medio de consultas diseñadas para ser vistas por la WEB.

Los procesos se llevan actualmente en forma manual, tanto para brindar información respecto a los estados de cuenta, como para el proceso de descuentos de nómina en ahorros y créditos. El sistema pretende agilizar dichos procedimientos mediante su automatización, generando mayor confianza y agilidad en los procesos.

La naturaleza WEB del proyecto exige perfiles de usuario, dónde se hace necesario realizar un estudio de perfiles, por lo tanto el producto final permitirá y por medio de un solo gestor de ingreso, identificar el perfil e ingresar a un sistema diferente para cada necesidad, es decir, en caso de ingresar un usuario del fondo de empleados le permitirá conocer la información respecto a ahorros y créditos y en caso de ingresar como administrador, a éste se le permitirá optar por cualquier opción del sistema.

Para su instalación será necesario configurar un servidor con características WEB,

que permita acceso desde cualquier punto de red del Instituto Colombiano de Bienestar Familiar sede Manizales.

Como producto final se obtiene un software orientado a la WEB, el cual permitirá su gestión en Internet, permitiendo agilizar los procesos y procedimientos que competen con el Fondo de Empleados del Instituto Colombiano de Bienestar Familiar sede Manizales.

1. ÁREA PROBLEMÁTICA

El Fondo de Empleados del Instituto Colombiano de Bienestar Familiar sede Manizales es una entidad interna que presta los servicios de ahorro y crédito para los empleados.

Su gestión principal radica en la administración integral de todos los recursos financieros, los cuales ingresan a manera de recaudo voluntario, mediante un descuento por nómina.

Nómina es un departamento del Instituto Colombiano de Bienestar Familiar sede Manizales, el cual es el encargado de procesar los pagos de los empleados. Dicho departamento en una de sus funciones, se encarga de registrar las novedades de los empleados, y cabe anotar que una de las novedades importantes para el fondo de empleados es el descuento mensual sobre sus ingresos, y para el caso del Fondo es la cuota de ahorro que será abonada al saldo del empleado.

El problema radica en demoras en los procedimientos, debido a que todo el proceso de verificación y generación de informes mensuales para nómina se hacen en forma manual, aumentando estadísticamente el índice de errores, lo que genera la necesidad de reproceso para su validación.

Otra tarea cumplida por el Fondo de Empleados del Instituto Colombiano de Bienestar Familiar sede Manizales es el servicio de préstamo, dónde se pacta un compromiso de pago en cuotas e interés con el empleado, iniciando así un descuento mensual hasta pagar el total de la deuda.

Según el sistema de crédito bajo las políticas del Fondo, es necesario reportar las novedades al departamento de nómina para su posterior descuento, dicho proceso se hace en forma manual y sin un control que permita fácil auditoría, lo que implica un problema que genera demoras e información no disponible en tiempo real, generando desconfianza en los empleados, los cuales hacen su propio control para su verificación debido a las inconsistencias que históricamente han presentado sus descuentos¹.

La problemática genera un alto índice de solicitudes para consultas de saldos en ahorro, cuotas de créditos pagadas y saldos pendientes. Teniendo en cuenta que

¹ Conforme ha pasado el tiempo se han registrado errores no intencionados, dichas inconsistencias generan aumento de cuotas para pagar los créditos o en algunos casos y por errores en los registros el no descuento de las cuotas para su ahorro.

los procesos y procedimientos se hacen en forma manual, la prestación del servicio se hace cada vez con menos eficiencia y eficacia.

Las quejas y reclamos hacen del Fondo de Empleados del Instituto Colombiano de Bienestar Familiar sede Manizales una entidad saturada en procesos de verificación y validación de la información; donde uno de los factores que influyen negativamente es que solo se cuenta con un funcionario para todos los menesteres, hecho que demanda la necesidad sentida de un sistema de gestión de descuentos y consultas en tiempo real.

Debido a la falta de interconexión entre las herramientas de software utilizadas para realizar las actividades necesarias en el fondo de empleados el flujo de información entre el origen y los usuarios tiene algunos retrasos y en algunos casos redundancia de datos.

A parte de todo lo anterior, la contabilidad del Fondo de Empleados de Bienestar Familiar es llevada en una aplicación que no cumple con todas las necesidades, ya que en algunos casos se requieren realizar actividades que no están incluidas dentro de la herramienta.

2. OBJETIVOS

OBJETIVO GENERAL

Realizar e implementar un sistema de Información integrado vía Web, para el Fondo de Empleados del Instituto Colombiano de Bienestar Familiar, que le permita tener alta agilidad y disponibilidad sobre la información, e interactuar con usuarios y vinculados.

OBJETIVOS ESPECÍFICOS

- Analizar la situación actual de las aplicaciones que posee el Fondo de Empleados del Instituto Colombiano de Bienestar Familiar.
- Diseñar e Implementar el Sistema de Información bajo la metodología orientada a objetos con la notación UML 2.0.
- Crear módulos de Gestión de asociados, Gestión de servicios, Gestión de aportes y/o ahorros, Crédito y Cartera, Servicio al Cliente, Gestión de Deducciones.
- Diseñar la pagina Web del Fondo de Empleados del Instituto Colombiano de Bienestar Familiar que le permita al usuario manejar desde Internet diferentes actividades realizadas por el Fondo.

3. JUSTIFICACIÓN

Durante los últimos años las tendencias en sistemas de información se han dirigido a garantizar los servicios y herramientas necesarias para que los usuarios, desarrollen sus actividades de forma óptima y segura.

Actualmente el Fondo de Empleados del Instituto Colombiano de Bienestar Familiar tiene problemas en el manejo óptimo de la información, ya que este se ha venido desarrollando de manera aislada, cada dependencia realiza sus actividades independientemente y no existe una forma ágil para organizar toda la información, debido a esto, a la hora de unir toda la información no existe compatibilidad entre las aplicaciones utilizadas para cada actividad, generando con esto aislamiento entre las divisiones del Fondo de Empleados, ausencia rápida de respuestas ante eventos y en ocasiones, errores en la integridad de los datos.

Además, algunas actividades propias del Fondo de Empleados son realizadas de forma manual, lo que conlleva a que el trabajo requiera más tiempo de ejecución, impidiendo con esto tomar decisiones en el momento adecuado y necesario; por otro lado el manejo de la información en el momento de consultar o modificar datos es demasiado complejo y esto debido a que los errores encontrados en determinado trabajo significan volver a realizar la misma actividad manualmente lo que conlleva a la sobre utilización del tiempo, recursos materiales, humanos y la inutilización de la tecnología la cual podría ser de gran ayuda para realizar este tipo de trabajos. El Fondo de Empleados del Instituto Colombiano de Bienestar Familiar no cuenta con un sistema de información adecuado para el manejo y control de la información en cuanto a Gestión de asociados, Gestión de servicios, Gestión de aportes y/o ahorros, Crédito y Cartera, Servicio al Cliente, Generación Contable; pues en este momento los datos son manejados de una forma manual lo cual requiere de más tiempo no sólo en la búsqueda sino también en la actualización de cada dato.

Por medio del sistema lo que se busca es brindar a los usuarios un medio más eficaz para acceder a cualquier tipo de información ya sea del usuario como tal o del funcionamiento administrativo del Fondo de Empleados, de tal forma que los usuarios van a poder utilizar los servicios que brinda la entidad.

Es novedoso ya que presenta la característica de poder interactuar con el usuario de una forma más directa por medio de una conexión Web, la cual serviría para realizar tareas como obtener información general sobre el fondo de empleados, líneas de crédito, simulación de capacidad de pago y de un crédito, hacer una presolicitud de crédito, consultar el estado de sus créditos actuales y eventos.

Por otro lado, se implementó una herramienta de WorkFlow para la gestión de la automatización de los flujos de datos en las solicitudes de crédito y los procesos administrativos que este tipo de actividades conlleva como son los controles de autorizaciones, verificación de datos y la relación de las personas involucradas en cada actividad; con esta herramienta de gestión le estamos dando al sistema una funcionalidad que no se encuentra comúnmente en este tipo de software para los Fondos de Empleados garantizando con esto su novedad y una nueva característica de interés para los usuarios.

Finalmente, el Sistema de Información, es un desarrollo que va a beneficiar a todo un grupo de asociados y trabajadores en cuanto a la comodidad, agilidad y eficiencia que se requieren en una entidad que maneja un amplio número de información la cual tiene que ser exacta, concisa y además en tiempo real.

4. MARCO TEÓRICO

4.1 SISTEMAS DE INFORMACIÓN

El sistema de información es la razón de ser para la tecnología de cómputo. Debido a que hay muchos tipos de información y muchos usos para ella, han sido desarrollados varias clases de sistemas de información.

Los sistemas de información hacen mucho más que almacenar y recuperar datos. Ayuda a la gente a usar información de formas innumerables, ya sea que eso involucre clasificar listas, imprimir reportes, comparar una sola huella digital, etc., en la creación de cualquier sistema de información entra una gran cantidad de planeación; un sistema complejo puede tomar meses para desarrollarse y después poner a producir a los talentos de docenas de hábiles profesionales.

Los sistemas de información no son nuevos. Mucho antes de la automatización de las computadoras, las compañías reunían, almacenaban y actualizaban información en lo que era el curso normal de hacer negocios. En el pasado como ahora, los sistemas de información consistían en los procedimientos y reglas establecidas para entregar información a la gente dentro de una organización. Diferentes personas requieren distinta información para realizar su trabajo, y las reglas del sistema gobiernan qué información debería ser distribuida a cada persona, cuando y en que formato.

4.1.1 ELEMENTOS DEL SISTEMA DE INFORMACIÓN

Un sistema de información, según Peralta, está compuesto de los siguientes elementos, que actúan entre sí para dar soporte a las actividades empresariales:

- **Documentos:** Manuales, los artículos impresos, u otra información que describe el uso que explica el uso y/o la forma de operar el sistema.
- **Procedimiento:** Conjunto de procedimientos que exponen el uso de cada elemento o parte del sistema de información o desempeño general.
- **Bases de Datos:** Es la colección de datos que se relacionan entre si los cuales son accedidos por el software y es el resultado de la relación completa de la información.
- **Personas:** Todos los individuos que son usuarios u operadores del sistema o se relacionan con el sistema.
- **Hardware:** Dispositivos electrónicos que dan soporte a la capacidad de computación y los dispositivos electromecánicos de donde se saca la información contenida en los soportes magnéticos.

- Software: Son los programas encargados de administrar la información que es requerida por el cliente o administrativo, de acuerdo a los roles que realiza cada uno dentro del sistema.²

4.1.2 TIPOS DE SISTEMAS DE INFORMACIÓN

Conforme mas y mas funciones de negocios han sido automatizadas, los sistemas de información se han vuelto crecientemente especializados. Tipos especializados de sistemas de información:

- Sistemas de automatización para oficinas: Las tareas rutinarias de oficina: enviar cartas o rastrear horarios, ahora están automatizadas por los sistemas de computo.
- Sistemas para procesamiento de transacciones: Las transacciones son eventos como tomar una orden o pagar una factura, transacciones para procedimiento de negocios significa cubrir, recuperar y rastrear datos acerca de los eventos.
- Sistemas de apoyo a las decisiones: estos sistemas con frecuencia dan a los administradores acceso directo a datos en el sistema de procesamiento para transacciones de la compañía.
- Sistemas de información para la administración: están diseñados para respaldar las necesidades de tres categorías diferentes de administradores: ejecutivos, administradores medios y de producción; estos sistemas generalmente producen una gama de reportes estandarizados.
- Sistemas expertos: automatiza el proceso para toma de decisiones en un área específica, como diagnósticos médicos o revisión de historias de crédito para aprobaciones de préstamos. Estos sistemas analizan datos y luego producen una recomendación para un curso de acción.

4.2 FONDOS DE EMPLEADOS³

4.2.1 DEFINICIÓN

"Son empresas asociativas, de derecho privado, sin ánimo de lucro, constituidas por trabajadores dependientes y subordinados". (Decreto 1481/89, artículo 2).

² Sistemas de Información.[En Línea]. Fecha de consulta 06.09.2001. Disponible en: <http://www.monografias.com/trabajos7/sisinf/sisinf.shtml>

³ Fondo de Empleados del Instituto Colombiano de Bienestar Familiar Regional Caldas.

4.2.2 VÍNCULO DE ASOCIACIÓN

Los asociados de un fondo de empleados, deberán tener un vínculo común de asociación, determinado por la calidad de trabajadores dependientes, en una de las siguientes modalidades:

De una misma institución o empresa.

De varias sociedades en las que se declare la unidad de empresa, de matrices o subordinados, o de entidades principales y adscritas y vinculadas, o de empresas que se encuentren integradas conformando un grupo empresarial.

De varias instituciones o empresas independientes entre sí, siempre que éstas desarrollen la misma actividad económica.

4.2.3 HISTORIA DE LOS FONDO DE EMPLEADOS

En Colombia este tipo de organizaciones nacieron con el siglo XX; al principio de manera informal, fuera de normas legales y posteriormente reglamentadas por diferentes normas; sin embargo el entramamiento burocrático, experiencias negativas en el manejo de algunas cooperativas fueron superadas por el interés de algunas personas, patronos o trabajadores, buscando formas no convencionales de cooperación, desde las cajas de obreros, las organizaciones conocidas como "natilleras", hasta llegar a lo que hoy se conoce como los Fondos de Empleados.

La posibilidad que los patronos colaboren en la creación o el desarrollo de los Fondos y puedan participar en la inspección y vigilancia de los recursos asignados, por una parte, y la cohesión que este tipo de asociados genera, han colocado a los Fondos de Empleados en un lugar muy importante dentro de los órganos asociativos del Sector Solidario.

4.2.4 OBJETIVOS

Contribuir en la solución de las necesidades de sus asociados, a través de los servicios de ahorro y crédito, y los demás que sean complementarios y que contribuyan a elevar la calidad de vida de los asociados.

4.2.5 CARACTERÍSTICAS

Que se integren básicamente con trabajadores asalariados.

Que la asociación y retiro sea voluntarios.

Que se garantice la igualdad de los derechos de participación y decisión de los asociados, sin consideración a sus aportes.

Que presten servicios en beneficio de sus asociados.

Que establezcan la irrepartibilidad de las reservas sociales y, en caso de liquidación, la del remanente patrimonial.

Que destinen los excedentes a la prestación de servicios de carácter social y al crecimiento de sus reservas y fondos.

Que su patrimonio sea variable e ilimitado.

Que se constituyan con duración indefinida.

Que fomenten la solidaridad y los lazos de compañerismo entre los asociados.

4.3 ICBF

El Instituto Colombiano de Bienestar Familiar, es la entidad adscrita al Ministerio de la Protección Social, es una de las instituciones más representativas del país. Fue creada en 1968 dando respuesta a problemáticas, tales como la deficiencia nutricional, la desintegración e inestabilidad de la familia, la pérdida de valores y la niñez abandonada.

4.4 ANTECEDENTES

4.4.1 FONDO FROSST. El sistema de información para los asociados podrá encontrar información específica de todos los datos necesarios para mantenerse informado de los respectivos descuentos, aportes y ahorros, como un sistema para acceder a todos los formularios necesarios para sus solicitudes de crédito.⁴

4.4.2 SICOOPWEB. Es un Sistema de Información para Cooperativas y Fondos de Empleados, desarrollado por JORJWEB LTDA en Colombia.

SICOOPWEB realiza automáticamente la mayoría de las operaciones en áreas tales como: Contabilidad, Cálculos Financieros, Depreciaciones, Causaciones, Conciliaciones, Liquidación de créditos, Reclasificación y Calificación de cartera, Recibos de caja y notas de contabilidad, Módulo de Transportes y otras operaciones en cada uno de los módulos. Posee un sistema de reconocimiento de usuarios por medio de password y huella lo cual permite mayor seguridad en el sistema.⁵

4.4.3 LINIX. Desarrollado por Sistemas En línea S.A. en Colombia, este es un sistema de información que permite la actualización en línea entre módulos y

⁴ Fondo Frosst. Fondo de Empleados Laboratorios Merck Sharp & Dohme [En Línea]. Bogotá. Fecha de consulta 03-06-2008. Catalogo de Software > Inicio > Sectores Específicos > Sector Solidario > Cooperativas / Fondos de Empleados. Disponible en: <http://www.fondofrosst.com/>

⁵ Jarzsotf Ltda. "SICOOPWEB" [En Línea]. Bogotá. Fecha de consulta 03-06-2008. Catalogo de Software > Inicio > Sectores Específicos > Sector Solidario > Cooperativas / Fondos de Empleados. Disponible en: <http://www.catalogodesoftware.com/producto.aspx?pid=327>

generación simultánea de documentos soportes. Es un software multiusuario que brinda agilidad seguridad e integridad. El sistema se compone de módulos como Contabilidad, Tesorería, Crédito Y Cartera, Aportes Y Servicios, Asociados, Descuentos Por Nomina, Cuentas De Ahorro, Cdts, Nomina, Activos Fijos, Inventarios, Compras, Ventas.⁶

4.4.4 SAFIX y SX. Es un Sistema de Información integrado, desarrollado por Sector Solidario que esta especializado en operaciones Comerciales, Administrativas y Financieras de Cooperativas y Fondos de Empleados. Realiza pagos electrónicos, permite el manejo del extracto del cliente, liquidación definitiva del asociado, manejo integrado de captación y colocación, descuentos, cuotas extras, control de capacidad de pago, entre otros.⁷

4.4.5 GESTION COOPERATIVO CAFÉ. Permite la integración de todos los procesos de la empresa, generando automáticamente la contabilidad, desde donde se originan los documentos fuentes, generación de informes generales y detallados para la toma de decisiones de las gestiones solidarias, comerciales, financieras, además proporciona todo el soporte operativo desde la creación de la empresa hasta obtener estados financieros.⁸

4.5 UML

El lenguaje Unificado de Modelado es la herramienta más utilizada para la definición y construcción de proyectos, debido a la facilidad de uso. Sus componentes son adecuados y además su aplicación es posible en todas las fases en el desarrollo de software.

UML y el proceso unificado son la composición más adecuada para el desarrollo de software, ya que tiene una adaptación para cada fase de construcción, es decir, tiene aplicación en la fase de requisitos, análisis, diseño, implementación, implantación y pruebas.

⁶ Sistemas Enlínea S.A. "LINIX" [En Línea]. Bogotá. Fecha de consulta 03-06-2008. Catalogo de Software > Inicio > Sectores Específicos > Sector Solidario > Cooperativas / Fondos de Empleados. Disponible en: <http://www.catalogodesoftware.com/producto.aspx?pid=673>

⁷ Xenco S.A. Sector Solidario "SAFIX Y SX". [En Línea]. Bogotá. Fecha de consulta 03-06-2008. Catalogo de Software > Inicio > Sectores Específicos > Sector Solidario > Cooperativas / Fondos de Empleados. Disponible en: <http://www.catalogodesoftware.com/producto.aspx?pid=1084>

⁸ SYSCAFE S.A. "Gestion Cooperativo Café". [En Línea]. Bogotá. Fecha de consulta 03-06-2008. Catalogo de Software > Inicio > Sectores Específicos > Sector Solidario > Cooperativas / Fondos de Empleados. Disponible en: <http://www.catalogodesoftware.com/producto.aspx?pid=1652>

Una de las características principales son los modelos, donde un modelo es una descripción de (parte de) un sistema, descrito en un lenguaje bien definido. Un lenguaje bien definido es un lenguaje con una sintaxis y semántica precisa, y que puede ser interpretado y manipulado por un computador.

Según consultas hechas en diferentes sitios de Internet los lenguajes de modelado que define OMG (Object Management Group) el más conocido y usado es sin duda UML (Unified Modelling Language). UML es un lenguaje gráfico para especificar, construir y documentar los artefactos que modelan un sistema. UML fue diseñado para ser un lenguaje de modelado de propósito general, por lo que puede utilizarse para especificar la mayoría de los sistemas basados en objetos o en componentes, y para modelar aplicaciones de muy diversos dominios de aplicación (telecomunicaciones, comercio, sanidad, etc.) y plataformas de objetos distribuidos (como por ejemplo J2EE, .NET o CORBA)⁹.

4.6 JAVA

Entre las características principales del lenguaje Java se encuentra entre los catalogados como orientado a objetos (P.O.O) cuya principal ventaja es su gran adaptabilidad a gran cantidad de plataformas como Windows, Linux, teléfonos móviles, webs, PDAS, etc. Cuando se escribe un programa en un lenguaje orientado a objetos, definimos una plantilla o clase que describe las características y el comportamiento de un conjunto de objetos similares. La clase automóvil describe las características comunes de todos los automóviles: sus atributos y su comportamiento. Los atributos o propiedades se refieren a la marca o fabricante, el color, las dimensiones, si tienen dos, tres, cuatro o más puertas, la potencia, si utiliza como combustible la gasolina o gasoil, etc. El comportamiento se refiere a la posibilidad de desplazarse por una carretera, frenar, acelerar, cambiar de marcha, girar, etc.

Según la OMG, los objetos encapsulan métodos y atributos, éstos se agrupan en una clase, dichos atributos son las características individuales que diferencian un objeto de otro y determinan su apariencia, estado u otras cualidades. Los atributos se guardan en variables denominadas de instancia, es decir, con una sola clase se pueden construir un sin número de objetos, y cada objeto particular puede tener valores distintos para estas variables¹⁰.

⁹ Revista Novatica–Asociación de Técnicos de Informática. Introducción a perfiles UML. [En Línea]. Fecha de Consulta: 30-01-2007. Disponible en : <http://www.lcc.uma.es/~av/Publicaciones>

¹⁰ Introducción. Conceptos básicos de la Programación Orientada a Objetos. [En Línea]. Fecha de Consulta: 28-04-2007. Disponible en : <http://www.sc.ehu.es/sbweb/fisica/cursoJava/fundamentos/clases1/clases.htm>

Las variables de instancia también denominados miembros dato, son declaradas en la clase pero sus valores son fijados y cambiados en el objeto, en algunos casos y para efectos de diseño se comportan como clases en el modelo de datos.

También se tiene como característica las variables de clase, las cuales se aplican a la clase y a todas sus instancias. Por ejemplo, el número de ruedas de un automóvil es el mismo cuatro, para todos los automóviles.

El comportamiento de los objetos de una clase se implementa mediante funciones miembro o métodos. Un método es un conjunto de instrucciones que realizan una determinada tarea y son similares a las funciones de los lenguajes estructurados.

Del mismo modo que hay variables de instancia y de clase, también hay métodos de instancia y de clase. En el primer caso, un objeto llama a un método para realizar una determinada tarea, en el segundo, el método se llama desde la propia clase¹¹.

4.7 JSP

Según las necesidades planteadas en el proyecto, es requerida su implementación e implantación en la WEB, hecho que obliga a seleccionar una herramienta que permita el desarrollo de software orientado a Internet.

Como herramienta de desarrollo permite la distribución de recursos cliente servidor, utilizando como servidor TOMCAT¹² y clientes en navegadores que soporten dicho servidor.

Debido a su característica WEB Java Server Pages (JSP) es una tecnología que nos permite mezclar HTML estático con HTML generado dinámicamente. Muchas páginas Web que están construidas con programas CGI son casi estáticas, con la parte dinámica limitada a muy pocas localizaciones. Pero muchas variaciones CGI, incluyendo los SERVLETS, hacen que generemos la página completa mediante nuestro programa, incluso aunque la mayoría de ella sea siempre lo mismo. JSP nos permite crear dos partes de forma separada¹³. Aquí tenemos un ejemplo:

¹¹ Introducción. Conceptos básicos de la Programación Orientada a Objetos. [En Línea]. Fecha de Consulta: 28-04-2007. Disponible en : <http://www.sc.ehu.es/sbweb/fisica/cursoJava/fundamentos/clases1/clases.htm>

¹² TOMCAT es una versión de apache group fundation que procesa archivos con extensión JSP, permitiendo su interpretación, compilación y posterior ejecución, por su soporte con Java permite la interacción con JDBC (Objeto para Conectar Bases de Datos)

¹³ Servlets y JSP. Conceptos de Marty Hall. [En Línea]. Fecha de Consulta: 12-01-2008. Disponible en : http://www.programacion.net/java/tutorial/servlets_jsp

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD><TITLE>I C B F</TITLE></HEAD>
<BODY>
<H1>Sistema de Prueba</H1>
<SMALL>Bienbenido,
<!--Ésto es un comentario en JSP -->
<% out.println(Utils.getUserNameFromCookie(request)); %>
Haga click, para llegar al archivo
<A HREF="Account-Settings.html">AQUI!!!</A></SMALL>
<P>
Aqui se puede escribir HTML normalmente (ICBF).
</BODY></HTML>

```

4.8 BASES DE DATOS

El concepto de base de datos manejado en éste apartado está orientado a su definición universal, se explicarán algunos términos para la aplicación del proyecto en el apartado de POSTGRES.

Básicamente es definida como una colección de archivos interrelacionados, son creados con un DBMS¹⁴. El contenido de una base de datos engloba a la información concerniente (almacenadas en archivos) de una organización, de tal manera que los datos estén disponibles para los usuarios, una finalidad de la base de datos es eliminar la redundancia o al menos minimizarla. Los tres componentes principales de un sistema de base de datos son el hardware, el software DBMS y los datos a manejar, así como el personal encargado del manejo del sistema.

Los sistemas de base de datos se diseñan para manejar grandes cantidades de información, la manipulación de los datos involucra tanto la definición de estructuras para el almacenamiento de la información como la provisión de mecanismos para la manipulación de la información, además un sistema de base de datos debe de tener implementados mecanismos de seguridad que garanticen la integridad de la información, a pesar de caídas del sistema o intentos de accesos no autorizados¹⁵.

¹⁴ Muy comúnmente conocido como el motor, el cual es el programa o conjunto de programas que se usan para almacenar y acceder a la información en forma rápida y eficaz.

¹⁵ Bases de Datos. Introducción a los Conceptos de Bases de Datos. [En Línea]. Fecha de Consulta: 10.11.2007. Disponible en : <http://academicos.cualtos.udg.mx/Informatica/Ceneval2003/Bases%20de%20Datos1.htm>

Se traza como objetivo principal en un sistema de base de datos el hecho de proporcionar a los usuarios finales una visión abstracta de los datos, esto se logra escondiendo ciertos detalles por ejemplo su almacenamiento.

En las bases de datos existen conceptos para hacer relación a su estructura interna, por ejemplo entidad, entidades y conjunto de entidades, concepto de relación, conjunto de relaciones, tipos de relaciones, limitantes de mapeo, qué es una llave primaria, diagrama Entidad Relación, reducción de diagramas E-R a tablas. Agregación. Estos términos se pueden ampliar ingresando a la página que se hace referencia al pie de página¹⁶.

Otro concepto es el ciclo de vida del software, ciclo de vida lineal, ciclo de vida en cascada, ciclo de vida en V, ciclo de vida tipo Sashimi, ciclo de vida iterativo, ciclo de vida por prototipos, ciclo de vida evolutivo, ciclo de vida en espiral.

También se entiende el modelo relacional como estructura de los datos, integridad de los datos, manipulación de los datos. Tipos de relaciones, las cuales se basan en claves primarias y dependencias funcionales, y para lograr su correcto funcionamiento es necesario plantear una correcta normalización¹⁷.

4.9 JDBC¹⁸

Es conocido como el puente entre Java y las bases de datos, su función primordial es lograr la conexión mediante DRIVERS¹⁹ y lograr el desarrollo de software útil y con almacenes de datos grandes.

JDBC es usado en el proyecto para lograr la conexión con POSTGRES²⁰.

4.10 DREAMWEAVER

Es la aplicación de este tipo más usada en el sector de diseño y programación web. Posee, como toda la línea Macromedia/Adobe, excelentes funcionalidades e integración con otras herramientas.

¹⁶ Bases de Datos. Introducción a los Conceptos de Bases de Datos. [En Línea]. Fecha de Consulta: 10.11.2007. Disponible en :

<http://academicos.cualtos.udg.mx/Informatica/Ceneval2003/Bases%20de%20Datos1.htm>

¹⁷ Proceso de formas normales para llegar de un conjunto de datos a la estructura deseada, son cinco formas normales pero en su aplicación se puede llegar a la cuarta sin generar problemas de diseño.

¹⁸ Java Data Base Conectivy

¹⁹ Archivos generados por los desarrolladores de los motores de bases de datos para la conexión con el lenguaje Java

²⁰ Motor de bases de datos seleccionado para el desarrollo del proyecto

La aplicación permite crear sitios de forma totalmente gráfica, y dispone de funciones para acceder al código HTML generado. Permite la conexión a un servidor, a base de datos, soporte para programación en ASP, PHP, Javascript, cliente FTP integrado, etc. Principal competencia de Microsoft FrontPage.

Debido a sus características que facilitan el desarrollo estético de la aplicación para el Fondo de Empleados del Instituto Colombiano de Bienestar Familiar sede Manizales fue seleccionado como IDE²¹ de desarrollo para JSP

4.11 POSTGRES

Los sistemas de mantenimiento de Bases de Datos relacionales tradicionales (DBMS,s) soportan un modelo de datos que consisten en una colección de relaciones con nombre, que contienen atributos de un tipo específico. En los sistemas comerciales actuales, los tipos posibles incluyen numéricos de punto flotante, enteros, cadenas de caracteres, cantidades monetarias y fechas. Está generalmente reconocido que este modelo será inadecuado para las aplicaciones futuras de procesamiento de datos. El modelo relacional sustituye modelos previos en parte por su "simplicidad espartana". Sin embargo, como se ha mencionado, esta simplicidad también hace muy difícil la implementación de ciertas aplicaciones. POSTGRES ofrece una potencia adicional sustancial al incorporar los siguientes cuatro conceptos adicionales básicos en una vía en la que los usuarios pueden extender fácilmente el sistema.²²

4.12 SQL

(Structured Query Language) Lenguaje utilizado para base de datos desarrollado entre 1974 y 1975 en IBM Research. Por esos años se llamaba SEQUEL (Structured English QUERY Language) y servía como interfaz para un sistema experimental de base de datos llamado SYSTEM R.

SQL fue introducido por Oracle por primera vez en una base de datos comercial en el año 1979.

Se convirtió en un estándar ANSI e ISO dando lugar al SQL-86 o SQL1, que se fue actualizando a SQL-92 o SQL2, luego a SQL3 y actualmente (2004) a SQL4.

²¹ Entorno de desarrollo gráfico, normalmente son de gran ayuda para la creación de aplicaciones porque su función principal es la generación de código, para el caso del proyecto se usa básicamente en la generación de códigos HTML y JAVASCRIPT

²² Manual del usuario de PostgreSQL. Introducción. [En Línea]. Fecha de Consulta: 12-01-2008. Disponible en : <http://es.tldp.org/Postgresql-es/web/navegable/user/intro.html#AEN34>

El lenguaje se utiliza para definir, consultar y actualizar la base de datos, y es el más popular de su estilo.

La estructura básica para consultas en SQL es select-from-where. Conceptualmente, SQL es un lenguaje de definición de datos (LDD), un lenguaje de definiciones de vistas (LDV) y un lenguaje de manipulación de datos (LMD), que posee también capacidad para especificar restricciones y evolución de esquemas²³.

4.13 TOMCAT

TOMCAT empezó siendo una implementación de la especificación de los servlets comenzada por James Duncan Davidson, que trabajaba como arquitecto de software en Sun Microsystems y que posteriormente ayudó a hacer el proyecto open source y en su donación a la Apache Software Foundation.

TOMCAT es un servidor WEB con soporte de SERVLETS y JSPS. Incluye el compilador JASPER, que compila JSPS convirtiéndolas en SERVLETS. El motor de SERVLETS de TOMCAT a menudo se presenta en combinación con el servidor web Apache²⁴.

TOMCAT puede funcionar como servidor web por sí mismo. En sus inicios existió la percepción de que el uso de TOMCAT de forma autónoma era sólo recomendable para entornos de desarrollo y entornos con requisitos mínimos de velocidad y gestión de transacciones.

Hoy en día ya no existe esa percepción y TOMCAT es usado como servidor web autónomo en entornos con alto nivel de tráfico y alta disponibilidad.

TOMCAT es mantenido y desarrollado por miembros de la Apache Software Foundation y voluntarios independientes. Los usuarios disponen de libre acceso a su código fuente y a su forma binaria en los términos establecidos en la Apache Software Licence. Las primeras distribuciones de TOMCAT fueron las versiones 3.0.x. Las versiones más recientes son las 6.x, que implementan las especificaciones de SERVLET 2.4 y de JSP 2.0. A partir de la versión 4.0, JAKARTA TOMCAT utiliza el contenedor de SERVLETS Catalina²⁵.

²³ Diccionario Informático. SQL. [En Línea]. Fecha de Consulta: 12-12-2007. Disponible en : <http://www.alegsa.com.ar/Dic/sql.php>

²⁴ Es una copia textual de la definición global de TOMCAT, básicamente vislumbra los conceptos internos de estructura, debe anotarse que es una herramienta de software libre.

²⁵ Tomcat. Definiciones. [En Línea]. Fecha de Consulta: 14-10-2007. Disponible en : http://es.wikipedia.org/wiki/Apache_Tomcat

4.14 RATIONAL MODELER

Visión general del producto: Los usuarios de las herramientas de modelado de software de Rational necesitan herramientas personalizables para especificar todos los aspectos de la arquitectura de una aplicación y para gestionar la complejidad de las aplicaciones. Rational Software Modeler soporta el modelado visual con Unified Modeling Language (UML) para documentar distintas vistas de un sistema. Puede utilizar Rational Software Modeler para diseñar visualmente una aplicación de software mediante modelos UML. A través del ciclo de vida de desarrollo, el equipo completa las actividades de recopilación de requisitos, análisis de casos de uso y análisis de dominio seguidas por el diseño detallado de la arquitectura en las fases de desarrollo recomendadas de Rational Unified Process (RUP).

Modelar en un entorno de equipo: Puede dividir los modelos en un entorno de desarrollo en equipo para evitar que distintos usuarios trabajen en la misma parte de un modelo al mismo tiempo y para reducir la frecuencia y la complejidad de las fusiones de archivo entre miembros de equipo. También puede gestionar cambios en archivos de modelado mediante la gestión de configuraciones para controlar y compartir los archivos y comparando o fusionando archivos de forma efectiva cuando sea necesario.

Capturar los requisitos de la aplicación en un modelo de caso de uso: Un proyecto de desarrollo se empieza identificando los usuarios del mismo, los objetivos que el usuario quiere conseguir o un problema determinado a resolver y después descubriendo la funcionalidad y el entorno del sistema que conseguirá estos objetivos o resolverá el problema. Hay que recopilar información sobre los requisitos de funcionamiento y documentarlos, así como identificar qué requisitos están asociados a funciones, tareas o comportamientos específicos del sistema. Cuando haya identificado los requisitos, puede crear un modelo de caso de uso para describir las funciones en términos de utilización del sistema.

Durante la recopilación de requisitos, éstos se estructuran y detallan. Los requisitos pueden tener distintos orígenes, pero los documentos de requisito y las entrevistas de usuario son comunes. Puede gestionar la recopilación y la documentación de requisitos mediante Rational RequisitePro.

Diseño de software con Rational Software Modeler: La forma más eficiente de construir sistemas y más satisfactoria de desplegarlos se da cuando la especificación de la arquitectura es clara y todos los participantes se ponen de acuerdo sobre el diseño en una fase temprana del ciclo de vida del proyecto.

5. METODOLOGÍA

5.1 TIPO DE TRABAJO

El desarrollo completo y satisfactorio de sistemas de información depende de las metodologías utilizadas, de la interacción de los usuarios responsables del desarrollo y de las herramientas utilizadas para el proceso completo de un proyecto de tipo informático como lo es el Sistema de Información Integrado para el Fondo de Empleados del Instituto Colombiano de Bienestar Familiar.

Con el desarrollo de este Sistema de Información realizamos una gestión administrativa del Fondo de Empleados, buscando que las operaciones administrativas se ejecuten con fluidez y eficacia.

5.2 PROCEDIMIENTO

Para la elaboración de este sistema de información se realizaron las siguientes fases de desarrollo:

5.2.1 FASE 1: ELABORACIÓN DEL ANÁLISIS Y DISEÑO. En esta fase es utilizado el Lenguaje de Modelado Unificado UML para poder definir todo el diseño tanto de la aplicación como de la Base de datos. Se realizaron las siguientes actividades:

- Actividad 1.1: Diagrama de clases.
- Actividad 1.2: Diagrama de casos de usos.
- Actividad 1.3: Diagrama de componentes.
- Actividad 1.4: Diagrama de despliegue.
- Actividad 1.5: Diagrama de estados.
- Actividad 1.6: Diagrama de actividad.
- Actividad 1.7: Diagrama de secuencia.
- Actividad 1.8: Diagrama de colaboración.

5.2.2 FASE 2: INSTALACIÓN DE LOS SERVICIOS. Se buscaron y se instalaron los servicios necesarios tanto para el desarrollador como para el usuario final de la aplicación.

- ACTIVIDAD 2.1: Se determinó cuales son las mejores aplicaciones y cual nos brindaba un mejor desempeño para este tipo de aplicaciones.
- ACTIVIDAD 2.2: Se instalaron y configuraron cada uno de los servicios que se escogieron para el desarrollo.

Los servicios y aplicaciones instaladas son:

Java

Tomcat

Postgres

Los drivers de conexión

- ACTIVIDAD 3.3: Se integraron todos los servicios de manera que se permitió la unificación del control de todos.

5.2.3 FASE 3: ANÁLISIS Y DISEÑO DE LA BASE DE DATOS.

- ACTIVIDAD 3.1: Se realizó el análisis de la BD, y la normalización de la Base de Datos.
- ACTIVIDAD 3.2: Se crearon todos los elementos de la Base de Datos: tablas, vistas, consultas, entre otros.
- ACTIVIDAD 3.3: Se implementaron los join, triggers y vistas necesarias para el buen funcionamiento de la Base de Datos con los módulos.
- ACTIVIDAD 3.4: Se realizaron de pruebas de funcionamiento para la BD.

5.2.4 FASE 4: DESARROLLO DEL MÓDULO DE GESTIÓN DE ASOCIADOS.

- ACTIVIDAD 4.1: Se codificaron los strings de conexión del módulo.
- ACTIVIDAD 4.2: Se codificaron los servlets: insertar, eliminar, consultar para el módulo de Gestión de Asociados.
- ACTIVIDAD 4.3: Se crearon los JSP para el módulo de Gestión de Asociados.

5.2.5 FASE 5: DESARROLLO DEL MÓDULO DE GESTIÓN DE SERVICIOS.

- ACTIVIDAD 5.1: Se codificaron los strings de conexión del módulo.

- ACTIVIDAD 5.2: Se codificaron los servlets: insertar, eliminar, consultar para el módulo de Gestión de servicios.
- ACTIVIDAD 5.3: Se crearon los JSP para el módulo de Gestión de servicios.

5.2.6 FASE 6: DESARROLLO DEL MÓDULO DE GESTIÓN DE APORTES Y/O AHORROS.

- ACTIVIDAD 6.1: Se codificaron los strings de conexión del módulo.
- ACTIVIDAD 6.2: Se codificaron los servlets: insertar, eliminar, consultar para el módulo de Gestión de aportes y/o ahorros.
- ACTIVIDAD 6.3: Se crearon los JSP para el módulo de Gestión de aportes y/o ahorros.

5.2.7 FASE 7: DESARROLLO DEL MÓDULO DE CRÉDITO Y CARTERA.

- ACTIVIDAD 7.1: Se codificaron los strings de conexión del módulo.
- ACTIVIDAD 7.2: Se codificaron los servlets: insertar, eliminar, consultar para el módulo de Gestión de Crédito y Cartera.
- ACTIVIDAD 7.3: Se crearon los JSP para el módulo de Gestión de Crédito y Cartera.

5.2.8 FASE 8: DESARROLLO DEL MÓDULO DE SERVICIO AL CLIENTE.

- ACTIVIDAD 8.1: Se codificaron los strings de conexión del módulo.
- ACTIVIDAD 8.2: Se codificaron los servlets: insertar, eliminar, consultar para el módulo de Servicio al Cliente
- ACTIVIDAD 8.3: Se crearon los JSP para el módulo de Servicio al Cliente.

6. RESULTADOS

6.1 DESCRIPCIÓN DE RESULTADOS

El proyecto construyó un software como producto final para el Fondo de Empleados del Instituto Colombiano de Bienestar Familiar regional Caldas, éste se adaptó a los requerimientos planteados con base a las visitas programadas.

Las características generales de la aplicación indican que su uso mejora ostensiblemente los problemas generados al momento de hacer consultas por parte de los asociados ya que si antes los asociados tenían que recurrir a una llamada telefónica o a viajar para poder tener cierta información ahora estos gastos son reducidos en mucha parte, ya que simplemente el asociado recurre al sistema para realizar las consultas correspondientes, debido a que cuentan con una plataforma orientada a la WEB que permite conocer todos los datos que los relacionan a ellos con respecto al fondo como lo son pormenores de su estado en el Fondo ya sea para estados de cuenta en ahorros o en estado de deducciones en créditos., datos de créditos y servicios, entre otros.

Para determinar esta ganancia en cuanto a tiempo y dinero se realizó en un principio una relación de costo tiempo de ciertos factores influyentes en el tema y se encontraron los siguientes resultados:

RELACIÓN COSTO/BENEFICIO				
NOMBRE		1ER AÑO	2DO AÑO	3ER AÑO
TELEFONIA:	180.000,00	185.400,00	190.962,00	196.690,86
CORREO	2.400.000,00	2.472.000,00	2.546.160,00	2.622.544,80
PAPELERIA		-	-	-
1. Formatos de créditos	72.000,00	74.160,00	76.384,80	78.676,34
TIEMPO	1.020.000,00	1.050.600,00	1.082.118,00	1.114.581,54
AFILIACIONES	587.250,00	604.867,50	623.013,53	641.703,93
PRESTAMOS			114.577,20	118.014,52

	108.000,00	111.240,00		
TOTAL	4.367.250,00	4.498.267,50	4.633.215,53	4.772.211,99

NOTA: SUPONIENDO UNA INFLACION DEL
3% CADA AÑO.

RELACIÓN:	
COSTO	485.000,00
BENEFICIO	4.398.150,00
	0,110273638

6.2 DISCUSIÓN DE RESULTADOS

A traves del analisis anterioro se determino que aunque no se realizo ningun tipo de inversion en la aplicaci3n, este puede ayudar a reducir los costos de tiempo y de recursos ya que estamos hablando de la sistematizacion de actividades realizadas al interior del Fondo de Empleados del Instituto Colombiano de Bienestar Familiar – Regional caldas.

7. CONCLUSIONES

El desarrollo del proyecto aportó experiencias que enriquecen la vida profesional de los actores del mismo, la investigación aplicada necesaria para el desarrollo tecnológico es punto de partida para una dinámica profesional, debido a que la naturaleza de las temáticas manejadas por los ingenieros de sistemas y telecomunicaciones son muy cambiantes, es necesario obtener una cultura de actualización y adaptación de técnicas y tecnologías no cursadas durante el transcurso de la carrera.

Como aporte altamente significativo se incluyó un almacén de datos para los activos fijos manejados por el Fondo, pese a la no inclusión como parte inicial de los objetivos específicos, dicho módulo se establece como novedad.

El uso de tecnologías WEB en el desarrollo del proyecto fue un factor clave de éxito para su aceptación y aplicación, debido a que es necesario y ponderante hacer uso de los recursos con los que cuenta el Instituto Colombiano de Bienestar Familiar regional Caldas. El uso de plataformas existentes en las empresas hace de los proyectos de desarrollo efectivos en el uso y explotación de recursos, convirtiéndose en una fortaleza importante de éxito.

El uso de una herramienta de gestión de procesos como es el WorkFlow, llegó a ser de una gran ayuda para todo el proceso de sistematización de solicitudes de crédito, además estamos recurriendo a la utilización de herramientas que permiten no solo realizar la actividad propia de un fondo como son la manipulación de créditos y ahorros si no también interactuar de forma clara y eficiente con el usuario final en este caso el asociado.

8. RECOMENDACIONES

Todo proyecto orientado a la WEB requiere un alto índice en niveles de seguridad, ya que el uso compartido de recursos, tanto para las bases de datos como para el acceso a la información se hace en la red y con el uso de protocolos base (TCP/IP), se recomienda la definición de políticas de seguridad al interior del departamento de sistemas del Instituto Colombiano de Bienestar Familiar regional Caldas.

Teniendo en cuenta que el Instituto Colombiano de Bienestar Familiar regional Caldas hace presencia con oficinas en cinco municipios del Departamento, se hace necesaria la adaptación y implantación del proyecto en un ambiente que permita la consulta desde dichos puntos, por lo tanto se recomienda la posterior instalación del software en un servidor con acceso externo a la intranet, es decir, que los usuarios que requieran solicitar un servicio lo puedan hacer desde cualquier punto en Internet.

Para el correcto uso de la aplicación y con miras de tener éxito en la aplicación de la herramienta se recomienda implementar un plan de capacitación permanente a los asociados, para que éstos en cualquier momento de formar parte del Fondo no tengan inconvenientes en el uso de todos los recursos ofrecidos.

BIBLIOGRAFÍA

BRUEGGE, Bernd y H. Dutoit, Allen. Ingeniería del software orientado a objetos. México. 576 p.

GILL, Harjinder and RAO, Prakash. Data Warehousing. (2000). La Integración de la Información para la Toma de Decisiones. México: Prentice Hall. 382 p.

JACOBSON, Ivar; BOOCH, Grady; RUMBAUGH, James. El proceso unificado de desarrollo de software. España. Addison Wesley. 438 p

LARMAN, Graig. (2000). UML y Patrones: Introducción al Análisis y Diseño Orientado a Objetos. México: Prentice Hall. 507 p.

MEYER, Bertrand. Reusable software. Great Britain. 514 p.

PEREZ ZAVALA, Moises. Publicación en Internet. Naucalpan de Juárez, Edo de México. 1997. Prentice-Hall Hispanoamérica S.A.

PEREZ ZAVALA, Moises. Programación en Web. Naucalpan de Juárez, Edo de México. 1998. Prentice-Hall Hispanoamérica S.A.

RUMBAUGH, James; BLAHA, Michael; PREMERLANI, William; EDDY, Frederick; LORENSEN, William. Modelado y diseño orientado a objetos. España. Prentice Hall. 643 p.

SOMMERVILLE, Ian. Ingeniería de software. España. Addison Wesley. 712 p

WIKERT, Joseph B. Java script a través de ejemplos. Naucalpan de Juárez, Edo de México. 1997. Prentice-Hall Hispanoamérica S.A.

REFERENCIAS

Bases de Datos. Introducción a los Conceptos de Bases de Datos. [En Línea]. Fecha de Consulta: 10.11.2007. Disponible en: <http://academicos.cualtos.udg.mx/Informatica/Ceneval2003/Bases%20de%20Datos1.htm>

Diccionario Informático. SQL. [En Línea]. Fecha de Consulta: 12-12-2007. Disponible en: <http://www.alegsa.com.ar/Dic/sql.php>

Introducción. Conceptos básicos de la Programación Orientada a Objetos. [En Línea]. Fecha de Consulta: 28-04-2007. Disponible en: <http://www.sc.ehu.es/sbweb/fisica/cursoJava/fundamentos/clases1/clases.htm>

Manual del usuario de PostgreSQL. Introducción. [En Línea]. Fecha de Consulta: 12-01-2008. Disponible en: <http://es.tldp.org/Postgresql-es/web/navegable/user/intro.html#AEN34>

Revista Novatica–Asociación de Técnicos de Informática. Introducción a perfiles UML. [En Línea]. Fecha de Consulta: 30-01-2007. Disponible en: <http://www.lcc.uma.es/~av/Publicaciones>

Servlets y JSP. Conceptos de Marty Hall. [En Línea]. Fecha de Consulta: 12-01-2008. Disponible en: http://www.programacion.net/java/tutorial/servlets_jsp

Tomcat. Definiciones. [En Línea]. Fecha de Consulta: 14-10-2007. Disponible en: http://es.wikipedia.org/wiki/Apache_Tomcat

ANEXOS

ANEXO A MODELO DE REQUISITOS

El modelo de caso de uso

El modelo de caso de uso proporciona información detallada acerca de los comportamientos del sistema o de la aplicación que se está desarrollando respecto al proyecto de informatización del Fondo de Empleados del Instituto Colombiano de Bienestar Familiar. Contiene diagramas de caso de uso y diagramas de actividad que describen cómo interactúan los usuarios con el sistema.

El modelo de caso de uso identifica los requisitos del sistema en términos de la funcionalidad que debe existir para conseguir los objetivos establecidos por el usuario o para resolver un problema identificado por éste. Los casos de uso describen los comportamientos principales identificados en los requisitos y describen el valor que los resultados dan a los usuarios; no describen cómo funciona internamente el sistema. Los actores son los usuarios del sistema y representan los diferentes roles que la gente y otros sistemas representan cuando interactúan con el sistema.

Los diagramas de casos de uso representan las relaciones entre los casos de uso y los actores y los diagramas de actividad describen el flujo de objetos y el control en cada comportamiento identificado.

Caso de uso (Contexto: Actor Asociado)

En dicho contexto se vislumbran los requisitos iniciales desde el punto de vista del Asociado, el cual se convierte en ésta instancia en un actor del sistema requerido, es necesario anotar que dichos casos de uso se convierten en casos de uso en el lenguaje del modelador.

En dicho diagrama se contextualizan las necesidades de consulta de estados de cuenta, consultas de movimiento, consulta de deducciones respecto a los créditos preestablecidos y demás acciones que el asociado como actor del software requiere.

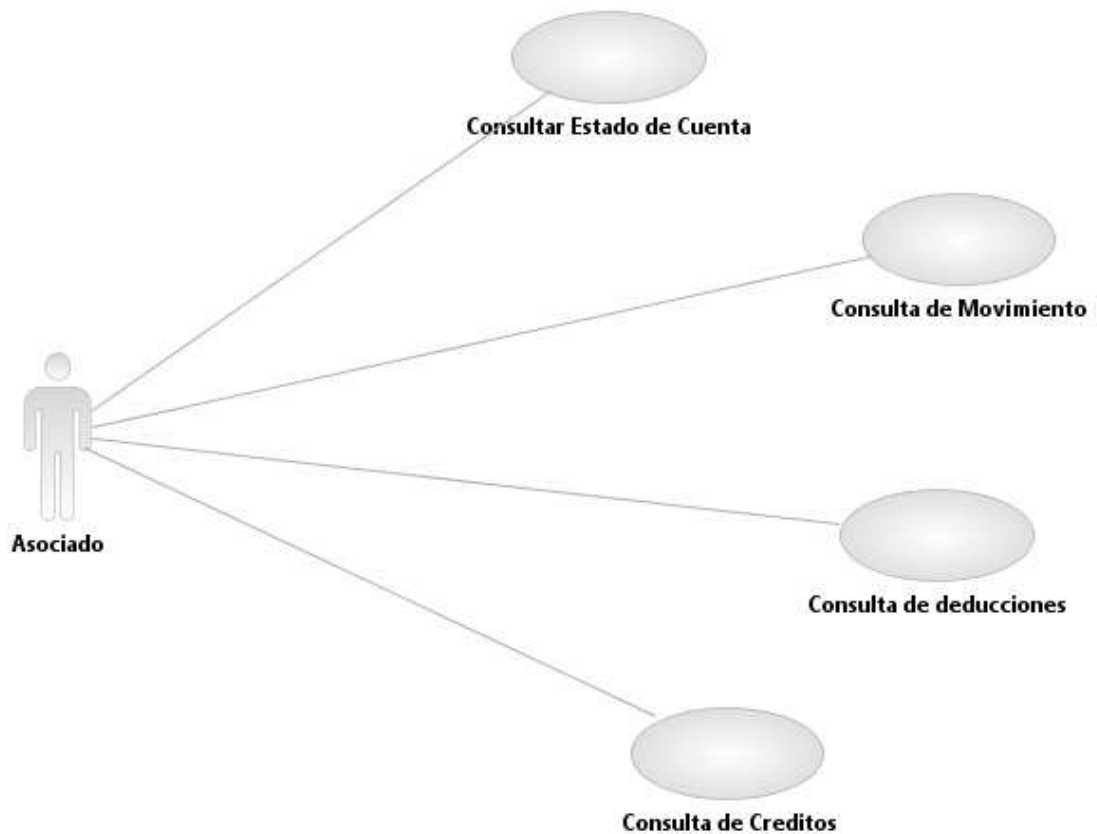


Figura 1: Casos de Uso (Actor Asociado)

Caso de uso (Contexto: Actor Administrador)

Según el punto de vista del responsable de la información y el procesamiento de la información se genera un actor llamado Administrador, el cual tiene como objetivo principal la mantención de la información en el orden de procesamiento y ofrecimiento para su consulta.

El administrador tiene la responsabilidad de mantener una base de datos simple de activos fijos, el cual le permitirá tener un inventario de los bienes del Fondo.

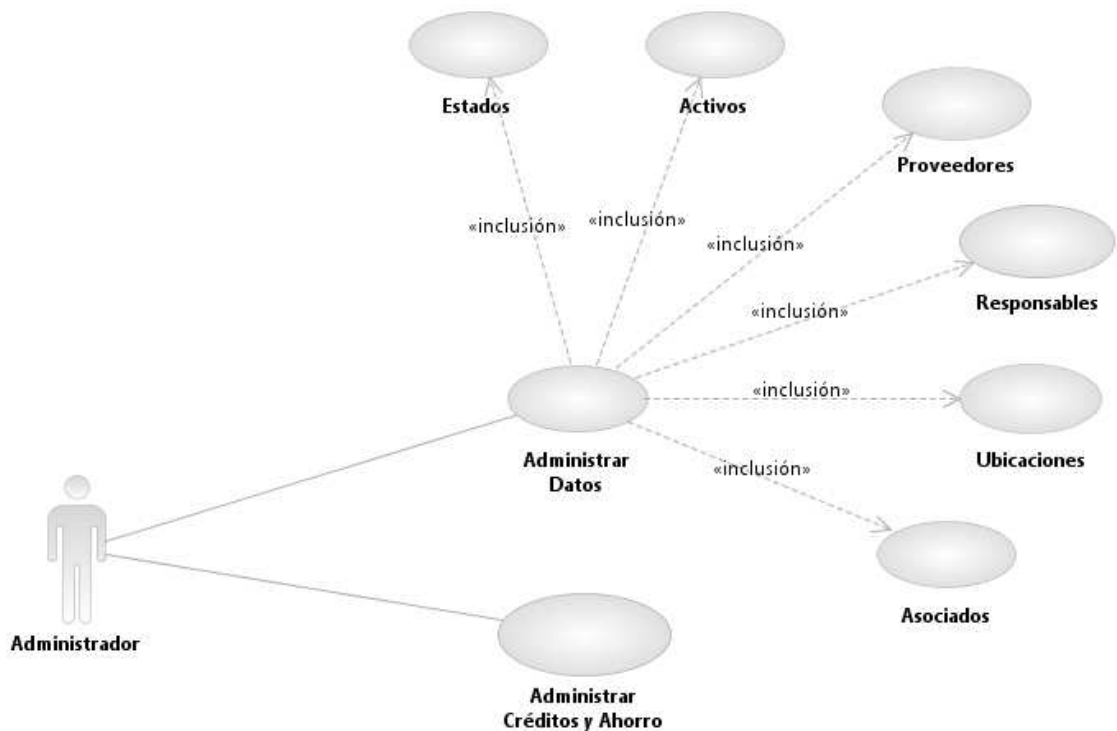


Figura 2 Caso de uso (Contexto: Actor Administrador)

Una de las características del Administrador tal y como se observa en el diagrama de casos de uso, son los datos a mantener en cuanto a inserción, eliminación, modificación y consulta ya que estos corresponden a: estados, activos, proveedores, responsables, ubicaciones y asociados.

Según los requerimientos del sistema es necesario crear perfiles de usuario para diferenciar el ingreso al sistema. Dichos perfiles de deben estudiar más adelante en el modelo de análisis para establecer si es necesario ampliar los perfiles administrativos en el orden de procesamiento y mantenimiento, es decir, separar las operaciones administrativas de las tablas con la gestión de procesamiento para nómina.

Diagrama de Actividad (Administrar)

Las acciones más relevantes y enmarcadas como requerimientos para la administración del sistema están relacionadas con la necesidad de solventar la cantidad de solicitudes de consulta, tanto para estados de cuenta, como para extractos y saldos de los créditos.

En el diagrama de actividad se expresan los procedimientos generales para ser soportados por el software a realizar, indicando un simple ciclo de procedimientos y procesos.

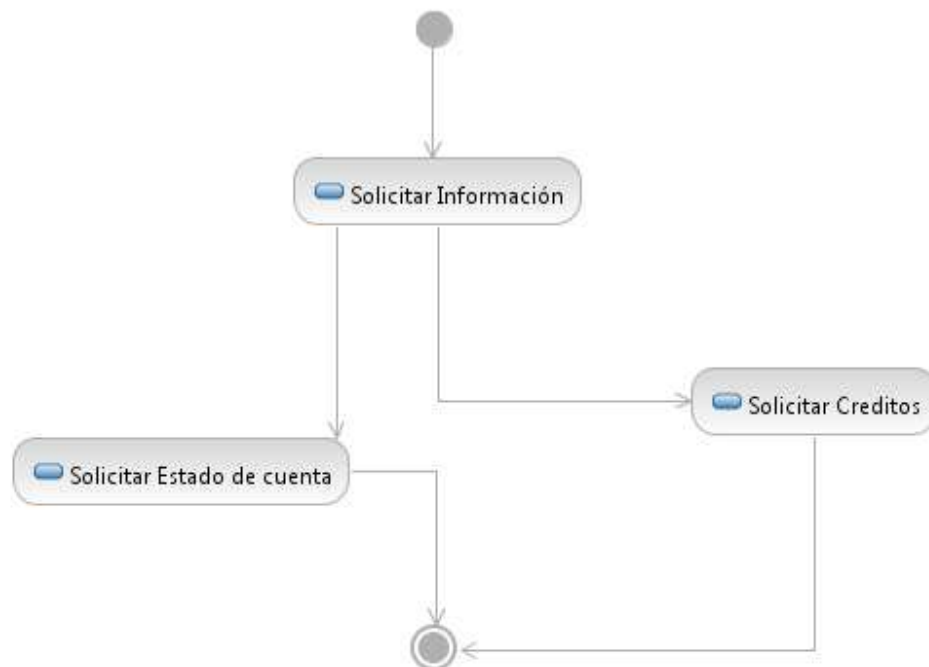


Figura 3 Diagrama de Actividad (Administrar)

Diagrama de Actividad (Procesar consultas)

El diagrama de actividad que esboza, los requerimientos en el orden de consultas son tratados tal y como se hacen en forma manual (con apoyo de hojas de cálculo).

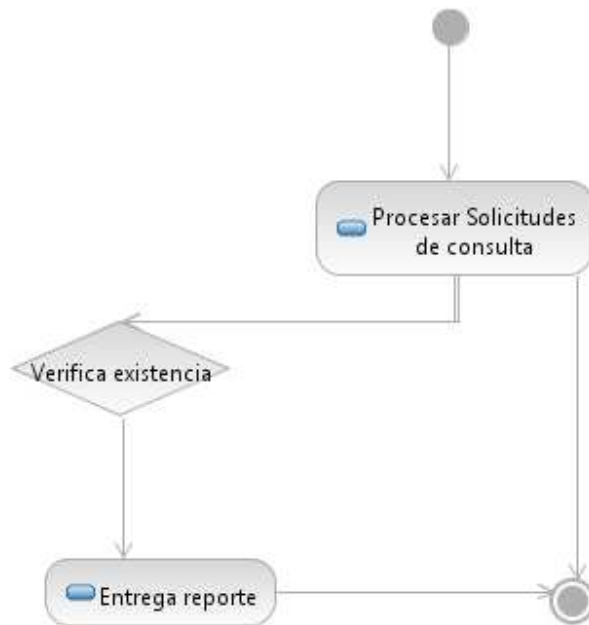


Figura 4 Diagrama de Actividad (Procesar consultas)

Diagrama de Actividad (Procesar créditos)



Figura 5 Diagrama de Actividad (Procesar créditos)

Diagrama de Actividad (Procesar nómina)



Figura 6 Diagrama de Actividad (Procesar nómina)

Anexo: Modelo de Análisis

El modelo de análisis tiene como función primordial la construcción de términos más técnicos para que el ingeniero constructor esboce todas las características técnicas en el lenguaje apropiado²⁶.

En esta fase se inician a construir las clases de Entidad, Control e interfaz, ayudando a la elaboración de las tablas que se usarán en la base de datos y en la etapa de diseño en forma de objetos.

Para cada elemento y después de un proceso de construcción y elaboración se indicarán por medio de diagramas de análisis cada uno de los elementos que conforman la aplicación con base a el modelo de requisitos.

Para el presente proyecto se utilizan las técnicas de desarrollo de software adaptada para las necesidades escasas del Fondo de Empleados del Instituto de Bienestar Familiar sede Manizales.

²⁶ El lenguaje utilizado para los modelos en las fases de construcción es UML 2.0, basados en las ayudas que ofrece la herramienta RATIONAL MODELER 2007, la cual ayuda a validar la correspondencia y concordancia con el modelo de requisitos.

Diagrama de Actividad: Página principal

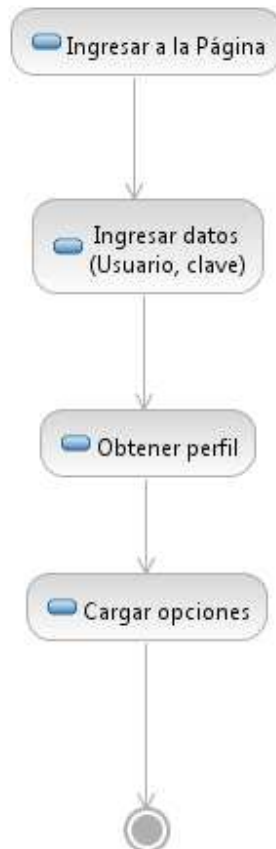


Figura 7 Diagrama de Actividad: Página principal

Diagrama de Clases Análisis: Página principal

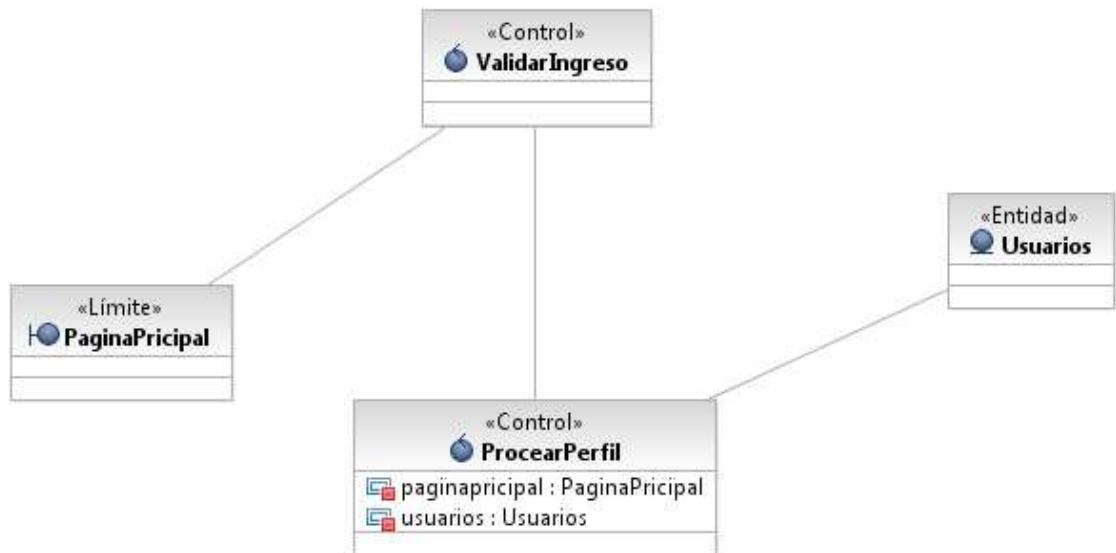


Figura 8 Diagrama de Clases Análisis: Página principal

Diagrama de Actividad: Página administrativa

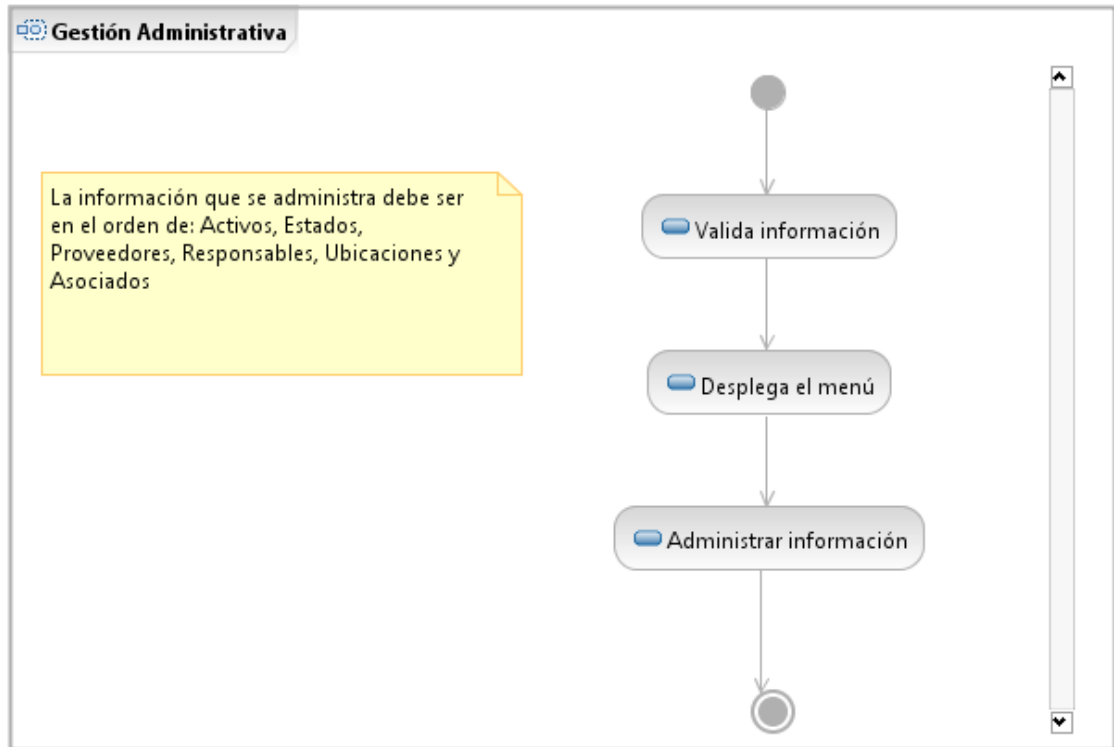


Figura 9 Diagrama de Actividad: Página administrativa

Diagrama de Actividad: Procesos de ahorro



Figura 10 Diagrama de Actividad: Página de Procesos

Diagrama de Clases Análisis: Procesos de ahorro

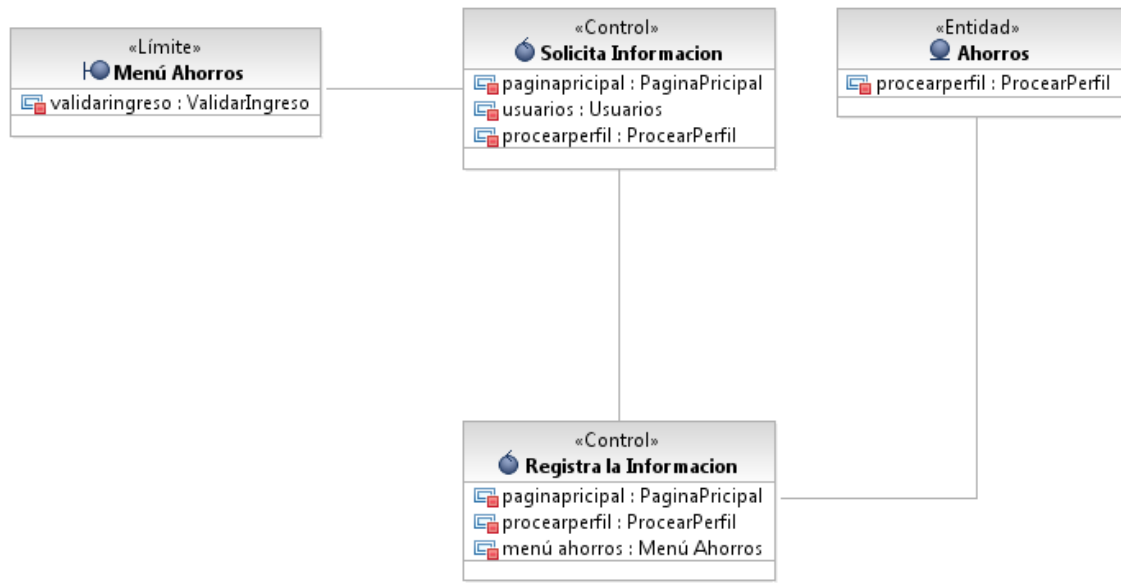


Figura 11 Diagrama de Clases Análisis: Procesos de ahorro

Diagrama de Actividad: Proceso de crédito

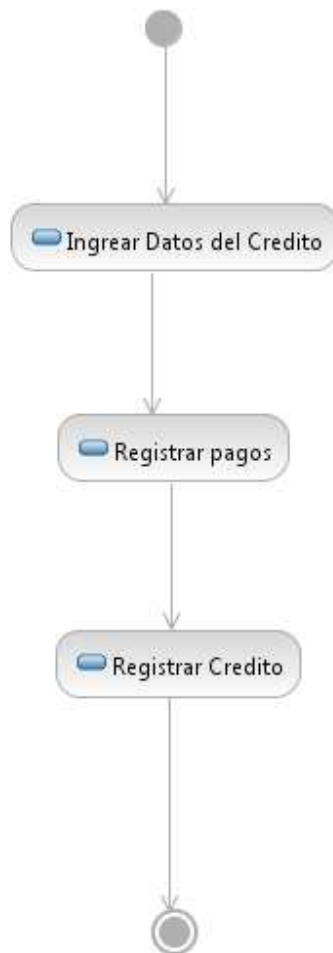


Figura 12 Diagrama de Actividad: Proceso de crédito

Diagrama de Clases Análisis: Proceso de crédito



Figura 13 Diagrama de Clases Análisis: Proceso de crédito

Diagrama de Actividad: Proceso de reporte para nómina (Ahorro)

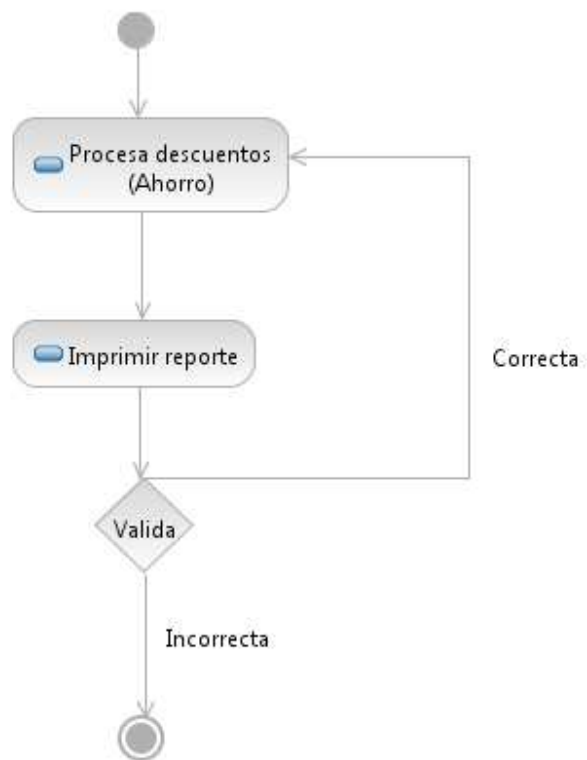


Figura 14 Proceso de reporte para nómina (Ahorro)

Diagrama de Clases Análisis: Proceso de reporte para nómina (Ahorro)

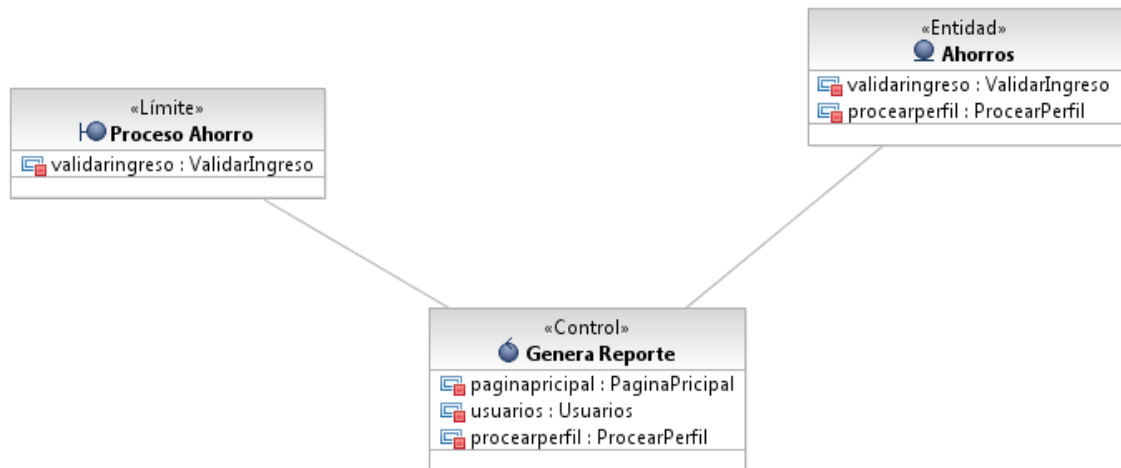


Figura 15 Diagrama de Clases Análisis: Proceso de reporte para nómina (Ahorro)

Diagrama de Actividad: Proceso de reporte para nómina (Crédito)

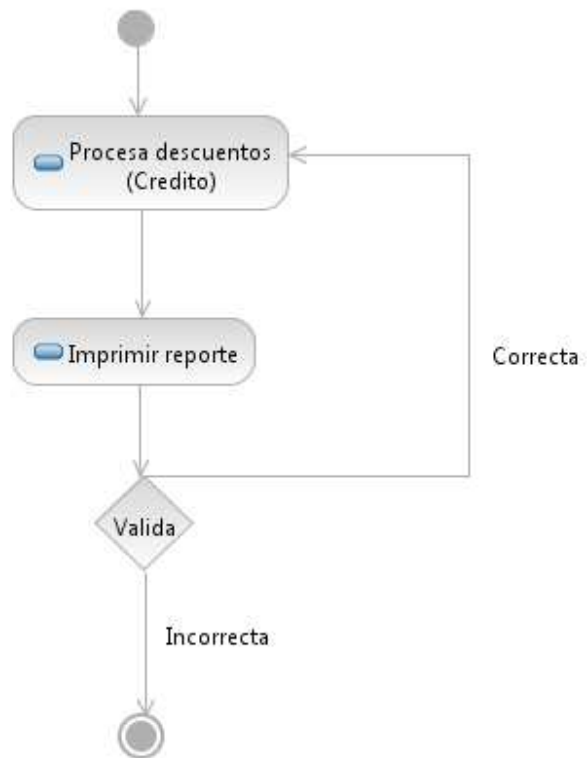


Figura 16 Proceso de reporte para nómina (Crédito)

Diagrama de Clases Análisis: Proceso de reporte para nómina (Crédito)

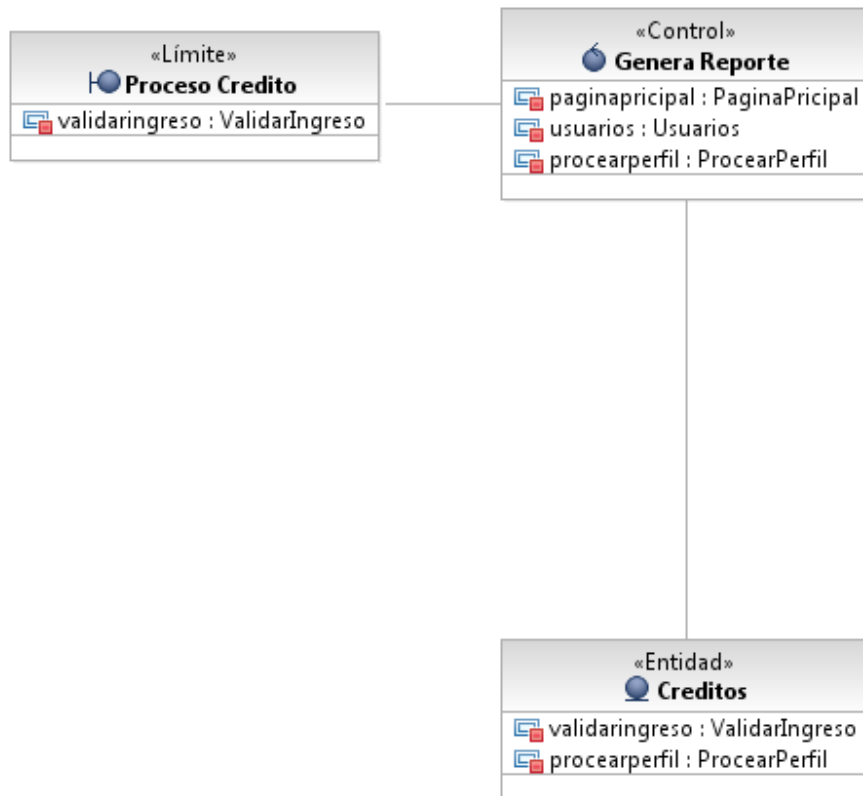


Figura 17 Proceso de reporte para nómina (Crédito)

Diagrama de Clases Núcleo: Activos Fijos

Para efectos de diseño en la base de datos se toman en cuenta las relaciones de multiplicidad del diagrama de clases núcleo como diagrama relacional, es decir, al momento de la construcción de la integridad en el modelo relacional de las bases de datos es necesario hacer referencia a los diagramas de clases núcleo expresados en UML 2.0.

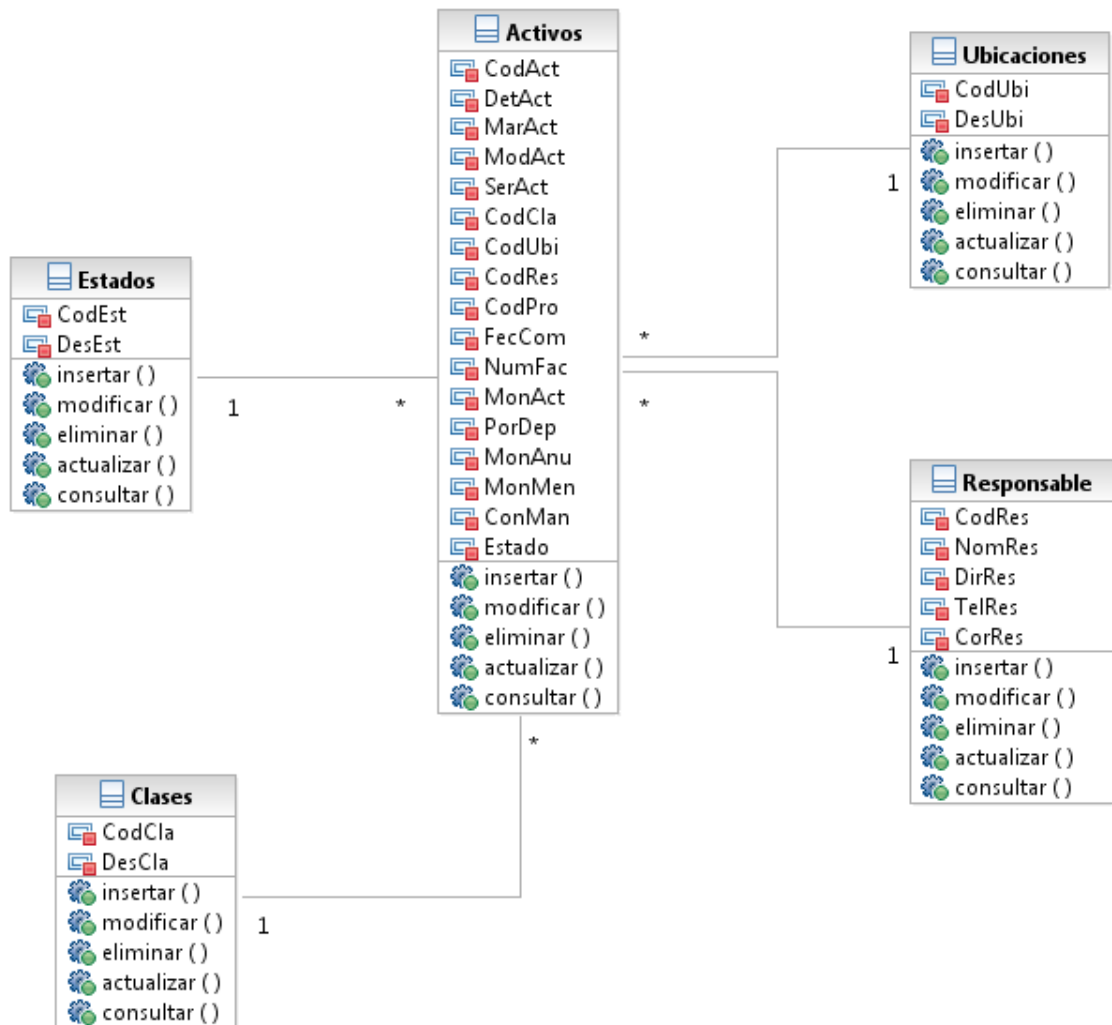


Figura 18 Diagrama de Clases Núcleo: Activos Fijos

Diagrama de Clases Núcleo: Administrativas y Procedimentales

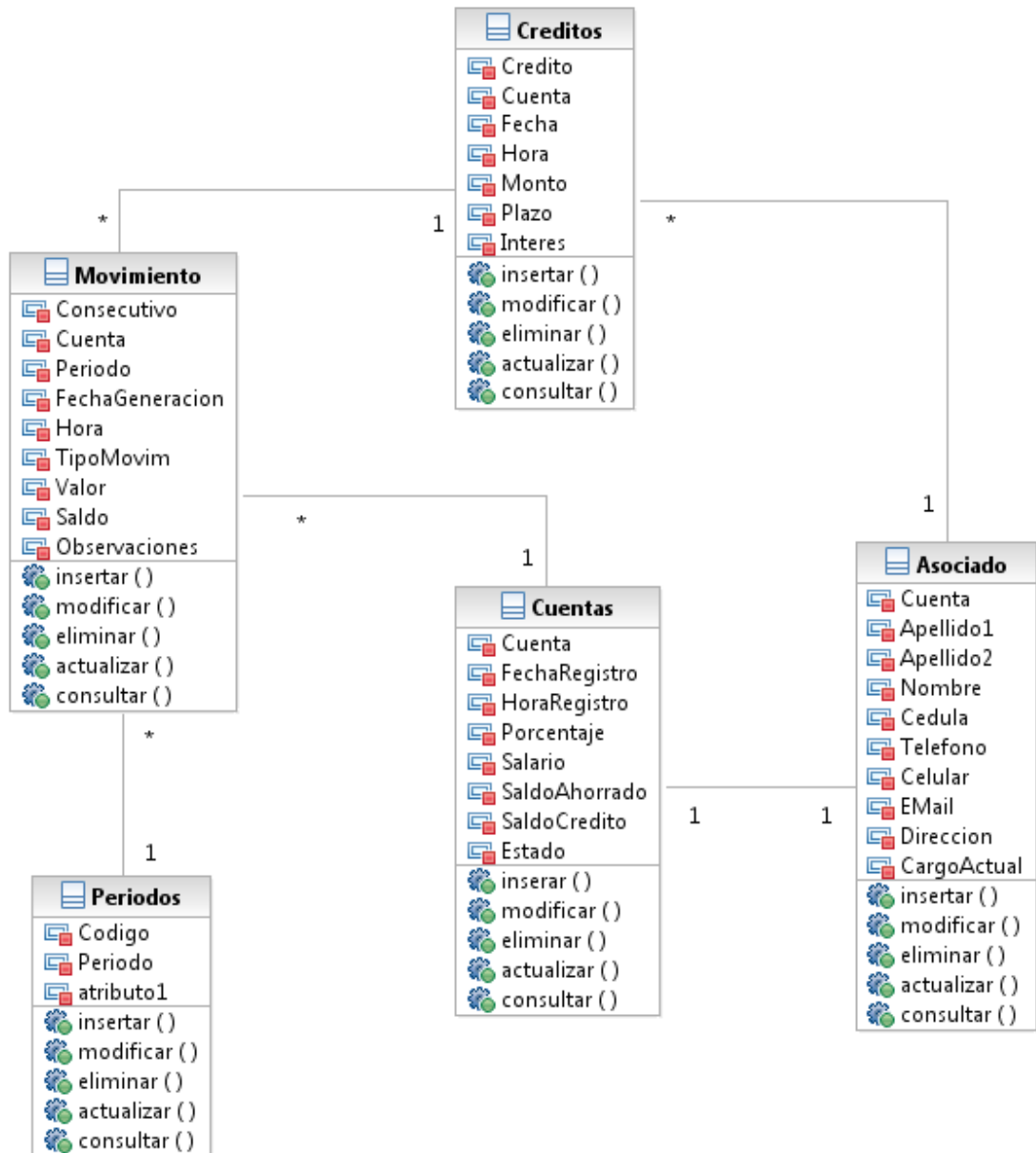


Figura 19 Diagrama de Clases Núcleo: Administrativas y Procedimentales

ANEXO B MODELO DE DISEÑO

Se ha creado el modelo de caso de uso que describe los requisitos de la aplicación y ha creado un modelo de análisis que identifica los requisitos del dominio de la aplicación y que describe la forma en que se estructurará la aplicación. El paso siguiente del proceso de desarrollo consiste en construir sobre el modelo de análisis explorando la arquitectura y las opciones de implementación de la aplicación.

La fase de diseño toma el modelo de alto nivel creado durante el análisis y añade detalles a la implementación actual del sistema. Mediante varios diagramas, incluyendo los diagramas de interfaz, componentes y máquina de estado si aplica, puede crear un modelo que describe cómo se estructura y cómo se implementará la aplicación. Se describen construcciones y tecnologías de programación como las utilizadas para persistencia y registro de información. Se puede seguir refinando el modelo de diseño aplicando los patrones de diseño que capturan estructuras y procesos utilizados frecuentemente o complejos.

Cuando está preparado para pasar a la fase de implementación, el modelo de diseño puede transformarse en código de aplicación

Diagrama de Componentes: Distribución de recursos

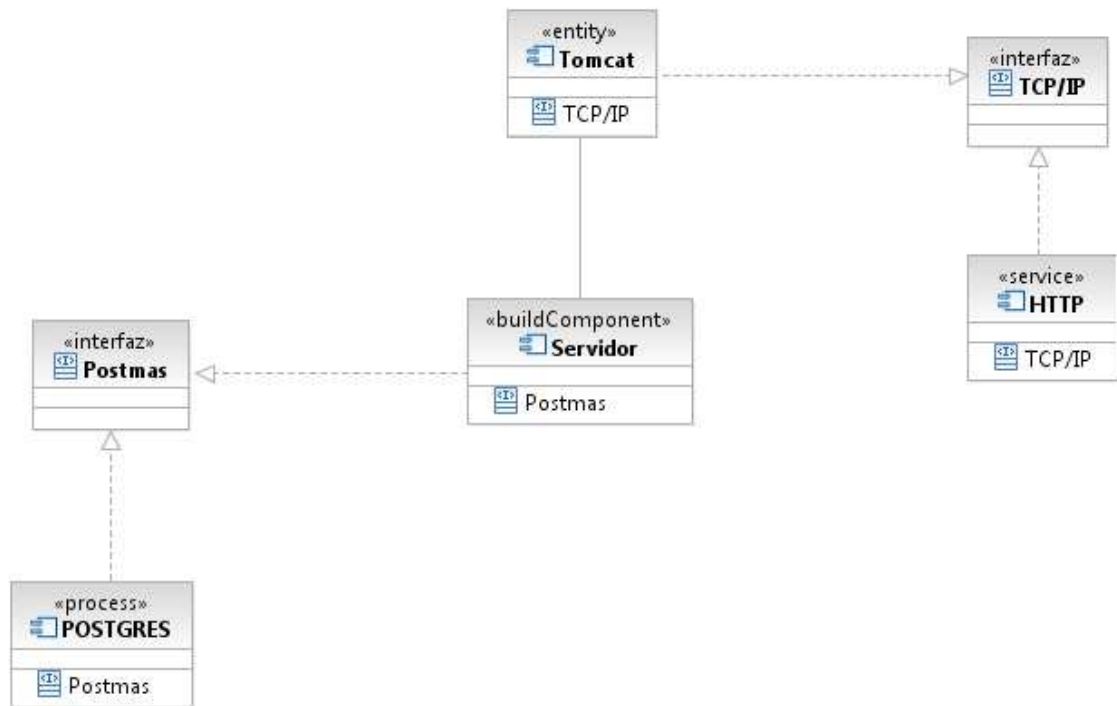


Figura 20 Diagrama de Componentes: Distribución de recursos

Diagrama de Interfaz: Página Principal



Figura 21 Diagrama de Interfaz: Página Principal

Diagrama de Componentes: Página Principal

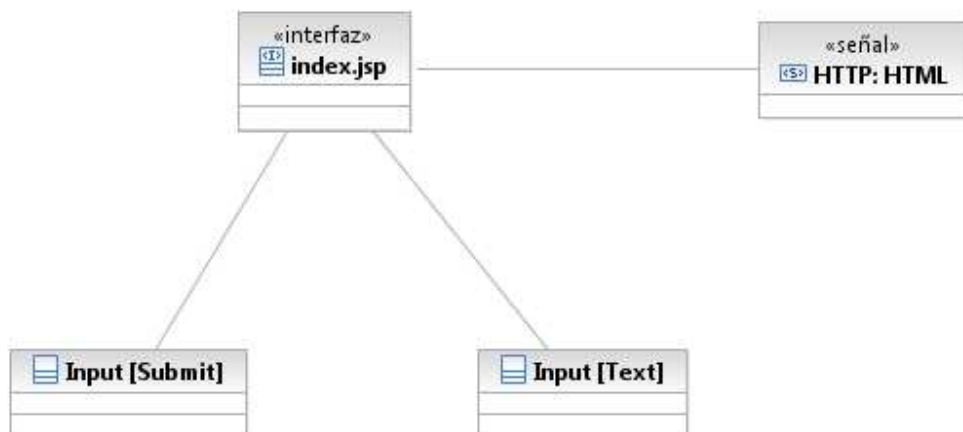


Diagrama de Interfaz: Menú Administrador

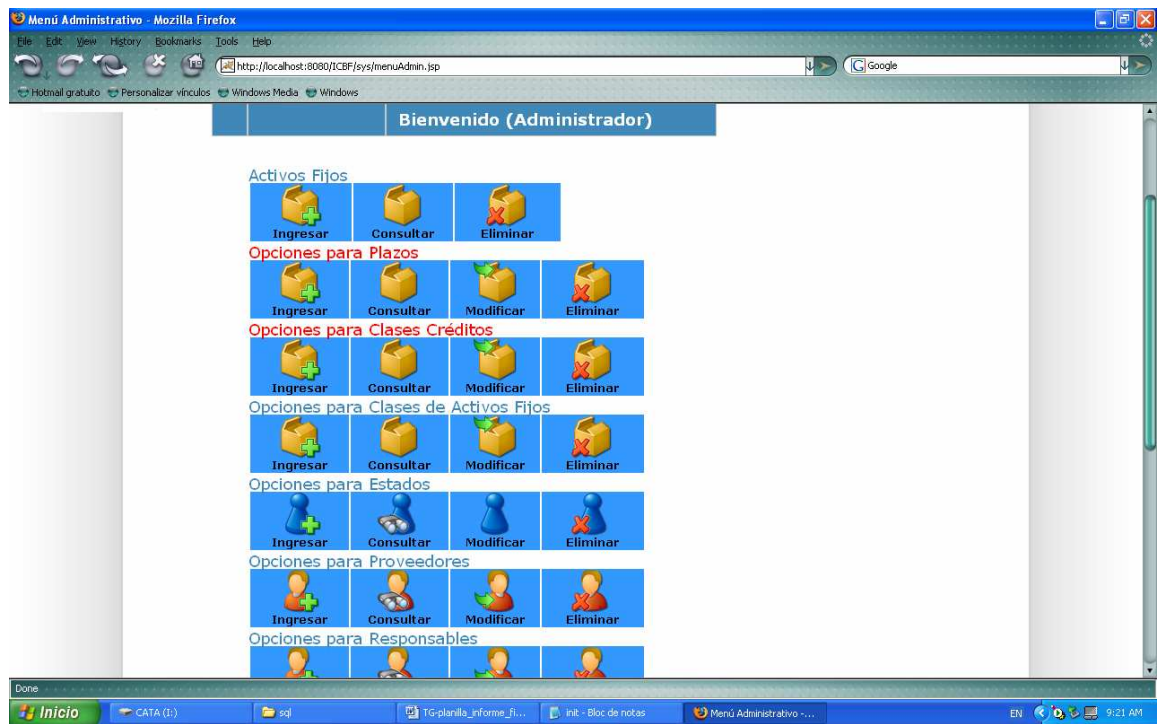


Figura 22 Diagrama de Interfaz: Menú Administrador

Diagrama de Componentes: Menú Administrador

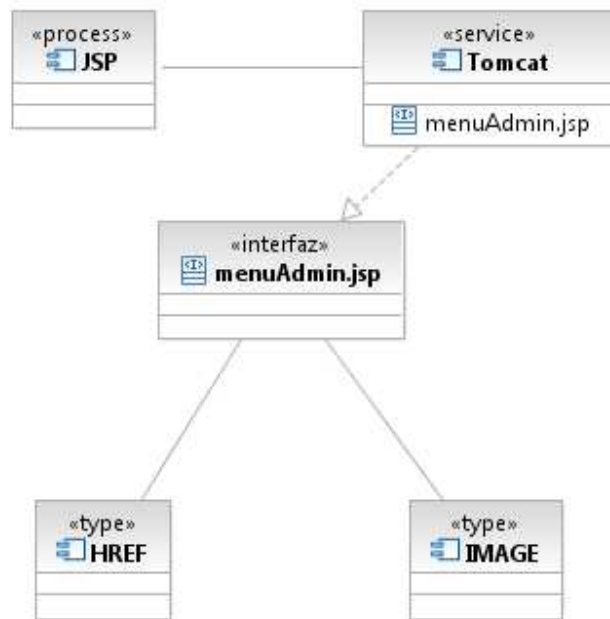


Figura 23 Diagrama de Componentes: Menú Administrador

Diagrama de Interfaz: Ingresar Clase de Activos

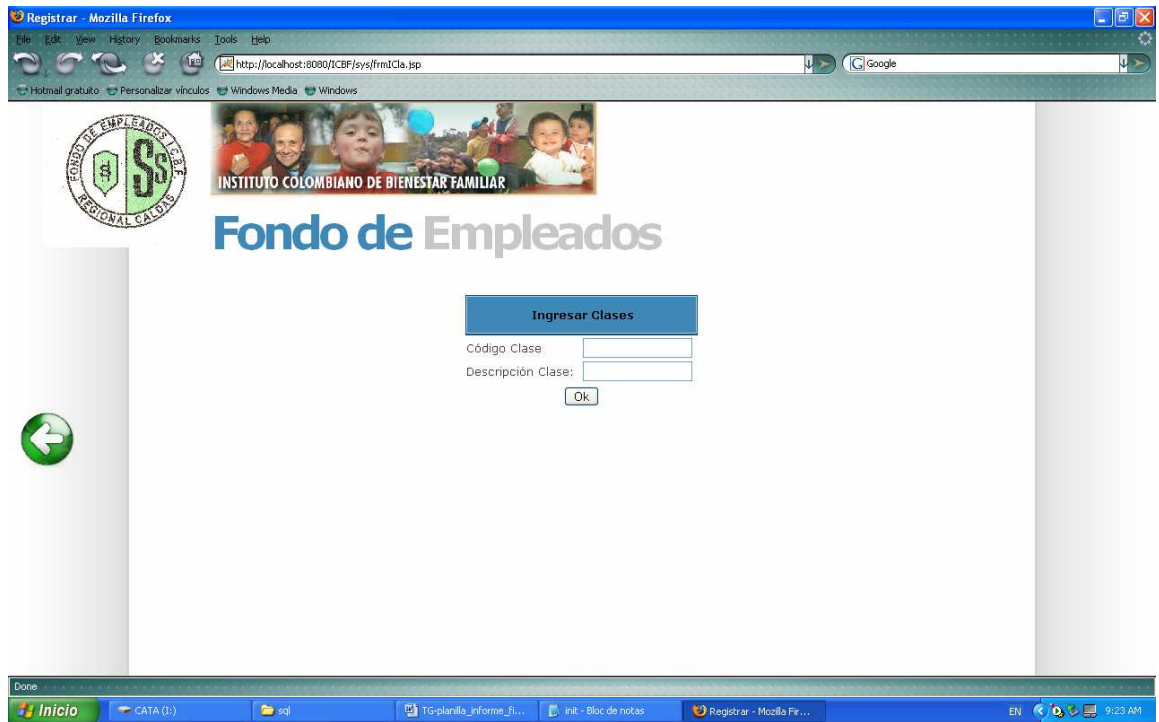


Figura 24 Diagrama de Interfaz: Ingresar Clase de Activos

Diagrama de Componentes: Ingresar Clase de Activos

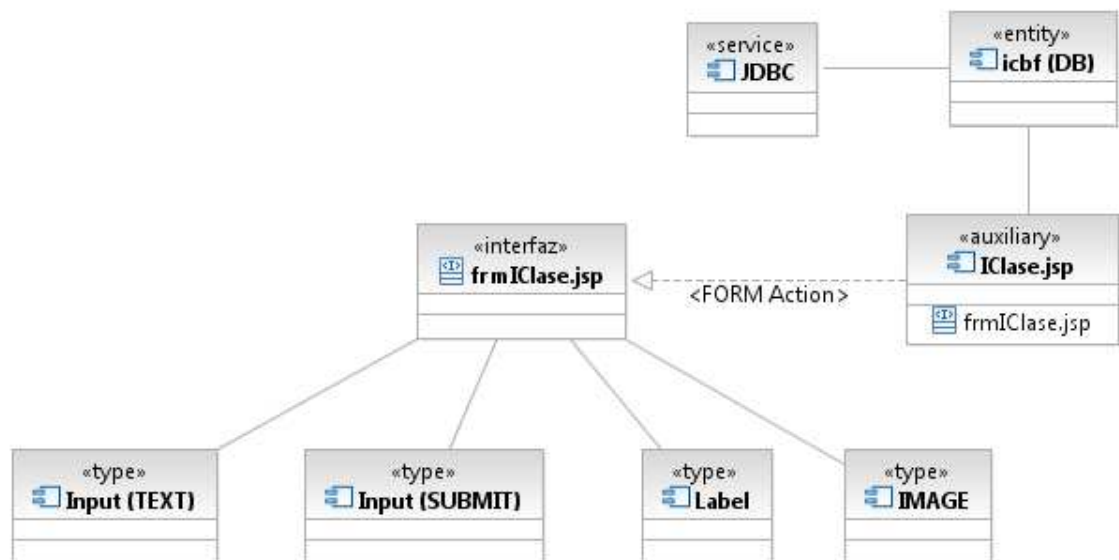


Figura 25 Diagrama de Componentes: Ingresar Clase de Activos

Código: Ingresar Clase de Activos

```
String cCod=request.getParameter("cCod");
String cNom=request.getParameter("cNom");

if ((cCod != null && cCod.equals("") != true) && (cNom != null && cNom.equals("")
!= true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4><center>Problemas en
conexion</center></h4>"+x);}
        String sql="select * from "+"\"tblClases\""+ " where
"+"\"cCodCla\""+ "="+cCod+"";
        if (objC.verificar(sql) == 0){
            sql="insert into "+"\"tblClases\""+ " values
('"+cCod+"', '"+cNom+"')";
            if (objC.insertar(sql)==null){
                out.println("<h4>Datos Insertados</h4>");
                out.print("<p><a
href=\" javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>");
            }
            else{
                out.println("<h2><center>Problemas para
Registrar</center></h2>"+sql+objC.insertar(sql));
                out.print("<p><a
href=\" javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>");
            }
        }
        else{
            out.println("<h4>El Código ya se encuentra
registrado</center></h4>");
            out.print("<p><a href=\" javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>");
        }
    } catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4>Debera ingresar informacion completa</center></h4>");
    out.print("<center><p><a href=\" javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>");
}
}
```

Diagrama de Interfaz: Consulta Clase de Activos

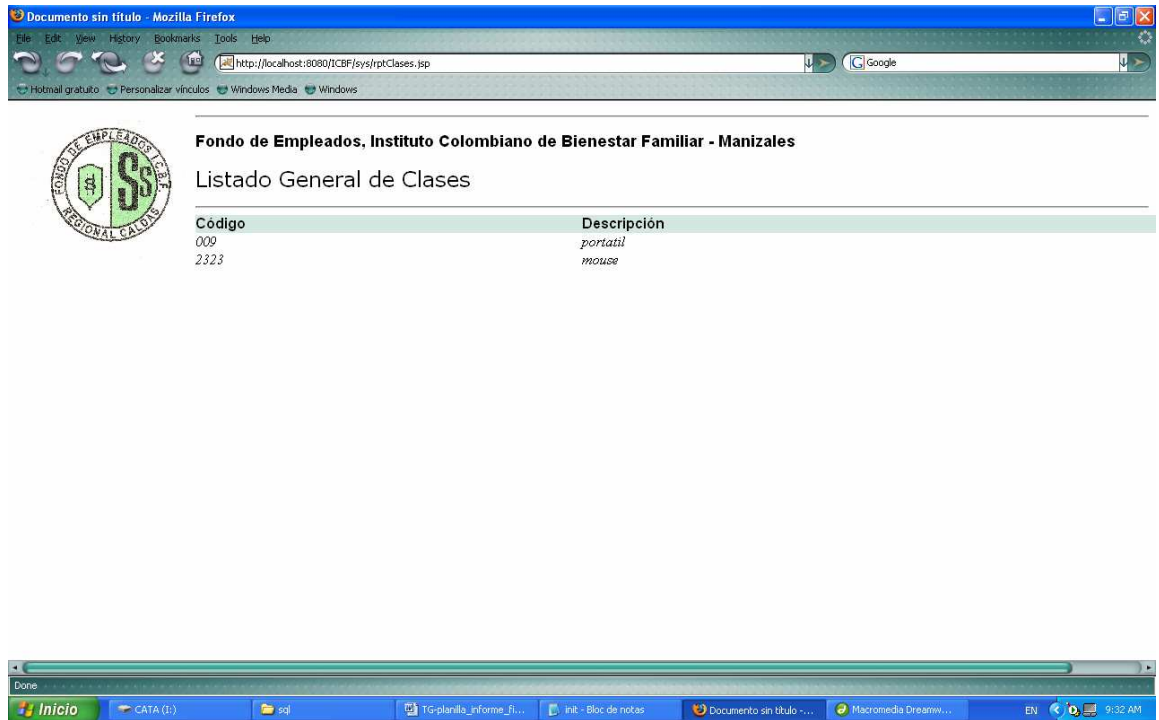


Figura 26 Diagrama de Interfaz: Consulta Clase de Activos

Diagrama de Componentes: Consulta Clase de Activos

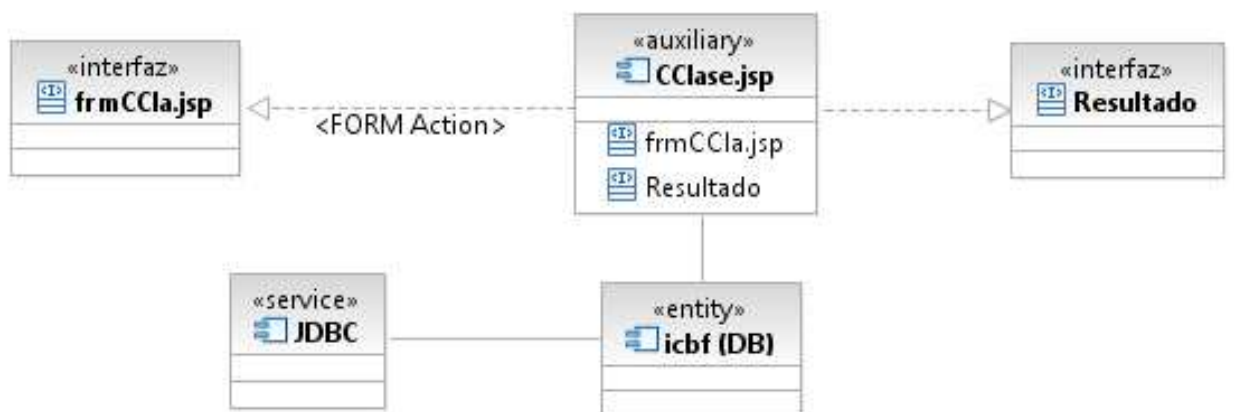


Figura 27 Diagrama de Componentes: Consulta Clase de Activos

Código: Consultar Clase Activos

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblClases\""+ " where
"+"\"cCodCla\""+\"=\"+\"cCod+\"";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append(" <table width=\"57%\" border=\"0\"> ");
            sb.append(" <tr> ");
            sb.append(" <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Descripcion:</font></div></td> ");
            sb.append(" <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append(" </tr> ");
            sb.append(" </table> ");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de Interfaz: Modifica Clase de Activos



Figura 28 Diagrama de Interfaz: Modifica Clase de Activos

Diagrama de Componentes: Modifica Clase de Activos

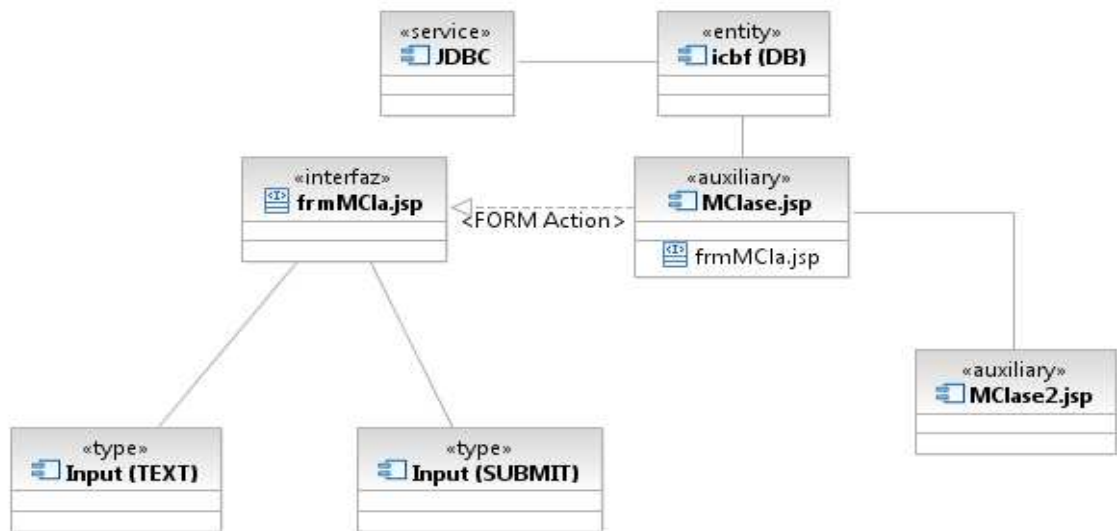


Figura 29 Diagrama de Componentes: Modifica Clase de Activos

Código: Modifica Clase de Activos

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblClases\""+" where
"+"\"cCodCla\""+\"=\"'+cCod+'\"";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                                <form method=\"post\"
action=\"Modificacion.jsp\" target=\"desc\" >");
            sb.append(" <center> <table width=\"57%\" border=\"0\" > ");
            sb.append(" <tr> ");
            sb.append(" <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Descripcion:</font></div></td> ");
            sb.append(" <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+"\" size=\"26\" maxlength=\"26\"></td>
");
            sb.append(" </tr> ");
            sb.append(" <tr> ");

            sb.append(" <td><input name=\"cCod\" type=\"hidden\"
id=\"cCod2\" value=\""+cCod+"\" ></td> ");
            sb.append(" <tr> <td height=\"31\" colspan=\"2\"> <div
align=\"center\"> ");
            sb.append(" <input name=\"ModificarClase\" type=\"submit\"
target=\"desc\" value=\"Modificar Clase\" width=\"91\" height=\"27\" > ");
            sb.append(" </div></td> </tr> ");
            sb.append(" </tr> ");
            sb.append(" </table> </center> ");
            sb.append(" </form > ");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de Interfaz: Elimina Clase de Activos



Figura 30 Diagrama de Interfaz: Elimina Clase de Activos

Diagrama de Componentes: Elimina Clase de Activos

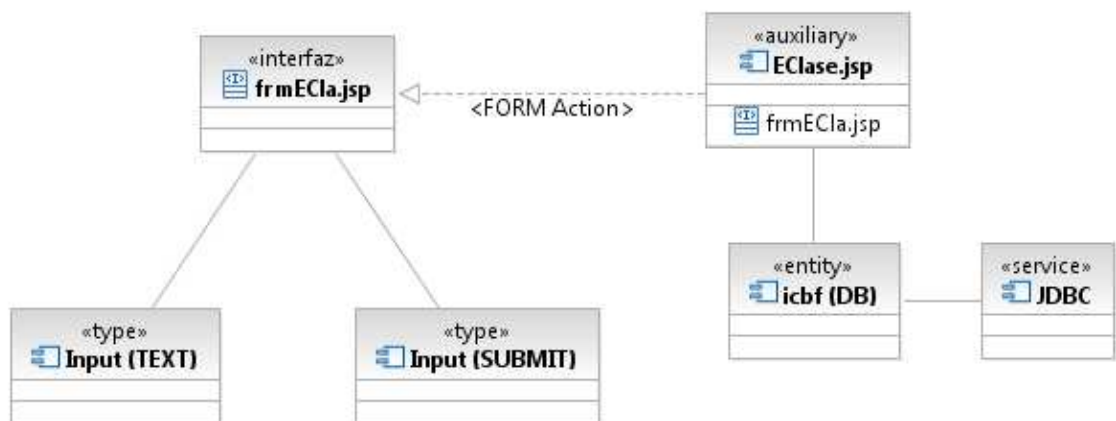


Figura 31 Diagrama de Componentes: Elimina Clase de Activos

Código: Elimina Clase de Activos

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from "+"\"tblClases\""+" where
"+"\"cCodCla\""+"='"+cCod+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from "+"\"tblClases\""+" where
"+"\"cCodCla\""+"='"+cCod+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para eliminar </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }

            else{
                out.println("<h4> Informacion eliminada </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }

        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print(" <p><a href=\"javascript:history.back()\"><font face=\"Verdana,
Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
}
```


Diagrama de Interfaz: Ingresar Estados

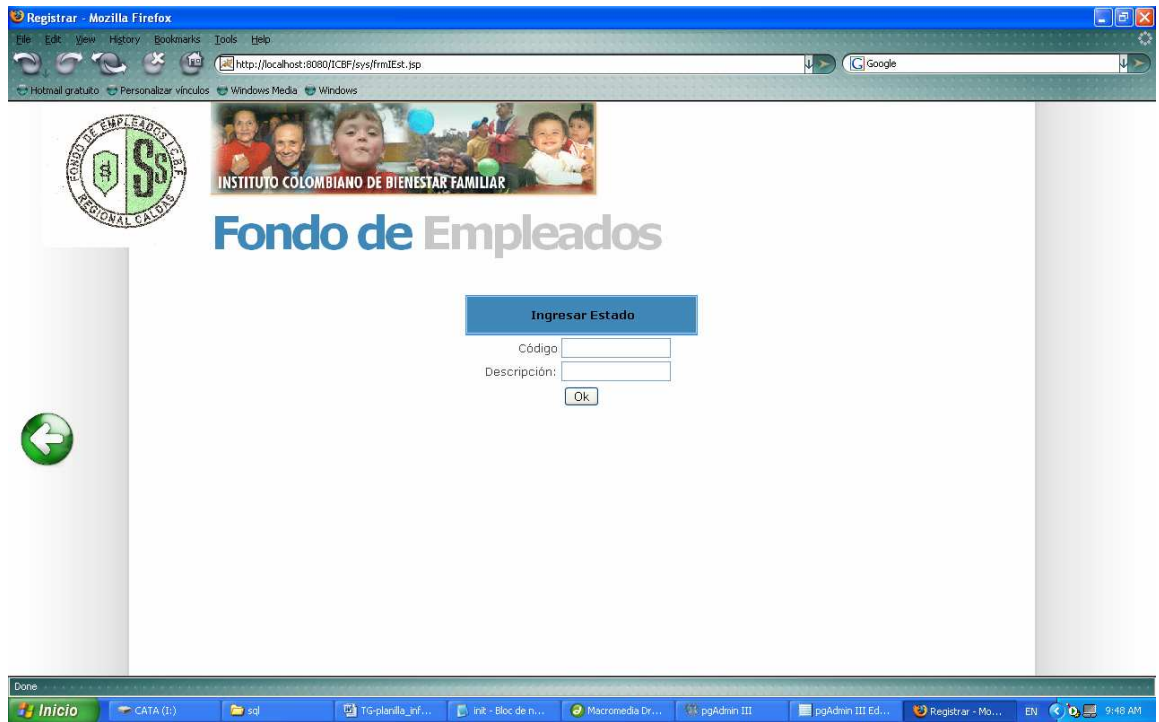


Figura 32 Diagrama de Interfaz: Ingresar Estados

Diagrama de Componentes: Ingresar Estados

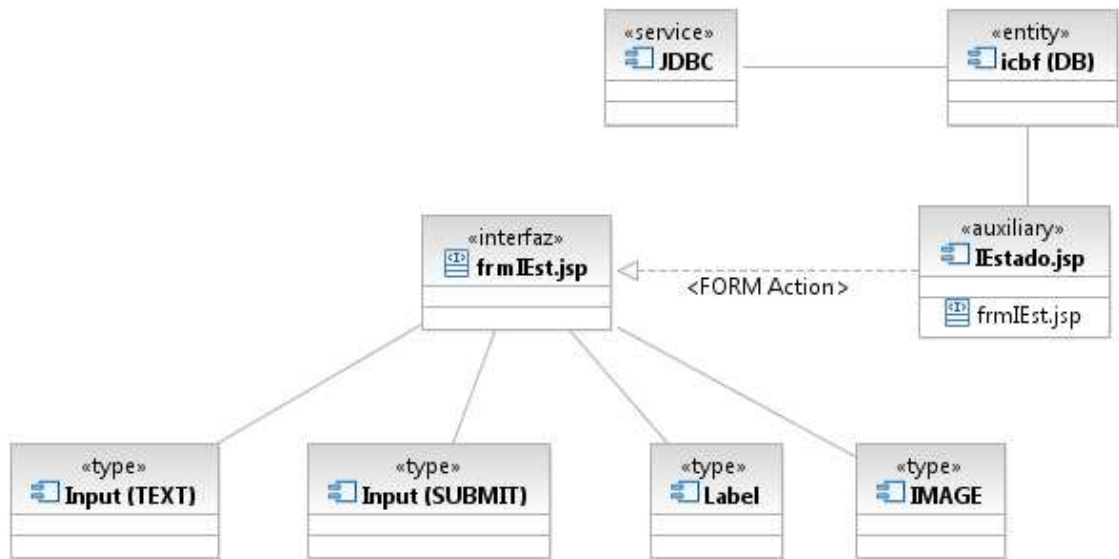


Figura 33 Diagrama de Componentes: Ingresar Estados

Código: Ingresar Estados

```
String cCod=request.getParameter("cCod");
String cNom=request.getParameter("cNom");

if ((cCod != null && cCod.equals("") != true) && (cNom != null && cNom.equals("")
!= true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblEstados\""+ " where
"+"\"cCodEst\""+"='"+cCod+"'";
        if (objC.verificar(sql) == 0){
            sql="insert into "+"\"tblEstados\""+ " values
('"+cCod+"', '"+cNom+"')";
            if (objC.insertar(sql)==null){
                out.println("<h4>Datos Insertados</h4>");
                out.print("<p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>");
            }
            else{
                out.println("<h2>Problemas para
Registrar</h2>"+sql+objC.insertar(sql));
                out.print("<p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>");
            }
        }
        else{
            out.println("<h4> El Código ya se encuentra registrado </h4>");
            out.print(" <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar informacion completa </h4>");
}
```

Diagrama de Interfaz: Consulta Estados

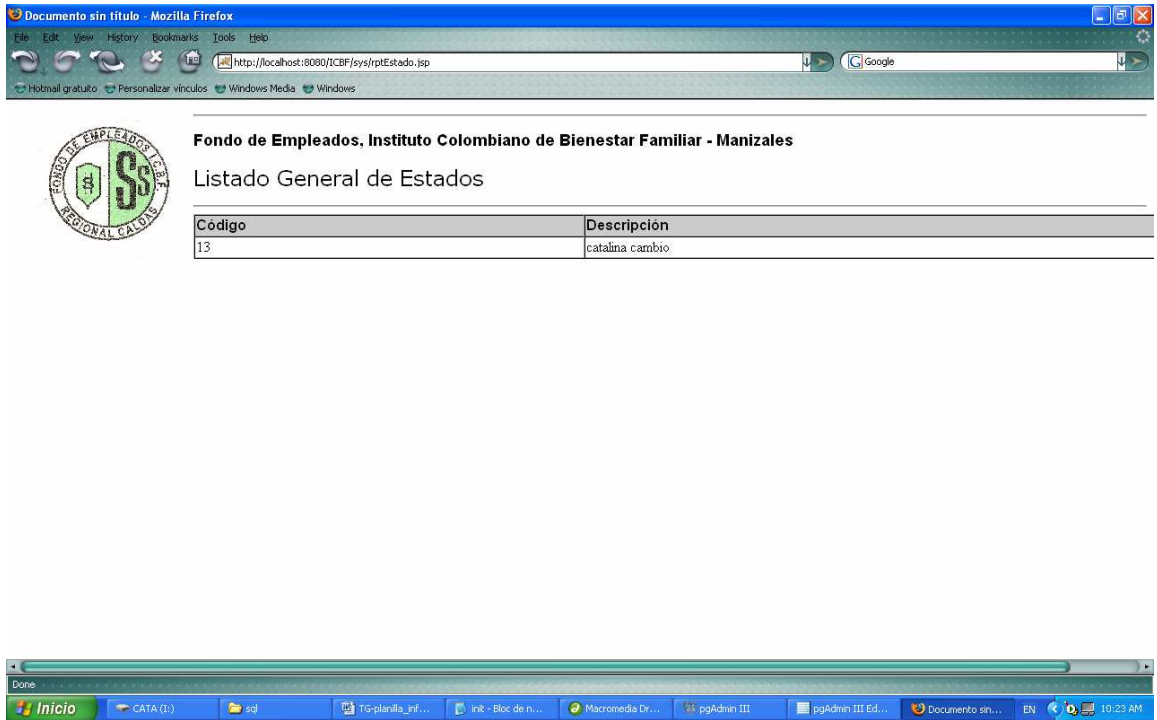


Figura 34 Diagrama de Interfaz: Consulta Estados

Diagrama de Componentes: Consulta Estados

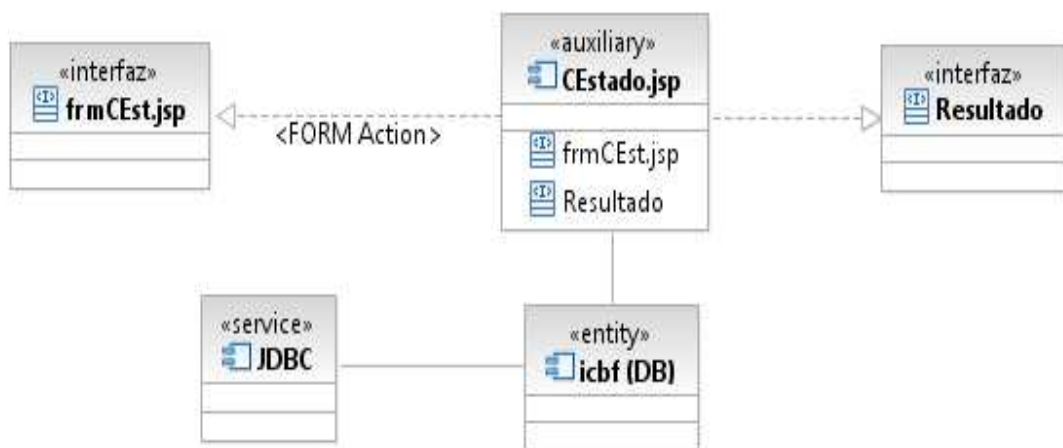


Figura 35 Diagrama de Componentes: Consulta Estados

Código: Consulta Estados

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblEstados\""+" where
"+"\"cCodEst\""+\"=\"+cCod+\"";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append(" <table width=\"57%\" border=\"0\"> ");
            sb.append(" <tr> ");
            sb.append(" <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Descripcion:</font></div></td> ");
            sb.append(" <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append(" </tr> ");
            sb.append(" </table> ");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de Interfaz: Modifica Estados

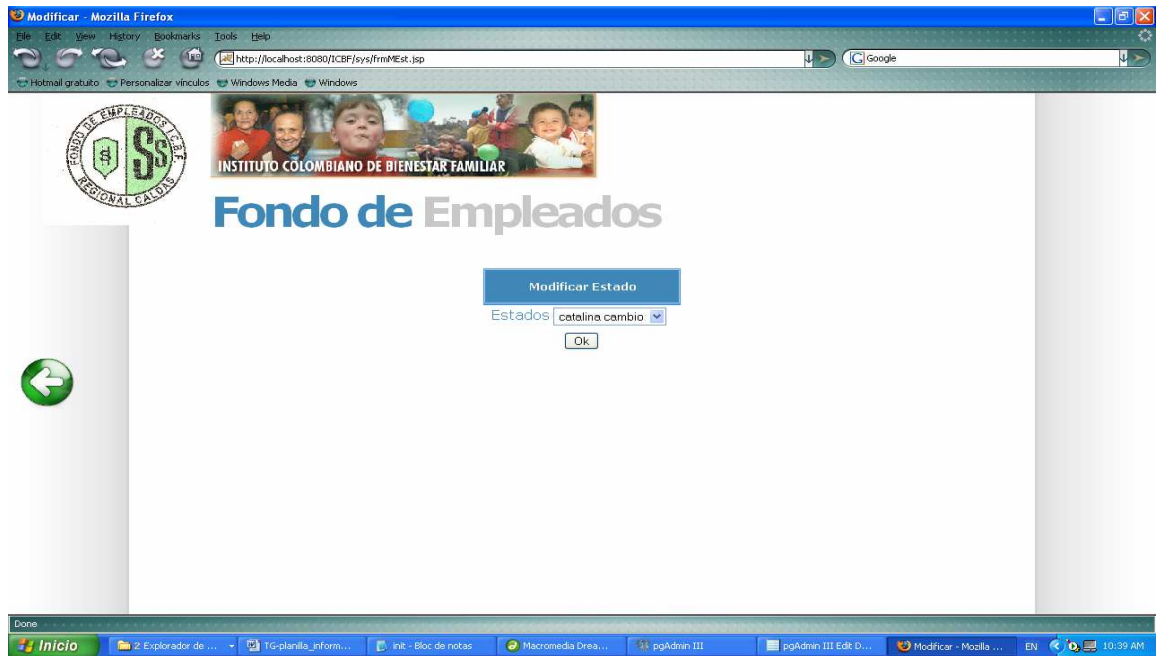


Figura 36 Diagrama de Interfaz: Modifica Estados

Diagrama de Componentes: Modifica Estados

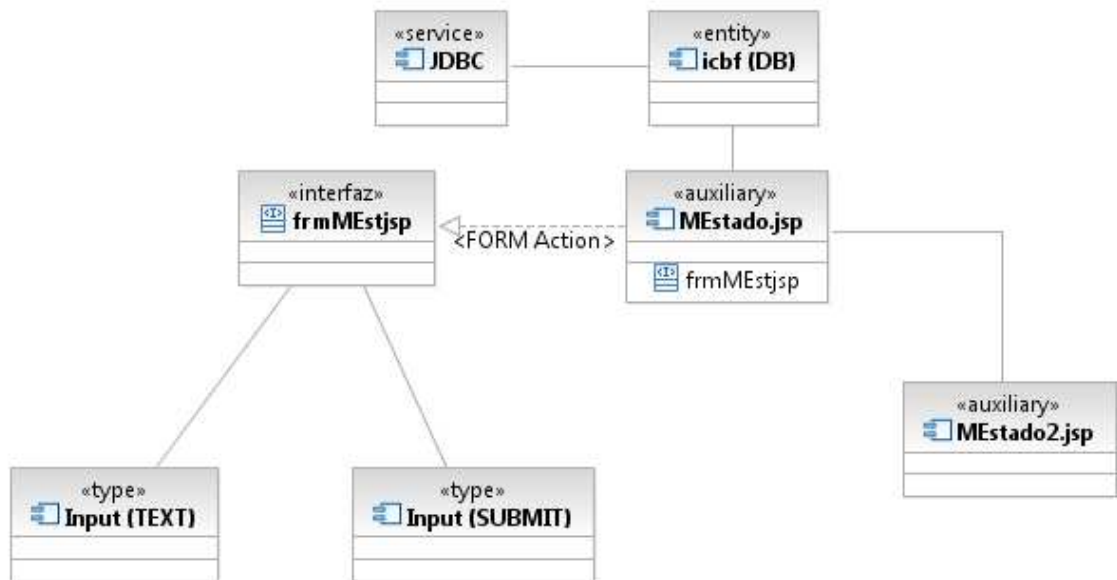


Figura 37 Diagrama de Componentes: Modifica Estados

Código: Modifica Estados

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblEstados\""+" where
"+"\"cCodEst\""+\"=\"'+cCod+'\"";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                                <form method=\"post\"
action=\"Modificacion.jsp\" target=\"desc\" >");
            sb.append(" <center> <table width=\"57%\" border=\"0\" > ");
            sb.append(" <tr>                                ");
            sb.append(" <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Descripcion:</font></div></td> ");
            sb.append(" <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+"\" size=\"26\" maxlength=\"26\"></td>
");
            sb.append(" </tr>                                ");
            sb.append(" <tr>                                ");

            sb.append(" <td><input name=\"cCod\" type=\"hidden\"
id=\"cCod2\" value=\""+cCod+"\" ></td> ");
            sb.append(" <tr> <td height=\"31\" colspan=\"2\"> <div
align=\"center\"> ");
            sb.append(" <input name=\"ModificarEstado\" type=\"submit\"
target=\"desc\" value=\"Modificar Estado\" width=\"91\" height=\"27\" > ");
            sb.append(" </div></td> </tr> ");
            sb.append(" </tr> ");
            sb.append(" </table> </center> ");
            sb.append(" </form >                                ");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}
```


Diagrama de Interfaz: Elimina Estados

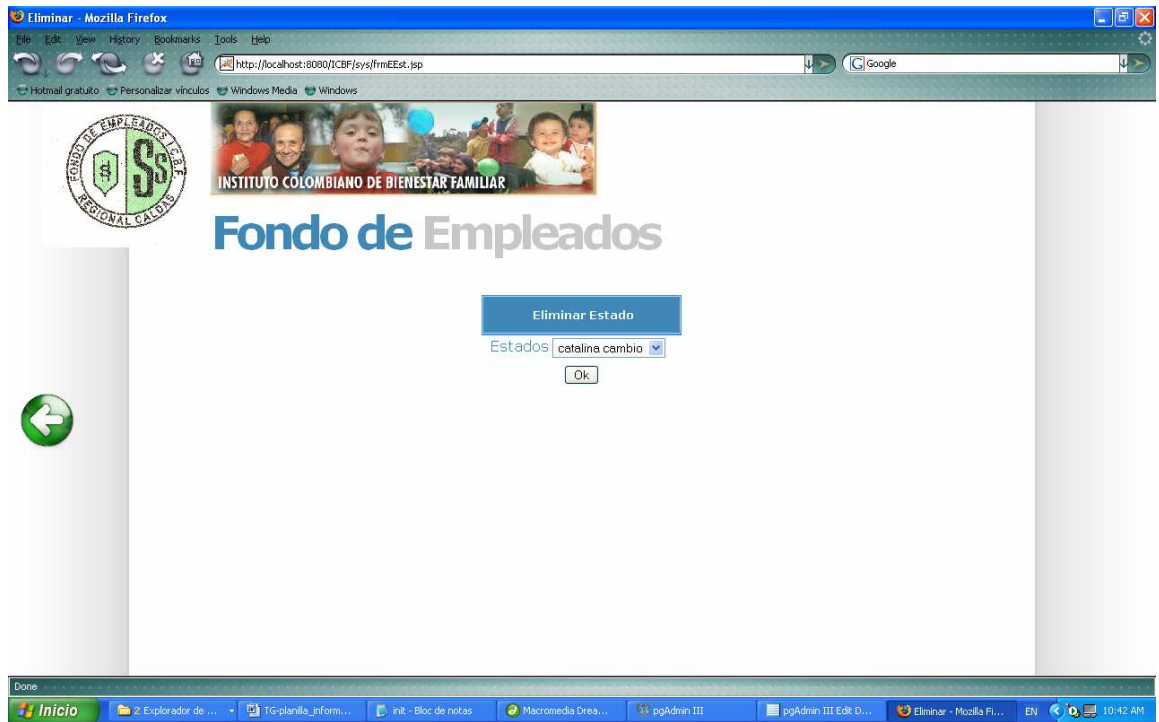


Figura 38 Diagrama de Interfaz: Elimina Estados

Diagrama de Componentes: Elimina Estados

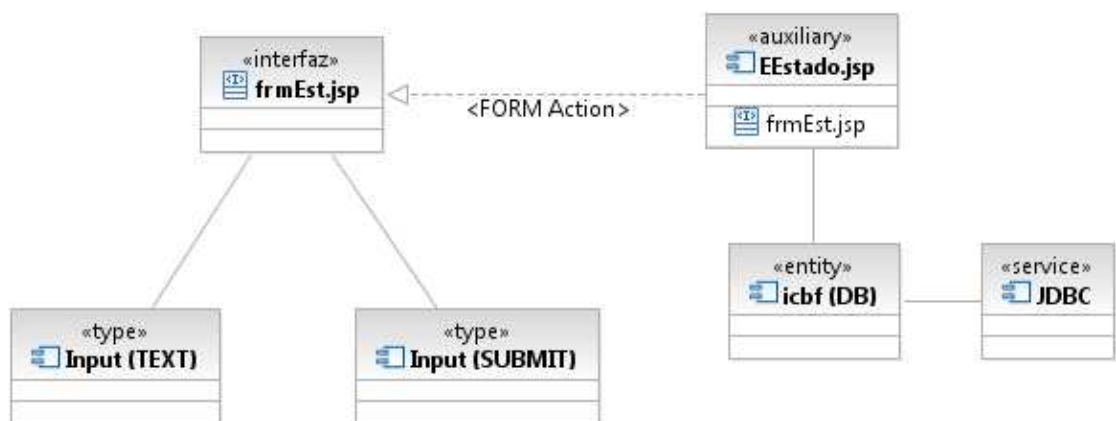


Figura 39 Diagrama de Componentes: Elimina Estados

Código: Elimina Estados

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from "+"\"tblEstados\""+
        "+"\"cCodEst\""+="'"+cCod+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from "+"\"tblEstados\""+
            "+"\"cCodEst\""+="'"+cCod+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para eliminar </h4>");
                out.print("
                href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
                serif\">&lt;&lt;</font></a></p> ");
            }

            else{
                out.println("<h4> Informacion eliminada </h4>");
                out.print("
                href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
                serif\">&lt;&lt;</font></a></p> ");
            }

        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("
            href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
            serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("
    href=\"javascript:history.back()\"><font face=\"Verdana,
    Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de Interfaz: Ingresar Proveedores

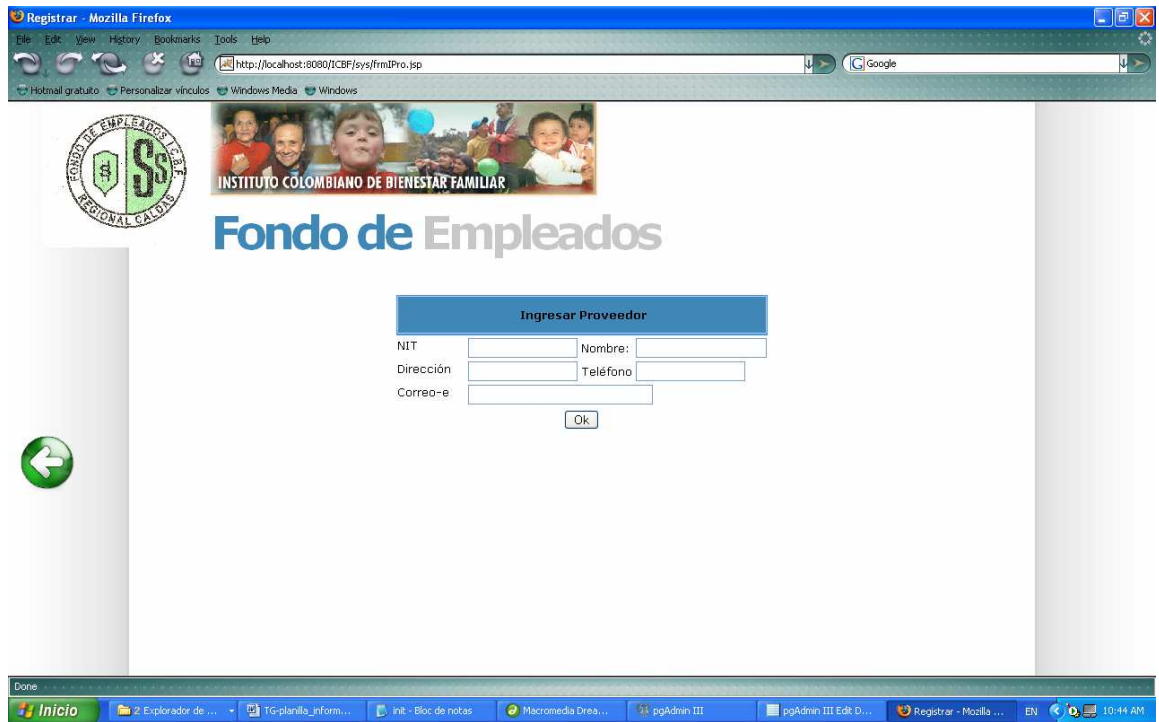


Figura 40 Diagrama de Interfaz: Ingresar Proveedores

Diagrama de Componentes: Ingresar Proveedores

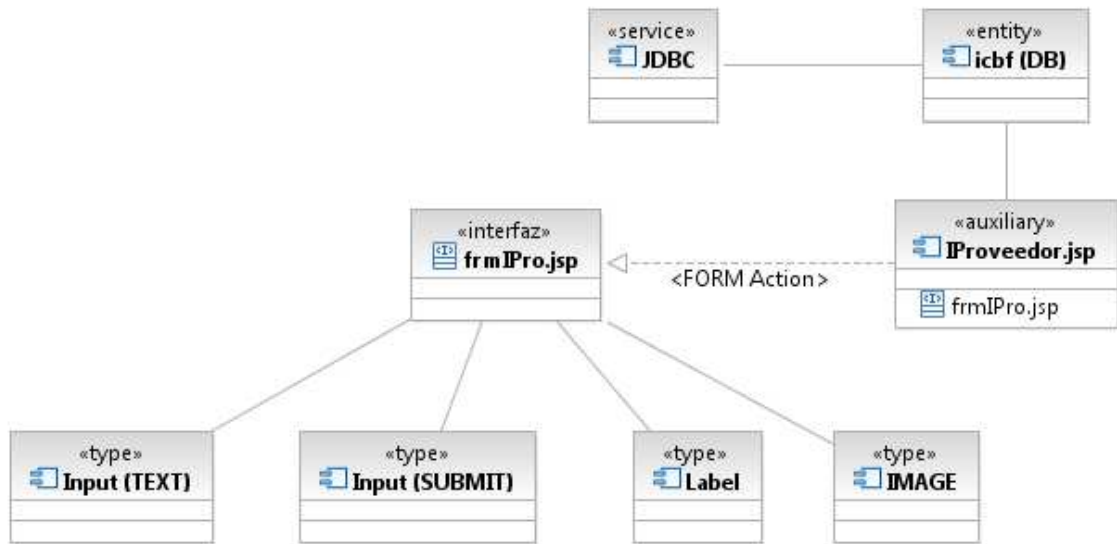


Figura 41 Diagrama de Componentes: Ingresar Proveedores

Código: Ingresar Proveedor

```
String cCod=request.getParameter("cCod");
String cNom=request.getParameter("cNom");
String cDir=request.getParameter("cDir");
String cTel=request.getParameter("cTel");
String cCor=request.getParameter("cCor");

if ((cCod != null && cCod.equals("") != true) && (cNom != null && cNom.equals("")
!= true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblProveedores\""+" where
"+"\"cNitPro\""+"='"+cCod+"'";
        if (objC.verificar(sql) == 0){
            sql="insert into "+"\"tblProveedores\""+" values
('"+cCod+"', '"+cNom+"', '"+cDir+"', '"+cTel+"', '"+cCor+"')";
            if (objC.insertar(sql)==null){
                out.println("<h4> Datos Insertados </h4>");
                out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
            else{
                out.println("<h2> Problemas para Registrar
</h2>"+sql+objC.insertar(sql));
                out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
        }
        else{
            out.println("<h4> El Código ya se encuentra registrado </h4>");
            out.print("                <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar informacion completa </h4>");
}
```

Diagrama de Interfaz: Consulta Proveedor

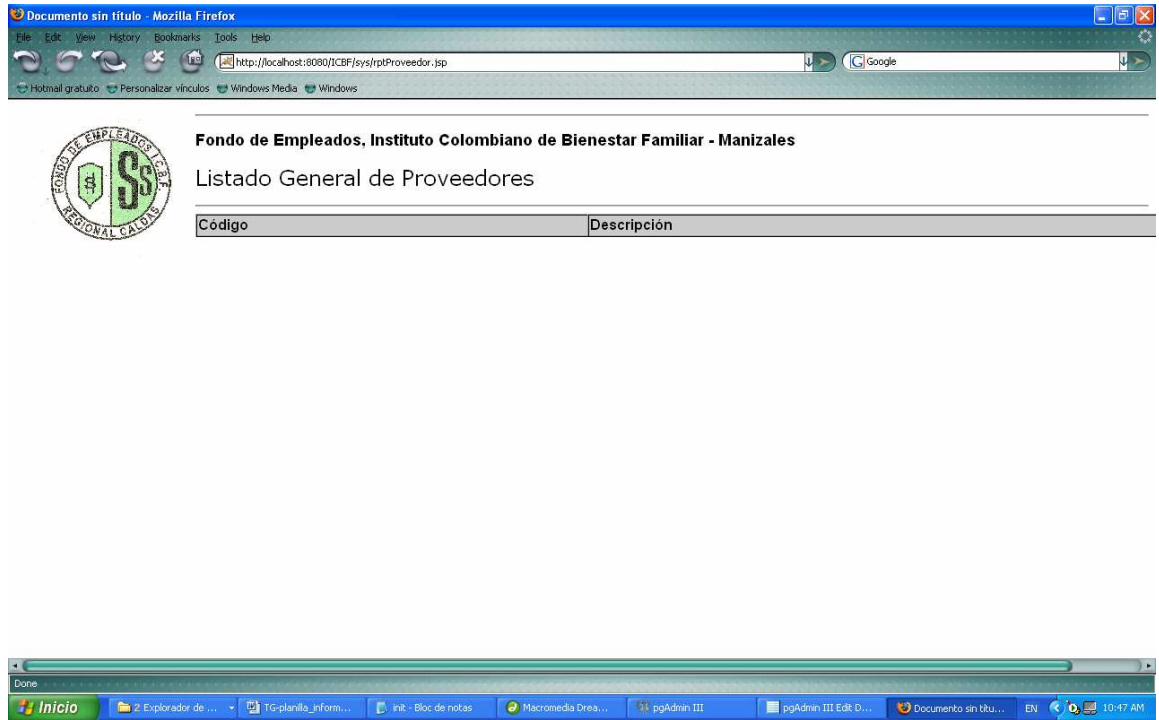


Figura 42 Diagrama de Interfaz: Consulta Proveedor

Diagrama de Componentes: Consulta Proveedor

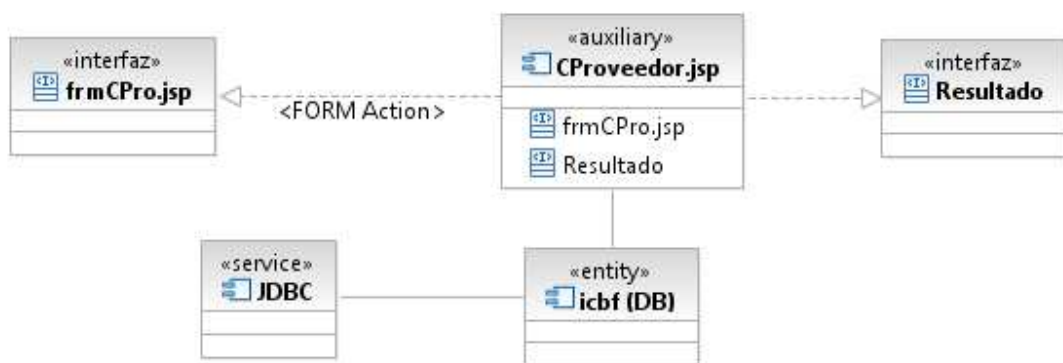


Figura 43 Diagrama de Componentes: Consulta Proveedor

Código: Consulta Proveedor

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblProveedores\""+" where
"+"\"cNitPro\""+"='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                <table width=\"57%\" border=\"0\"> ");
            sb.append("                <tr>");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Nombre:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>");
            sb.append("                <tr>");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Direccion:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(3)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>");
            sb.append("                <tr>");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Telefono:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(4)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>");
            sb.append("                <tr>");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Correo-e:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(5)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>");
            sb.append("                </table>");
            sb.append("                <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
    }
}
```

```

        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>
");
}

```


Diagrama de Interfaz: Modifica Proveedor



Figura 44 Diagrama de Interfaz: Modifica Proveedor

Diagrama de Componentes: Modifica Proveedor

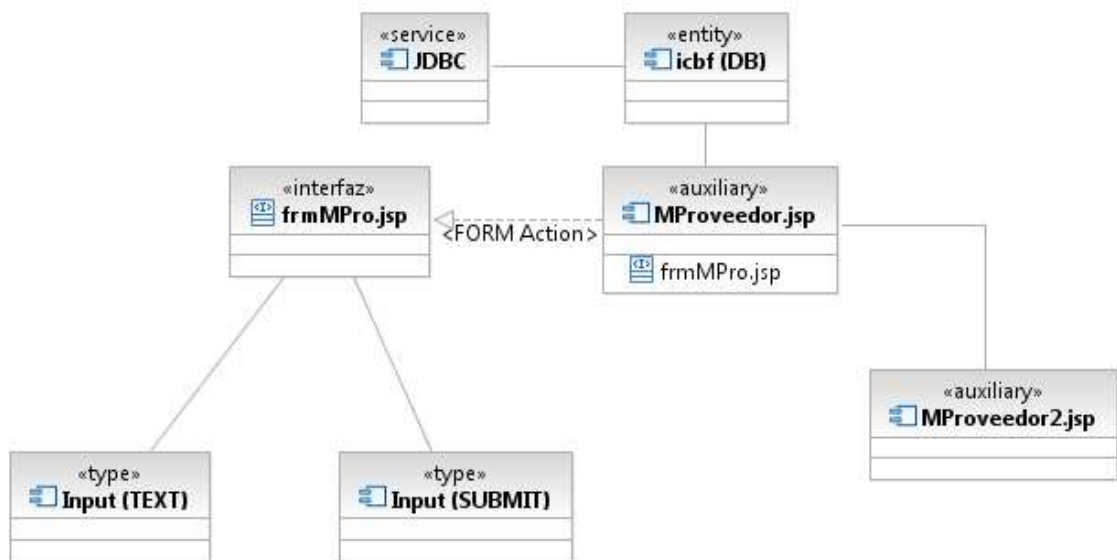


Figura 45 Diagrama de Componentes: Modifica Proveedor

Código: Modifica Proveedor

```

String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblProveedores\""+ " where
"+"\"cNitPro\""+"='"+cCod+'";
        rs=objC.consultar(sql);

        if (rs.next()){%>
            <form method="post" action="Modificacion.jsp" target="desc" >
                <table width="370" border="0" align="center">
                    <tr>
                        <td>&nbsp;</td>
                        <td>&nbsp;</td>
                        <td>&nbsp;</td>
                        <td>&nbsp;</td>
                    </tr>
                    <tr>
                        <td>&nbsp;</td>
                        <td><div align="right">Nombre:</div></td>
                        <td><input name="cNom" type="text" id="cNom"
value="<%=rs.getString(2)%>"></td>
                        <td>&nbsp;</td>
                    </tr>
                    <tr>
                        <td>&nbsp;</td>
                        <td><div align="right">Direccion:</div></td>
                        <td><input name="cDir" type="text" id="cDir"
value="<%=rs.getString(3)%>"></td>
                        <td>&nbsp;</td>
                    </tr>
                    <tr>
                        <td>&nbsp;</td>
                        <td><div align="right">Telefono:</div></td>
                        <td><input name="cTel" type="text" id="cTel"
value="<%=rs.getString(4)%>"></td>
                        <td>&nbsp;</td>
                    </tr>
                    <tr>
                        <td>&nbsp;</td>
                        <td><div align="right">Correo:</div></td>
                        <td><input name="cCor" type="text" id="cCor"
value="<%=rs.getString(5)%>"></td>
                        <td>&nbsp;</td>
                    </tr>
                    <tr>
                        <td>&nbsp;</td>
                        <td colspan="2"><div align="center">
                            <input name="ModificarProveedor" type="submit"
id="ModificarProveedor" value="Modificar Proveedor">
                        </div></td>
                    </tr>
                </table>
            </form>
        }
    }
}

```

```

                                <td><input    name="cCod"    type="hidden"    id="cCod"
value="<%=rs.getString(1)%>"></td>
                                </tr>
                                </table>
</form>
    <%
    }
    else{
        out.println("<h4> Código no encontrado </h4>");
        out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
    }
    rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}

```

Diagrama de Interfaz: Elimina Proveedor

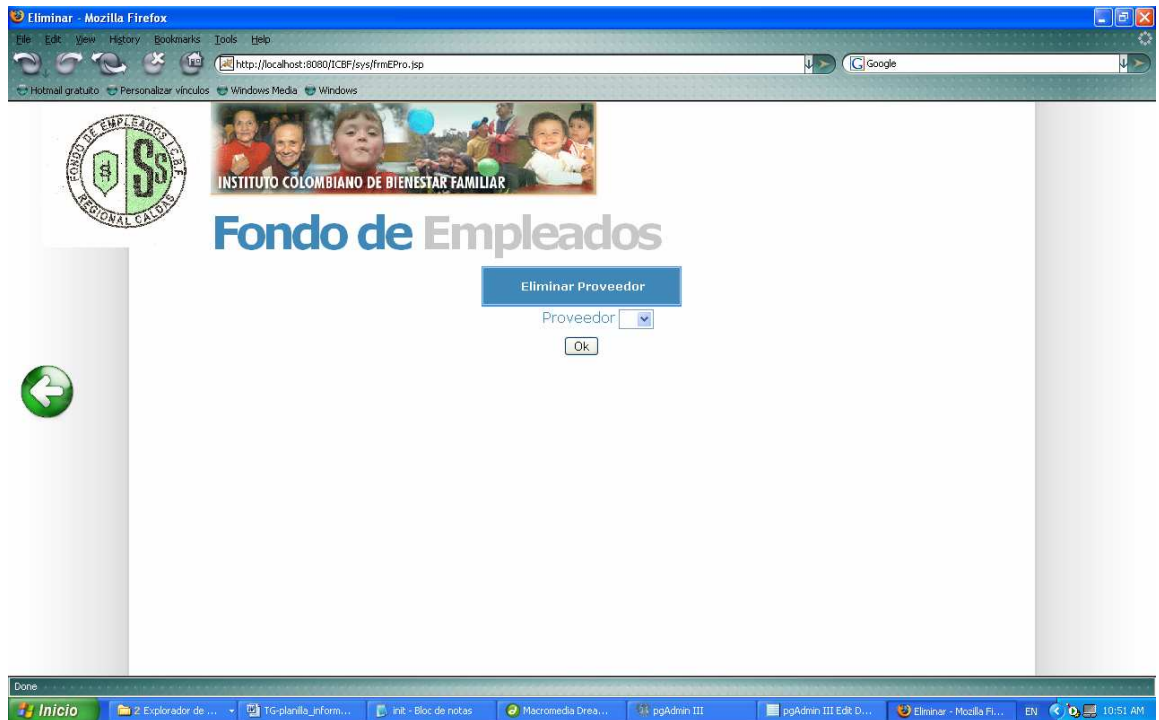


Figura 46 Diagrama de Interfaz: Elimina Proveedor

Diagrama de Componentes: Elimina Proveedor

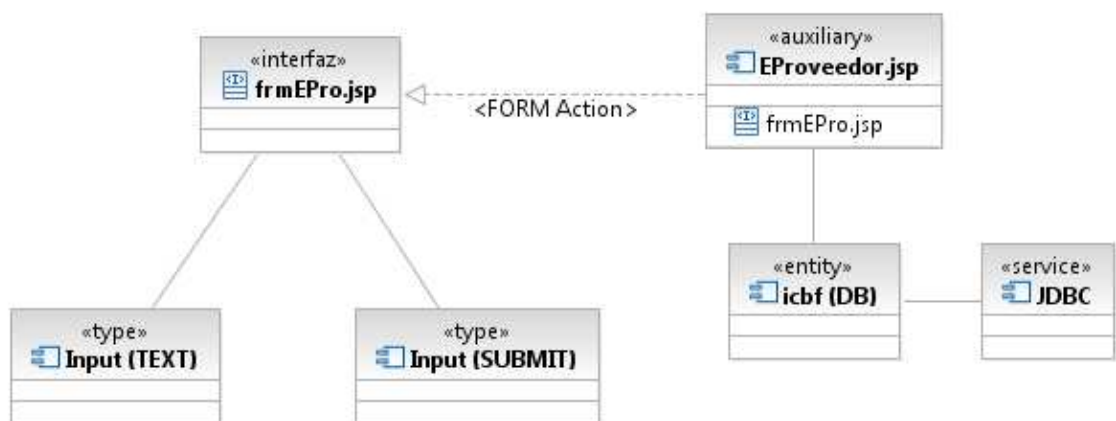


Figura 47 Diagrama de Componentes: Elimina Proveedor

Código: Elimina Proveedor

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from "+"\"tblProveedores\""+ where
        "+"\"cNitPro\""+="'"+cCod+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from "+"\"tblProveedores\""+ where
            "+"\"cNitPro\""+="'"+cCod+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para eliminar</center></h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p></center>");
            }
            else{
                out.println("<h4> Informacion eliminada</center></h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p></center>");
            }
        }
        else{
            out.println("<h4> Código no encontrado</center></h4>");
            out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p></center>");
        }
    } catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código</center></h4>");
    out.print(" <p><a
href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p></center>");
}
```

Diagrama de Interfaz: Ingresar Responsable

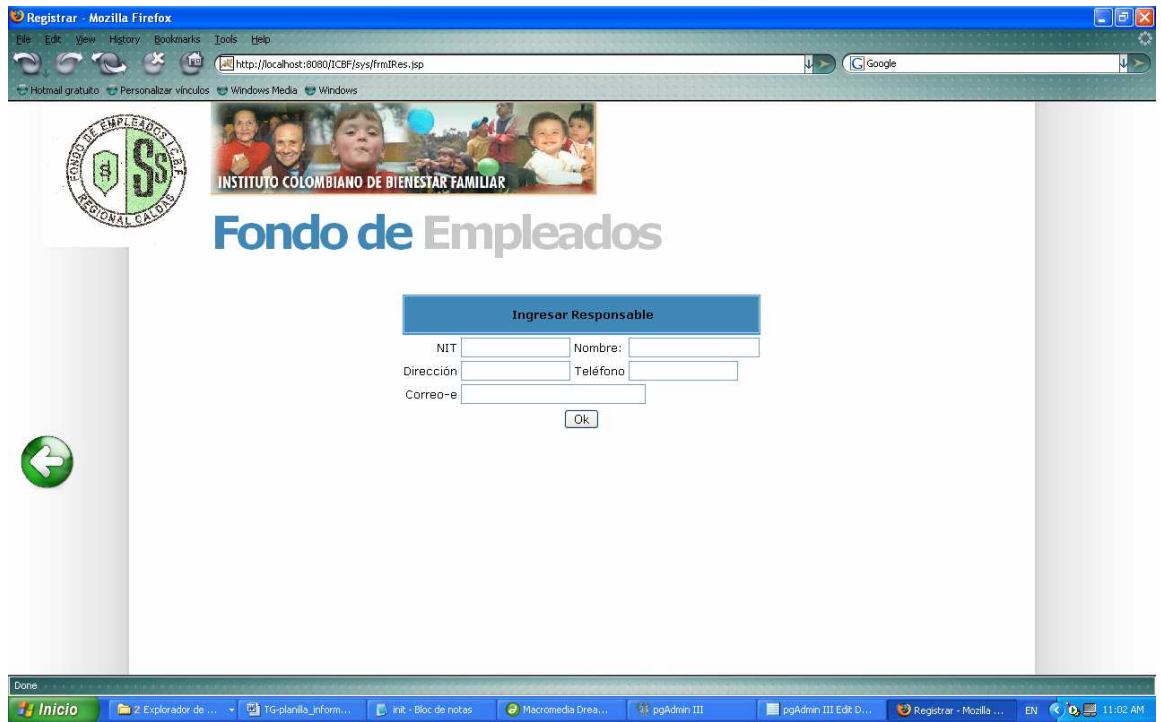


Figura 48 Diagrama de Interfaz: Ingresar Responsable

Diagrama de Componentes: Ingresar Responsable

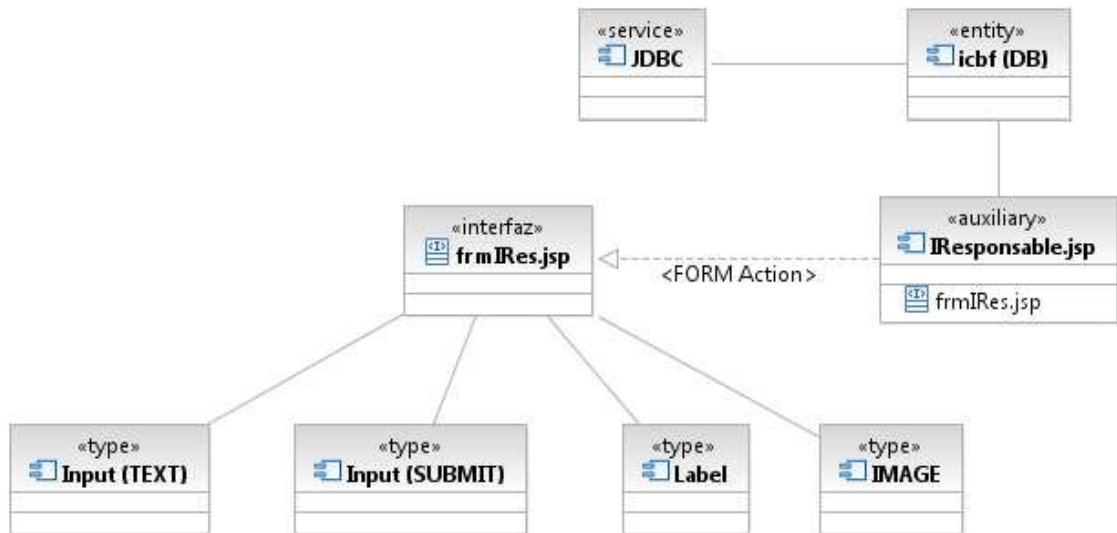


Figura 49 Diagrama de Componentes: Ingresar Responsable

Código: Ingresar Responsable

```
String cCod=request.getParameter("cCod");
String cNom=request.getParameter("cNom");
String cDir=request.getParameter("cDir");
String cTel=request.getParameter("cTel");
String cCor=request.getParameter("cCor");

if ((cCod != null && cCod.equals("") != true) && (cNom != null && cNom.equals("")
!= true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblResponsable\""+" where
"+"\"cCodRes\""+"='"+cCod+"'";
        if (objC.verificar(sql) == 0){
            sql="insert into "+"\"tblResponsable\""+" values
('"+cCod+"', '"+cNom+"', '"+cDir+"', '"+cTel+"', '"+cCor+"')";
            if (objC.insertar(sql)==null){
                out.println("<h4> Datos Insertados </h4>");
                out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
            else{
                out.println("<h2> Problemas para Registrar
</h2>"+sql+objC.insertar(sql));
                out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
        }
        else{
            out.println("<h4> El Código ya se encuentra registrado </h4>");
            out.print("                <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar informacion completa </h4>");
}
}
```

Diagrama de Interfaz: Consulta Responsable



Figura 50 Diagrama de Interfaz: Consulta Responsable

Diagrama de Componentes: Consulta Responsable

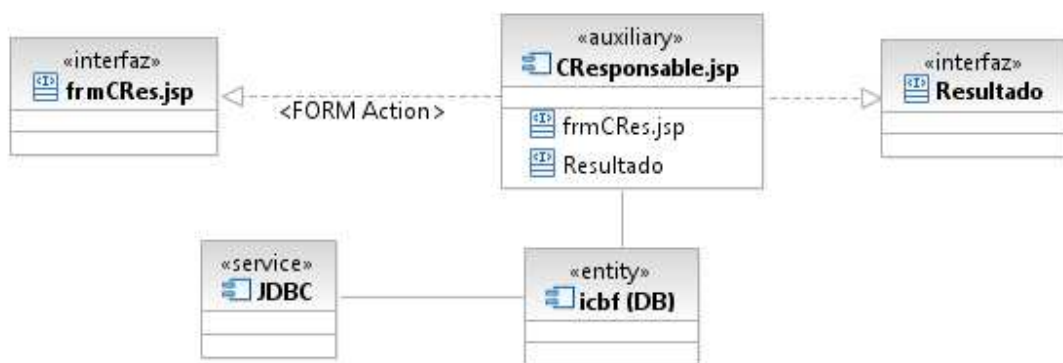


Figura 51 Diagrama de Componentes: Consulta Responsable

Código: Consulta Responsable

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblResponsable\""+" where
"+"\"cCodRes\""+"='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                <table width=\"57%\" border=\"0\"> ");
            sb.append("                <tr>");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Nombre:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>");
            sb.append("                <tr>");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Direccion:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(3)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>");
            sb.append("                <tr>");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Telefono:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(4)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>");
            sb.append("                <tr>");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Correo-e:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(5)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>");
            sb.append("                </table>");
            sb.append("                <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
    }
}
```

```

        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>
");
}

```

Diagrama de Interfaz: Modifica Responsable



Figura 52 Diagrama de Interfaz: Modifica Responsable

Diagrama de Componentes: Modifica Responsable

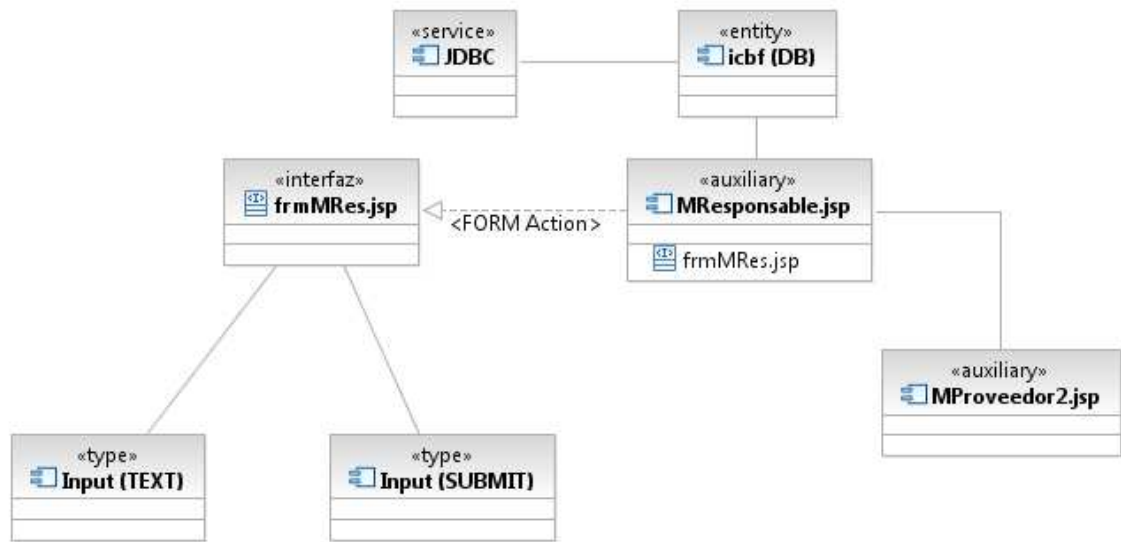


Figura 53 Diagrama de Componentes: Modifica Responsable

Código: Modifica Responsable

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){
            out.println("<h4> Problemas en conexion </h4>"+x);
        }
        String sql="select * from "+"\"tblResponsable\" "+" where
"+"\"cCodRes\""+"\"="+cCod+" ";
        rs=objC.consultar(sql);

        if (rs.next())
        {
            %>
            <table width="370" border="0" align="center">
            <tr>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
                <td>&nbsp;</td>
            </tr>
            <tr>
                <td>&nbsp;</td>
                <td><div align="right">Nombre:</div></td>
                <td><input name="cNom" type="text" id="cNom"
value="<%=rs.getString(2)%>"></td>
                <td>&nbsp;</td>
            </tr>
            <tr>
                <td>&nbsp;</td>
                <td><div align="right">Direccion:</div></td>
                <td><input name="cDir" type="text" id="cDir"
value="<%=rs.getString(3)%>"></td>
                <td>&nbsp;</td>
            </tr>
            <tr>
                <td>&nbsp;</td>
                <td><div align="right">Telefono:</div></td>
                <td><input name="cTel" type="text" id="cTel"
value="<%=rs.getString(4)%>"></td>
                <td>&nbsp;</td>
            </tr>
            <tr>
                <td>&nbsp;</td>
                <td><div align="right">Correo:</div></td>
                <td><input name="cCor" type="text" id="cCor"
value="<%=rs.getString(5)%>"></td>
                <td>&nbsp;</td>
            </tr>
            <tr>
                <td>&nbsp;</td>
            </tr>
        }
    }
}
```



```

                <td colspan="2"><div align="center">
                    <input      name="ModificarProveedor"      type="submit"
id="ModificarProveedor" value="Modificar Proveedor">
                </div></td>
                <td><input      name="cCod"      type="hidden"      id="cCod"
value="<%=rs.getString(1)%>"></td>
            </tr>
        </table>
        %>
    }
    else{
        out.println("<h4> Código no encontrado </h4>");
        out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
    }
    rs.close();
}
catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}

```

Diagrama de Interfaz: Elimina Responsable

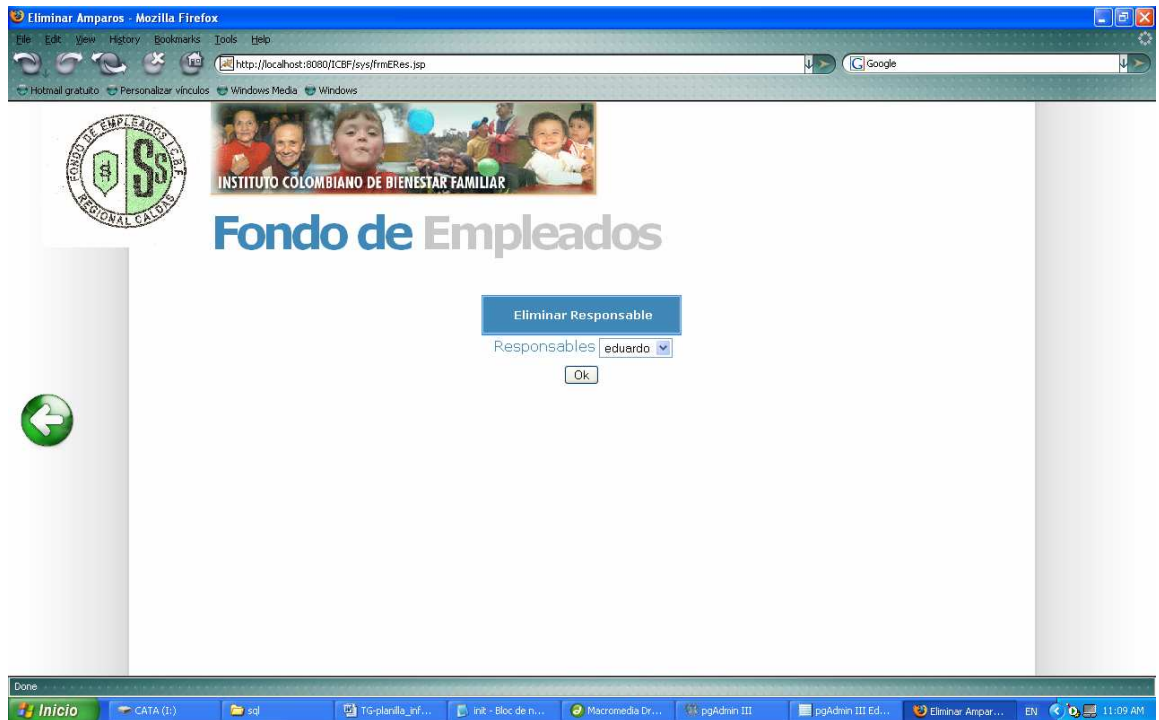


Figura 54 Diagrama de Interfaz: Elimina Responsable

Diagrama de Componentes: Elimina Responsable

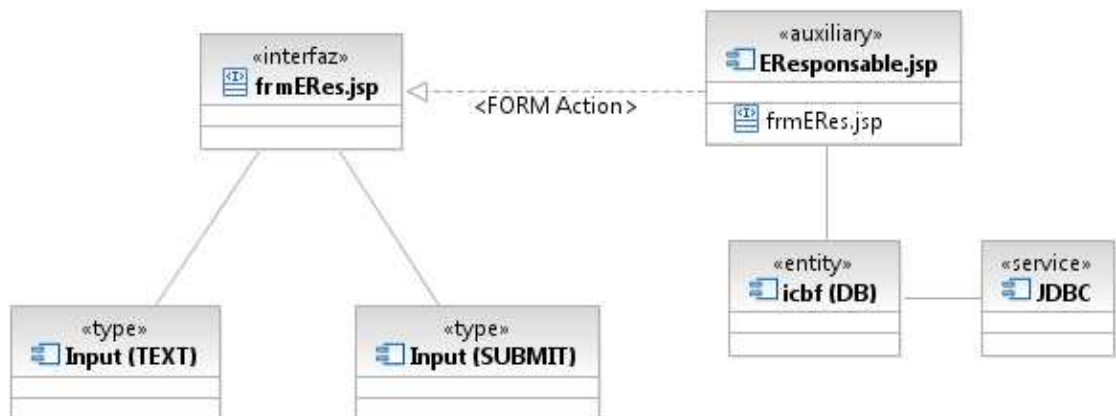


Figura 55 Diagrama de Componentes: Elimina Responsable

Código: Elimina Responsable

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from "+"\"tblResponsable\""+ " where
"+"\"cCodRes\""+"='"+cCod+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from "+"\"tblResponsable\""+ " where
"+"\"cCodRes\""+"='"+cCod+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para eliminar </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
        }
        else{
            out.println("<h4> Informacion eliminada </h4>");
            out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
    }
    else{
        out.println("<h4> Código no encontrado </h4>");
        out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print(" <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
}
}
```

Diagrama de Interfaz: Ingresar Ubicación

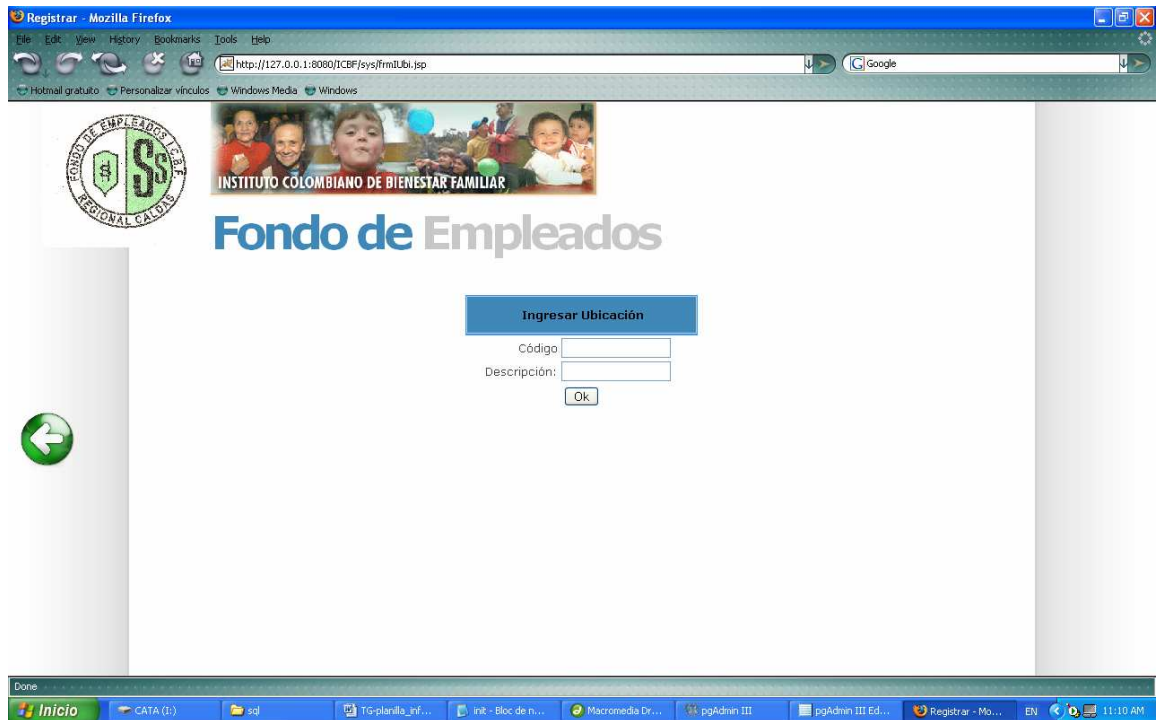


Figura 56 Diagrama de Interfaz: Ingresar Ubicación

Diagrama de Componentes: Ingresar Ubicación

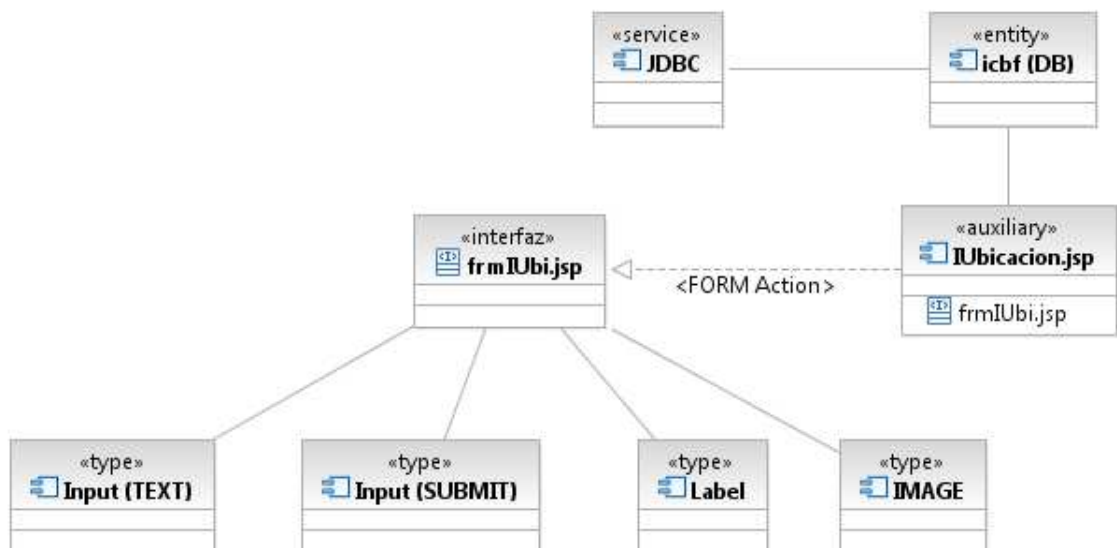


Figura 57 Diagrama de Componentes: Ingresar Ubicación

Código: Ingresar Ubicación

```
String cCod=request.getParameter("cCod");
String cNom=request.getParameter("cNom");

if ((cCod != null && cCod.equals("") != true) && (cNom != null && cNom.equals("")
!= true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblUbicaciones\""+" where
"+"\"cCodUbi\""+"='"+cCod+"'";
        if (objC.verificar(sql) == 0){
            sql="insert into "+"\"tblUbicaciones\""+" values
('"+cCod+"','"+cNom+"'");
            if (objC.insertar(sql)==null){
                out.println("<h4> Datos Insertados </h4>");
                out.print("
                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
            else{
                out.println("<h2> Problemas para Registrar
</h2>"+sql+objC.insertar(sql));
                out.print("
                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
        }
        else{
            out.println("<h4> El Código ya se encuentra registrado </h4>");
            out.print("
            <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar informacion completa </h4>");
}
```

Diagrama de Interfaz: Consulta Ubicación

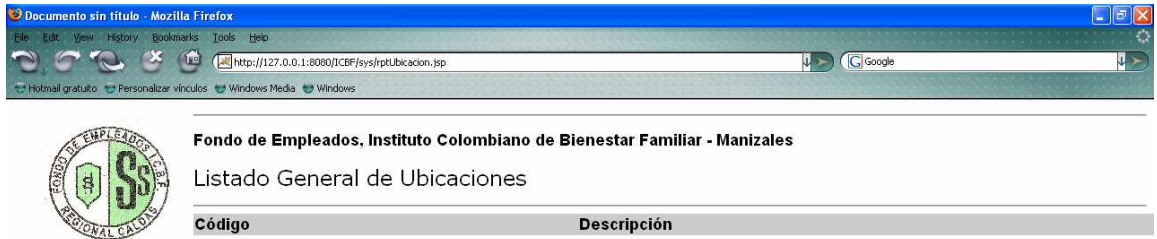


Figura 58 Diagrama de Interfaz: Consulta Ubicación

Diagrama de Componentes: Consulta Ubicación

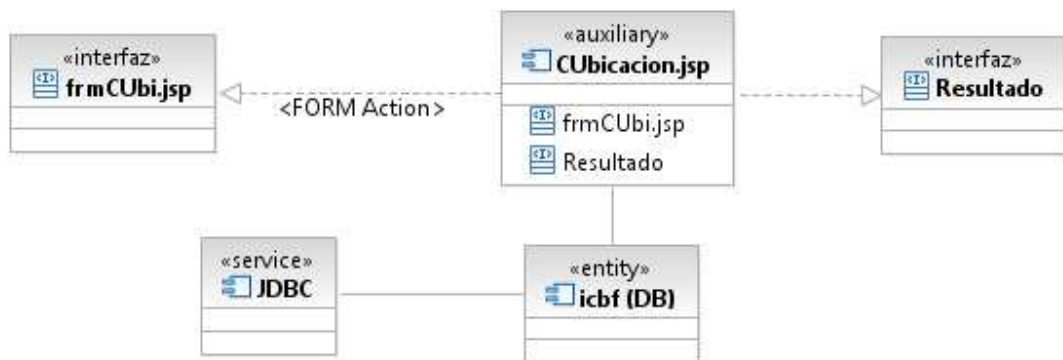


Figura 59 Diagrama de Componentes: Consulta Ubicación

Código: Consulta Ubicación

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblUbicaciones\""+ " where
        "+"\"cCodUbi\""+\"=\"+\"+cCod+\"";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append(" <table width=\"57%\" border=\"0\"> ");
            sb.append(" <tr> ");
            sb.append(" <td align=\"right\"><font size=\"2\"
            face=\"VERDANA\">Descripcion:</font></div></td> ");
            sb.append(" <td><input name=\"cNom\" type=\"text\"
            id=\"cNom2\" value=\""+rs.getString(2)+" \" size=\"26\" maxlength=\"26\"></td>
            ");
            sb.append(" </tr> ");
            sb.append(" </table> ");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print(" <p><a
            href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
            serif\">&lt;&lt;</font></a></p> ");
        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print(" <p><a
    href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
    serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de Interfaz: Modificar Ubicación



Figura 60 Diagrama de Interfaz: Modifica Ubicación

Diagrama de Componentes: Modifica Ubicación

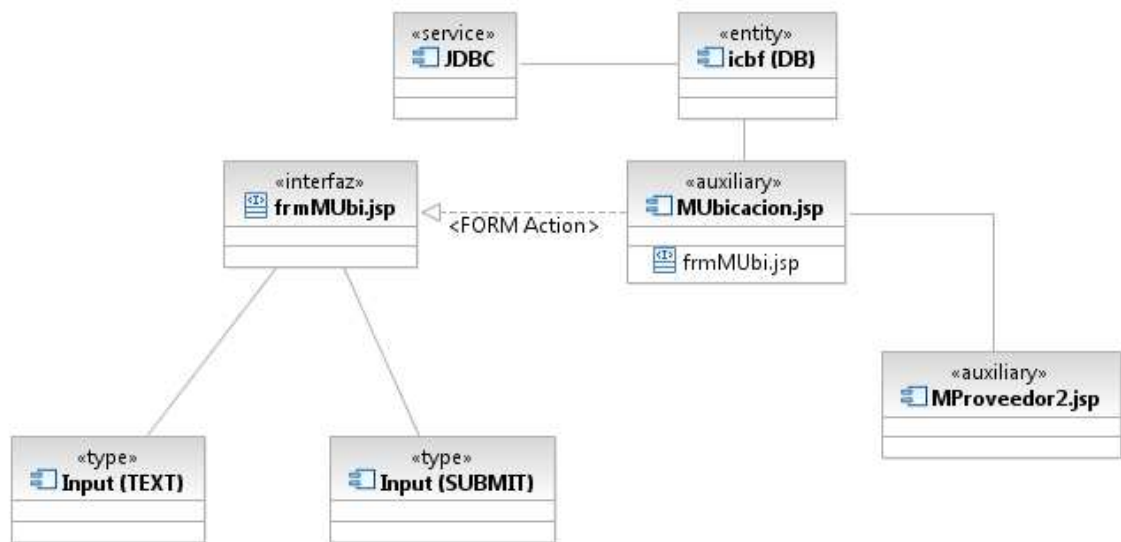


Figura 61 Diagrama de Componentes: Modifica Ubicación

Código: Modificar Ubicación

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from "+"\"tblUbicaciones\""+" where
"+"\"cCodUbi\""+\"=\""+cCod+"\"";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                                <form method=\"post\"
action=\"Modificacion.jsp\" target=\"desc\" >");
            sb.append("                                <table width=\"57%\" border=\"0\" > ");
            sb.append("                                <tr>                                ");
            sb.append("                                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Descripcion:</font></div></td>                                ");
            sb.append("                                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+"\" size=\"26\" maxlength=\"26\"></td>
");
                sb.append("                                </tr>                                ");
                sb.append("                                <tr>                                ");

                sb.append("                                <td><input name=\"cCod\" type=\"hidden\"
id=\"cCod2\" value=\""+cCod+"\" ></td>                                ");
                sb.append("                                <tr>                                <td height=\"31\" colspan=\"2\"> <div
align=\"center\">                                ");
                sb.append("                                <input name=\"ModificarUbicacion\" type=\"submit\"
target=\"desc\" value=\"Modificar Ubicacion\" width=\"91\" height=\"27\" >                                ");
                sb.append("                                </div></td> </tr>                                ");
                sb.append("                                </tr>                                ");
                sb.append("                                </table>                                ");
                sb.append("                                </form >                                ");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de Interfaz: Elimina Ubicación



Figura 62 Diagrama de Interfaz: Elimina Ubicación

Diagrama de Componentes: Elimina Ubicación

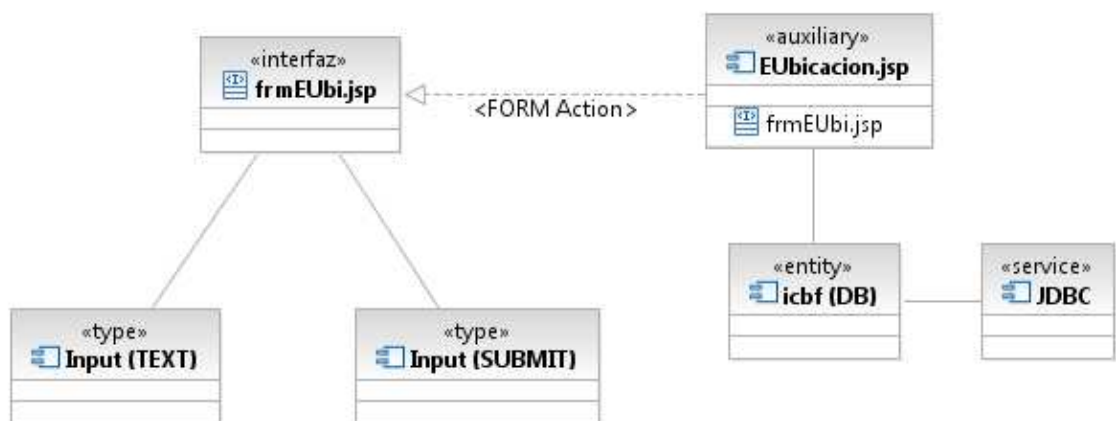


Figura 63 Diagrama de Componentes: Elimina Ubicación

Código: Eliminar Ubicación

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from "+"\"tblUbicaciones\""+
        "+"\"cCodUbi\""+="'"+cCod+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from "+"\"tblUbicaciones\""+
            "+"\"cCodUbi\""+="'"+cCod+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para eliminar </h4>");
                out.print("
                href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
                serif\">&lt;&lt;</font></a></p> ");
            }

            else{
                out.println("<h4> Informacion eliminada </h4>");
                out.print("
                href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
                serif\">&lt;&lt;</font></a></p> ");
            }

        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("
            href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
            serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("
    href=\"javascript:history.back()\"><font
    face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
}
}
```

Diagrama de Interfaz: Ingresar Asociados

The screenshot shows a Mozilla Firefox browser window displaying the website of the Fondo de Empleados Regional Caldas. The page features a header with the organization's logo and a banner for the Instituto Colombiano de Bienestar Familiar. The main content area contains a registration form titled 'Ingresar Asociados' with the following fields:

Ingresar Asociados	
Cuenta:	<input type="text"/>
Primer Apellido:	<input type="text"/>
Segundo Apellido:	<input type="text"/>
Celular:	<input type="text"/>
Correo-e:	<input type="text"/>
Cargo Actual:	<input type="text"/>
<input type="checkbox"/> Acceso WEB	Clave: <input type="text"/>
	Repetir: <input type="text"/>
<input type="button" value="OK"/>	

Figura 64 Diagrama de Componentes: Ingresar Asociados

Diagrama de Componentes: Ingresar Asociados

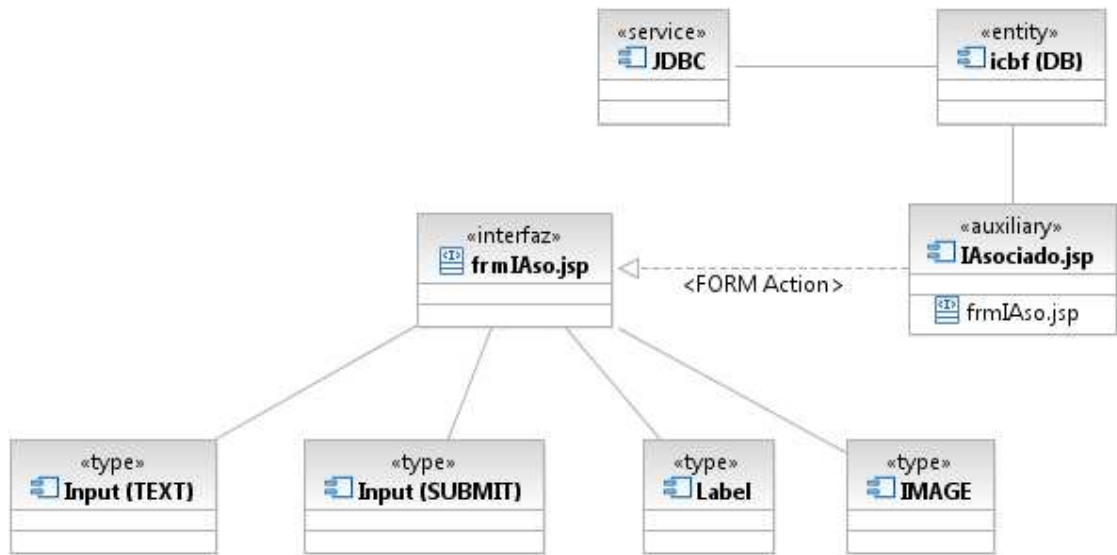


Figura 65 Diagrama de Componentes: Ingresar Asociados

Código: Ingresar Asociado

```
String Cue=request.getParameter("Cue");
String Apel=request.getParameter("Apel");
String Ape2=request.getParameter("Ape2");
String Nom=request.getParameter("Nom");
String Ced=request.getParameter("Ced");
String Tel=request.getParameter("Tel");
String Cel=request.getParameter("Cel");
String Cor=request.getParameter("Cor");
String Dir=request.getParameter("Dir");
String Car=request.getParameter("Car");
String chkWEB=request.getParameter("chkWEB");
String Cla=request.getParameter("Cla");
String Cla2=request.getParameter("Cla2");

if ((Cue != null && Cue.equals("") != true) && (Nom != null && Nom.equals("") != true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Asociado where Cuenta='"+Cue+"'";
        if (objC.verificar(sql) == 0){
            sql="insert into Asociado values ('"+Cue+"', '"+Apel+"', '"+Ape2+"', '"+Nom+"', '"+Ced+"', '"+Tel+"', '"+Cel+"', '"+Cor+"', '"+Dir+"', '"+Car+"')";
            if (objC.insertar(sql)==null){
                out.println("<h4> Datos Insertados </h4>");
                out.print("
                href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
            }
            else{
                out.println("<h2> Problemas para Registrar
                </h2>"+sql+objC.insertar(sql));
                out.print("
                href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
            }
            if (chkWEB.equals("si")){
                sql="insert into Usuarios values ('"+Cue+"', '"+Apel+"
                "+Ape2+" "+Nom+"', '"+Cla+"', '3')";
                if (objC.insertar(sql)==null){
                    out.println("<h4> Usuario WEB Registrado </h4>");
                }
                else{
                    out.println("<h2>El usuario WEB no fue
                    registrado</h2>"+sql+objC.insertar(sql));
                }
            }
            else{
                out.println("<h4> El Código ya se encuentra registrado </h4>");
                out.print("
                href=\"javascript:history.back()\"><font
                face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
            }
        }
    }
}
```

```
        catch(Exception e){out.println("Atencion: "+e);}
    }
    else{ out.println("<h4> Debera ingresar informacion completa </h4>");}
```


Diagrama de Interfaz: Consulta Asociado

The screenshot shows a Mozilla Firefox browser window displaying a web application. The page title is 'Fondo de Empleados, Instituto Colombiano de Bienestar Familiar - Manizales' and the main heading is 'Listado General de Asociados'. A circular logo for 'FONDO DE EMPLEADOS C.B.F.B. REGIONAL CALDAS' is visible on the left. The main content is a table with the following data:

Cuenta	Nombre	Cédula	Dirección	Teléfono	Estado	Correo-e
4	Catalina Aranzazu Aranzazu	123456	Bogota	09876	a	coor@gmail.com
5	Lorena Echeverry Echeverry	010101	Bogota	09876	i	coor@gmail.com

The browser's taskbar at the bottom shows several open applications, including 'Inicio', 'CATA (D...)', 'sql', 'TG-planilla_inf...', 'nit - Bloc de n...', 'Macromedia Dr...', 'pgAdmin III', 'pgAdmin III Ed...', and 'Documento sin...'. The system clock indicates the time is 11:18 AM.

Figura 66 Diagrama de Interfaz: Consulta Asociado

Diagrama de Componentes: Consulta Asociado

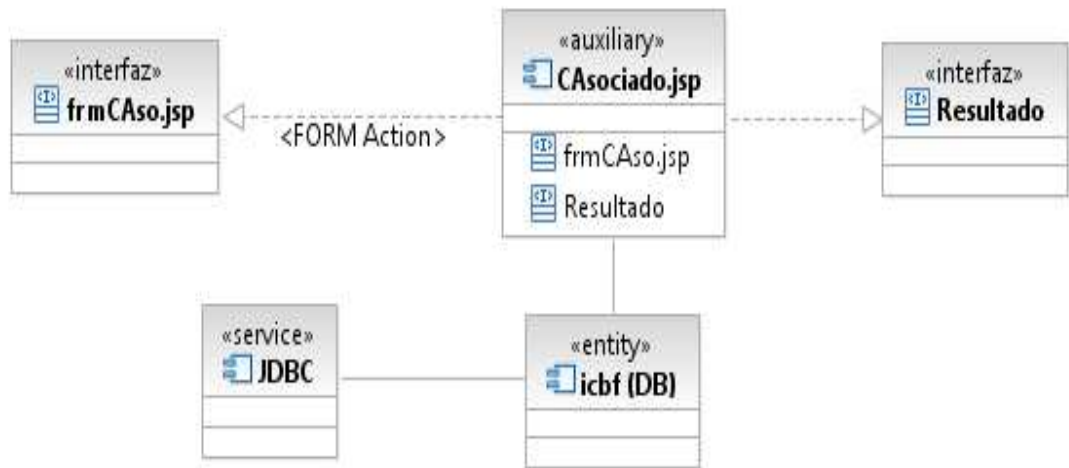


Figura 67 Diagrama de Componentes: Consulta Asociado

Código: Consulta Asociado

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Asociado where Cuenta='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                <table width=\\"57%\" border=\\"0\"> ");

            sb.append("                <tr>                ");
            sb.append("                <td><div align=\\"right\"><font size=\\"2\">
face=\\"VERDANA\">Primer Apellido:</font></div></td>                ");
            sb.append("                <td><input name=\\"cNom\" type=\\"text\"
id=\\"cNom2\" value=\\""+rs.getString(2)+" \" size=\\"26\" maxlength=\\"26\"></td>
                ");
            sb.append("                </tr>                ");

            sb.append("                <tr>                ");
            sb.append("                <td><div align=\\"right\"><font size=\\"2\">
face=\\"VERDANA\">Segundo Apellido:</font></div></td>                ");
            sb.append("                <td><input name=\\"cNom\" type=\\"text\"
id=\\"cNom2\" value=\\""+rs.getString(3)+" \" size=\\"26\" maxlength=\\"26\"></td>
                ");
            sb.append("                </tr>                ");

            sb.append("                <tr>                ");
            sb.append("                <td><div align=\\"right\"><font size=\\"2\">
face=\\"VERDANA\">Nombre:</font></div></td>                ");
            sb.append("                <td><input name=\\"cNom\" type=\\"text\"
id=\\"cNom2\" value=\\""+rs.getString(4)+" \" size=\\"26\" maxlength=\\"26\"></td>
                ");
            sb.append("                </tr>                ");

            sb.append("                <tr>                ");
            sb.append("                <td><div align=\\"right\"><font size=\\"2\">
face=\\"VERDANA\">Cedula:</font></div></td>                ");
            sb.append("                <td><input name=\\"cNom\" type=\\"text\"
id=\\"cNom2\" value=\\""+rs.getString(5)+" \" size=\\"26\" maxlength=\\"26\"></td>
                ");
            sb.append("                </tr>                ");

            sb.append("                <tr>                ");
            sb.append("                <td><div align=\\"right\"><font size=\\"2\">
face=\\"VERDANA\">Telefono:</font></div></td>                ");
            sb.append("                <td><input name=\\"cNom\" type=\\"text\"
id=\\"cNom2\" value=\\""+rs.getString(6)+" \" size=\\"26\" maxlength=\\"26\"></td>
                ");
        }
    }
}
```

```

        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Celular:</font></div></td>                ");
        sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(7)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">E-Mail:</font></div></td>                ");
        sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(8)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Direccion:</font></div></td>                ");
        sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(9)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Cargo Actual:</font></div></td>                ");
        sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(10)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("        </table>                ");
        sb.append("        <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>                ");
        out.println(sb);
    }
    else{
        out.println("<h4> Código no encontrado </h4>");
        out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                ");
    }
    rs.close();
}
catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                ");
}
}

```

Diagrama de Interfaz: Modifica Asociado

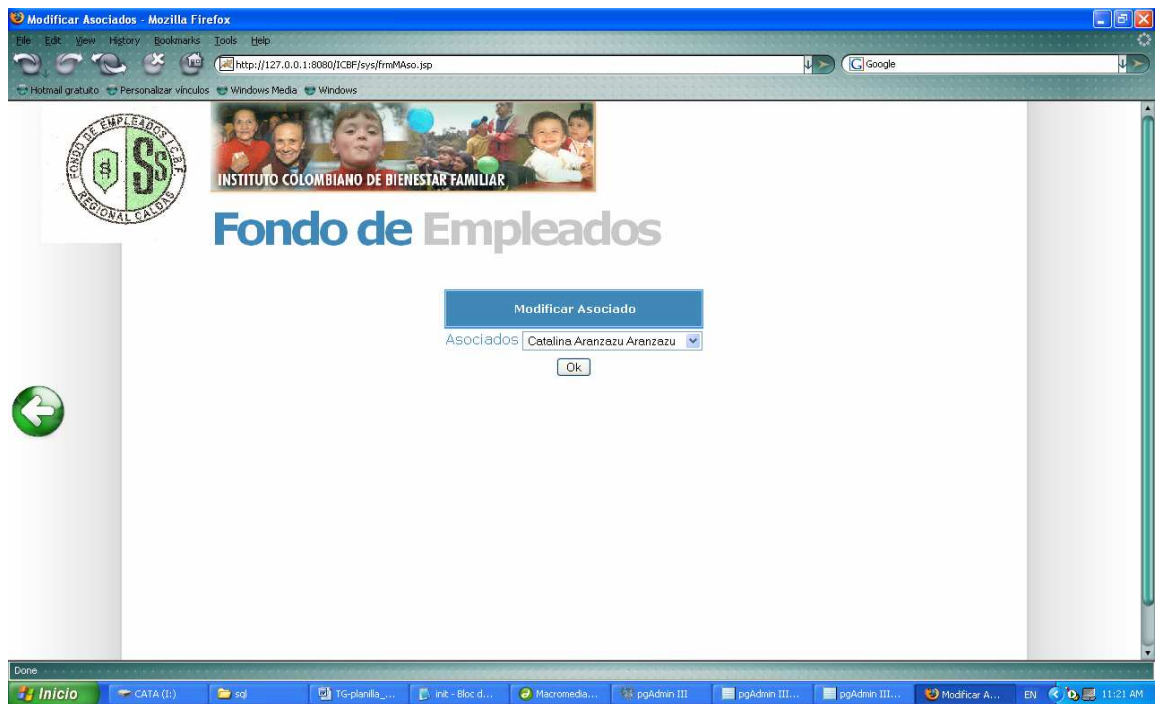


Figura 68 Diagrama de Componentes: Modificar Asociado

Diagrama de Componentes: Modifica Asociado

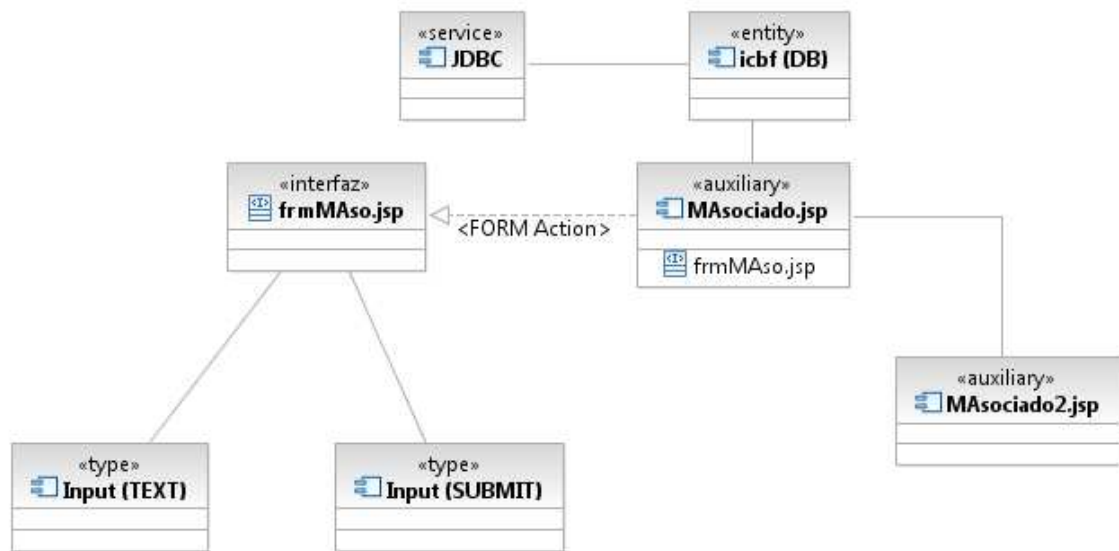


Figura 69 Diagrama de Componentes: Modifica Asociado

Código: Modifica Asociado

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Asociado where Cuenta='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                                <form method=\"post\"
action=\"Modificacion.jsp\" target=\"desc\" >");
            sb.append("                                <table width=\"57%\" border=\"0\" > ");

            sb.append("                                <tr>                                ");
            sb.append("                                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Primer Apellido:</font></div></td>                                ");
            sb.append("                                <td><input name=\"Ape1\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+"\" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                                </tr>                                ");

            sb.append("                                <tr>                                ");
            sb.append("                                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Segundo Apellido:</font></div></td>                                ");
            sb.append("                                <td><input name=\"Ape2\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(3)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                                </tr>                                ");

            sb.append("                                <tr>                                ");
            sb.append("                                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Nombre:</font></div></td>                                ");
            sb.append("                                <td><input name=\"Nom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(4)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                                </tr>                                ");

            sb.append("                                <tr>                                ");
            sb.append("                                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Cedula:</font></div></td>                                ");
            sb.append("                                <td><input name=\"Ced\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(5)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                                </tr>                                ");

            sb.append("                                <tr>                                ");
            sb.append("                                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Telefono:</font></div></td>                                ");
```

```

        sb.append("                <td><input name=\"Tel\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(6)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Celular:</font></div></td>                ");
        sb.append("                <td><input name=\"Cel\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(7)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">E-Mail:</font></div></td>                ");
        sb.append("                <td><input name=\"Cor\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(8)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Direccion:</font></div></td>                ");
        sb.append("                <td><input name=\"Dir\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(9)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Cargo Actual:</font></div></td>                ");
        sb.append("                <td><input name=\"Car\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(10)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("        </table>        ");
        sb.append("        <tr>                ");
        sb.append("                <td><input name=\"cCod\" type=\"hidden\"
id=\"cCod2\" value=\""+cCod+"\" ></td>                ");
        sb.append("        <tr>                <td height=\"31\" colspan=\"2\"> <div
align=\"center\">                ");
        sb.append("                <input name=\"ModificarAsociado\" type=\"submit\"
target=\"desc\" value=\"Modificar Asociado\" width=\"91\" height=\"27\" >                ");
        sb.append("                </div></td> </tr>                ");
        sb.append("        </tr>                ");
        sb.append("        </table>                ");
        sb.append("        </form >                ");
        sb.append("        <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>                ");

out.println(sb);
}
else{
    out.println("<h4> Código no encontrado </h4>");
}

```



```

        out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
    }
    rs.close();
}
catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}

```

Diagrama de Interfaz: Elimina Asociado

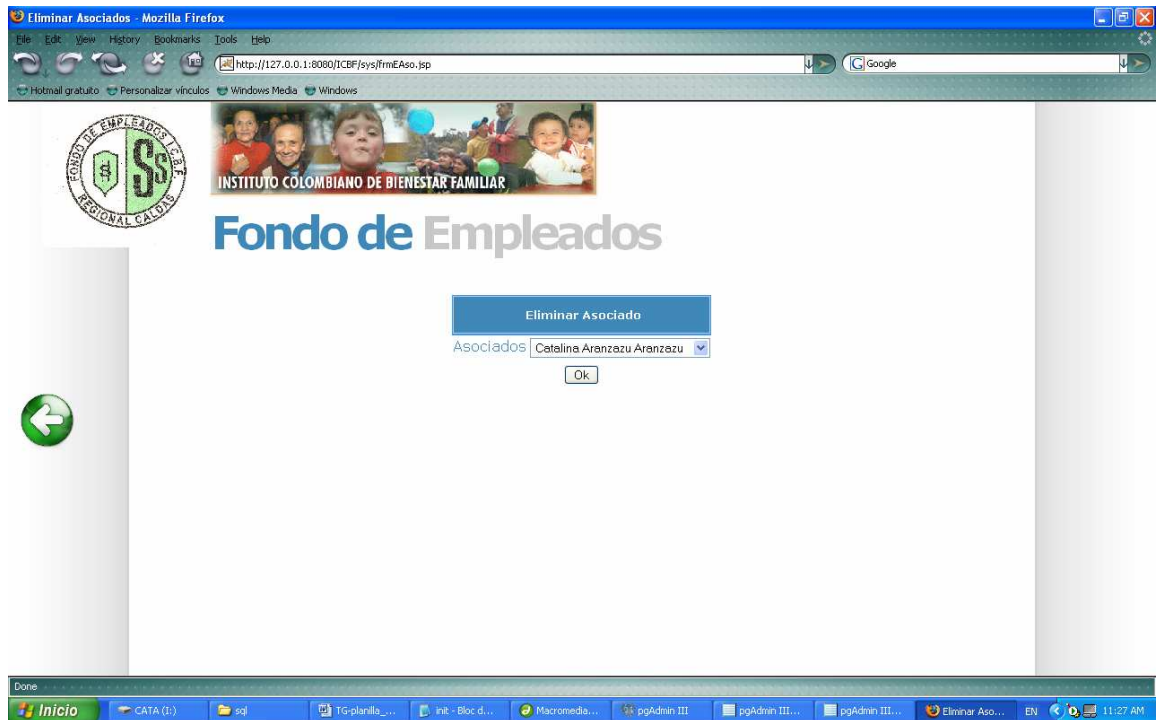


Figura 70 Diagrama de Interfaz: Elimina Asociado

Diagrama de Componentes: Elimina Asociado

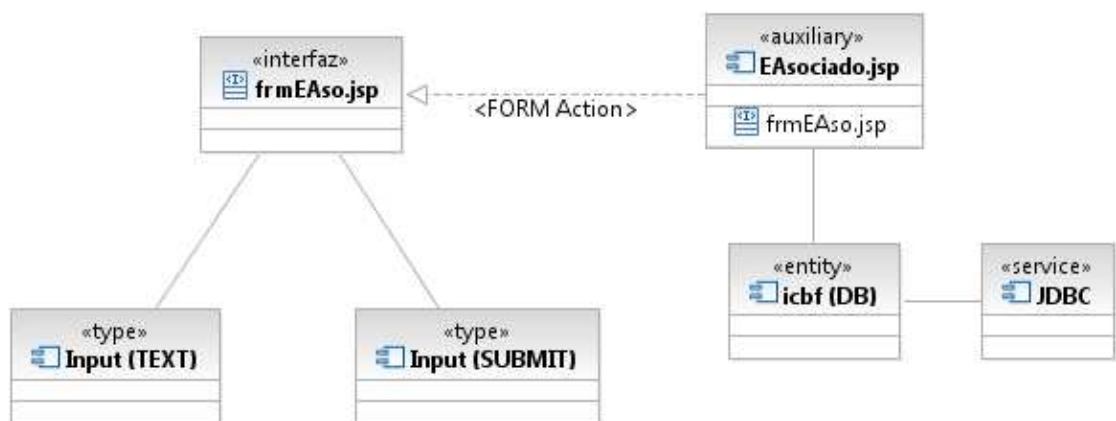


Figura 71 Diagrama de Componentes: Elimina Asociado

Código: Elimina Asociado

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from Asociado where Cuenta='"+cCod+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from Asociado where Cuenta='"+cCod+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para eliminar </h4>");
                out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
            else{
                out.println("<h4> Informacion eliminada </h4>");
                out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("
href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
}
}
```

Diagrama de interfaz: Ingresar Plazos

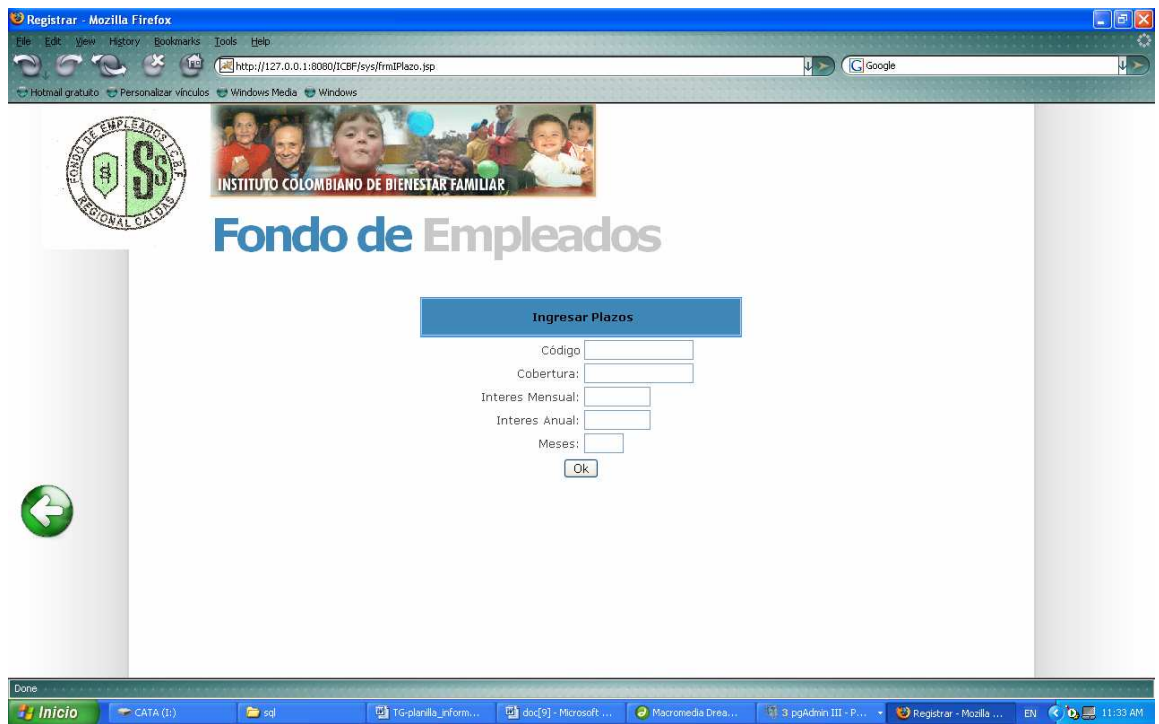


Figura 72 Diagrama de Interfaz: Ingresar Plazos

Diagrama de Componentes: Ingresar Plazos

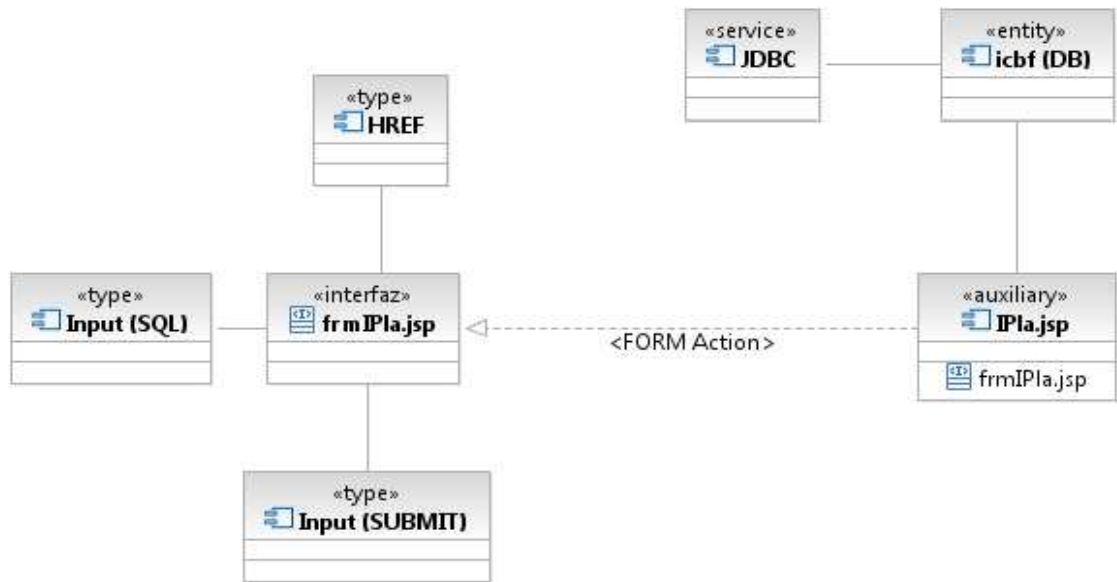


Figura 73 Diagrama de Componentes: Menú Procesador

Código: Ingresar Plazos

```
String cCod=request.getParameter("cCod");
String cNom=request.getParameter("cNom");
String intMensual=request.getParameter("intMensual");
String intAnual=request.getParameter("intAnual");
String Mes=request.getParameter("Mes");

if ((cCod != null && cCod.equals("") != true) && (cNom != null && cNom.equals("")
!= true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4><center>Problemas en
conexion</center></h4>"+x);}
        String sql="select * from Plazos where idPlazos='"+cCod+"'";
        if (objC.verificar(sql) == 0){
            sql="insert into Plazos values
('"+cCod+"', '"+cNom+"', '"+intMensual+"', '"+intAnual+"', '"+Mes+"')";
            if (objC.insertar(sql)==null){
                out.println("<h4>Datos Insertados</h4>");
                out.print("<p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>");
            }
        }
        else{
            out.println("<h2><center>Problemas para
Registrar</center></h2>"+sql+objC.insertar(sql));
            out.print("<p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>");
        }
    }
    else{
        out.println("<h4>El Código ya se encuentra
registrado</center></h4>");
        out.print("<p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>");
    }
}
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4>Debera ingresar informacion completa</center></h4>");
    out.print("<center><p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>");
}
}
```

Diagrama de interfaz: Consultar Plazos

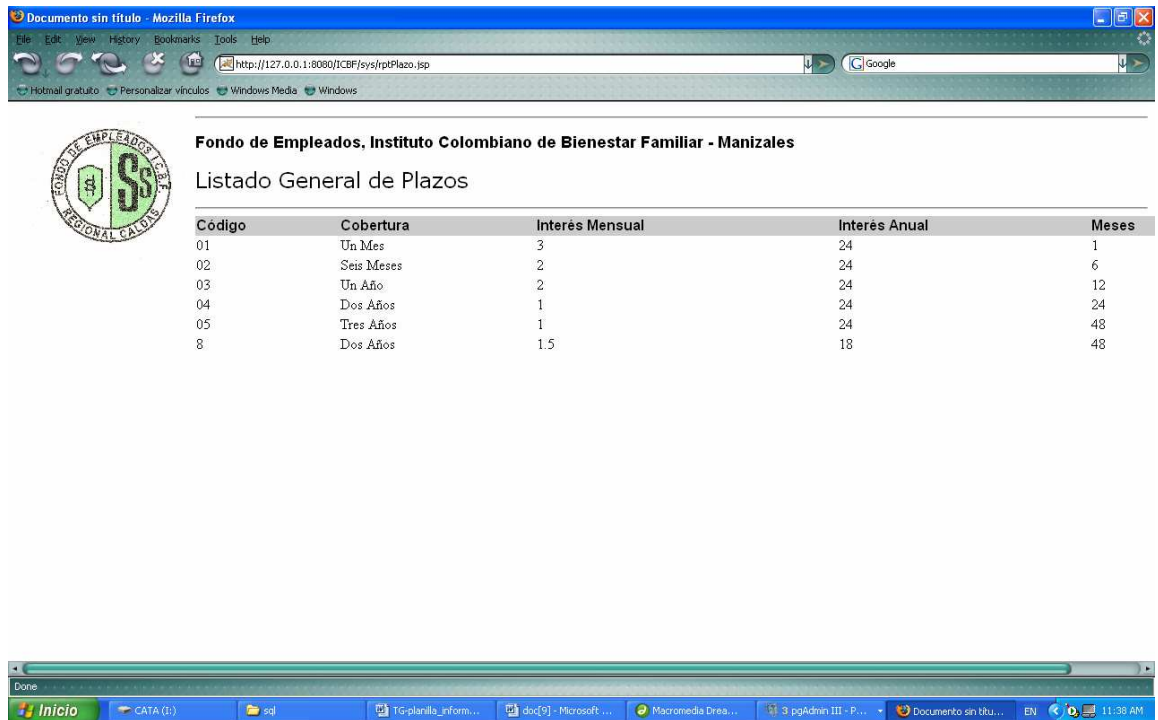


Figura 74 Diagrama de Interfaz: Consultar Plazos

Diagrama de Componentes: Consultar Plazos

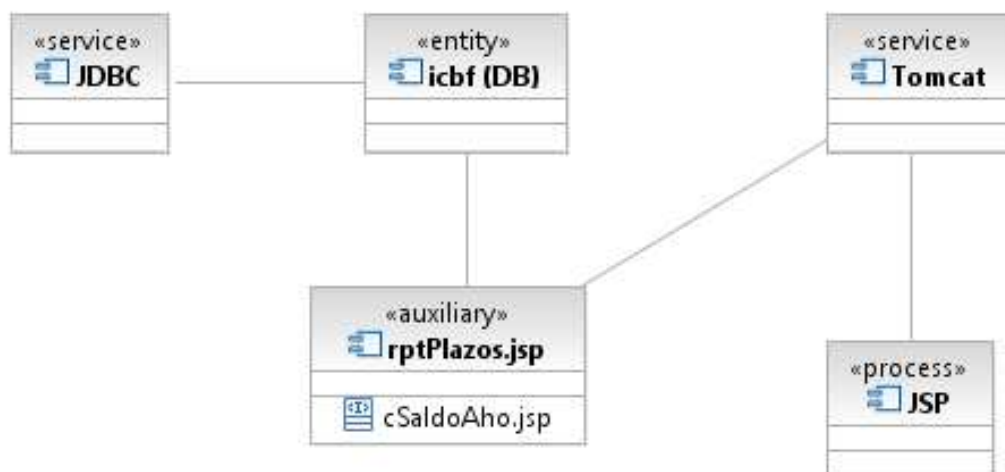


Figura 75 Diagrama de Componentes: Consultar Plazos

Código: Consultar Plazos

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Plazos where idPlazos='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                <table width=\"57%\" border=\"0\"> ");
            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Cobertura:</font></div></td>                                ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>                                ");
            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">int Mensual:</font></div></td>                                ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(3)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">int Anual:</font></div></td>                                ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(4)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>                                ");

            sb.append("                </table>                                ");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                ");
        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                ");
}
```


Diagrama de interfaz: Modificar Plazos

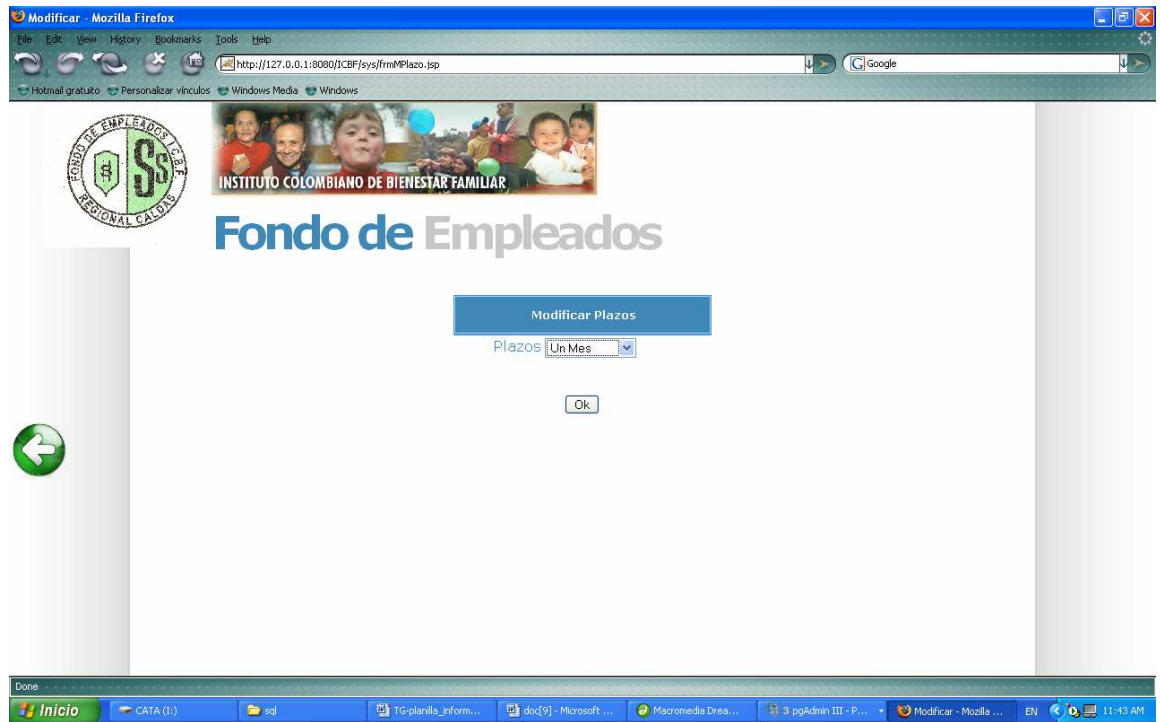


Figura 76 Diagrama de Interfaz: Modificar Plazos

Diagrama de Componentes: Modificar Plazos

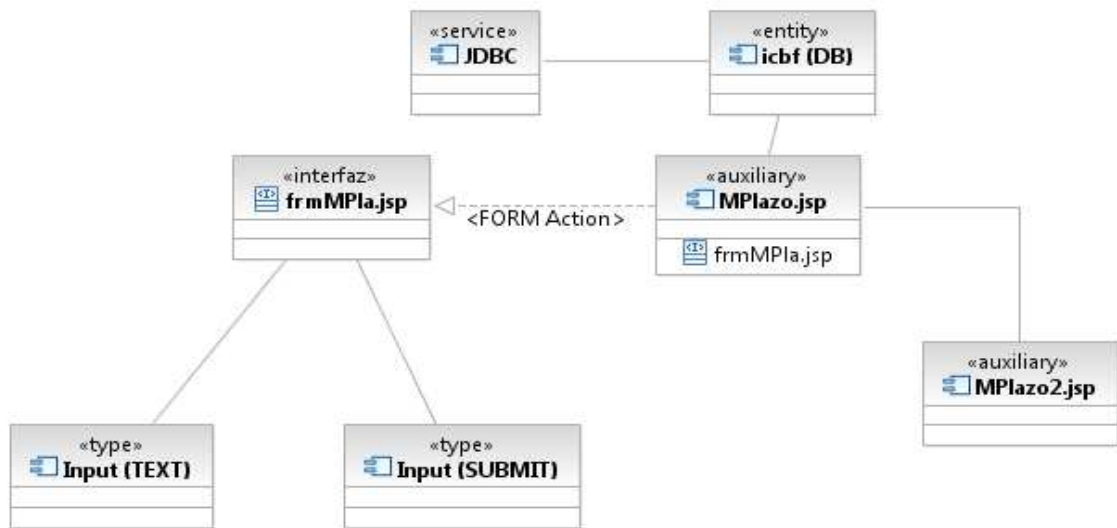


Figura 77 Diagrama de Componentes: Modificar Plazos

Código: Modificar Plazos

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Plazos where idPlazos='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            %>

Cobertura:</div></td>
        <td><input name="Cob" type="text" id="Cob"
value="<%=rs.getString(2)%>"></td>
        <td>&nbsp;</td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td><div align="right">Interes Mensual:</div></td>
        <td><input name="IntM" type="text" id="IntM"
value="<%=rs.getString(3)%>"></td>
        <td>&nbsp;</td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td><div align="right">Interes Anual:</div></td>
        <td><input name="IntA" type="text" id="IntA" value="<%=rs.getString(4)%>"
        <input name="cCod" type="hidden" value="<%=cCod%>"></td>
        <td>&nbsp;</td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td colspan="2">
            <div align="center">
                <input name="modPlazo" type="submit" id="modPlazo" value="Modificar
Plazo">
            </div></td>
        <td>&nbsp;</td>
    </tr>
</table>
</form >
<br>
<br>
<%
    }
    else{
        out.println("<h4> Código no encontrado </h4>");
        out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
    }
}
```

```
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de Interfaz: Eliminar Plazos

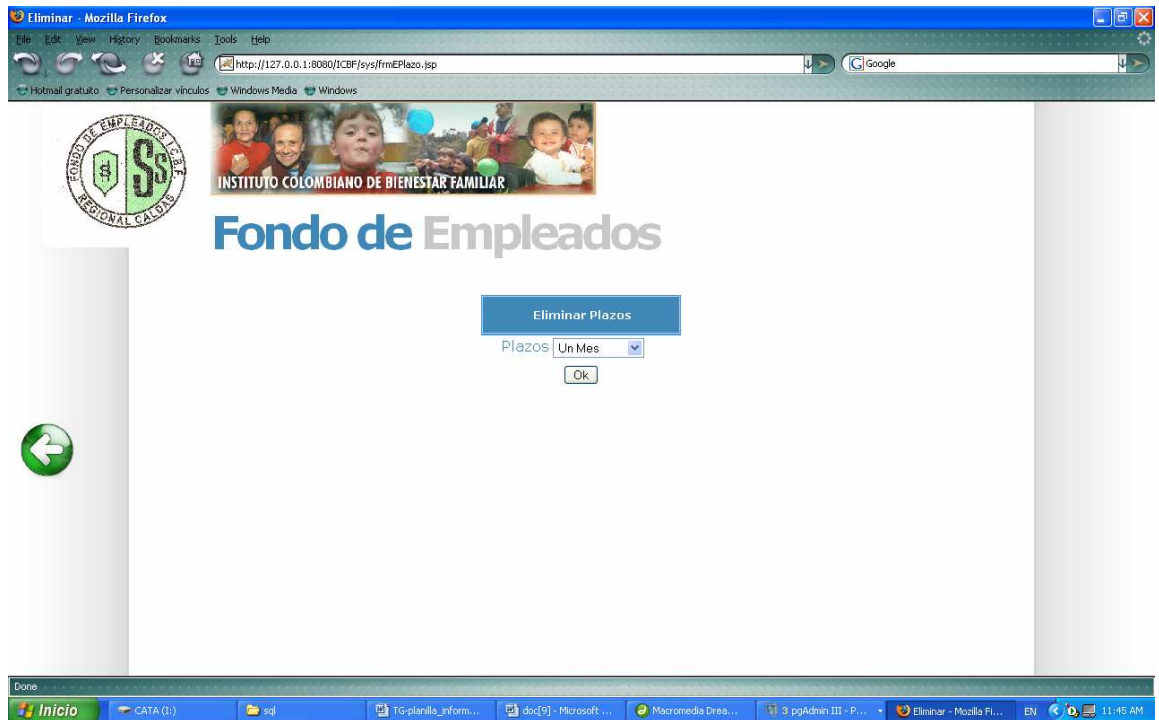


Figura 78 Diagrama de Interfaz: Eliminar Plazos

Diagrama de Componentes: Eliminar Plazos

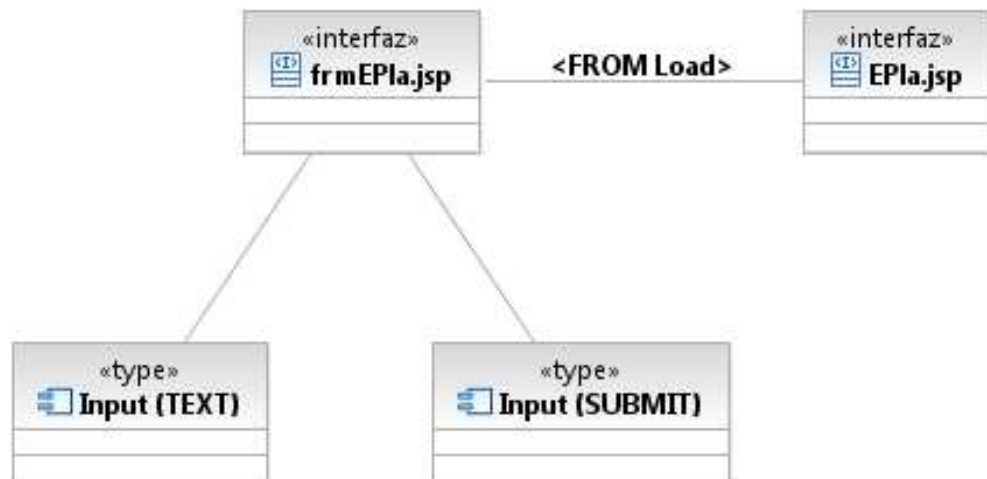


Figura 79 Diagrama de Interfaz: Eliminar Plazos

Código: Eliminar Plazos

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from Plazos where idPlazos='"+cCod+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from Plazos where idPlazos='"+cCod+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para eliminar </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }

            else{
                out.println("<h4> Informacion eliminada </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }

        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print(" <p><a href=\"javascript:history.back()\"><font face=\"Verdana,
Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de Interfaz: Ingresar Clases Créditos

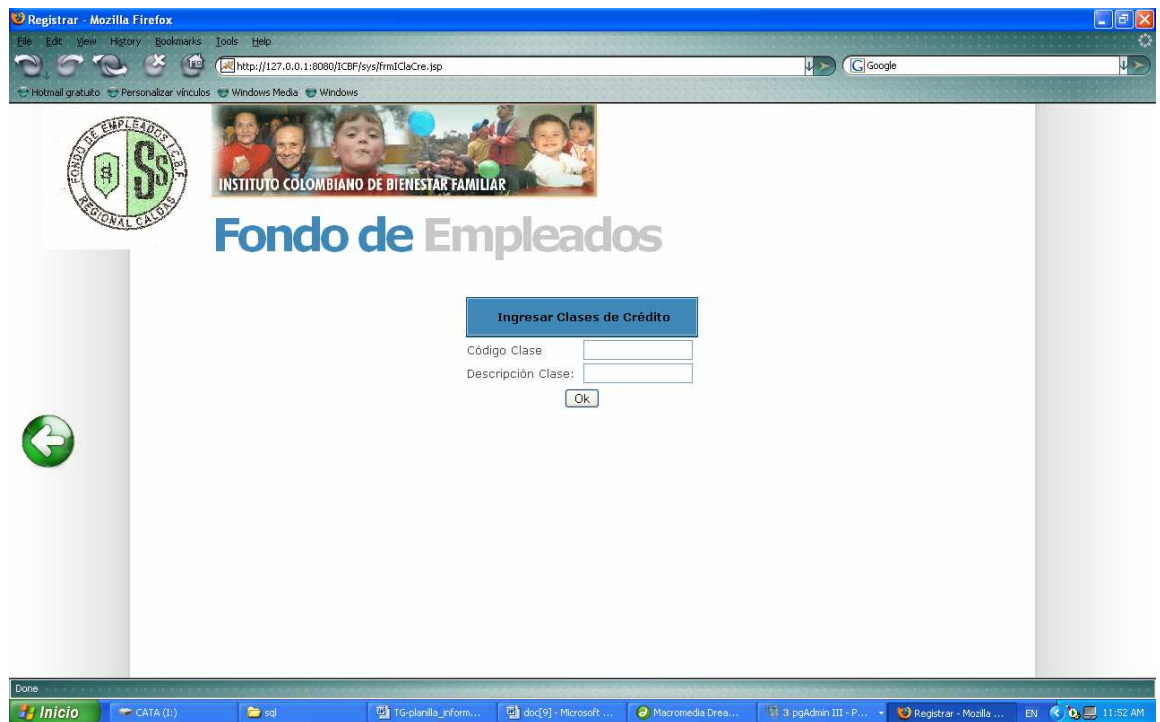


Figura 80 Diagrama de Interfaz: Ingresar Clases Creditos

Diagrama de Componentes: Ingresar Clases Créditos

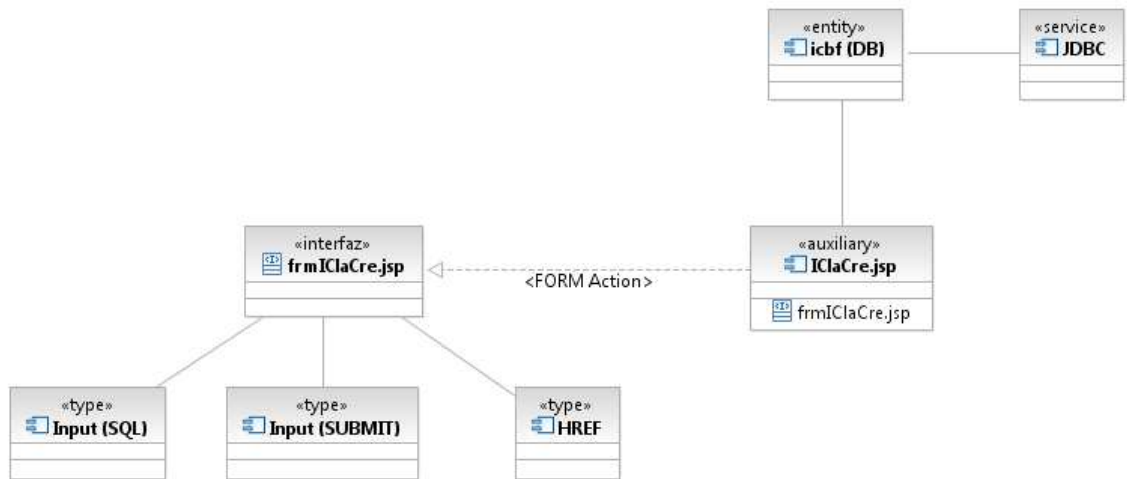


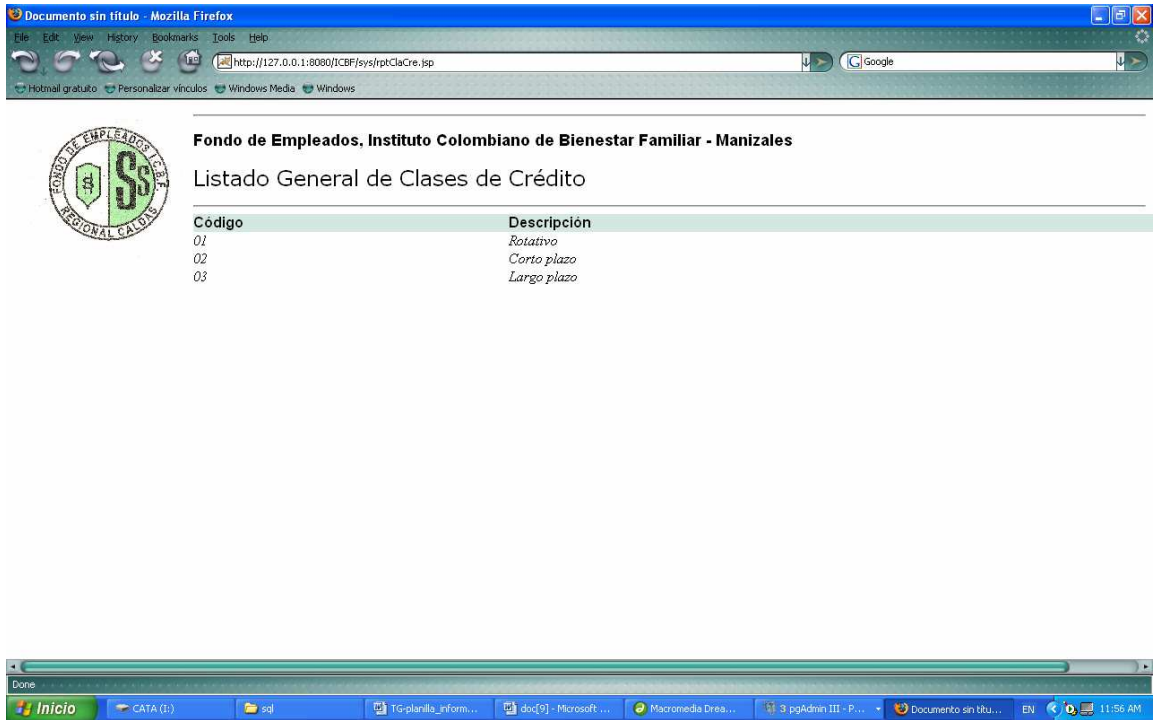
Figura 81 Diagrama de Componentes: Ingresar Clases Creditos

Código: Ingresar Clases Créditos

```
String cCod=request.getParameter("cCod");
String cNom=request.getParameter("cNom");

if ((cCod != null && cCod.equals("") != true) && (cNom != null && cNom.equals("")
!= true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4><center>Problemas en
conexion</center></h4>"+x);}
        String sql="select * from ClaseCreditos where
idClaseCreditos='"+cCod+"'";
        if (objC.verificar(sql) == 0){
            sql="insert into ClaseCreditos values ('"+cCod+"', '"+cNom+"')";
            if (objC.insertar(sql)==null){
                out.println("<h4>Datos Insertados</h4>");
                out.print("<p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>");
            }
            else{
                out.println("<h2><center>Problemas para
Registrar</center></h2>"+sql+objC.insertar(sql));
                out.print("<p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>");
            }
        }
        else{
            out.println("<h4>El Código ya se encuentra
registrado</center></h4>");
            out.print("<p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>");
        }
        catch(Exception e){out.println("Atencion: "+e);}
    }
    else{
        out.println("<h4>Debera ingresar informacion completa</center></h4>");
        out.print("<center><p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>");
    }
}
```

Diagrama de Interfaz: Consultar Clases Credito



Documento sin título - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://127.0.0.1:8080/fcebf/sys/rptClaCre.jsp

Hotmail gratuito Personalizar vínculos Windows Media Windows

Fondo de Empleados, Instituto Colombiano de Bienestar Familiar - Manizales

Listado General de Clases de Crédito

Código	Descripción
01	Rotativo
02	Corto plazo
03	Largo plazo

Done

Inicio CATA (I:) sql TG-planilla_inform... doc[9] - Microsoft... Macromedia Drea... 3 pgAdmin III - P... Documento sin títu... EN 11:56 AM

Figura 82 Diagrama de Interfaz: Consultar Clases Creditos

Diagrama de Componentes: Consultar Clases Créditos

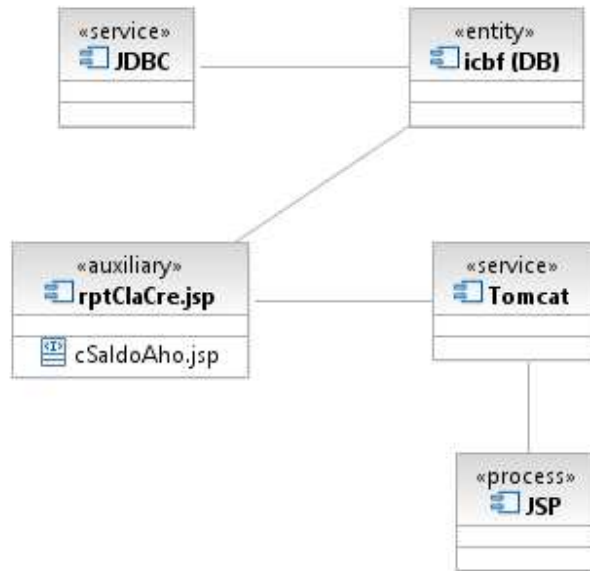


Figura 83 Diagrama de Componentes: Consultar Clases Creditos

Código: Consultar Clases Creditos

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from ClaseCreditos where
idClaseCreditos='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                <table width=\"57%\" border=\"0\"> ");
            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Descripcion:</font></div></td>    ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+" \" size=\"26\" maxlength=\"26\"></td>
");
                sb.append("                </tr>                                ");
            sb.append("                </table>                ");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                ");
        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                ");
}
```

Diagrama de Interfaz: Modificar Clases Créditos



Figura 84 Diagrama de Interfaz: Modificar Clases Crédito

Diagrama de Componentes: Modificar Clases Credito

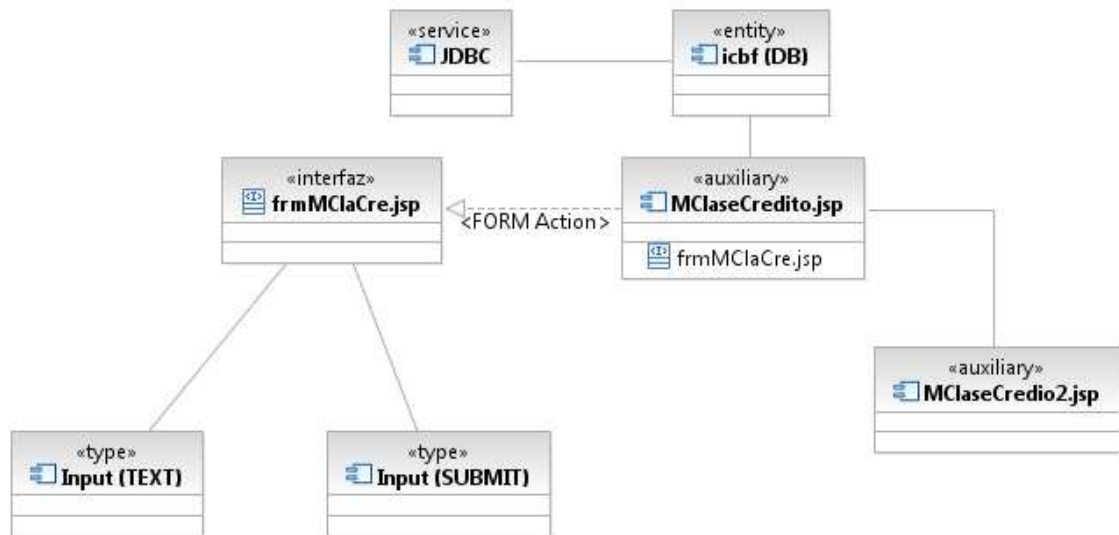


Figura 85 Diagrama de Componentes: Modificar Clases Crédito

Código: Modificar Clases Credito

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from ClaseCreditos where
idClaseCreditos='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            %>
                Nombre:</div></td>
                <td><input name="Nom" type="text" id="Nom"
value="<%=rs.getString(2)%>">
                <input name="cCod" type="hidden" value="<%=cCod%>"></td>
                <td>&nbsp;</td>
            </tr>

            <tr>
                <td>&nbsp;</td>
                <td colspan="2"><div align="center">
                    <input name="modClase" type="submit" id="modClase"
value="Modificar Clase">
                    </div></td>
                <td>&nbsp;</td>
            </tr>
        </table>
        <%
            }
            else{
                out.println("<h4> Código no encontrado </h4>");
                out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
            rs.close();
        }
        catch(Exception e){out.println("Atencion: "+e);}
    }
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de Interfaz: Eliminar Clases Credito

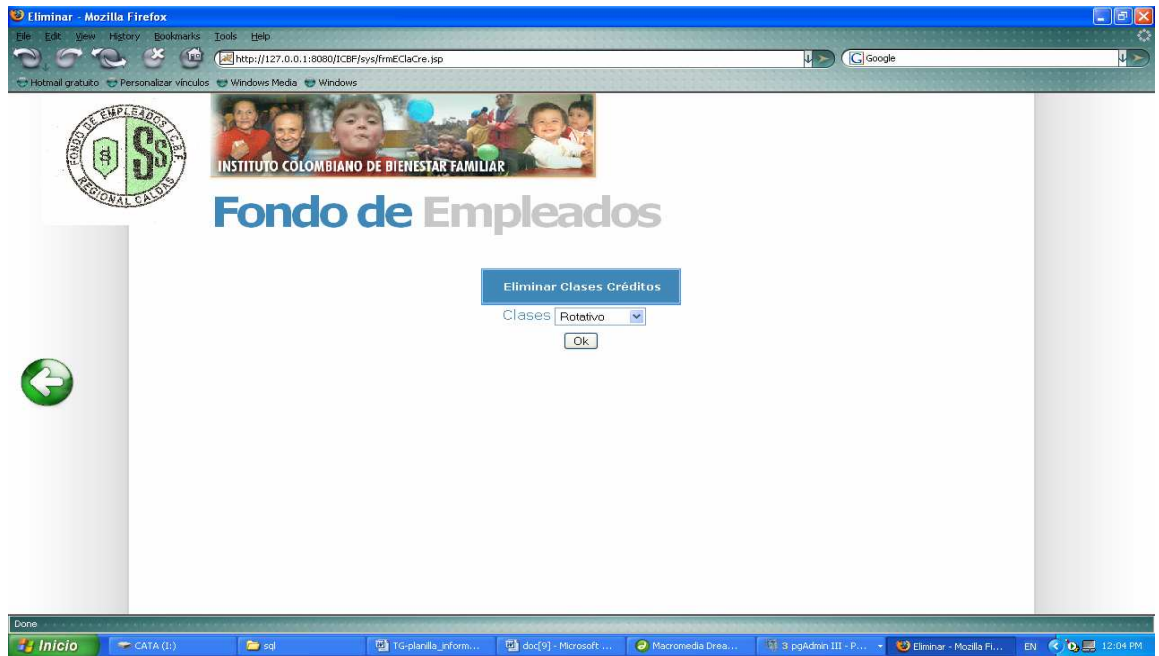


Figura 86 Diagrama de Interfaz: Eliminar Clases Crédito

Diagrama de Componentes: Eliminar Clases Credito

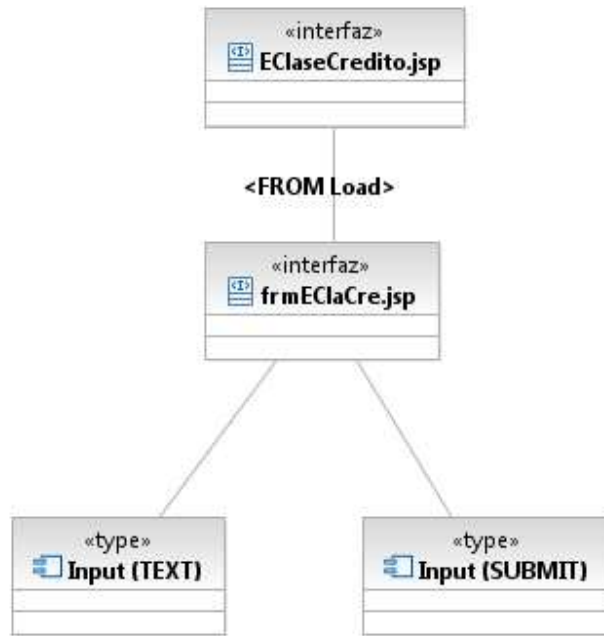


Figura 87 Diagrama de Componentes: Eliminar Clases Crédito

Código: Eliminar Clases Credito

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from ClaseCredito where
idClaseCredito='"+cCod+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from ClaseCredito where idClaseCredito='"+cCod+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para eliminar </h4>");
                out.print("
href=\"javascript:history.back()\"<font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }

            else{
                out.println("<h4> Informacion eliminada </h4>");
                out.print("
href=\"javascript:history.back()\"<font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }

        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("
href=\"javascript:history.back()\"<font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("
href=\"javascript:history.back()\"<font face=\"Verdana,
Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de Interfaz: Ingresar Actividades

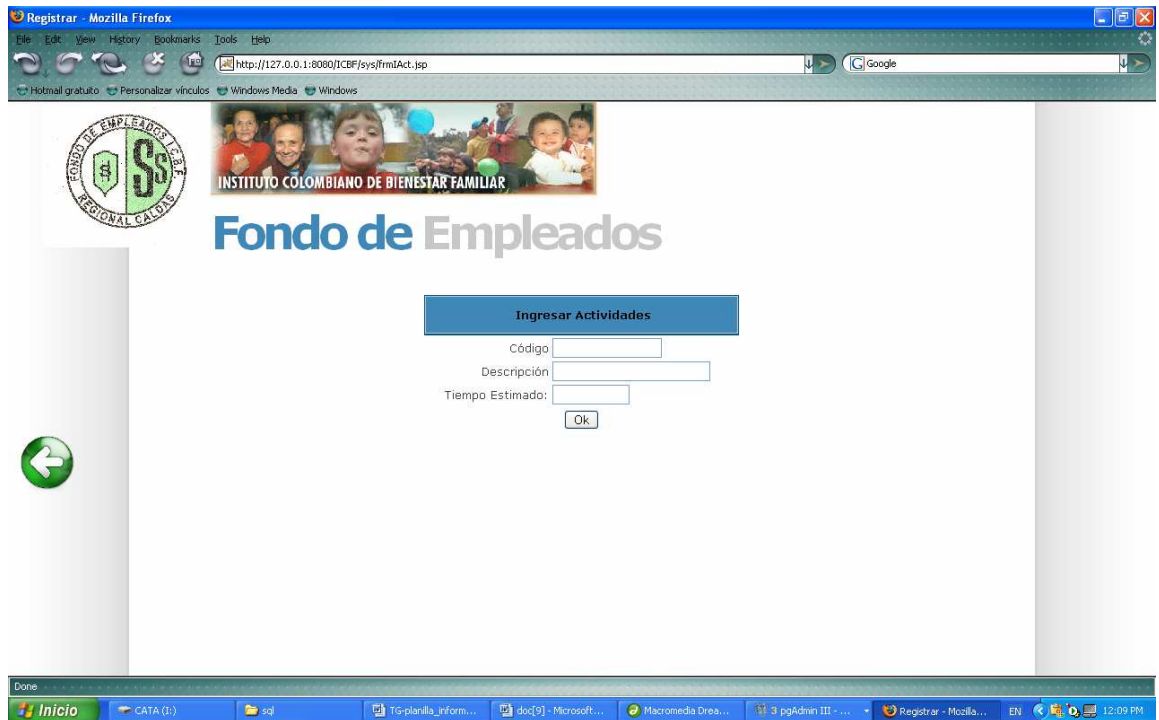


Figura 88 Diagrama de Interfaz: Ingresar Actividades

Diagrama de Componentes: Ingresar Actividades

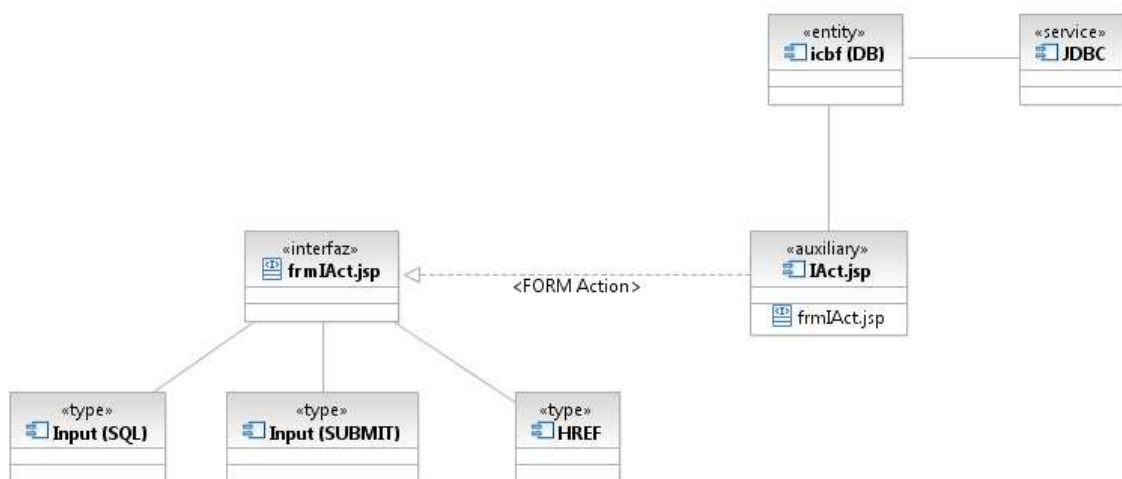


Figura 89 Diagrama de Componentes: Ingresar Actividades

Código: Ingresar Actividades

```
String cCod=request.getParameter("cCod");
String cNom=request.getParameter("cNom");
String Tiempo=request.getParameter("Tiempo");

if ((cCod != null && cCod.equals("") != true) && (cNom != null && cNom.equals("")
!= true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4><center>Problemas en
conexion</center></h4>"+x);}
        String sql="select * from Actividades where idActividad='"+cCod+"'";
        if (objC.verificar(sql) == 0){
            sql="insert into Actividades values
('"+cCod+"','"+cNom+"','"+Tiempo+"')";
            if (objC.insertar(sql)==null){
                out.println("<h4>Datos Insertados</h4>");
                out.print("<p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>");
            }
            else{
                out.println("<h2><center>Problemas para
Registrar</center></h2>"+sql+objC.insertar(sql));
                out.print("<p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>");
            }
        }
        else{
            out.println("<h4>El Código ya se encuentra
registrado</center></h4>");
            out.print("<p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4>Debera ingresar informacion completa</center></h4>");
    out.print("<center><p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>");
}
}
```

Diagrama de interfaz: Consultar Actividades

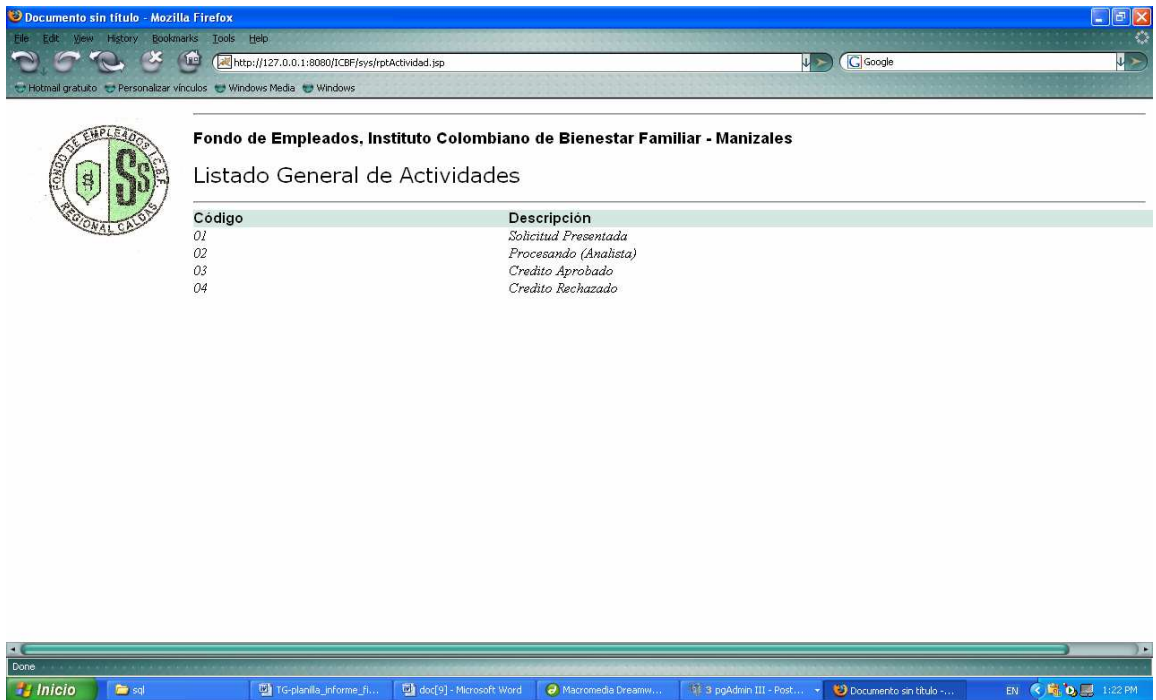


Figura 90 Diagrama de Componentes: Consultar Actividades

Diagrama de Componentes: Consultar Actividades

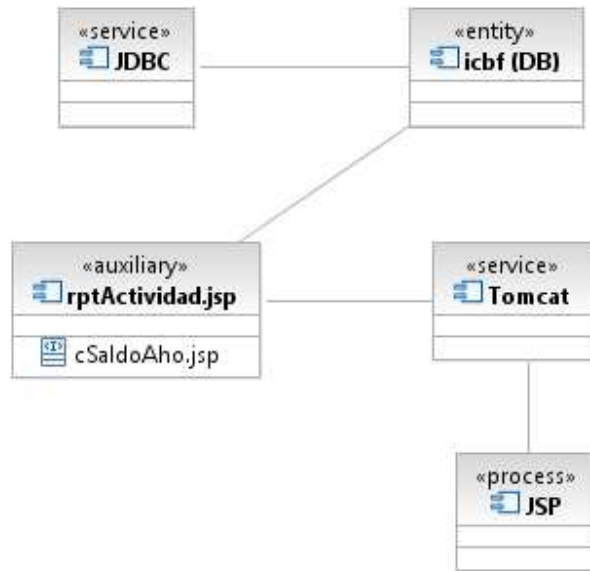


Figura 91 Diagrama de Componentes: Consultar Actividades

Código: Consultar Actividades

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Actividades where idActividad='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                <table width=\"57%\" border=\"0\"> ");
            sb.append("                <tr>");
            sb.append("                <td><div align=\"right\"><font size=\"2\">
face=\"VERDANA\">Descripcion:</font></div></td>");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+" \" size=\"26\" maxlength=\"26\"></td>
");
                sb.append("                </tr>");
            sb.append("                </table>");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>");
        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;&lt;/font></a></p>");
}
```

Diagrama de Interfaz: Modificar Actividades

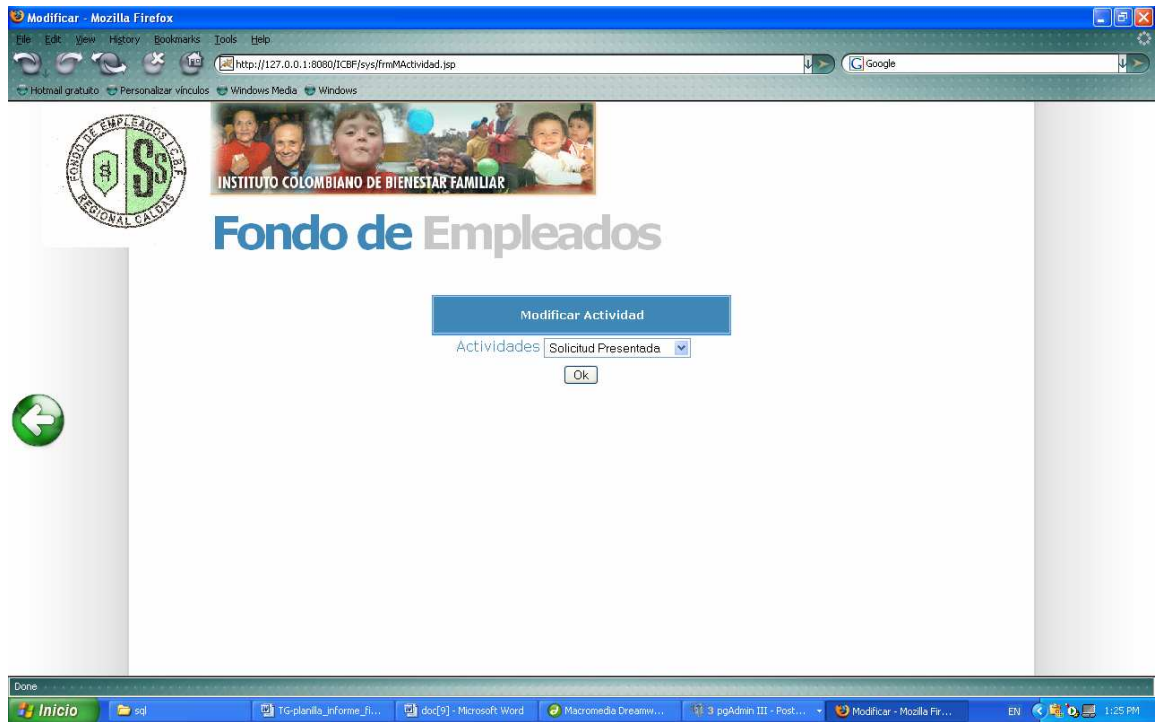


Figura 92 Diagrama de Interfaz: Modificar Actividades

Diagrama de Componentes: Modificar Actividades

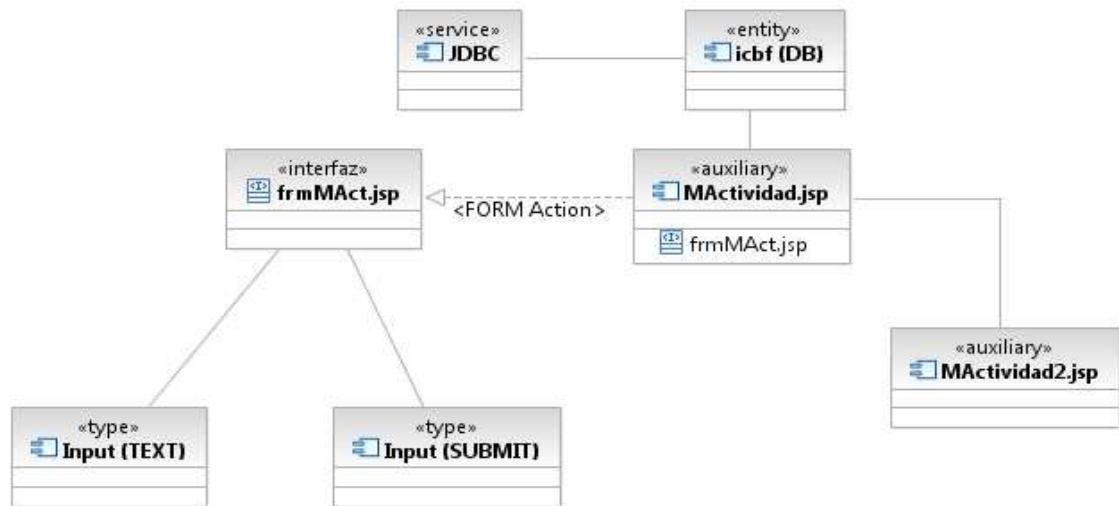


Figura 93 Diagrama de Componentes: Modificar Actividades

Código: Modificar Actividades

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
%><input name="cCod" type="hidden" value="<%=cCod%>"><%
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Actividades where idActividad='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("        <form method=\"post\" action=\"Modificacion.jsp\"
target\"desc\" >");
            sb.append("        <center>        <table width=\"57%\" border=\"0\" > ");
            sb.append("        <tr>                                ");
            sb.append("        <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Nombre:</font></div></td>        ");
            sb.append("        <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\"\"+rs.getString(2)+\"\" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("        </tr>                                ");
            sb.append("        <tr>                                ");

            sb.append("        <td><input name=\"cCod\" type=\"hidden\"
id=\"cCod2\" value=\"\"+cCod+\"\" ></td>        ");
            sb.append("        <tr>        <td height=\"31\" colspan=\"2\"> <div
align=\"center\">        ");
            sb.append("        <input name=\"ModificarAct\" type=\"submit\"
target=\"desc\" value=\"Modificar Actividad\" width=\"91\" height=\"27\" >        ");
            sb.append("        </div></td> </tr>                                ");
            sb.append("        </tr>                                ");
            sb.append("        </table> </center>        ");
            sb.append("        </form >                                ");
            out.println(sb);
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
        rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de interfaz: Eliminar Actividades



Figura 94 Diagrama de Interfaz: Eliminar Actividades

Diagrama de Componentes: Eliminar Actividades

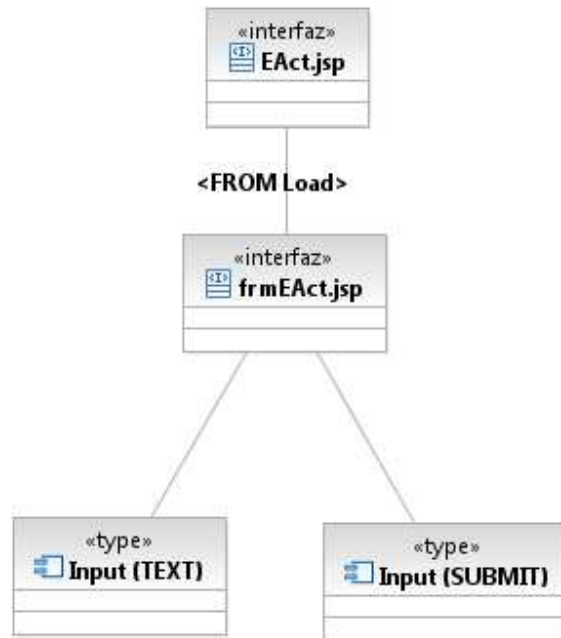


Figura 95 Diagrama de Componentes: Consultar Actividades

Código: Eliminar Actividades

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from Actividades where idActividad='"+cCod+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from Actividades where idActividad='"+cCod+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para eliminar </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }

            else{
                out.println("<h4> Informacion eliminada </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }

        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print(" <p><a href=\"javascript:history.back()\"><font face=\"Verdana,
Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
}
```

Diagrama de Interfaz: Ingresar Analista

The screenshot shows a Mozilla Firefox browser window displaying the website of the Fondo de Empleados Regional Caldas. The page features a logo on the left and a banner with children's faces at the top. The main content is a registration form titled 'Ingresar Analistas' with the following fields:

Ingresar Analistas	
Código:	<input type="text"/>
Primer Apellido:	<input type="text"/>
Segundo Apellido:	<input type="text"/>
Celular:	<input type="text"/>
Correo-e:	<input type="text"/>
Cargo Actual:	<input type="text"/>
Nombre:	<input type="text"/>
Cédula:	<input type="text"/>
Teléfono:	<input type="text"/>
Dirección:	<input type="text"/>
<input type="checkbox"/> Acceso WEB	Clave: <input type="text"/>
	Repetir: <input type="text"/>
<input type="button" value="Ingresar"/>	

Figura 96 Diagrama de Interfaz: Ingresar Analista

Diagrama de Componentes: Ingresar Analista

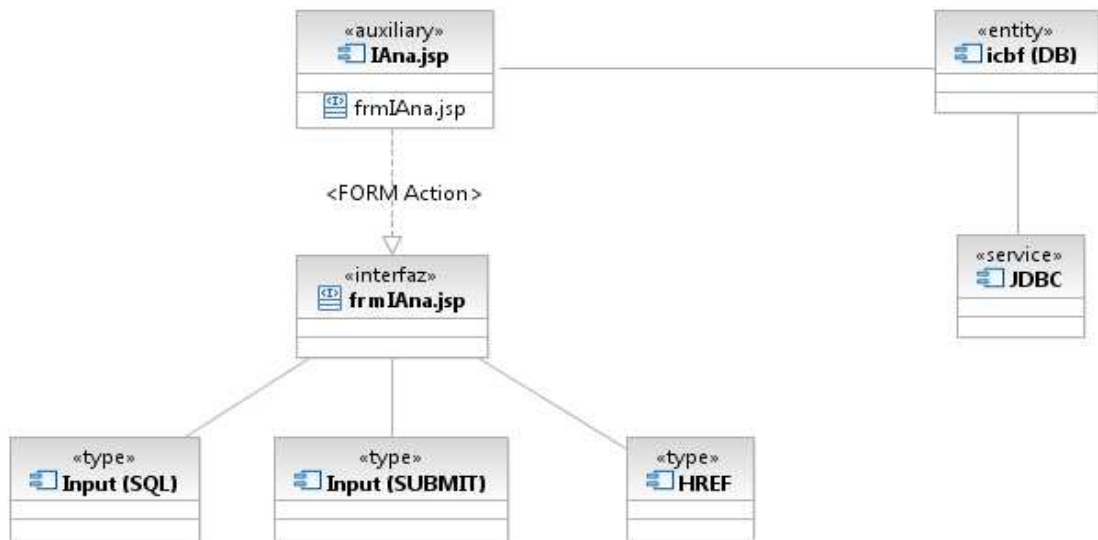


Figura 97 Diagrama de Componentes: Ingresar Analista

Código: Ingresar Analista

```
String Cue=request.getParameter("Cue");
String Apel=request.getParameter("Apel");
String Ape2=request.getParameter("Ape2");
String Nom=request.getParameter("Nom");
String Ced=request.getParameter("Ced");
String Tel=request.getParameter("Tel");
String Cel=request.getParameter("Cel");
String Cor=request.getParameter("Cor");
String Dir=request.getParameter("Dir");
String Car=request.getParameter("Car");
String chkWEB=request.getParameter("chkWEB");
String Cla=request.getParameter("Cla");
String Cla2=request.getParameter("Cla2");
if ((Cue != null && Cue.equals("") != true) && (Nom != null && Nom.equals("") !=
true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Analista where Código='"+Cue+"'";
        if (objC.verificar(sql) == 0){
            sql="insert into Analista values
('"+Cue+"', '"+Apel+"', '"+Ape2+"', '"+Nom+"', '"+Ced+'', '"+Tel+'', '"+Cel+'', '"+Cor+'
', '"+Dir+'', '"+Car+'')";
            if (objC.insertar(sql)==null){
                out.println("<h4> Datos Insertados </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
            else{
                out.println("<h2> Problemas para Registrar
</h2>"+sql+objC.insertar(sql));
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
            if (chkWEB.equals("si")){
                sql="insert into Usuarios values ('"+Cue+"', '"+Apel+"
"+Ape2+" "+Nom+"', '"+Cla+'', '3')";
                if (objC.insertar(sql)==null){
                    out.println("<h4> Usuario WEB Registrado </h4>");
                }
                else{
                    out.println("<h2>El usuario WEB no fue
registrado</h2>"+sql+objC.insertar(sql));
                }
            }
            else{
                out.println("<h4> El Código ya se encuentra registrado </h4>");
                out.print(" <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
            }
        }
        catch(Exception e){out.println("Atencion: "+e);}
    }else{ out.println("<h4> Debera ingresar informacion completa </h4>");}
```


Diagrama de Interfaz: Consultar Analista

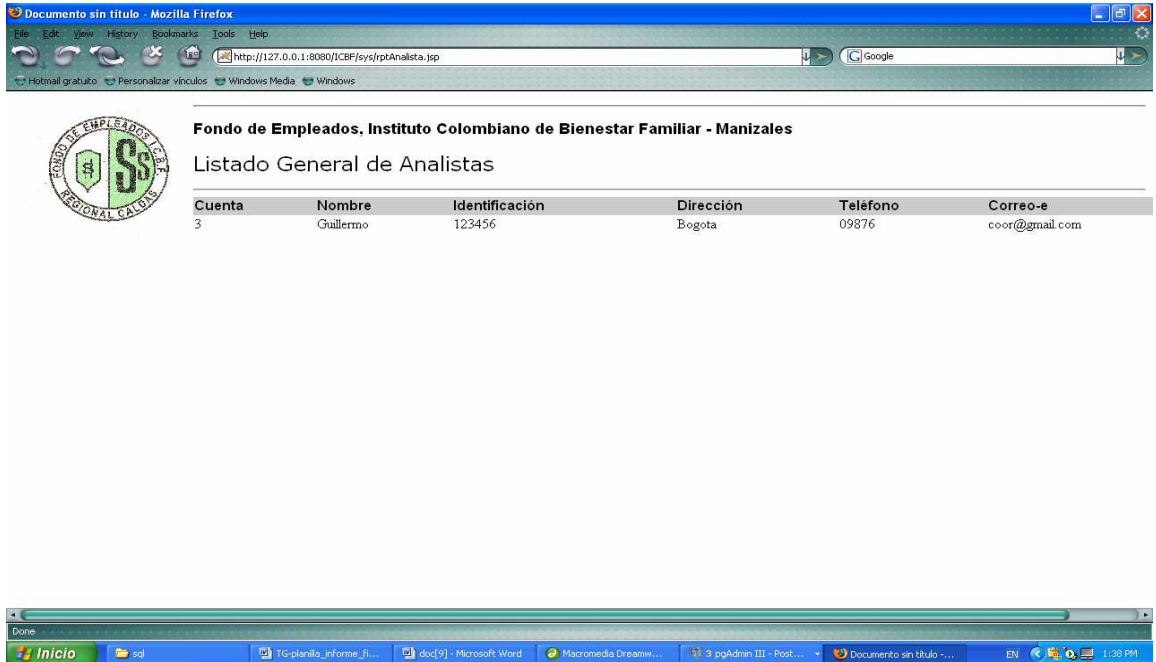


Figura 98 Diagrama de Interfaz: Consultar Analistas

Diagrama de Componentes: Consultar Analistas

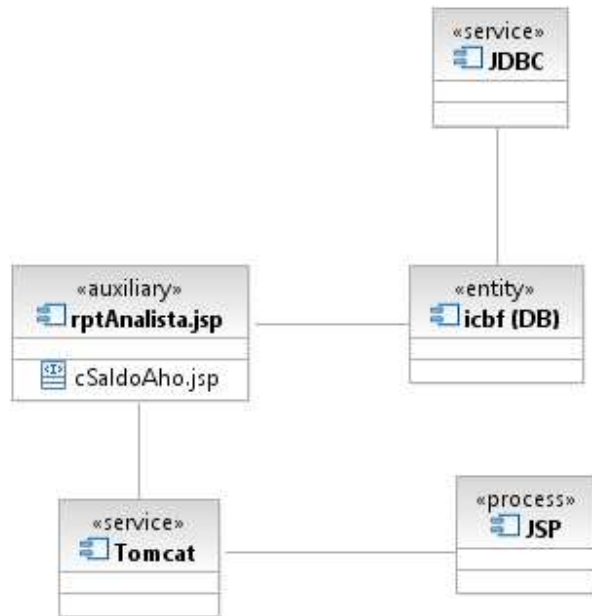


Figura 99 Diagrama de Componentes: Consultar Analistas

Código: Consultar Analistas

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Analista where Código='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                <table width=\"57%\" border=\"0\"> ");

            sb.append("                <tr>                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\">
face=\"VERDANA\">Primer Apellido:</font></div></td>                ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+" \" size=\"26\" maxlength=\"26\"></td>
                ");
            sb.append("                </tr>                ");

            sb.append("                <tr>                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\">
face=\"VERDANA\">Segundo Apellido:</font></div></td>                ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(3)+" \" size=\"26\" maxlength=\"26\"></td>
                ");
            sb.append("                </tr>                ");

            sb.append("                <tr>                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\">
face=\"VERDANA\">Nombre:</font></div></td>                ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(4)+" \" size=\"26\" maxlength=\"26\"></td>
                ");
            sb.append("                </tr>                ");

            sb.append("                <tr>                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\">
face=\"VERDANA\">Cedula:</font></div></td>                ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(5)+" \" size=\"26\" maxlength=\"26\"></td>
                ");
            sb.append("                </tr>                ");

            sb.append("                <tr>                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\">
face=\"VERDANA\">Telefono:</font></div></td>                ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(6)+" \" size=\"26\" maxlength=\"26\"></td>
                ");
            sb.append("                </tr>                ");
        }
    }
}
```

```

        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Celular:</font></div></td>                ");
        sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(7)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">E-Mail:</font></div></td>                ");
        sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(8)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Direccion:</font></div></td>                ");
        sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(9)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("                <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Cargo Actual:</font></div></td>                ");
        sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(10)+" \" size=\"26\" maxlength=\"26\"></td>
");
        sb.append("                </tr>                ");
        sb.append("        </table>                ");
        sb.append("        <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>                ");
        out.println(sb);
    }
    else{
        out.println("<h4> Código no encontrado </h4>");
        out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                ");
    }
    rs.close();
}
catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                ");
}
}

```

Diagrama de Interfaz: Modificar Analista

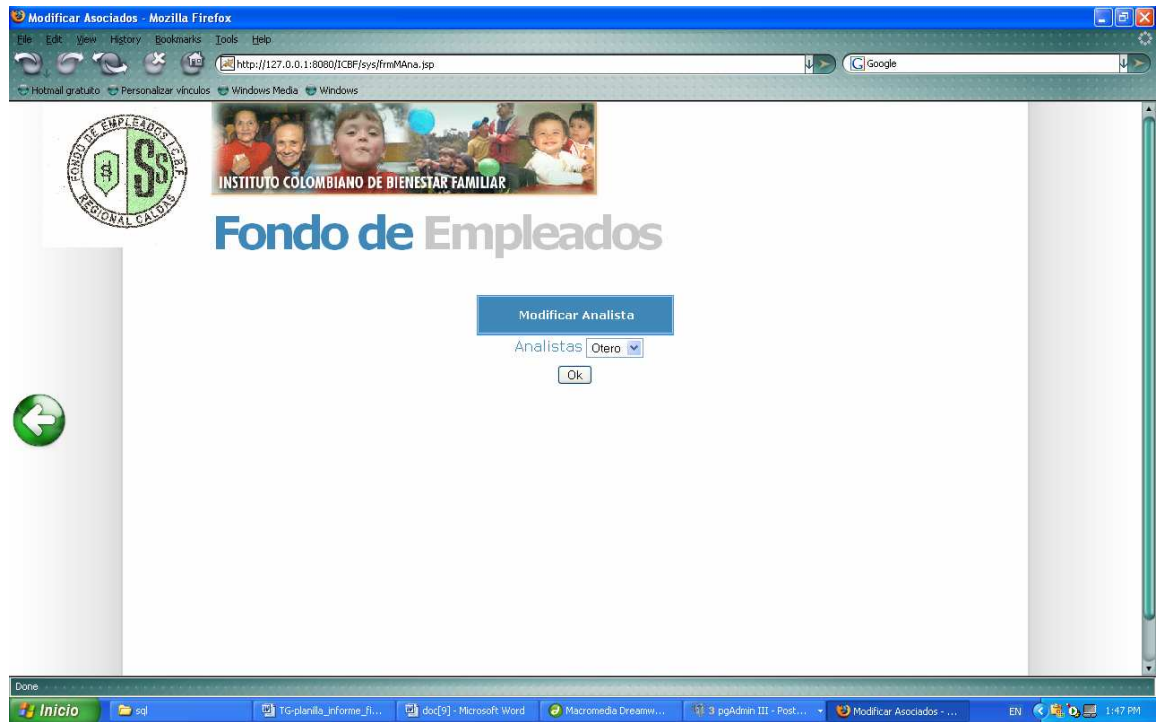


Figura 100 Diagrama de Interfaz: Modificar Analistas

Diagrama de Componentes: Modificar Analistas

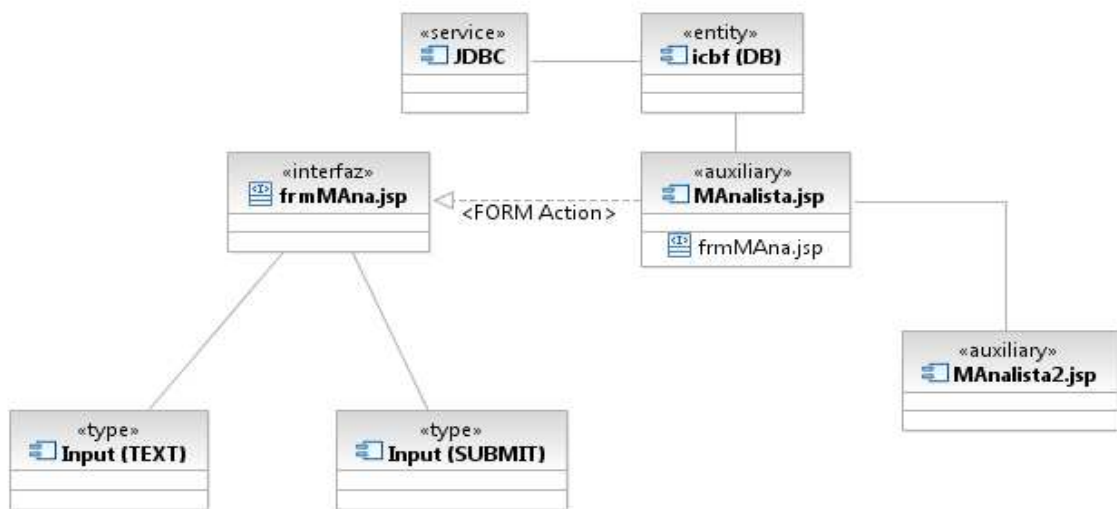


Figura 101 Diagrama de Componentes: Modificar Analistas

Código: Modificar Analistas

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Analista where Código='"+cCod+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                <form method=\"post\"
action=\"Modificacion.jsp\" target=\"desc\" >");
            sb.append("        <center>        <table width=\"57%\" border=\"0\" > ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Primer Apellido:</font></div></td>        ");
            sb.append("                <td><input name=\"Ape1\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+"\" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Segundo Apellido:</font></div></td>        ");
            sb.append("                <td><input name=\"Ape2\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(3)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Nombre:</font></div></td>        ");
            sb.append("                <td><input name=\"Nom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(4)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Cedula:</font></div></td>        ");
            sb.append("                <td><input name=\"Ced\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(5)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\"
face=\"VERDANA\">Telefono:</font></div></td>        ");
            sb.append("                <td><input name=\"Tel\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(6)+" \" size=\"26\" maxlength=\"26\"></td>
");
            sb.append("                </tr>                                ");
        }
    }
}
```

```

        sb.append("                </tr>                ");
        sb.append("        <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\" face=\"VERDANA\">Celular:</font></div></td>                ");
        sb.append("                <td><input name=\"Cel\" type=\"text\" id=\"cNom2\" value=\""+rs.getString(7)+" \" size=\"26\" maxlength=\"26\"></td>                ");
        sb.append("        </tr>                ");
        sb.append("        <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\" face=\"VERDANA\">E-Mail:</font></div></td>                ");
        sb.append("                <td><input name=\"Cor\" type=\"text\" id=\"cNom2\" value=\""+rs.getString(8)+" \" size=\"26\" maxlength=\"26\"></td>                ");
        sb.append("        </tr>                ");
        sb.append("        <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\" face=\"VERDANA\">Direccion:</font></div></td>                ");
        sb.append("                <td><input name=\"Dir\" type=\"text\" id=\"cNom2\" value=\""+rs.getString(9)+" \" size=\"26\" maxlength=\"26\"></td>                ");
        sb.append("        </tr>                ");
        sb.append("        <tr>                ");
        sb.append("                <td><div align=\"right\"><font size=\"2\" face=\"VERDANA\">Cargo Actual:</font></div></td>                ");
        sb.append("                <td><input name=\"Car\" type=\"text\" id=\"cNom2\" value=\""+rs.getString(10)+" \" size=\"26\" maxlength=\"26\"></td>                ");
        sb.append("        </tr>                ");
        sb.append("    </table>                ");
        sb.append("        <tr>                ");
        sb.append("                <td><input name=\"cCod\" type=\"hidden\" id=\"cCod2\" value=\""+cCod+"\" ></td>                ");
        sb.append("                <tr>                <td height=\"31\" colspan=\"2\"> <div align=\"center\">                ");
        sb.append("                <input name=\"ModificarAnalista\" type=\"submit\" target=\"desc\" value=\"Modificar Analista\" width=\"91\" height=\"27\" >                ");
        sb.append("                </div></td> </tr>                ");
        sb.append("                </tr>                ");
        sb.append("                </table>                ");
        sb.append("        </form >        </center>                ");
        sb.append("    </p><a href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
        out.println(sb);
    }
    else{
        out.println("<h4> Código no encontrado </h4>");
    }
}

```



```

                                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
                                }
                                rs.close();
                                }
                                catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}

```

Diagrama de Interfaz: Eliminar Analistas

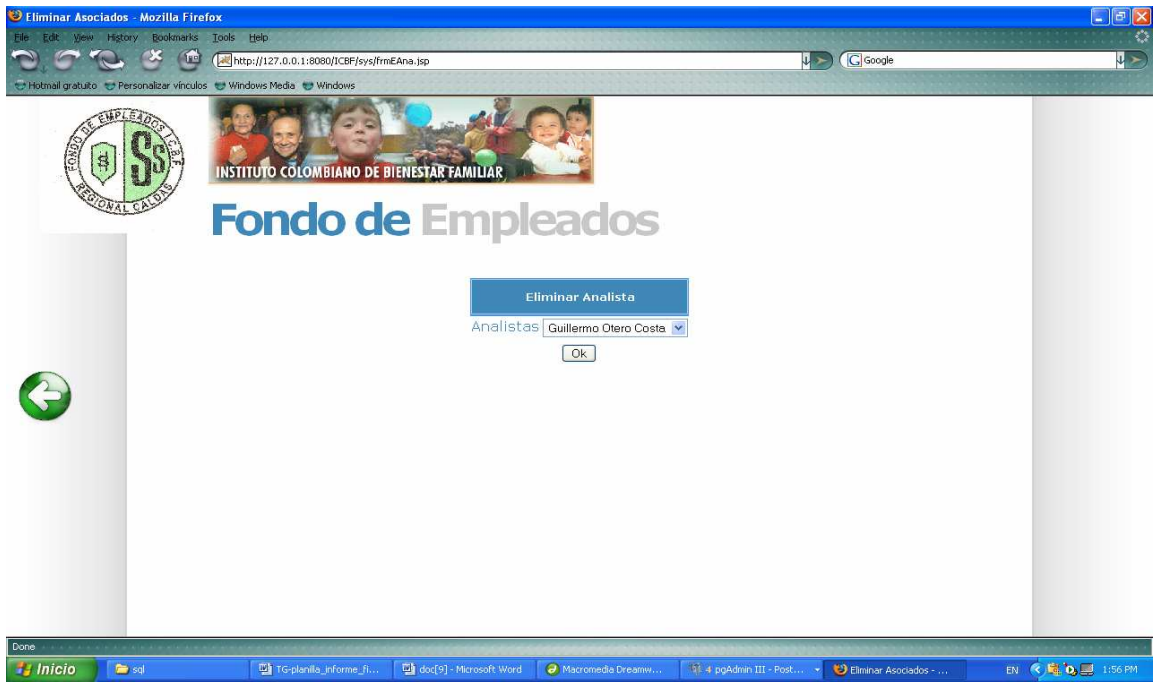


Figura 102 Diagrama de Interfaz: Eliminar Analistas

Diagrama de Componentes: Eliminar Analistas

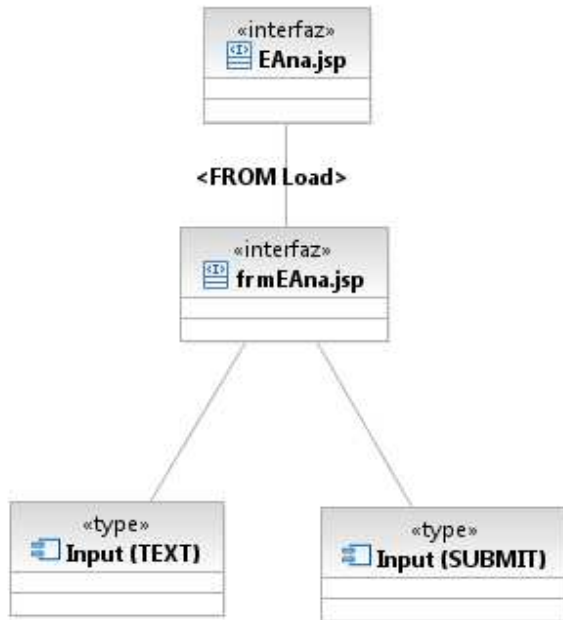


Figura 103 Diagrama de Componentes: Eliminar Analistas

Código: Eliminar Analistas

```
String cCod=request.getParameter("cCod");

if (cCod != null && cCod.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from Analista where Código='"+cCod+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from Analista where Código='"+cCod+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para eliminar </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
            else{
                out.println("<h4> Informacion eliminada </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print(" <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
}
}
```

Diagrama de Interfaz: Módulo Procesador



Figura 104 Diagrama de Interfaz: Modulo Procesador

Diagrama de Componentes: Modulo Procesador

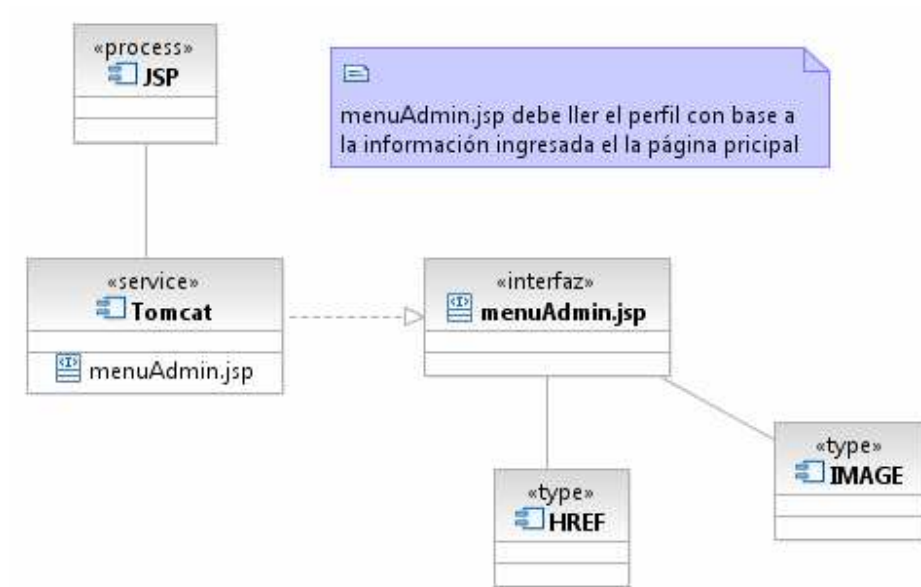


Figura 105 Diagrama de Componentes: Modulo procesador

Diagrama de Interfaz: Registro de Créditos

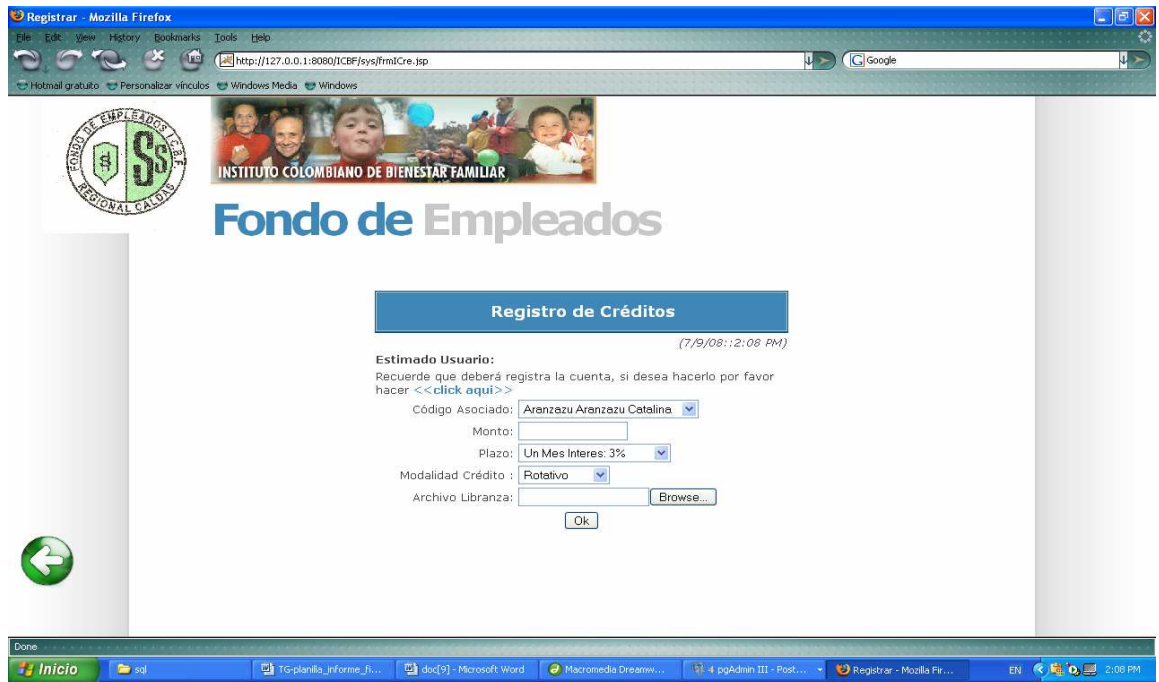


Figura 106 Diagrama de Interfaz: Registro de Créditos

Diagrama de Componentes: Registro de Créditos

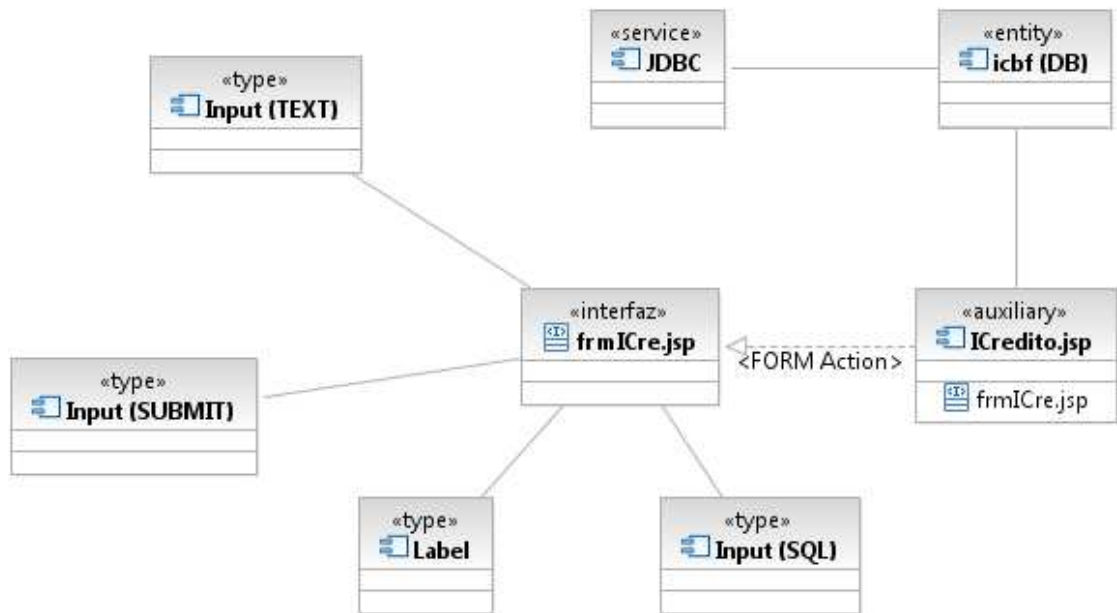


Figura 107 Diagrama de Componentes: Registro de Créditos

Código: Registro de Créditos

```
String LaFecha =
DateFormat.getDateInstance(DateFormat.SHORT,Locale.US).format(Calendar.getInstance().getTime());
String LaHora =
DateFormat.getTimeInstance(DateFormat.SHORT,Locale.US).format(Calendar.getInstance().getTime());
String Cue=request.getParameter("Cue");
String Mon=request.getParameter("Mon");
String Pla=request.getParameter("Pla");
String Arc=request.getParameter("Arc");
String Cla=request.getParameter("Cla");
ResultSet rs=null;
String sql="";
if ((Cue != null && Cue.equals("") != true) && (Mon != null && Mon.equals("") != true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}

        sql="select max(Credito) as fundillo from Creditos";
        rs=objC.consultar(sql);
        int NumCre=0;
        String sqlTmp="select InteresMensual from Plazos where
idPlazos='"+Pla+"'";
        //String ElInt=objC.buscarS(sqlTmp);
        Float ElInt=Float.parseFloat(objC.buscarS(sqlTmp).toString());
        sqlTmp="select meses from Plazos where idPlazos='"+Pla+"'";
        int Coutas=Integer.parseInt(objC.buscarS(sqlTmp).toString());

        Float Valor=Integer.parseInt(Mon)+(Integer.parseInt(Mon)*
(ElInt*Coutas) /100);
        if (rs.next()){
            if (rs.getString(1)==null)
                NumCre=1;
            else
                NumCre=Integer.parseInt(rs.getString(1))+1;}
        else
            NumCre=1;

        rs.close();
        sql="select * from Credito where Cuenta='"+Cue+"'";
        if (objC.verificar(sql) == 0){

            sql="insert into Creditos values
('"+NumCre+"','"+Cue+"','"+LaFecha+"','"+LaHora+"','"+Valor+"','"+Pla+"','"+Arc+"
','"+Cla+"',0)";
            if (objC.insertar(sql)==null){
                sql="update Cuentas set SaldoCredito=" + Valor + "
where cuenta=" + Cue + "'";
                objC.insertar(sql);

                sql="select max(Cuenta) as fundillo from Rutas";
                rs=objC.consultar(sql);
                if (rs.next()){
                    if (rs.getString(1)==null)
```

```

                NumCre=1;
            else
                NumCre=Integer.parseInt(rs.getString(1))+1;}
        else
            NumCre=1;
            sql="insert into Rutas values
('"+NumCre+"','03','"+Cue+"','Credito Aprobado','"+LaFecha+"','"+LaHora+"')";
            objC.insertar(sql);
            sql="update SolicitudCredito set Estado='Credito Aprobado'
where Cuenta='"+Cue+"'";
            objC.insertar(sql);

        out.println("<h4> ESTIMADO USUARIO </h4>");
        out.println("<h4> EL CRÉDITO FUE REGISTRADO
EXITOSAMENTE ..... </h4>");
        out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
    }
    else{
        out.println("<h2> Problemas para Registrar
</h2>"+sql+objC.insertar(sql));
        out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
    }
}
else{
    out.println("<h4> YA EXISTE SOLICITUD PREVIA EN TRÁMITE </h4>");
    out.print(" <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
}
}
catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> DEBERÁ INGRESAR LA INFORMACIÓN COMPLETA </h4>");
}
}

```

Diagrama de Interfaz: Consulta de Créditos



Figura 108 Diagrama de Interfaz: Consulta de Créditos

Diagrama de Componentes: Consulta de Créditos

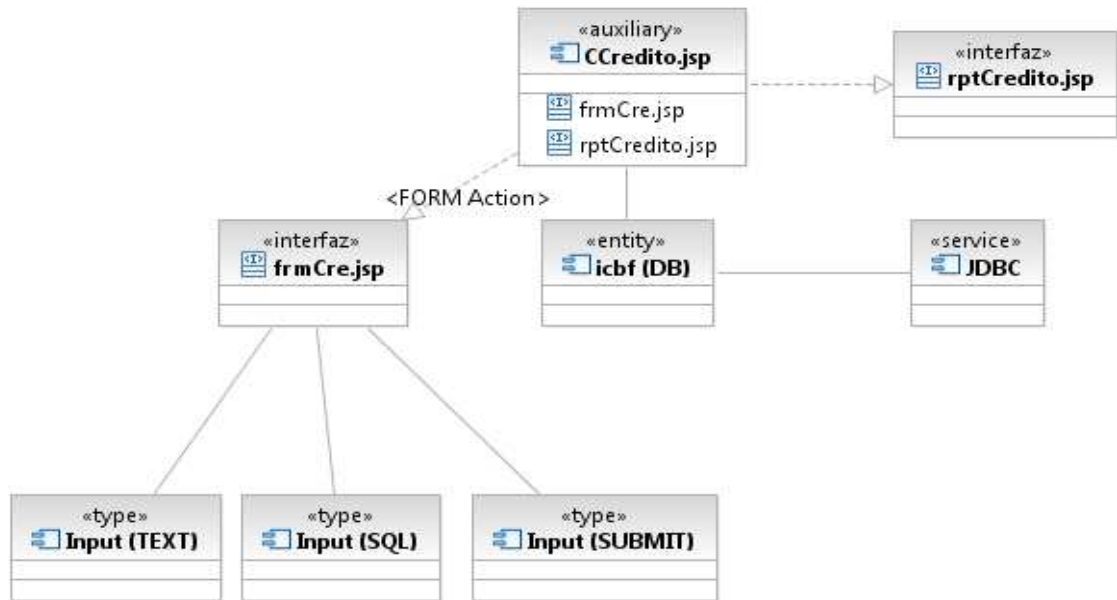


Figura 109 Diagrama de Componentes: Consulta de Créditos

Código: Consulta de Créditos

```
String Aso=request.getParameter("Aso");

if (Aso != null && Aso.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Creditos where Cuenta='"+Aso+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                <table width=\\"57%\" border=\\"0\"> ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\\"right\"><font size=\\"2\">
face=\\"VERDANA\">Fecha:</font></div></td>                ");
            sb.append("                <td><input name=\\"cNom\" type=\\"text\">
id=\\"cNom2\" value=\\""+rs.getString(3)+" \" size=\\"26\" maxlength=\\"26\"></td>
");
            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\\"right\"><font size=\\"2\">
face=\\"VERDANA\">Monto:</font></div></td>                ");
            sb.append("                <td><input name=\\"cNom\" type=\\"text\">
id=\\"cNom2\" value=\\""+rs.getString(5)+" \" size=\\"26\" maxlength=\\"26\"></td>
");
            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\\"right\"><font size=\\"2\">
face=\\"VERDANA\">Plazo:</font></div></td>                ");
            sb.append("                <td><input name=\\"cNom\" type=\\"text\">
id=\\"cNom2\" value=\\""+rs.getString(6)+" \" size=\\"26\" maxlength=\\"26\"></td>
");
            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\\"right\"><font size=\\"2\">
face=\\"VERDANA\">Interes:</font></div></td>                ");
            sb.append("                <td><input name=\\"cNom\" type=\\"text\">
id=\\"cNom2\" value=\\""+rs.getString(7)+" \" size=\\"26\" maxlength=\\"26\"></td>
");
            sb.append("                </tr>                                ");

            sb.append("                </table>                ");
            sb.append("                <p><a href=\\"javascript:history.back()\"><font
face=\\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>                ");

            out.println(sb);
        }
        else{
```

```

        out.println("<h4> Código no encontrado </h4>");
        out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
    }
    rs.close();
}
catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
}
}

```

Diagrama de Interfaz: Cancelar Créditos



Figura 110 109 Diagrama de Interfaz: Cancelar Créditos

Diagrama de Componentes: Cancelar Créditos

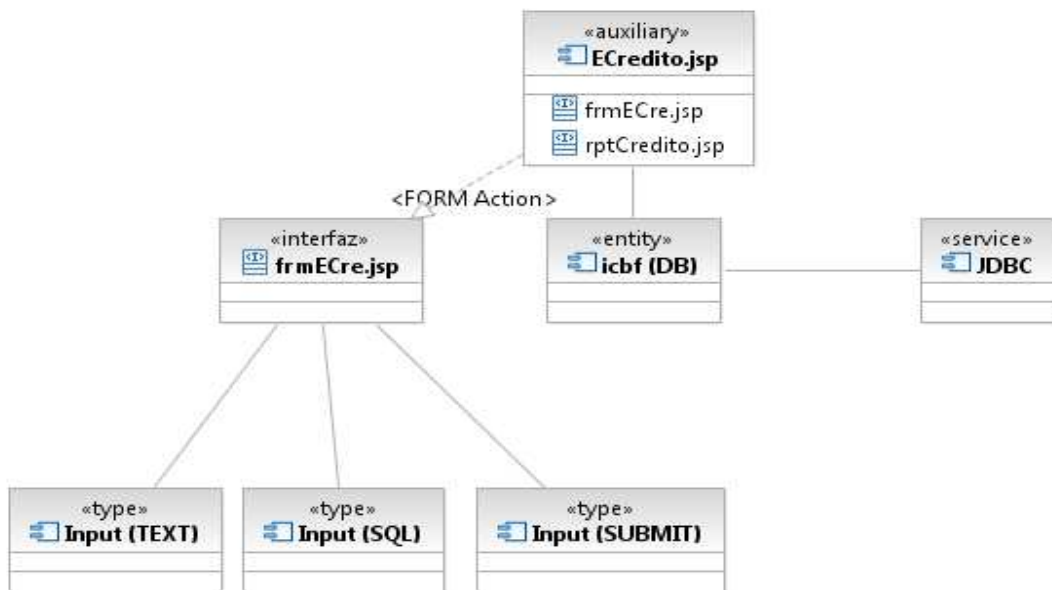


Figura 111 Diagrama de Componentes: Cancelar Créditos

Código: Cancelar créditos

```
String Cue=request.getParameter("Cue");

if (Cue != null && Cue.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from Cuentas where Cuenta='"+Cue+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from Cuentas where Cuenta='"+Cue+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para retirar </h4>");
                out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                                ");
            }
            else{
                out.println("<h4> La cuenta fue cancelada </h4>");
                out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                                ");
            }
        }
        else{
            out.println("<h4> Código no encontrado </h4>");
            out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                                ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                                <p><a
href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>                                ");
}
}
```

Diagrama de Interfaz: Registro de Cuentas

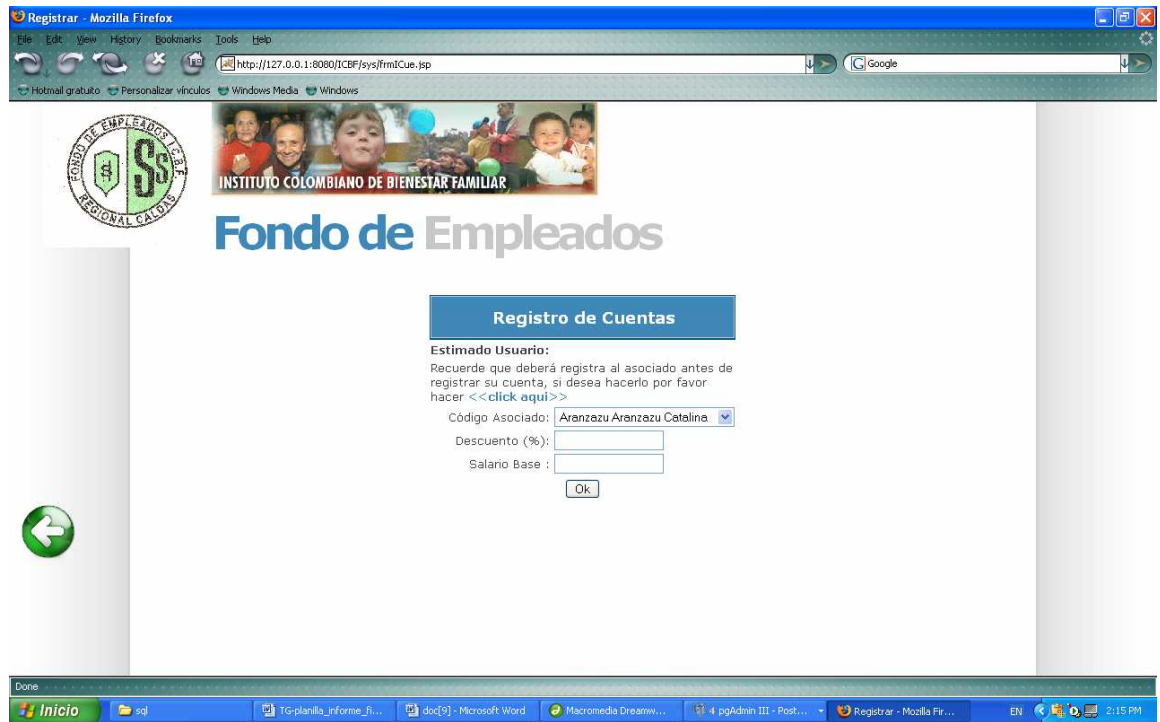
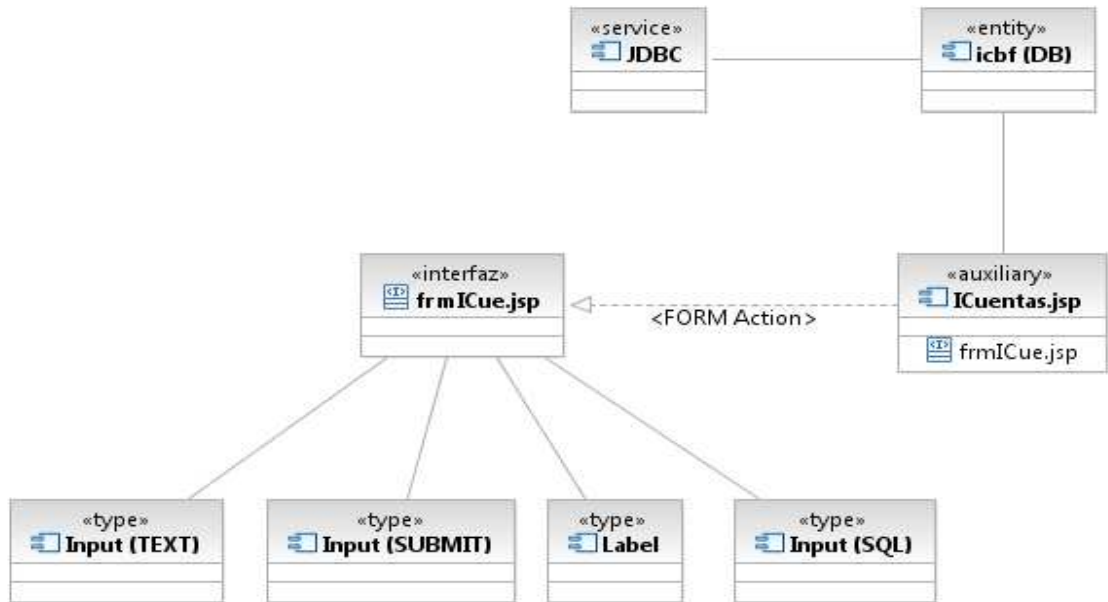


Figura 112 Diagrama de Interfaz: Registro de Cuentas

Diagrama de Componentes: Registro de Cuentas



IFigura 113 Diagrama de Componentes: Registro de Cuentas

Código: Registro de Cuentas

```
String LaFecha =
DateFormat.getDateInstance(DateFormat.SHORT,Locale.US).format(Calendar.getInstanc
e().getTime());
String LaHora =
DateFormat.getTimeInstance(DateFormat.SHORT,Locale.US).format(Calendar.getInstanc
e().getTime());
String Cue=request.getParameter("Cue");
String Des=request.getParameter("Des");
String Sal=request.getParameter("Sal");

if ((Cue != null && Cue.equals("") != true) && (Des != null && Des.equals("") !=
true )){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Cuentas where Cuenta='"+Cue+"'";
        if (objC.verificar(sql) == 0){
            sql="insert into Cuentas values
('"+Cue+"','"+LaFecha+"','"+LaHora+"','"+Des+"','"+Sal+"','0','0','A')";
            if (objC.insertar(sql)==null){
                out.println("<h4> Cuenta Registrada ..... </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
            else{
                out.println("<h2> Problemas para Registrar
</h2>"+sql+objC.insertar(sql));
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
        }
        else{
            out.println("<h4> La Cuenta ya se encuentra registrada </h4>");
            out.print(" <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
        }
        catch(Exception e){out.println("Atencion: "+e);}
    }
}
else{
    out.println("<h4> Debera ingresar informacion completa </h4>");
}
```

Diagrama de Interfaz: Consulta de Cuentas



Figura 114 Diagrama de Interfaz: Consulta de Cuentas

Diagrama de Componentes: Consulta de Cuentas

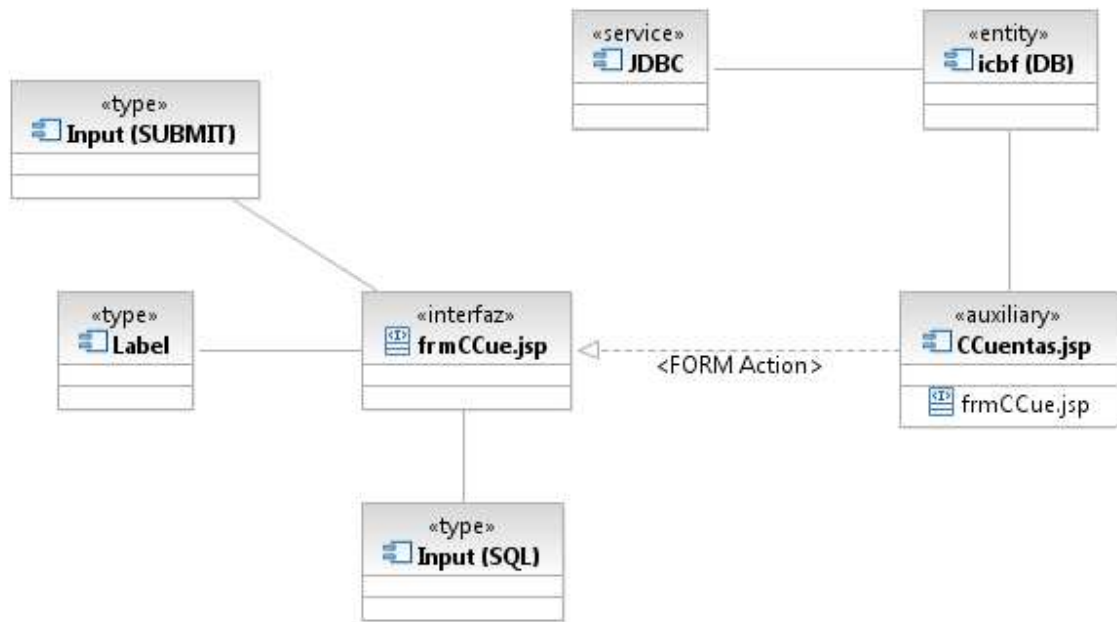


Figura 115 Diagrama de Componentes: Consulta de Cuentas

Código: Consulta de Cuentas

```
String Aso=request.getParameter("Aso");

if (Aso != null && Aso.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Cuentas where Cuenta='"+Aso+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            sb.append("                <table width=\"57%\" border=\"0\"> ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\" "
face=\"VERDANA\">Fecha Registro:</font></div></td> ");

            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(2)+" \" size=\"26\" maxlength=\"26\"></td>
");

            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\" "
face=\"VERDANA\">Porcentaje:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(4)+" \" size=\"26\" maxlength=\"26\"></td>
");

            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\" "
face=\"VERDANA\">Salario:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(5)+" \" size=\"26\" maxlength=\"26\"></td>
");

            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\" "
face=\"VERDANA\">Saldo Ahorrado:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(6)+" \" size=\"26\" maxlength=\"26\"></td>
");

            sb.append("                </tr>                                ");

            sb.append("                <tr>                                ");
            sb.append("                <td><div align=\"right\"><font size=\"2\" "
face=\"VERDANA\">Saldo Credito:</font></div></td> ");
            sb.append("                <td><input name=\"cNom\" type=\"text\"
id=\"cNom2\" value=\""+rs.getString(7)+" \" size=\"26\" maxlength=\"26\"></td>
");
        }
    }
}
```

```

        sb.append("                </tr>                ");
    sb.append("            </table>            ");
    sb.append("        <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>        ");

    out.println(sb);
    }
    else{
        out.println("<h4> Código no encontrado </h4>");
        out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                ");
    }
    rs.close();
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>                ");
}
}

```


Diagrama de Interfaz: Retiro de Cuentas



Figura 116 Diagrama de Interfaz: Retiro de Cuentas

Diagrama de Componentes: Retiro de Cuentas

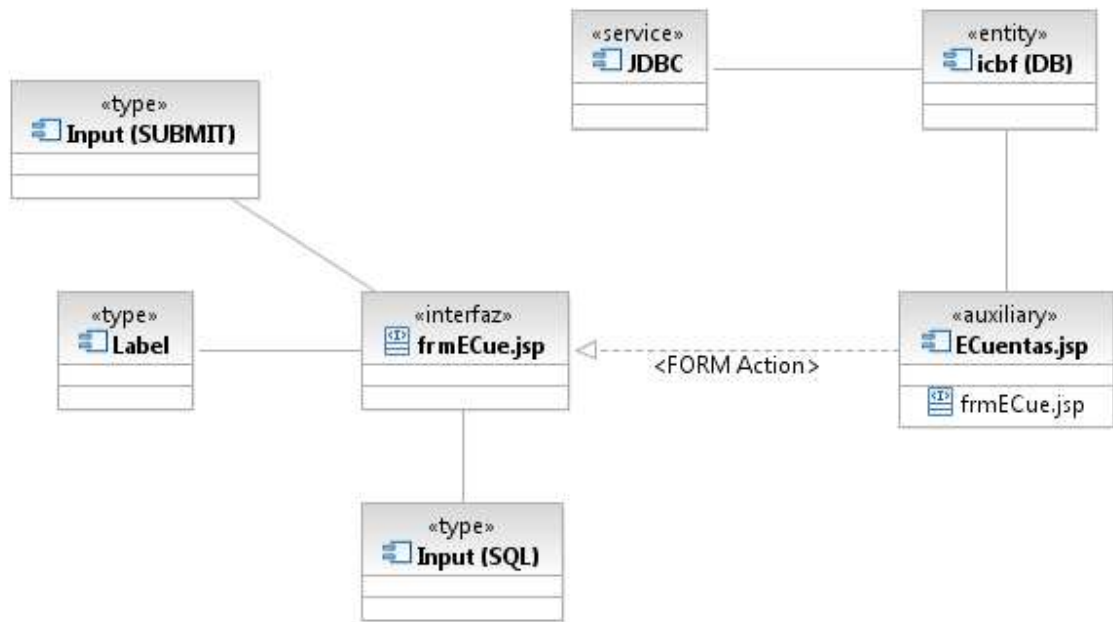


Figura 117 Diagrama de Componentes: Retiro de Cuentas

Código: Retiro de Cuentas

```
String Cue=request.getParameter("Cue");

if (Cue != null && Cue.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from Cuentas where Cuenta='"+Cue+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from Cuentas where Cuenta='"+Cue+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4> Problemas para retirar </h4>");
                out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
            }
        }
        else{
            out.println("<h4> La cuenta fue cancelada </h4>");
            out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
    }
    else{
        out.println("<h4> Código no encontrado </h4>");
        out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
    }
}
catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar un Código </h4>");
    out.print("                                <p><a
href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
}
}
```

Diagrama de Interfaz: Generación de Reporte Mensual



Figura 118 Diagrama de Interfaz: Generación de Reporte Mensual

Diagrama de Componentes: Generación de Reporte Mensual

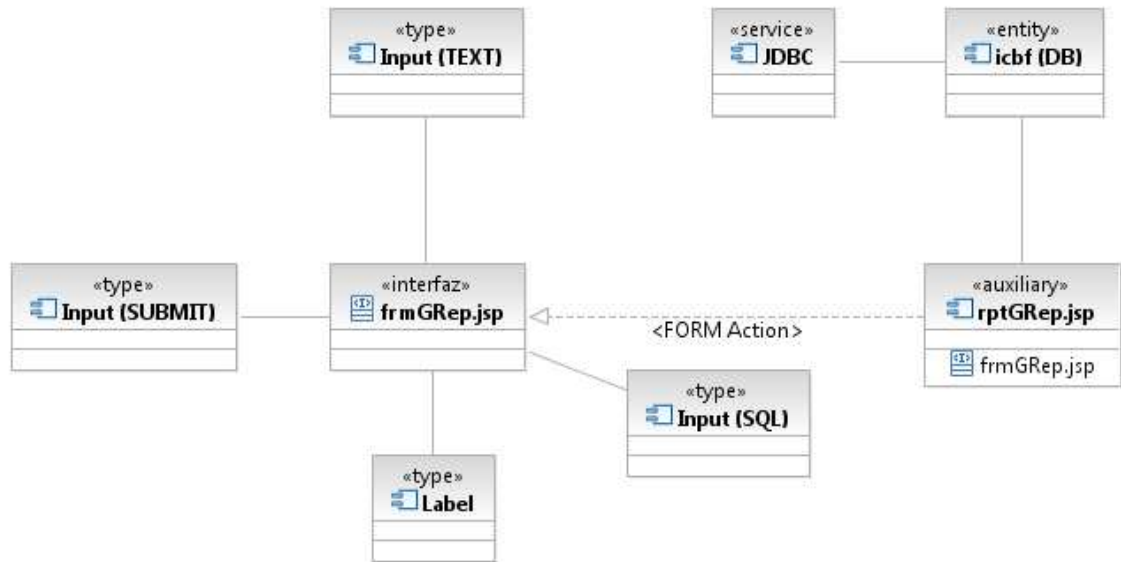


Figura 119 Diagrama de Componentes: Generación de Reporte Mensual

Código: Generación de Reporte Mensual

```
String LaFecha =
DateFormat.getDateInstance(DateFormat.SHORT,Locale.US).format(Calendar.getInstance().getTime());
String LaHora =
DateFormat.getTimeInstance(DateFormat.SHORT,Locale.US).format(Calendar.getInstance().getTime());
String Per=request.getParameter("Per");
String Obs=request.getParameter("Obs");
int consecutivo=0,ban=0;
ResultSet rs=null;
String sql="";
float Ahorro=0;
float Credito=0;
float Porcentaje=0,Salario=0,Monto=0;
if ((Per != null && Per.equals("") != true) ){
    co objC = new co();
    try{ int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        sql="select * from Movimiento where Periodo='"+Per+"'";
        if (objC.verificar(sql) == 0){
            ResultSet rs2=null;
            String sql2="select max(Consecutivo) as NroConsecutivo from
Movimiento";
            int ElMen=0;
            rs2=objC.consultar(sql2);
            if (rs2.next()){
                consecutivo=Integer.parseInt(rs2.getString(1))+1;
            }
            rs2.close();

            sql="select * from Cuentas";
            rs=objC.consultar(sql);
            while (rs.next()){
                //out.print("cuenta "+rs.getString(1)+" ccc "+consecutivo);
                Porcentaje=Integer.parseInt(rs.getString(4));
                Salario=Integer.parseInt(rs.getString(5));
                if (rs.getString(6).equals("Ahorro")){
                    sql="select SaldoAhorrado from Cuentas where
Cuenta="+rs.getString(1)+"";

                    Ahorro=Integer.parseInt(objC.buscarS(sql).toString());
                }
                sql="insert into Movimiento values
('"+consecutivo+"','"+rs.getString(1)+"','"+Per+"','"+LaFecha+"','"+LaHora+"','Ah
orro','"+Salario*Porcentaje/100+"','"+(Ahorro+(Salario*Porcentaje/100))+"','"+Obs
+"')";

                if (objC.insertar(sql)==null){
                    ban=1;
                }
                consecutivo++;
            }
            rs.close();
            sql="select * from Creditos";
            rs=objC.consultar(sql);
            while (rs.next()){
```

```

Monto=Integer.parseInt(rs.getString(5));
Porcentaje=Integer.parseInt(rs.getString(7));
if (rs.getString(6).equals("Ahorro")){
    sql="select SaldoCredito from Cuentas where
Cuenta="+rs.getString(2)+"";

    Credito=Integer.parseInt(objC.buscarS(sql).toString());
    }
    sql="insert into Movimiento values
('"+consecutivo+"','"+rs.getString(2)+"','"+Per+"','"+LaFecha+"','"+LaHora+"','Cr
edito','"+Monto*Porcentaje/100+"','"+(Ahorro-
(Monto*Porcentaje/100))+"','"+Obs+"')";
    if (objC.insertar(sql)==null){
        ban=1;
    }
    consecutivo++;
}
rs.close();
if (ban == 1){
    out.println("<p><h4> Informe Generado .....
</h4></p>");
    out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
    %>
    <p>Ahora usted podr imprimir el informe, si
desea<br>
                    hacer sto ya mismo por favor hacer <a
href=" ../frmGImp.jsp?q2e46y=eoB9i8u">&lt;&lt;clik aqui&gt;&gt;</a><br>
</p>
    <%
        }
        else
        {
            out.println("<h4> No hay cuentas para iniciar el
proceso ..... </h4>");
            out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
        }
    }
    else{
        out.println("<h4> El Periodo ya fue Generado </h4>");
        out.print(" <p><a href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p> ");
    }
}
catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4> Debera ingresar informacion completa </h4>");
}
}

```

Diagrama de Interfaz: Depuración



Figura 120 Diagrama de Interfaz: Depuración Movimiento

Diagrama de Componentes: Ajustes de información procesada

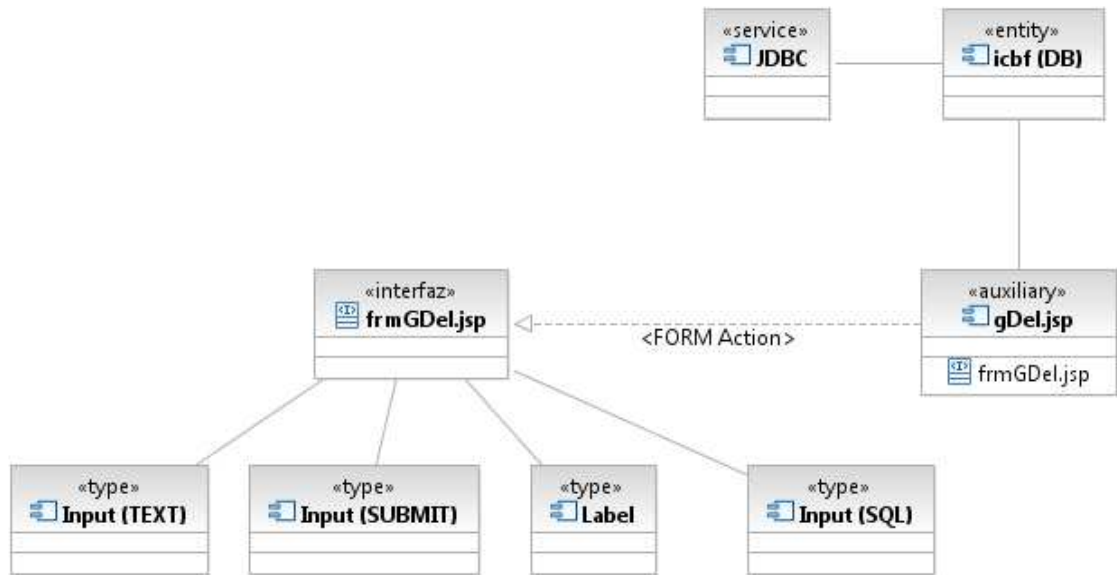


Figura 121 Diagrama de Componentes: Depuración Movimiento

Código: Depuración Movimiento

```
String Per=request.getParameter("Per");

if (Per != null && Per.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    try{
        objC.conectar();
        String sql="select * from Movimiento where Periodo='"+Per+"'";
        if (objC.verificar(sql) == 1){
            sql="delete from Movimiento where Periodo='"+Per+"'";
            if (objC.actualizar(sql) == -1){
                out.println("<h4>    No se encontro el Periodo seleccionado
</h4>");
                out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>    ");
            }
            else{
                out.println("<h4>    Periodo Eliminado </h4>");
                out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>    ");
            }
        }
        else{
            out.println("<h4>    Periodo no encontrado </h4>");
            out.print("                                <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>    ");
        }
    }
    catch(Exception e){out.println("Atencion: "+e);}
}
else{
    out.println("<h4>    Debera ingresar un Periodo </h4>");
    out.print("                                <p><a
href=\"javascript:history.back()\"><font
face=\"Verdana, Arial, Helvetica, sans-serif\">&lt;&lt;</font></a></p>    ");
}
}
```

Diagrama de Interfaz: Generación reporte nómina

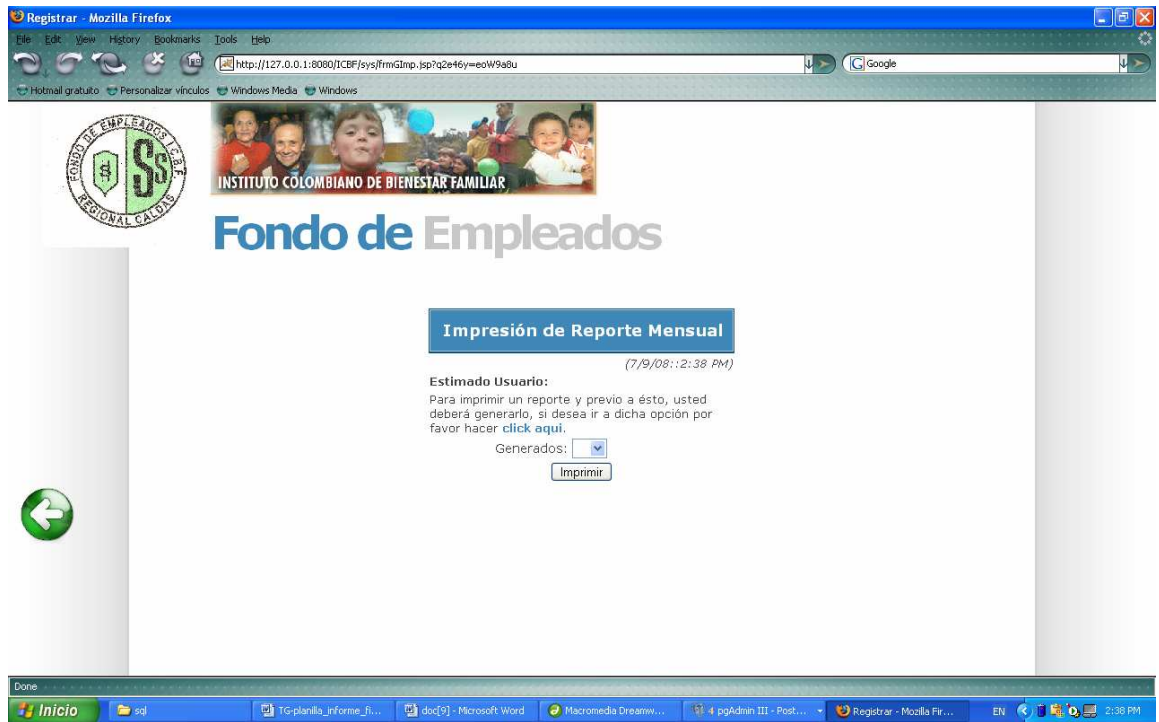


Figura 122 Diagrama de Interfaz: Generación reporte nómina

Diagrama de Componentes: Generación reporte nómina

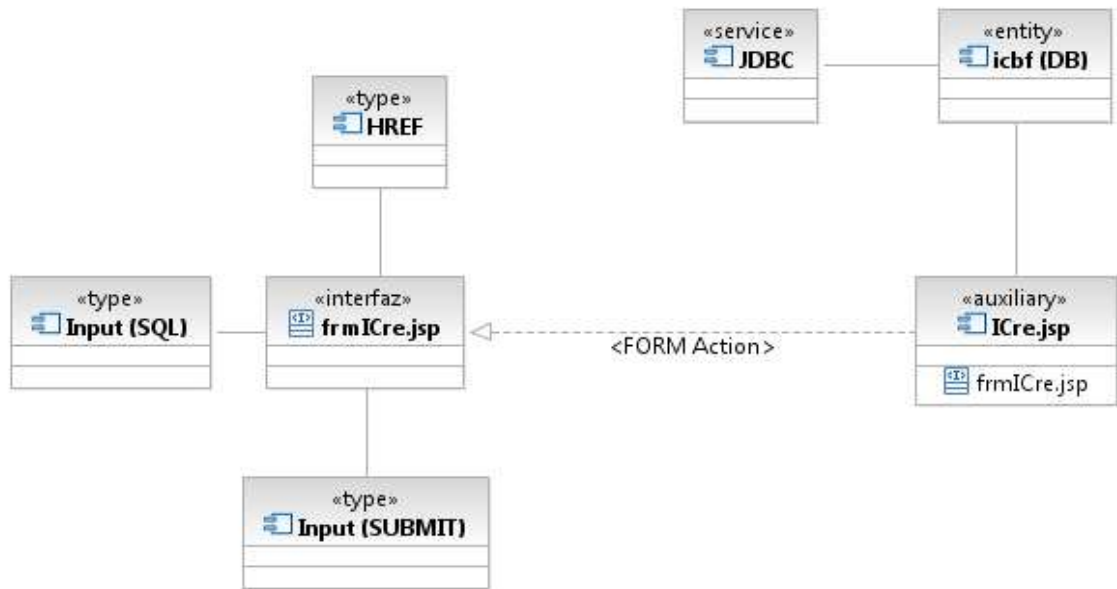


Figura 123 Diagrama de Componentes: Generación reporte nómina

Código: Generación Reporte Nominal

```
co objC = new co();
String sqlTmp="";
String ElNom="";
try{ int x=objC.conectar();
    if (x == -1){ out.println("No se conecto"+x);}
}
catch(Exception e){out.println("Atencion: "+e);}
String Per=request.getParameter("Per");
NumberFormat nf = NumberFormat.getCurrencyInstance(Locale.US);
sqlTmp="select Periodo from Periodos where Código='"+Per+"'";
ElNom=objC.buscarS(sqlTmp);
String sql="SELECT * From Movimiento WHERE Periodo ='"+Per+"' and
TipoMovim='Ahorro' order by Cuenta";
Driver DriverRecordset1 = (Driver)Class.forName(MM_dbCoo_DRIVER).newInstance();
Connection ConnRecordset1 =
DriverManager.getConnection(MM_dbCoo_STRING,MM_dbCoo_USERNAME,MM_dbCoo_PASSWORD);
PreparedStatement StatementRecordset1 = ConnRecordset1.prepareStatement(sql);
ResultSet Recordset1 = StatementRecordset1.executeQuery();
boolean Recordset1_isEmpty = !Recordset1.next();
boolean Recordset1_hasData = !Recordset1_isEmpty;
Object Recordset1_data;
int Recordset1_numRows = 0;
%>
<%
int Repeat1__numRows = 10;
int Repeat1__index = 0;
Recordset1_numRows += Repeat1__numRows;
%>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Listado de Ahorros</title>
</head>

<body>
<p></p>
<p>&nbsp;</p>
<hr />
<p><font size="4"><strong><font color="#000000" face="Arial, Helvetica, sans-
serif">Fondo de Empleados, Instituto Colombiano de Bienestar Familiar -
Manizales</font></strong></font></p>
<p><font size="5" face="Verdana, Arial, Helvetica, sans-serif">Listado de
Ahorros (<%=ElNom%>)</font></p>
<p>&nbsp;</p>
<hr />

<table width="90%" border="0" cellspacing="0" cellpadding="1">
<tr>
<td bgcolor="#CCCCCC"><div align="left"><font face="Arial, Helvetica, sans-
serif"><strong>Cuenta</strong></font></div></td>
<td bgcolor="#CCCCCC"><div align="left"><font face="Arial, Helvetica, sans-
serif"><strong>Nombre</strong></font></div></td>
```

```

        <td bgcolor="#CCCCCC"><div align="left"></div></td>
        <td bgcolor="#CCCCCC"><div align="left"><font face="Arial, Helvetica, sans-
serif"><strong>Movimiento</strong></font></div></td>
        <td bgcolor="#CCCCCC"><div align="left"><font face="Arial, Helvetica, sans-
serif"><strong>Valor</strong></font></div></td>
        <td bgcolor="#CCCCCC"><font face="Arial, Helvetica, sans-serif"><strong>Saldo
en ahorro</strong></font></td>
        <td bgcolor="#CCCCCC"><font face="Arial, Helvetica, sans-
serif"><strong>Observaciones</strong></font></td>
</tr>
<tr> </tr>
<% while ((Recordset1_hasData)&&(Repeat1__numRows-- != 0)) {
    Object cta=(((Recordset1_data = Recordset1.getObject("Cuenta"))==null ||
Recordset1.isNull())?"":Recordset1_data);
    %>
<tr>
<%
    Object Valo=Recordset1.getObject("Valor");
    Object Sald=Recordset1.getObject("Saldo");
    Object Valo2= nf.format( Double.parseDouble( Valo+" " ));
    Object Sald2= nf.format( Double.parseDouble( Sald+" " ));
    sqlTmp="select Apellido1 from Asociado where Cuenta='"+cta+"'";
    String ElApel1=objC.buscarS(sqlTmp);
    sqlTmp="select Apellido2 from Asociado where Cuenta='"+cta+"'";
    String ElApe2=objC.buscarS(sqlTmp);
    sqlTmp="select Nombre from Asociado where Cuenta='"+cta+"'";
    String ElNombre=objC.buscarS(sqlTmp);
    sqlTmp="select Cedula from Asociado where Cuenta='"+cta+"'";
    String Ced=objC.buscarS(sqlTmp);
    String Chilla=ElNombre+" "+ElApel1+" "+ElApe2;

    %>
<td><div align="left"><%=cta%></div></td>
<td><div align="left"><%=Chilla%></div></td>

    <td><div align="left"></div></td>
    <td><div align="left"><%=(((Recordset1_data = Recordset1.getObject("TipoMovim"))==null ||
Recordset1.isNull())?"":Recordset1_data)%></div></td>
    <td><div align="left"><%=Valo2%></div></td>
    <td><%=Sald2%></td>
    <td><%=(((Recordset1_data = Recordset1.getObject("Observaciones"))==null ||
Recordset1.isNull())?"":Recordset1_data)%></td>
</tr>
<%
    Repeat1__index++;
    Recordset1_hasData = Recordset1.next();
}
%>
</table>
</body>
</html>
<%
Recordset1.close();
StatementRecordset1.close();
ConnRecordset1.close();
%>

```

Diagrama de Interfaz: Menú Asociado



Figura 124 Diagrama de Interfaz: Menú Asociado

Diagrama de Componentes: Menú Asociado

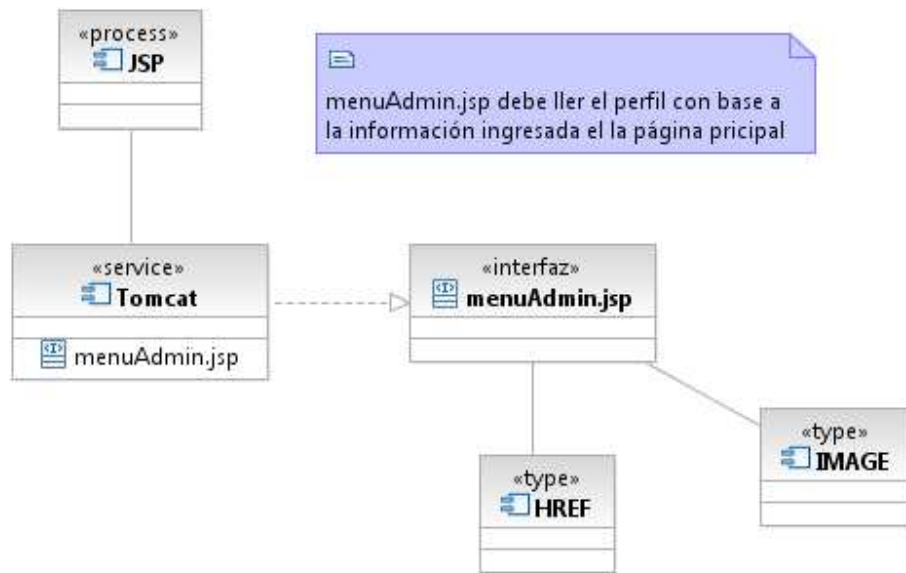


Figura 125 Diagrama de Componentes: Menú Asociado

Diagrama de Interfaz: Consulta Saldo Ahorros

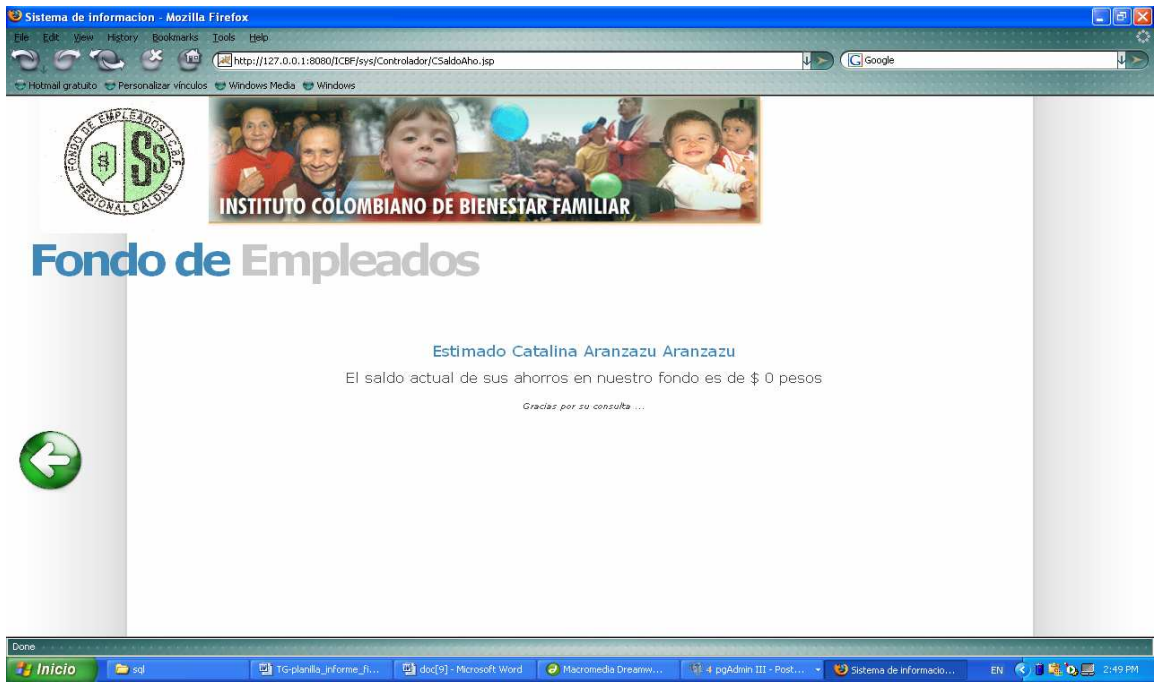


Figura 126 Diagrama de Interfaz: Consulta Saldo Ahorros

Diagrama de Componentes: Consulta Saldo Ahorros

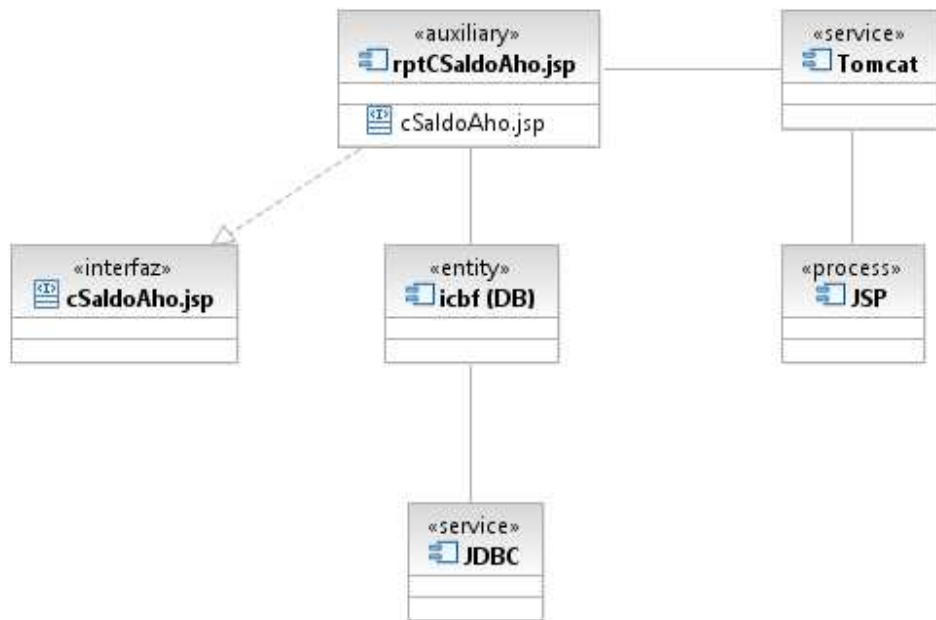


Figura 127 Diagrama de Componentes: Consulta Saldo Ahorros

Código: Consulta Saldo Ahorros

```
Object Aso=session.getAttribute("theMan");

if (Aso != null && Aso.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Asociado where Cuenta='"+Aso+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            String sqlTmp="select Apellido1 from Asociado where
Cuenta='"+Aso+"'";
            String ElApe1=objC.buscarS(sqlTmp);
            sqlTmp="select Apellido2 from Asociado where
Cuenta='"+Aso+"'";
            String ElApe2=objC.buscarS(sqlTmp);
            sqlTmp="select Nombre from Asociado where Cuenta='"+Aso+"'";
            String ElNom=objC.buscarS(sqlTmp);
            sqlTmp="select SaldoAhorrado from Cuentas where
Cuenta='"+Aso+"'";
            %>
            </p>
            <p align="center" class="blue Estilo1"> Estimado <%=ElNom%> <%=ElApe1%>
<%=ElApe2%> </p>
            <p align="center" class="Estilo2">El saldo actual de sus ahorros en nuestro
fondo es de $ <%=objC.buscarS(sqlTmp)%> pesos </p>
            <p align="center" class="Estilo3">Gracias por su consulta ... </p>
            <p class="Estilo3"><a href="javascript:back()"></a></p>
            <p>
            <%
                }
            else{
                out.println("<h4> El Asociado "+Aso+" no fue encontrado </h4>");
                out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
                }
                rs.close();
            }
            catch(Exception e){out.println("Atencion: "+e);}
        }
    }
    else{
        out.println("<h4> Debera ingresar un Código </h4>");
        out.print(" <p><a
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p> ");
    }
}
```

Diagrama de Interfaz: Consulta Saldo en Credito

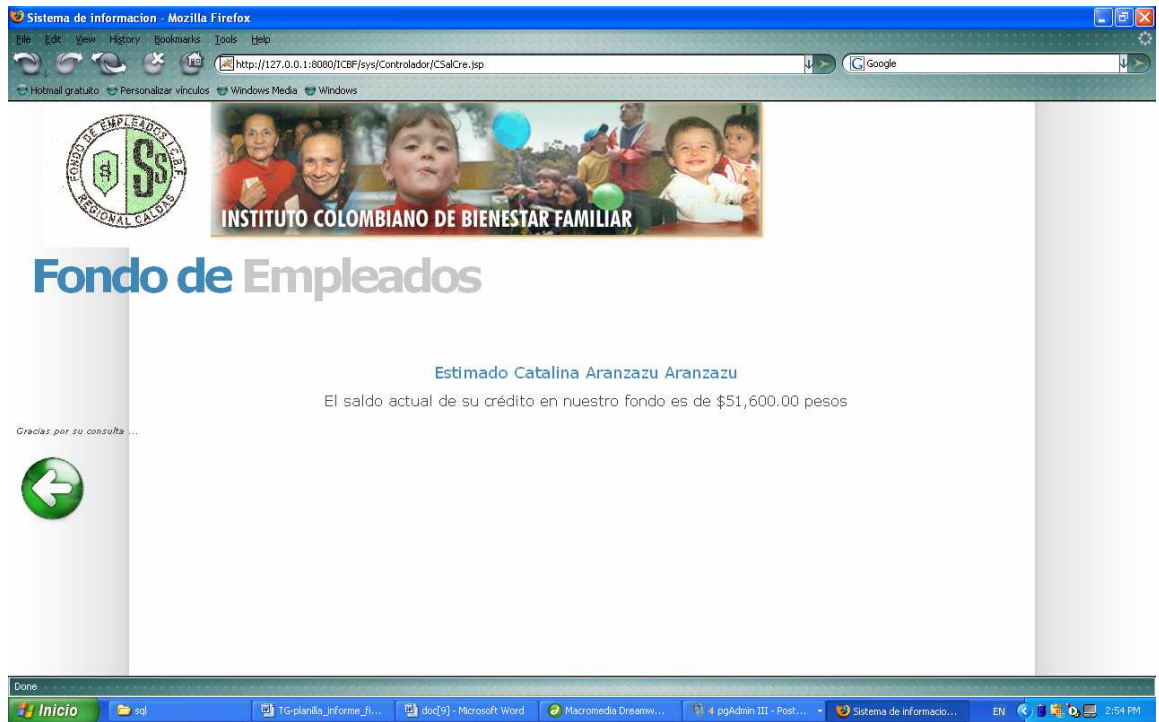


Figura 128 Diagrama de Interfaz: Consulta Saldo en crédito

Diagrama de Componentes: Consulta Saldo en crédito

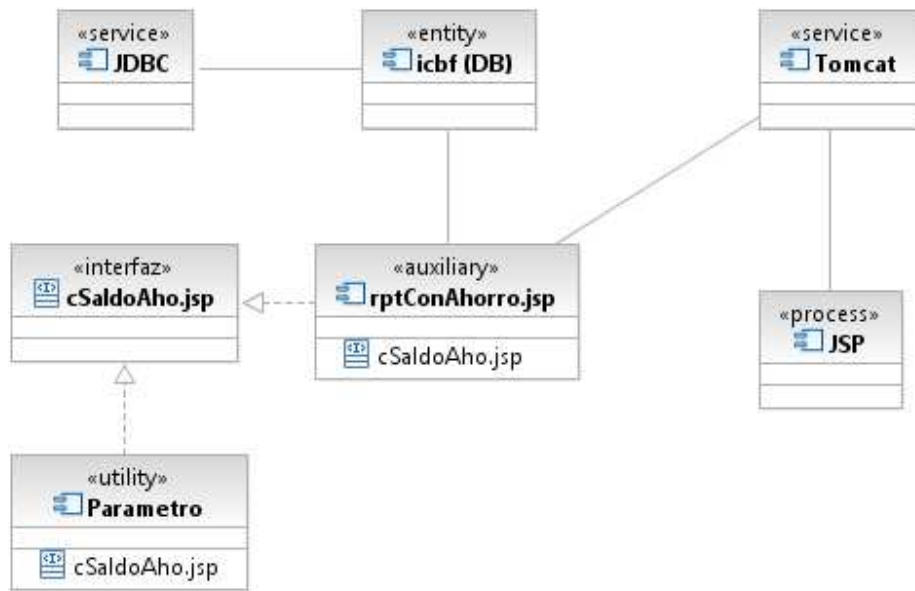


Figura 129 Diagrama de Componentes: Consulta Saldo en credito

Código: Consulta Saldo en crédito

```
Object Aso=session.getAttribute("theMan");
String ElApe1="";
String ElApe2="";
String ElNom="";
if (Aso != null && Aso.equals("") != true){
    co objC = new co();
    StringBuffer sb= new StringBuffer();
    ResultSet rs=null;
    try{
        int x=objC.conectar();
        if (x == -1){ out.println("<h4> Problemas en conexion </h4>"+x);}
        String sql="select * from Creditos where Cuenta='"+Aso+"'";
        rs=objC.consultar(sql);

        if (rs.next()){
            String sqlTmp="select Apellido1 from Asociado where
Cuenta='"+Aso+"'";
            ElApe1=objC.buscarS(sqlTmp);
            sqlTmp="select Apellido2 from Asociado where
Cuenta='"+Aso+"'";
            ElApe2=objC.buscarS(sqlTmp);
            sqlTmp="select Nombre from Asociado where Cuenta='"+Aso+"'";
            ElNom=objC.buscarS(sqlTmp);
            sqlTmp="select SaldoCredito from Cuentas where
Cuenta='"+Aso+"'";
            NumberFormat nf = NumberFormat.getCurrencyInstance(Locale.US);
            Object salidaF= nf.format( Double.parseDouble(
objC.buscarS(sqlTmp) ));
            %>
            </p>
            <p align="center" class="blue Estilo1"> Estimado <%=ElNom%> <%=ElApe1%>
<%=ElApe2%> </p>
            <p align="center" class="Estilo2"> El saldo actual de su crédito en nuestro
fondo es de <%=salidaF%> pesos</p>
            <p class="Estilo3">Gracias por su consulta ... </p>
            <p class="Estilo3"><a href="javascript:back()"></a></p>
            <p>
                <%
                    }
                else{
                    String Datos = ElNom + " " + ElApe1 + " " + ElApe2;
                    out.println("<h4> El Asociado "+Aso+" no fue encontrado </h4>");
                    out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>
                    ");
                }
                rs.close();
            }
            catch(Exception e){out.println("Atencion: "+e);}
        }
    }
    else{
        out.println("<h4> Debera ingresar un Código </h4>");
        out.print("
href=\"javascript:history.back()\"><font face=\"Verdana, Arial, Helvetica, sans-
serif\">&lt;&lt;</font></a></p>
        ");
    }
}
```

ANEXO C MODELO DE PRUEBAS

Aplicación al sistema del Fondo de Empleados (ICBF)

Es necesario recordar que la informática tiene como función principal mejorar los procesos para hacer la vida más fácil. En calidad es necesario contestar las preguntas de forma correcta y adecuada para poder obtener buenos resultados.

Hay definiciones que toman como el software como un arte y otras la toman como ingeniería, lo que si es necesario, según algunos autores: el sistema de calidad es absolutamente necesario.

Si nos preguntamos ¿dónde se implementa la calidad?, obtendremos que la respuesta es en el proceso, ya que de ésta forma obtendremos un buen producto. Con estos lineamientos sabremos que en ninguna etapa tendremos errores, cabe anotar que todo el personal debe estar involucrado en todos esos procesos.

La calidad es objetiva en todas las empresas, se pueden obtener buenos resultados en factor de empresa, pasando por satisfacción de clientes, eficiencia en procesos y reducción de costos de operaciones. También puede representar para las empresas cierto impacto en su representación del mercado, aumentando sus costos y reputación con los clientes, en algunos casos se tienen implicaciones internacionales en aspectos de marcas.

Los sistemas de información empresarial se alimentan de datos, por lo que es necesario hacer hincapié en la información, analizando su calidad y disponibilidad. Hoy en día la mayor parte de los sistemas transaccionales, los cuales en su estructura son analíticos, en orden de generación de conocimiento. Las técnicas para generación de conocimiento tenemos minería de datos, la cual es una de las más usadas por los actores que implican dichas tecnologías.

El conocimiento es la base de la toma de decisiones, la idea es tomarlas en forma no intuitiva si no con base a datos reales.

Antes de ejecutar una prueba, se realiza una revisión del código centrada en todos los cambios en el código. Las revisiones de código son el método más efectivo y eficaz para validar la calidad del código y asegurarse contra defectos de código y errores en la ejecución. Las pruebas garantizan que se valide en primer lugar el área débil o crítica principal identificada mediante revisión del código o evaluación de riesgos, puesto que si falla, la prueba no puede continuar.

Código Error	Módulo	Fecha y Hora	Descripción	Hallazgo
1	Inicio	Feb 13 11:00 AM	Módulo para el ingreso al aplicativo. Login de usuarios	Al dar clic después de digitar el usuario sale un error, no se si es de instalación o de funcionamiento. Envío anexo con el error. Anexo1
2	Estado de cuenta	Feb. 19 7:40 PM	Módulo para consultar el estado de crédito y el estado de cuenta en un periodo	Selecciono un periodo o Todos o 0 y al dar clic en el botón en consultar me sale un error. Envío el error. Anexo2
3	Módulo Administrador	Mar 03 07:04 PM	Módulo para administrar el sistema	Al estar ingresado como Administrador el sistema no me deja continuar con ninguna consulta. Además cuando dice Bienvenido aparece un numero que me imagino es el código de cada uno de los usuarios. Envío el error. Anexo3.
4	Módulo Procesador	Jun 17 11:24 AM	Módulo para generar el reporte mensual	Al generar un mes determinado el mensaje de salida se esta duplicando
5	Módulo	Junio 17	Módulo para	Al generar este

	Procesador	11:24 AM	generar el reporte de las solicitudes	reporte sale un error y no me genera el reporte.
6	Módulo Usuario	Junio 17 11:24 AM	Módulo para consultar los créditos de un asociado	En la información de Cuotas Pagadas e Intereses están saliendo datos negativos (-1)
7	Módulo Usuario	Junio 17 11:30 AM	Módulo para consultar los ahorros de un asociado	Esta saliendo mal ubicada la información en la pantalla
8	Módulo Usuario	Junio 17 11:32 AM	Módulo para consultar el saldo de el crédito de un asociado	Cuando el asociado no es encontrado debe mostrar el nombre en el mensaje y no el numero de cuenta
9	Módulo Usuario	Junio 17 11:35 AM	Módulo para consultar el crédito de un asociado logueado en el sistema	La información esta saliendo duplicada en la pantalla
10	Módulo Analista	Junio 17 11:45 AM	Módulo para consultar todos los ahorros generados de los asociados	Se requiere que además del numero de Cuenta salga también el Nombre completo
11	Módulo Analista	Junio 17 11:50 AM	Módulo para consultar todos los créditos generados de los asociados	Se requiere que además del numero de Cuenta salga también el Nombre completo
12	Módulo Analista	Junio 17 12:00 PM	Módulo para sacar el reporte de los créditos y	Se requiere que además del número de Cuenta salga

			los ahorros generados por todos los asociados	también el Nombre completo.
13	Módulo Administrador	Junio 17 01:00 PM	Módulo para modificar todas las clases parametrizables del sistema	Al modificar un dato se modifican todos, falta un filtro en el query.
14	Módulo Usuario	Junio 17 01:10 PM	Módulo para registrar un crédito	Al solicitar un credito no permite que los intereses sean de tipo flotante, solo entero
15	Módulo Analista	Junio 17 01:20 PM	Módulo para consultar un numero de ticket de un credito registrado	Cuando no hay información del ticket solicitado no debe salir información negativa.
16	Módulo Procesador	Junio 17 03:00 PM	Módulo para registrar cartera de un asociado determinado	Cuando el usuario fue eliminado, la información de este en la pantalla aparece negativa y simplemente no debería aparecer nada ya que se pretende solo mostrar el nombre completo del asociado activo.

ANEXO D CREACIÓN DE LA BASE DE DATOS

```
DROP DATABASE icbf;  
CREATE DATABASE icbf  
  WITH OWNER = postgres  
  ENCODING = 'LATIN1'  
  TABLESPACE = pg_default;
```

```
\c icbf;
```

```
CREATE TABLE Creditos (  
  Creditor varchar NOT NULL,  
  Cuenta varchar NOT NULL,  
  Fecha varchar NOT NULL,  
  Hora varchar NOT NULL,  
  Monto varchar NOT NULL,  
  idPlazo varchar NOT NULL,  
  Archivo varchar NOT NULL,  
  idClaseCreditos varchar not null,  
  CuentaMes int4,  
  CONSTRAINT cCredito PRIMARY KEY (Credito, Cuenta) USING INDEX  
  TABLESPACE pg_default  
) WITHOUT OIDS  
TABLESPACE pg_default;  
ALTER TABLE Creditos OWNER TO postgres;
```

```
CREATE TABLE ClaseCreditos(  
  idClaseCreditos varchar not null,  
  Nombre varchar not null,  
  CONSTRAINT cidClaseCreditos PRIMARY KEY (idClaseCreditos) USING INDEX  
  TABLESPACE pg_default  
) WITHOUT OIDS  
TABLESPACE pg_default;  
ALTER TABLE ClaseCreditos OWNER TO postgres;
```

```
CREATE TABLE Actividades(  
  idActividad varchar not null,  
  Nombre varchar not null,  
  TiempoEstimado varchar,  
  CONSTRAINT cActividades PRIMARY KEY (idActividad) USING INDEX  
  TABLESPACE pg_default  
) WITHOUT OIDS
```

```

TABLESPACE pg_default;
ALTER TABLE Actividades OWNER TO postgres;

CREATE TABLE Rutas(
    idRutas varchar not null,
    idActividad varchar not null,
    Cuenta varchar NOT NULL,
    Descripcion varchar not null,
    Fecha varchar NOT NULL,
    Hora varchar NOT NULL,
    CONSTRAINT cRutas PRIMARY KEY (idRutas, idActividad, Cuenta) USING INDEX
TABLESPACE pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE Rutas OWNER TO postgres;

CREATE TABLE SolicitudCredito(
    Cuenta varchar NOT NULL,
    Fecha varchar NOT NULL,
    Hora varchar NOT NULL,
    Monto varchar NOT NULL,
    idPlazo varchar NOT NULL,
    idClaseCredito varchar not null,
    Estado varchar,
    CONSTRAINT cSolicitudCredito PRIMARY KEY (Cuenta) USING INDEX
TABLESPACE pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE SolicitudCredito OWNER TO postgres;

CREATE TABLE AnalistaCredito(
    idAnalista varchar not null,
    idSolCre varchar not null,
    Observaciones varchar,
    Estado varchar,
    CONSTRAINT cAnalistaCredito PRIMARY KEY (idAnalista, idSolCre) USING
INDEX TABLESPACE pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE AnalistaCredito OWNER TO postgres;

CREATE TABLE Cuentas (
    Cuenta varchar NOT NULL,

```

```

FechaRegistro varchar NOT NULL,
HoraRegistro varchar NOT NULL,
Porcentaje varchar NOT NULL,
Salario varchar NOT NULL,
SaldoAhorrado varchar not null,
SaldoCredito varchar not null,
Estado varchar Not Null,
CONSTRAINT cCuentas PRIMARY KEY (Cuenta) USING INDEX TABLESPACE
pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE Cuentas OWNER TO postgres;

```

```

CREATE TABLE Cartera (
Cartera varchar NOT NULL,
Cuenta varchar NOT NULL,
Fecha varchar NOT NULL,
Hora varchar NOT NULL,
Saldo varchar,
CONSTRAINT cCartera PRIMARY KEY (Cartera, Cuenta) USING INDEX
TABLESPACE pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE Cartera OWNER TO postgres;

```

```

CREATE TABLE Asociado (
Cuenta varchar NOT NULL,
Apellido1 varchar NOT NULL,
Apellido2 varchar NOT NULL,
Nombre varchar NOT NULL,
Cedula varchar NOT NULL,
Telefono varchar NOT NULL,
Celular varchar NOT NULL,
EMail varchar NOT NULL,
Direccion varchar NOT NULL,
CargoActual varchar NOT NULL,
estado varchar,
CONSTRAINT cCuenta PRIMARY KEY (Cuenta) USING INDEX TABLESPACE
pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE Asociado OWNER TO postgres;

```

```

CREATE TABLE Analista (
  Código varchar NOT NULL,
  Apellido1 varchar NOT NULL,
  Apellido2 varchar NOT NULL,
  Nombre varchar NOT NULL,
  Cedula varchar NOT NULL,
  Telefono varchar NOT NULL,
  Celular varchar NOT NULL,
  EMail varchar NOT NULL,
  Direccion varchar NOT NULL,
  CargoActual varchar NOT NULL,
  CONSTRAINT cAnalista PRIMARY KEY (Código) USING INDEX TABLESPACE
pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE Analista OWNER TO postgres;

```

```

CREATE TABLE Movimiento (
  Consecutivo varchar NOT NULL,
  Cuenta varchar NOT NULL,
  Periodo varchar not null,
  FechaGeneracion varchar NOT NULL,
  Hora varchar NOT NULL,
  TipoMovim varchar NOT NULL,
  Valor varchar NOT NULL,
  Saldo varchar NOT NULL,
  Observaciones varchar Not Null,
  CONSTRAINT cConsecutivo PRIMARY KEY (Consecutivo, Periodo, Cuenta) USING
INDEX TABLESPACE pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE Movimiento OWNER TO postgres;

```

```

CREATE TABLE Periodos(
  Código int4 not null,
  Periodo varchar not null,
  Estado varchar,
  CONSTRAINT cCódigo PRIMARY KEY (Código) USING INDEX TABLESPACE
pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE Periodos OWNER TO postgres;

```

```

CREATE TABLE Plazos(
    idPlazos varchar not null,
    Cobertura varchar not null,
    InteresMensual varchar not null,
    InteresAnual varchar not null,
    meses varchar,
    CONSTRAINT cPlazos PRIMARY KEY (idPlazos) USING INDEX TABLESPACE
pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE Plazos OWNER TO postgres;

```

```

CREATE TABLE "tblActivos" (
    "cCodAct" varchar NOT NULL,
    "cDetAct" varchar NOT NULL,
    "cMarAct" varchar NOT NULL,
    "cModAct" varchar NOT NULL,
    "cSerAct" varchar NOT NULL,
    "cCodCla" varchar NOT NULL,
    "cCodUbi" varchar NOT NULL,
    "cCodRes" varchar NOT NULL,
    "cCodPro" varchar NOT NULL,
    "cFecCom" varchar NOT NULL,
    "cNumFac" varchar NOT NULL,
    "cMonAct" varchar NOT NULL,
    "cPorDep" varchar NOT NULL,
    "cMonAnu" varchar NOT NULL,
    "cMonMen" varchar NOT NULL,
    "cConMan" varchar NOT NULL,
    "cEstado" varchar NOT NULL,
    CONSTRAINT "cCodAct" PRIMARY KEY ("cCodAct") USING INDEX
TABLESPACE pg_default

```

```

) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE "tblActivos" OWNER TO postgres;
COMMENT ON COLUMN "tblActivos"."cCodAct" IS 'Para asignar el Código';
COMMENT ON COLUMN "tblActivos"."cDetAct" IS 'se refiere a la Descripción del
Activo';
COMMENT ON COLUMN "tblActivos"."cMarAct" IS 'se refiere a la Marca del
Fabricante del Activo';
COMMENT ON COLUMN "tblActivos"."cModAct" IS 'se refiere al Modelo propio del
Activo';

```

```

COMMENT ON COLUMN "tblActivos"."cSerAct" IS 'se refiere al numero Serial del
Activo, en algunos casos este será igual al Código del Activo';
COMMENT ON COLUMN "tblActivos"."cCodCla" IS 'se refiere a la Clase o Familia a la
cual pertenece el Código digitado';
COMMENT ON COLUMN "tblActivos"."cCodUbi" IS 'se refiere a la Ubicación actual del
Activo procesado';
COMMENT ON COLUMN "tblActivos"."cCodPro" IS 'se refiere al proveedor o vendedor
del Activo';
COMMENT ON COLUMN "tblActivos"."cFecCom" IS 'fecha de compra del Activo';
COMMENT ON COLUMN "tblActivos"."cNumFac" IS 'Numero de Factura de
Adquisición del Activo';
COMMENT ON COLUMN "tblActivos"."cMonAct" IS 'monto que se pago por el Activo';
COMMENT ON COLUMN "tblActivos"."cPorDep" IS 'Depreciación. según la tabla de %
de Depreciación de Tributación';
COMMENT ON COLUMN "tblActivos"."cMonAnu" IS 'el Sistema calculara este monto
con los parámetros del Monto de Adquisición y el %';
COMMENT ON COLUMN "tblActivos"."cMonMen" IS 'el Sistema calculara este valor,
dividiendo el Monto Anual entre 12 meses';
COMMENT ON COLUMN "tblActivos"."cConMan" IS 'contrato de Mantenimiento: Sí o
No (marcar)';
COMMENT ON COLUMN "tblActivos"."cEstado" IS 'Estado del Activos';

```

```

CREATE TABLE "tblClases"
(
  "cCodCla" varchar NOT NULL,
  "cDesCla" varchar NOT NULL,
  CONSTRAINT "cCodCla" PRIMARY KEY ("cCodCla") USING INDEX
TABLESPACE pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE "tblClases" OWNER TO postgres;
COMMENT ON COLUMN "tblClases"."cCodCla" IS 'Código Tipos de Activos Fijos';
COMMENT ON COLUMN "tblClases"."cDesCla" IS 'Descripcion: Computadores,
Impresoras, Maquinas de Escribir, Maquinas de Fax, Vehículos, Mobiliario de Oficina, etc';

```

```

CREATE TABLE "tblEstados"
(
  "cCodEst" varchar NOT NULL,
  "cDesEst" varchar NOT NULL,
  CONSTRAINT "cCodEst" PRIMARY KEY ("cCodEst") USING INDEX
TABLESPACE pg_default
) WITHOUT OIDS

```



```

TABLESPACE pg_default;
ALTER TABLE "tblEstados" OWNER TO postgres;
COMMENT ON COLUMN "tblEstados"."cCodEst" IS 'Código del estado de los Activos Fijos';
COMMENT ON COLUMN "tblEstados"."cDesEst" IS 'Descripcion: en operación, en mantenimiento, fuera de uso, vendido, etc';

```

```

CREATE TABLE "tblUbicaciones"
(
  "cCodUbi" varchar NOT NULL,
  "cDesUbi" varchar NOT NULL,
  CONSTRAINT "cCodUbi" PRIMARY KEY ("cCodUbi") USING INDEX
TABLESPACE pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE "tblEstados" OWNER TO postgres;
COMMENT ON COLUMN "tblUbicaciones"."cCodUbi" IS 'Código de la ubicación de los Activos Fijos';
COMMENT ON COLUMN "tblUbicaciones"."cDesUbi" IS 'Descripcion: Computo, Contabilidad, Mercadeo y Ventas, Gerencia General, Bodega, etc';

```

```

CREATE TABLE "tblResponsable"
(
  "cCodRes" varchar NOT NULL,
  "cNomRes" varchar NOT NULL,
  "cDirRes" varchar NOT NULL,
  "cTelRes" varchar NOT NULL,
  "cCorRes" varchar NOT NULL,
  CONSTRAINT "cCodRes" PRIMARY KEY ("cCodRes") USING INDEX
TABLESPACE pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE "tblResponsable" OWNER TO postgres;
COMMENT ON COLUMN "tblResponsable"."cCodRes" IS 'Código del Usuario Responsable';
COMMENT ON COLUMN "tblResponsable"."cNomRes" IS 'Nombre del Usuario Responsable';

```

```

CREATE TABLE "tblProveedores"
(
  "cNitPro" varchar NOT NULL,
  "cNomPro" varchar NOT NULL,

```

```

        "cDirPro" varchar NOT NULL,
        "cTelPro" varchar NOT NULL,
        "cCorPro" varchar NOT NULL,
        CONSTRAINT "cNitPro" PRIMARY KEY ("cNitPro") USING INDEX TABLESPACE
pg_default
) WITHOUT OIDS
TABLESPACE pg_default;
ALTER TABLE "tblProveedores" OWNER TO postgres;
COMMENT ON COLUMN "tblProveedores"."cNitPro" IS 'Código del Proveedor';
COMMENT ON COLUMN "tblProveedores"."cNomPro" IS 'Nombre del Proveedor';
COMMENT ON COLUMN "tblProveedores"."cDirPro" IS 'Direccion del proveedir';
COMMENT ON COLUMN "tblProveedores"."cTelPro" IS 'Telefono del proveedir';

CREATE TABLE Usuarios
(
    idusu int4 primary key,
    tdesusu varchar(50),
    tpass varchar(30),
    nivel varchar
)
WITHOUT OIDS;
ALTER TABLE Usuarios OWNER TO postgres;

insert into Usuarios values (1,'Administrador','1','1');
insert into Usuarios values (2,'Procesador','2','2');
insert into Usuarios values (3,'Analista','3','3');
insert into Usuarios values (4,'Catalina aranzazu','4','4');
insert into Usuarios values (5,'Lorena Echeverry','5','4');

insert into Periodos values ('1','Ene - Feb 2008','a');
insert into Periodos values ('2','Feb - Mar 2008','a');
insert into Periodos values ('3','Mar - Abr 2008','a');
insert into Periodos values ('4','Abr - May 2008','a');
insert into Periodos values ('5','May - Jun 2008','a');
insert into Periodos values ('6','Jun - Jul 2008','a');
insert into Periodos values ('7','Jul - Ago 2008','a');
insert into Periodos values ('8','Ago - Sep 2008','a');
insert into Periodos values ('9','Sep - Oct 2008','a');
insert into Periodos values ('10','Oct - Nov 2008','a');
insert into Periodos values ('11','Nov - Dic 2008','a');
--insert into Movimiento values ('0','0','0','0','0','0','0','0','0');
insert
            into
            Analista
            values
('3','Otero','Costa','Guillermo','123456','09876','312','coor@gmail.com','Bogota','Jefe');

```

```

insert                into                Asociado                values
('4','Aranzazu','Aranzazu','Catalina','123456','09876','312','coor@gmail.com','Bogota','Jefe','
a');
insert                into                Asociado                values
('5','Echeverry','Echeverry','Lorena','010101','09876','312','coor@gmail.com','Bogota','Jefe','
a');
insert into Cuentas values ('5','01/01/2008','06:00','10','12000','0','0','A');
insert into Cuentas values ('4','01/01/2008','06:00','10','12000','0','0','A');

--insert into Creditos values ('0','0','0','0','0','0','0','01');
insert into ClaseCreditos values ('01','Rotativo');
insert into ClaseCreditos values ('02','Corto plazo');
insert into ClaseCreditos values ('03','Largo plazo');
insert into Actividades values ('01','Solicitud Presentada',2);
insert into Actividades values ('02','Procesando (Analista)',2);
insert into Actividades values ('03','Credito Aprobado',2);
insert into Actividades values ('04','Credito Rechazado',2);
insert into plazos values ('01','Un Mes', '3', '24','1');
insert into plazos values ('02','Seis Meses', '2', '24','6');
insert into plazos values ('03','Un Año', '2', '24','12');
insert into plazos values ('04','Dos Años', '1', '24','24');
insert into plazos values ('05','Tres Años', '1', '24','48');

```

ANEXO E MANUAL TECNICO

Instalación de la aplicación

Para la instalación de la aplicación es necesario configurar un servidor WEB de tal forma que permita la interacción de los usuarios con el Fondo de Empleados del Instituto Colombiano de Bienestar Familiar Regional Caldas en línea desde cualquier punto de acceso.

El servidor es un elemento esencial de distribución de recursos para el proyecto y para tales efectos se usa el software TOMCAT, el cual permite la interacción de la base de datos con los usuarios, los cuales cuentan con un BROWSER (navegador) que procese las transacciones.

Los siguientes pasos son requeridos para la instalación y configuración de la aplicación:

Se deben descargar el siguiente software:

Java jdk-6-beta-windows-i586

Apache-tomcat 5.5

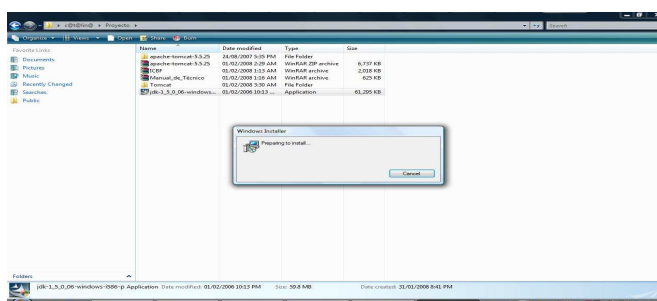
Postgresql-8.1

El driver de postgres (jdbc)

Se comienza con la instalación de cada uno, para eso siga los siguientes pasos:

Instalación de JAVA

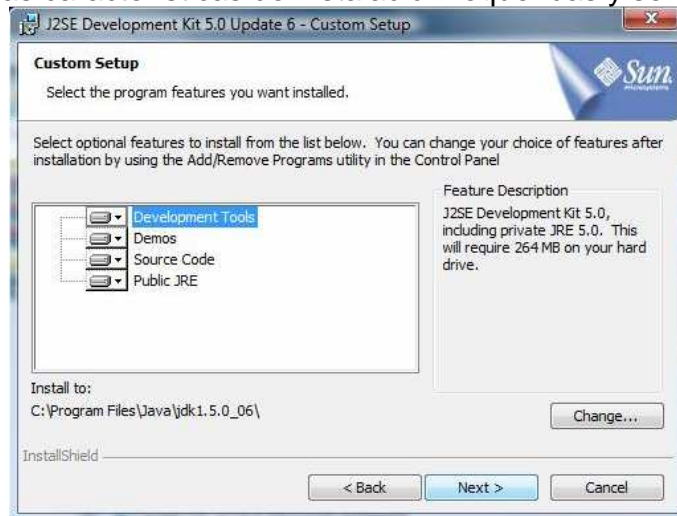
Buscamos la opción para instalar java llamada jdk-6-beta-windows-i586, ejecutamos esta opción:



En este cuadro aceptamos los términos del contrato



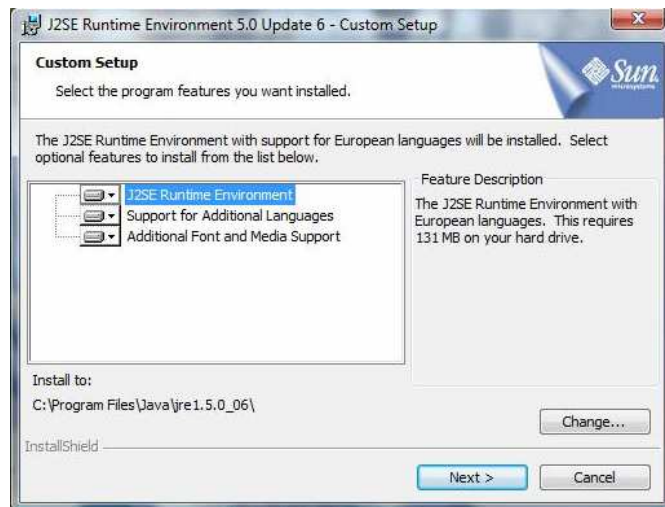
Se seleccionan las características de instalación requeridas y se da clic en Next



A continuación aparecerá el cuadro con el estado de la Instalación



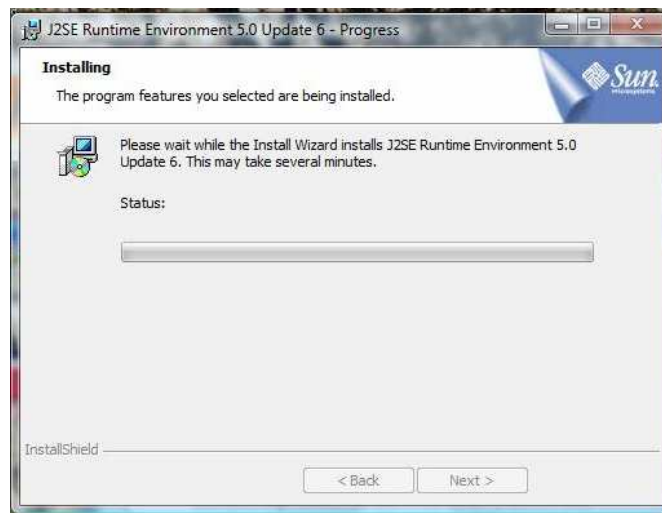
Seguimos el mismo proceso dándole siguiente (Next)



Se selecciona el browser que se quiere registrar con JAVA y se da clic en Next.



Se muestra el proceso de la instalación



La instalación ha finalizado

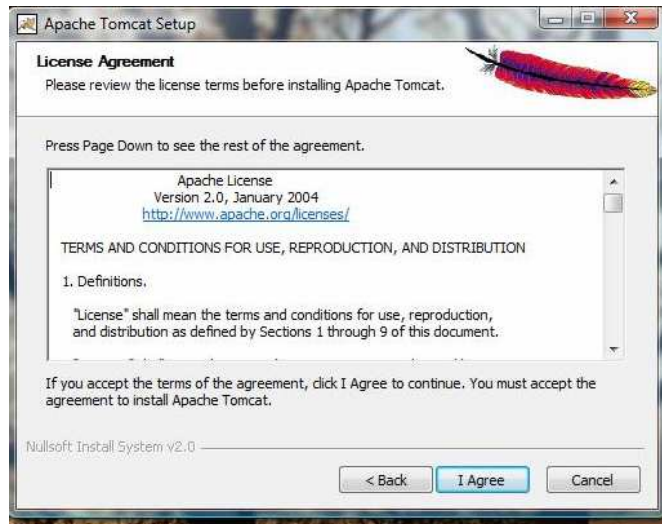


Instalar TOMCAT

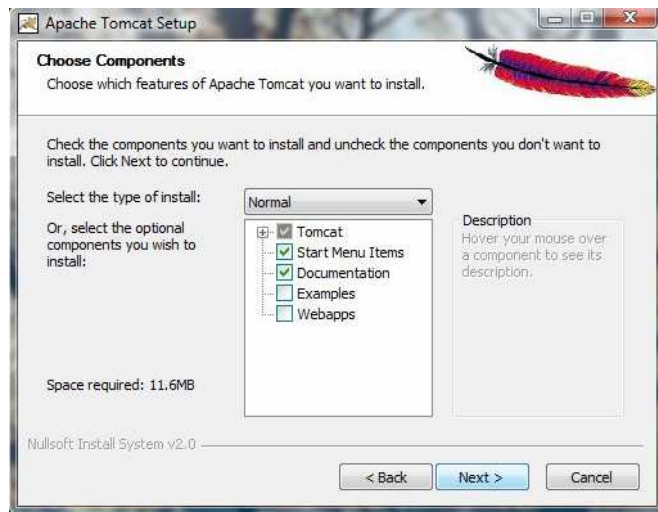
Esta es la opción para empezar a ejecutar el archivo de Tomcat 5.5



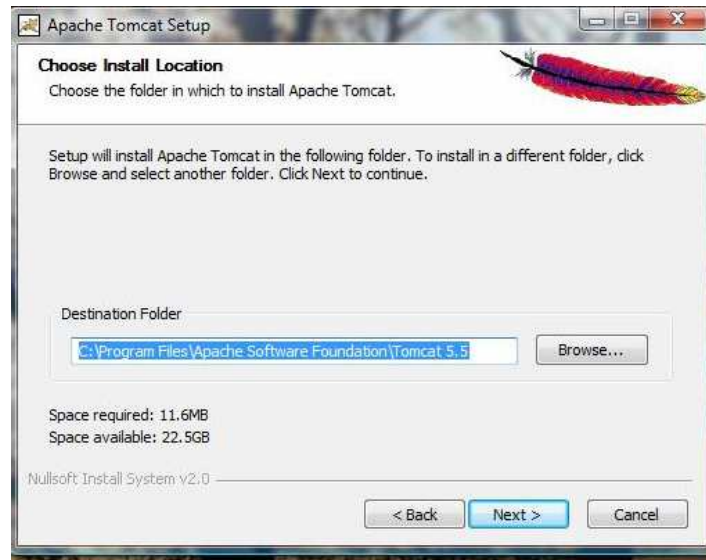
Se acepta la licencia del Apache Tomcat



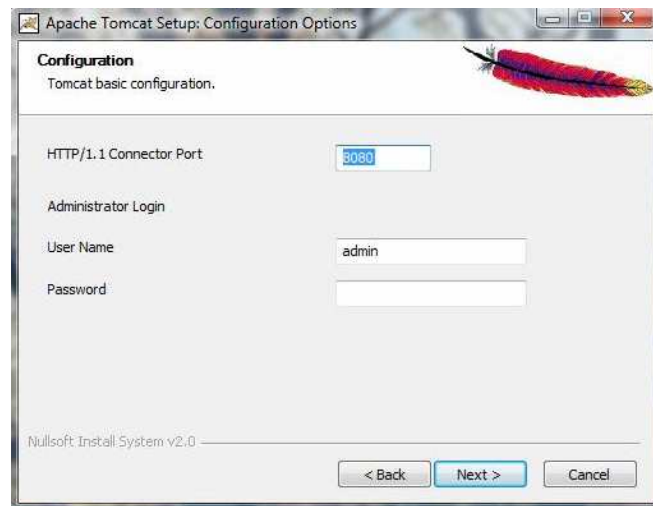
Se seleccionan los componentes y se da clic en Next



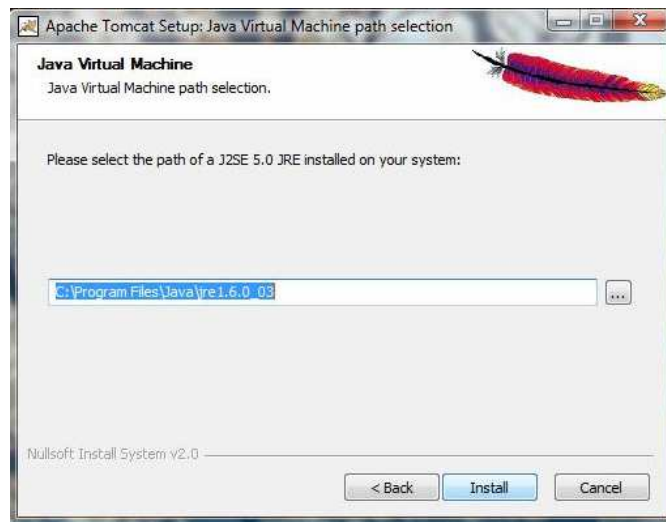
Se selecciona la ubicación de instalación del Apache Tomcat, en la mayoría de los casos se deja por defecto y se da clic en Next



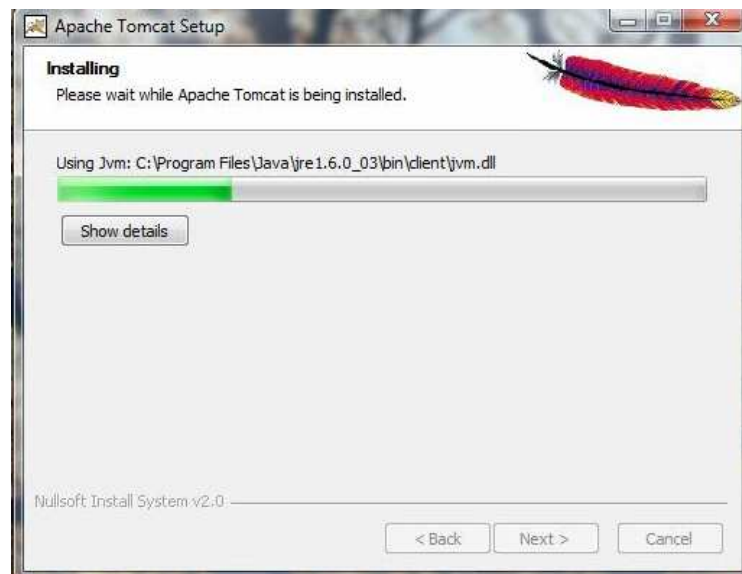
Se configure lo básico del Tomcat, se deja por defecto



Se selecciona la ruta



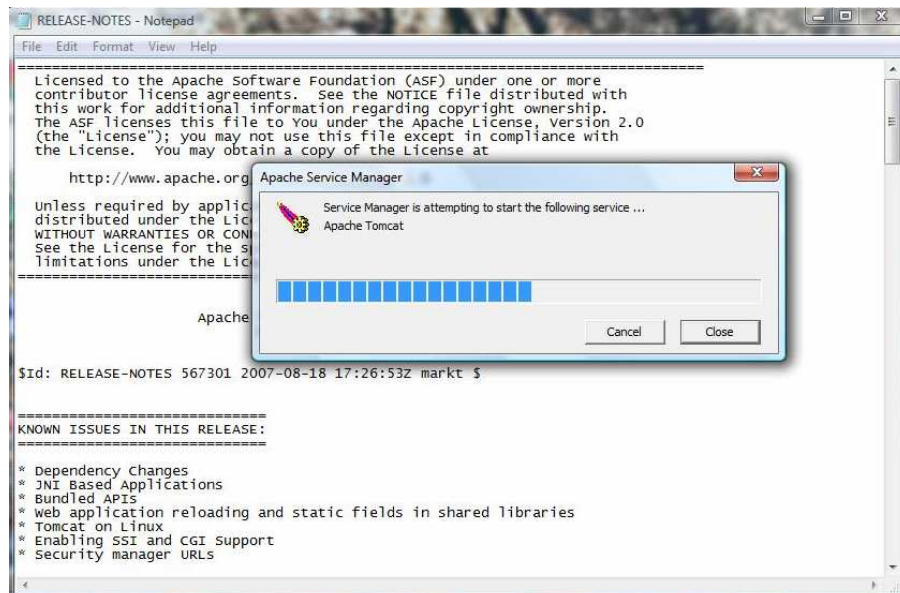
Aparece el proceso de instalación



Se completa la instalación y se da clic en Finish

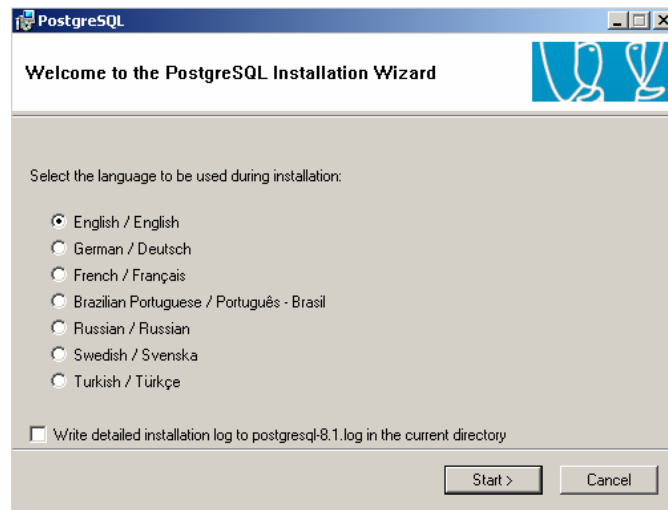


Se inicia el servicio y muestra el README del Tomcat

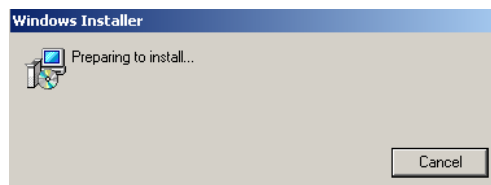


Instalación del Postgres

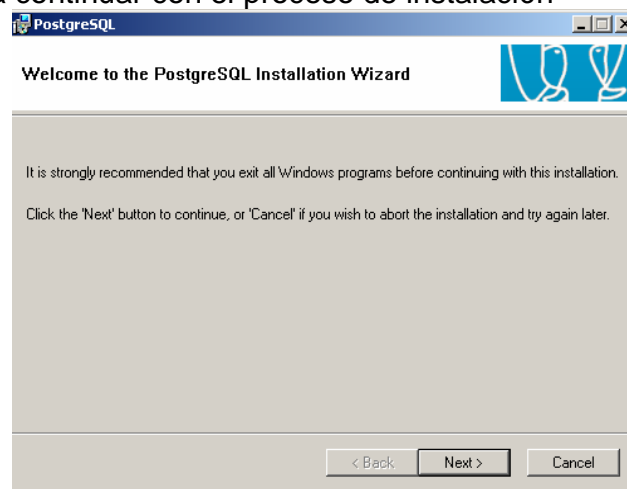
Se selecciona el lenguaje para ser usado durante la instalación



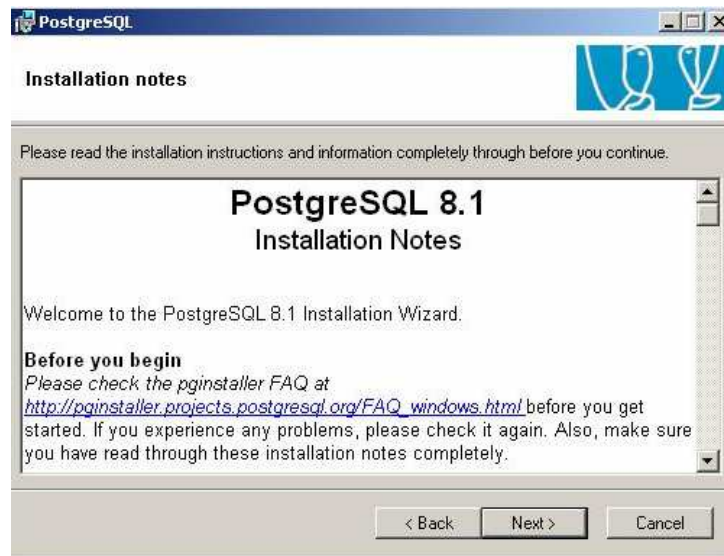
Preparando la instalación



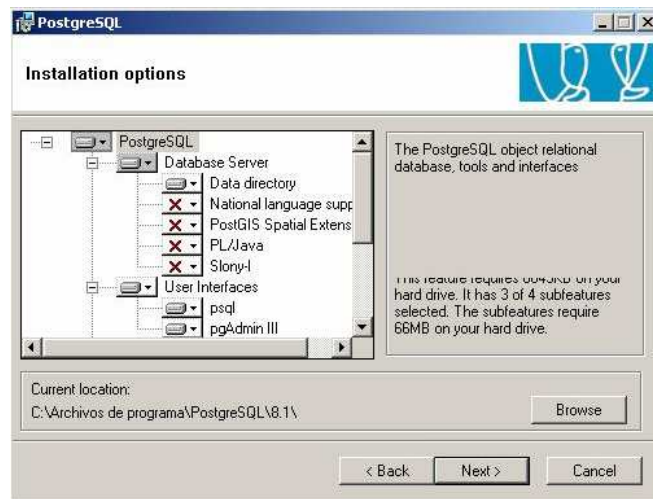
Se da clic en Next para continuar con el proceso de instalación



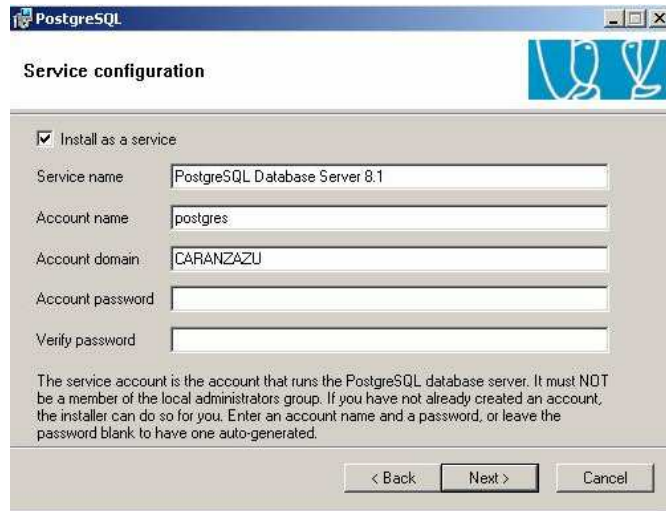
Aparecen las notas de Instalación



Opciones de Instalación y se debe seleccionar la ubicación para la instalación del postgres, se da clic en Next

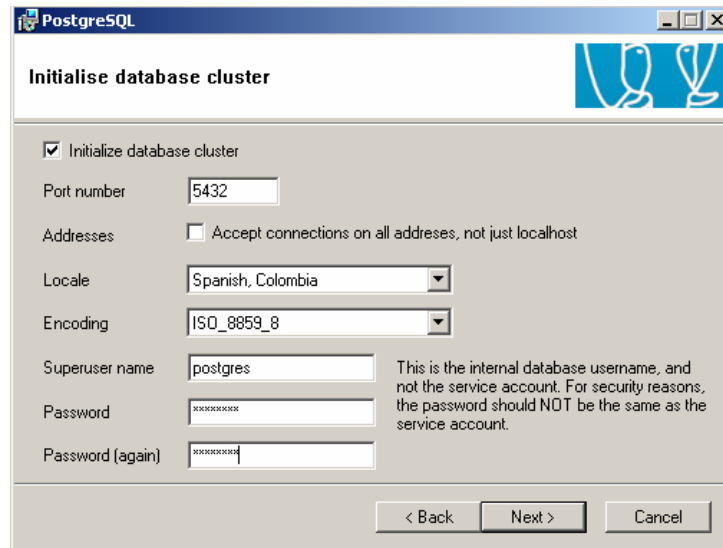


Configuración de servicios



The screenshot shows the 'Service configuration' window in the PostgreSQL installer. The 'Install as a service' checkbox is checked. The 'Service name' is 'PostgreSQL Database Server 8.1', 'Account name' is 'postgres', and 'Account domain' is 'CARANZAZU'. The 'Account password' and 'Verify password' fields are empty. A note at the bottom states: 'The service account is the account that runs the PostgreSQL database server. It must NOT be a member of the local administrators group. If you have not already created an account, the installer can do so for you. Enter an account name and a password, or leave the password blank to have one auto-generated.' Navigation buttons '< Back', 'Next >', and 'Cancel' are at the bottom.

La contraseña es el mismo usuario (postgres) y se da clic en NEXT



The screenshot shows the 'Initialise database cluster' window. The 'Initialize database cluster' checkbox is checked. The 'Port number' is 5432. The 'Addresses' checkbox 'Accept connections on all addresses, not just localhost' is unchecked. The 'Locale' is 'Spanish, Colombia' and the 'Encoding' is 'ISO_8859_8'. The 'Superuser name' is 'postgres' and the 'Password' and 'Password (again)' fields contain asterisks. A note on the right says: 'This is the internal database username, and not the service account. For security reasons, the password should NOT be the same as the service account.' Navigation buttons '< Back', 'Next >', and 'Cancel' are at the bottom.

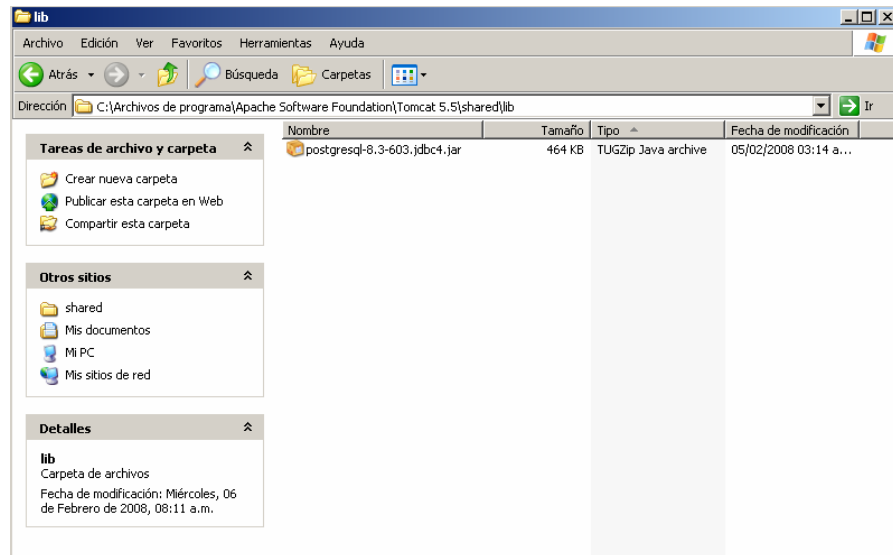
A continuación se instala Potgres.

Instalación del Software del ICBF

Tomar del CD el archivo ICBF que se encuentra adjunto y copiarlo en la carpeta C:\Archivos de programa\Apache Software Foundation\Tomcat 5.5\WEBapps\ROOT, y luego de descomprimirla allí mismo se deberá crear una carpeta llamada ICBF.

Instalación de DRIVERS

Para la puesta en marcha de la aplicación es necesario el uso de DRIVERS que permitan la conexión con las bases de datos. Para tales efectos es necesario copiar el archivo postgresql-8.3-603.jdbc4.jar en la carpeta C:\Archivos de programa\Apache Software Foundation\Tomcat 5.5\shared\lib, tal y como lo indica el siguiente gráfico:

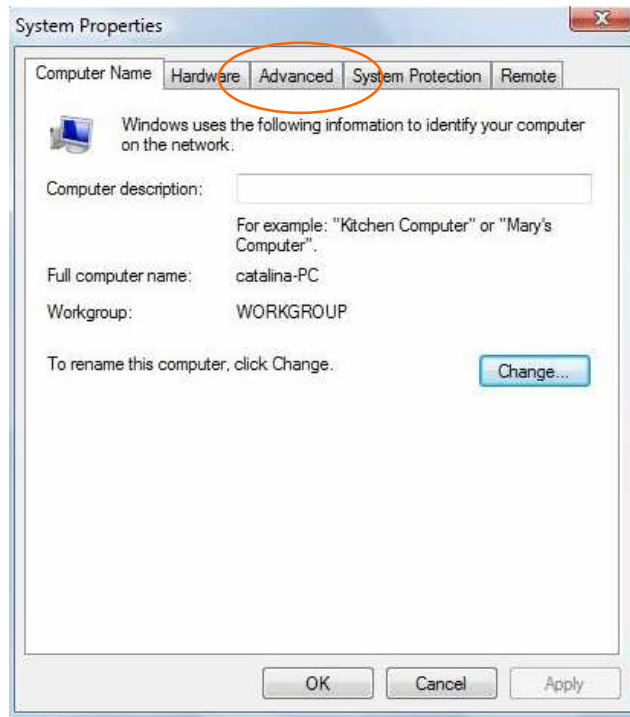


Configuración Variables de Entorno

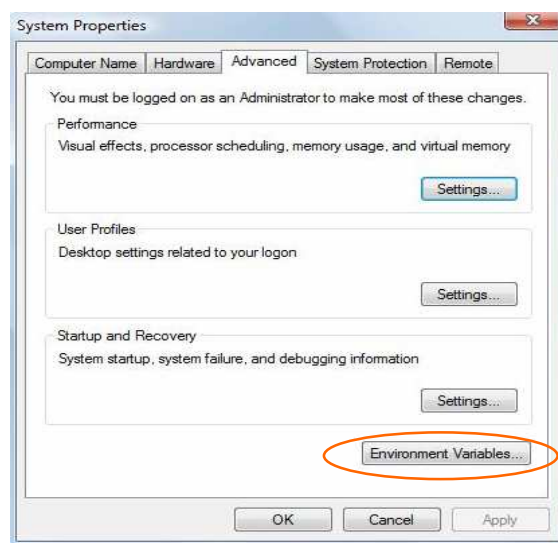
Para la instalación de las bases de datos es necesario ejecutar el SCRIP SQL que permita la creación de las bases de datos del proyecto.

Primero que todo es necesario ingresar una variable de entorno para que se pueda ejecutar el comando psql, y se logra mediante los siguientes pasos

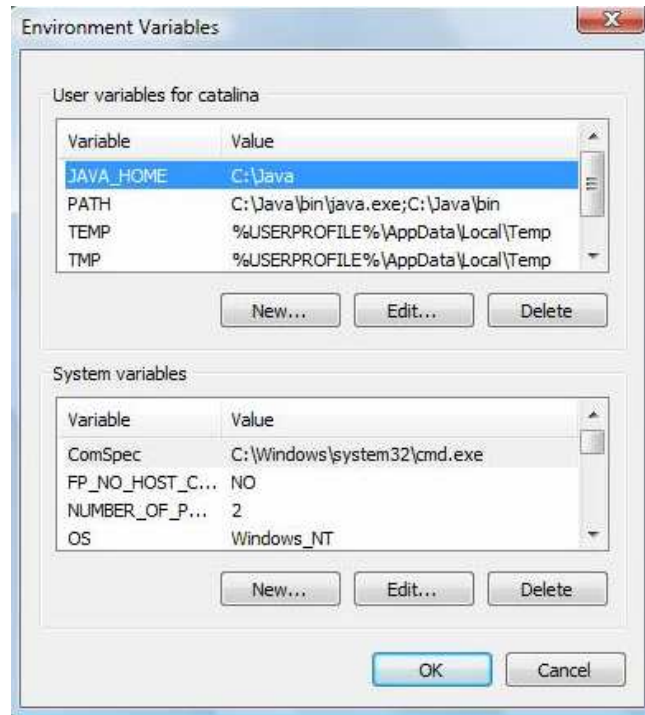
Abrir la ventana de propiedades de MI PC



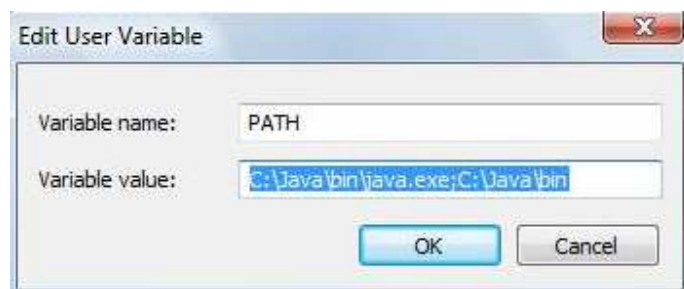
Luego de lograr abrir la ventana anterior es necesario hacer clic en la pestaña de opciones avanzadas, tal y como aparece en el siguiente gráfico:



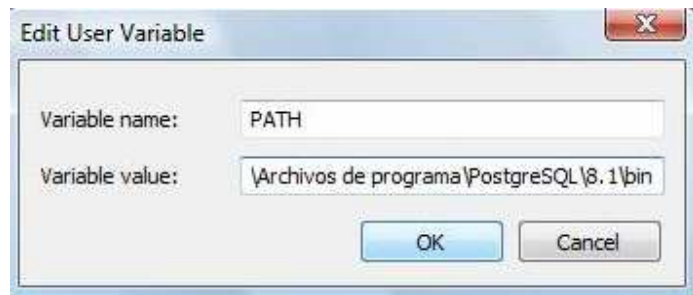
Lo anterior mostrará la siguiente ventana con las variables para el usuario y las variables del sistema.



El paso a seguir es buscar en el área de Variables del sistema la variable Path, todo con el fin de incluir el PATH del POSTGRES, para realizar esto será necesario hacer clic en el botón modificar para hacer el nuevo ingreso, observe el siguiente gráfico y aparecerá la siguiente ventana:



Para lograr incluir el PATH de Postgres es necesario agregar la cadena “;C:\Archivos de programa\PostgreSQL\8.1\bin; ” tal y como aparece en el resalto sobre el siguiente gráfico:



Para terminar el primer paso es necesario aceptar en todas las ventanas.

Ejecución del SQL

Por otro lado es necesario ingresar al símbolo de sistema (cmd) para la ejecución del programa_PSQL.

En el momento que se ingrese al símbolo de sistema es necesario entrar en la carpeta C:\Archivos de programa\Apache Software Foundation\Tomcat 5.5\WEBapps\ROOT\ICBF\sys\sql por medio de los comandos correspondientes.

Luego se deberá ingresar el siguiente comando, Donde la U deberá estar dispuesta en mayúsculas:

Psql -U postgres < init.sql

Compilación clase base

El software que se está instalando requiere de la configuración de la clase que permite la conexión con la base de datos, ésta deberá ser compilada y copiada en la carpeta C:\Archivos de programa\Apache Software Foundation\Tomcat 5.5\webapps\ROOT\WEB-INF\classes\icbf, es importante anotar que es necesario crearla, ya que en el sistema no se encuentra.

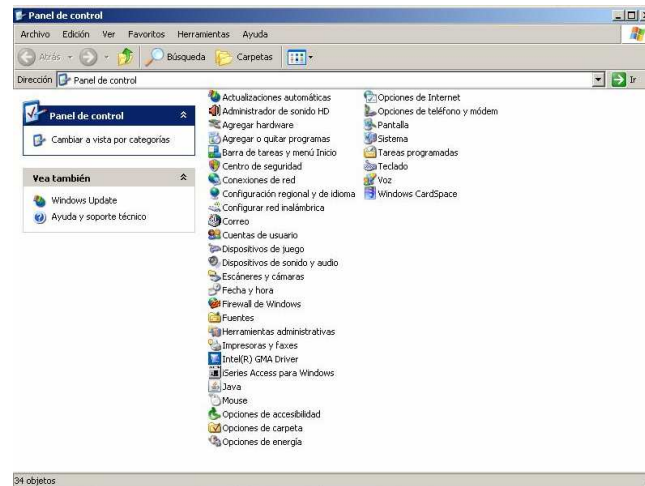
Luego, y por medio del símbolo de sistema se compilará utilizando un BAT; dicho archivo se encuentra en C:\Archivos de programa\Apache Software Foundation\Tomcat 5.5\webapps\ROOT\ICBF\class, el cual tiene como nombre **c.bat**, el cual contiene las siguientes instrucciones:

```
echo off
javac co.java
copy co.class ..\..\WEB-INF\classes\ICBF
```

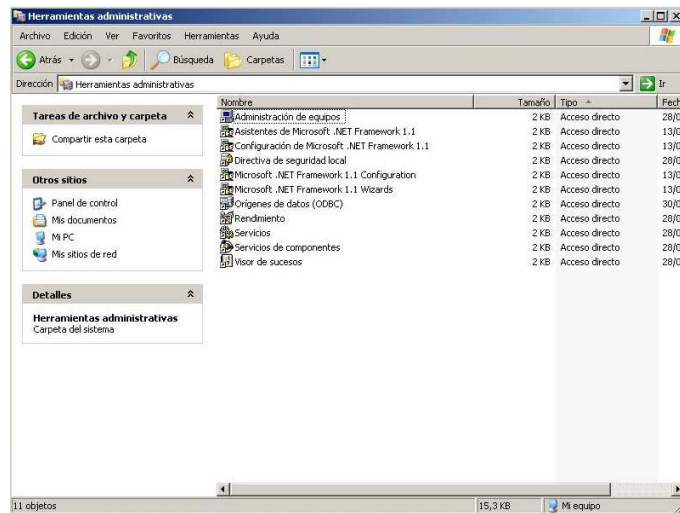
La ejecución se logra mediante el programa C y se logra ingresando simplemente **c** en dicho símbolo, tal y como lo indica la siguiente ventana:

Configuración de los DRIVES para los REPORTES

Los reportes hacen conexión con la base de datos mediante ODBC, y para tales efectos es necesario ingresar al panel de control y buscar el icono de Herramientas administrativas, abajo se ilustra un ejemplo de cómo identificarlo:



Al momento de hacer clic en dicho icono se desplegará la siguiente ventana:



Y luego se deberá hacer clic en el icono Orígenes de datos (ODBC), para luego ingresar un nuevo elemento mediante la siguiente ventana:



Luego de debe hacer clic en la pestaña DNS de sistema para que aparezca la siguiente ventana:



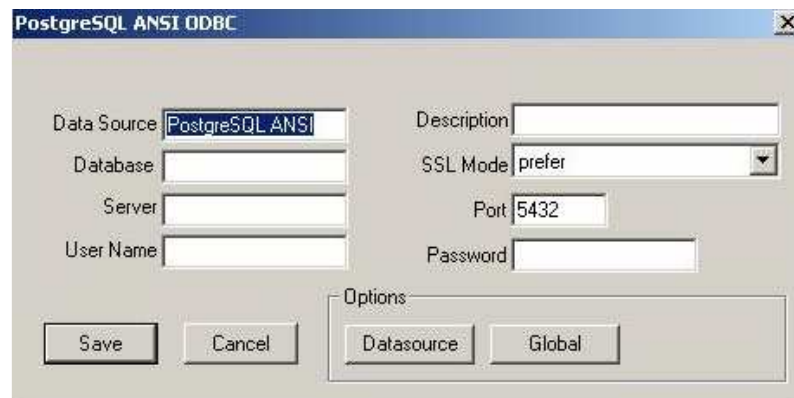
Luego hacer clic en el botón agregar, esto nos dará como resultado la siguiente ventana:



El paso a seguir buscar el nombre: PostgreSQL ANSI y presionar el botón finalizar



Esto no da como resultado la siguiente interfaz



En cada campo debe digitar:

Database: icbf

Descripción: BaseInstituto

Server: 127.0.0.1

User Name: postgres

Password: postgres