

UNIVERSIDAD TÉCNICA DEL NORTE



FACULTAD DE INGENIERÍA EN CIENCIAS
APLICADAS

CARRERA DE INGENIERÍA EN SISTEMAS
COMPUTACIONALES

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN SISTEMAS COMPUTACIONALES**

TEMA:

Desarrollo de un Sistema Web para Gestionar Cajas de Ahorro

APLICATIVO:

Sistema de Caja de Ahorro Comunitario para la Parroquia de Ambuquí

AUTOR:

Christian Giovany Ipiales Moreta.

DIRECTOR:

Ing. José Luis Rodríguez

Ibarra – Ecuador

2012

CERTIFICACIÓN

El Señor Christian Giovany Ipiales Moreta ha trabajado en el desarrollo del Proyecto de Desarrollo de un Sistema Web para Gestionar Cajas de Ahorro con el aplicativo “Sistema de Caja de Ahorro Comunitario para la Parroquia de Ambuquí”, previo a la obtención del Título de Ingeniero en Sistemas Computacionales, realizándolo con interés profesional y responsabilidad, lo cual certifico en honor a la verdad.

Ing. José Luis Rodríguez

DIRECTOR DE TESIS

CERTIFICACIÓN

Ibarra, 11 de Septiembre del 2012

Señores
UNIVERSIDAD TÉCNICA DEL NORTE
Presente

De mis consideraciones.-

Siendo auspiciantes del proyecto de tesis del Egresado CHRISTIAN GIOVANY IPIALES MORETA con CI: 1003187851 quien desarrolló su trabajo con el tema “DESARROLLO DE UN SISTEMA WEB PARA GESTIONAR CAJAS DE AHORRO” con el aplicativo “SISTEMA DE CAJA DE AHORRO COMUNITARIO PARA LA PARROQUIA DE AMBUQUÍ”, me es grato informar que se han superado con satisfacción las pruebas técnicas y la revisión de cumplimiento de los requerimientos funcionales, por lo que se recibe el proyecto como culminado y realizado por parte del egresado CHRISTIAN GIOVANY IPIALES MORETA. Una vez que hemos recibido la capacitación y documentación respectiva, nos comprometemos a continuar utilizando el mencionado aplicativo en beneficio de la parroquia de Ambuquí.

El egresado CHRISTIAN GIOVANY IPIALES MORETA puede hacer uso de este documento para los fines pertinentes en la Universidad Técnica del Norte.

Atentamente,

.....
MSC. ARMANDO FLORES

PRESIDENTE DE LA PARROQUIA DE AMBUQUÍ.



UNIVERSIDAD TÉCNICA DEL NORTE
CESION DE DERECHOS DE AUTOR DEL TRABAJO
DE INVESTIGACIÓN
A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL
NORTE

Yo, Christian Giovany Ipiates Moreta, con cédula de identidad Nro. 1003187851, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la ley de propiedad intelectual del Ecuador, artículos 4, 5 y 6, en calidad de autor del trabajo de grado denominado: **“DESARROLLO DE UN SISTEMA WEB PARA GESTIONAR CAJAS DE AHORRO”**, que ha sido desarrollado para optar por el título de: **Ingeniero en Sistemas Computacionales**, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la biblioteca de la Universidad Técnica del Norte.

Firma.....

Nombre: Christian Giovany Ipiates Moreta

Cédula: 1003187851

Ibarra al 1 día del mes de octubre del 2012



UNIVERSIDAD TÉCNICA DEL NORTE
BIBLIOTECA UNIVERSITARIA AUTORIZACIÓN
DE USO Y PUBLICACIÓN A FAVOR DE LA
UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA.

La UNIVERSIDAD TÉCNICA DEL NORTE dentro del proyecto Repositorio Digital institucional determina la necesidad de disponer los textos completos en formato digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual ponemos a disposición la siguiente información.

DATOS DE CONTACTO	
CÉDULA DE IDENTIDAD	1003187851
APELLIDOS Y NOMBRES	IPIALES MORETA CHRISTIAN GIOVANY
DIRECCIÓN	San Antonio de Ibarra Hns Mideros y Sucre 661
EMAIL	christiangiovany@hotmail.es
TELÉFONOS	2932580,093498593

DATOS DE LA OBRA	
TÍTULO	DESARROLLO DE UN SISTEMA WEB PARA GESTIONAR CAJAS DE AHORRO.
AUTOR	IPIALES MORETA CHRISTIAN GIOVANY
FECHA	1 de Octubre del 2012
PROGRAMA	PREGRADO
TITULO POR EL QUE OPTA	INGENIERÍA EN SISTEMAS COMPUTACIONALES
DIRECTOR	ING. JOSÉ LUIS RODRÍGUEZ

1. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD.

Yo, Christian Giovany Ipiales Moreta, con cédula de identidad Nro. 1003187851, en calidad de autor y titular de los derechos patrimoniales de la obra o trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en forma digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y uso del archivo digital en la biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión, en concordancia con la ley de educación Superior Artículo 143.

Firma:

Nombre: Christian Giovany Ipiales Moreta

Cédula: 1003187851

DEDICATORIA

A mi Padre, por ser un ejemplo a seguir por ser una persona trabajadora que con mucho esfuerzo nos pudo sacar adelante brindándonos el apoyo incondicional para cumplir los objetivos en la vida.

A mi Madre, por ser una persona de gran iniciativa y emprendimiento que realizo muchos sacrificios para brindarnos lo mejor. Por su cariño infinito que me permitió seguir adelante y cumplir las metas propuestas.

A mis hermanos y hermana por que han sido los amigos, los compañeros fieles en el camino hasta aquí ayudándonos mutuamente para lograr nuestros objetivos.

AGRADECIMIENTO

A Dios por bendecirme y ser mi guía durante todo en el camino de la vida ayudándome a superar los obstáculos que se me presentaron.

A mi familia por ser la base fundamental para la culminación de mis estudios quienes sin escatimar esfuerzo alguno, han sacrificado gran parte de su vida para otorgarme una formación integral.

A la Universidad Técnica del Norte y en especial a la Facultad de Ingeniería en Ciencias Aplicadas y a la Carrera de Ingeniería en Sistemas Computacionales por permitirme realizarme como profesional.

A las Señoras integrantes de las cajas de ahorro de la parroquia de Ambuquí por darme apertura y ayuda en el desarrollo del proyecto.

ÍNDICE DE CONTENIDOS

DEDICATORIA	i
AGRADECIMIENTO	ii
RESUMEN	viii
SUMMARY	ix
CAPÍTULO I	1
ANTECEDENTES GENERALES	1
1. ANTECEDENTES GENERALES	2
1.1. Introducción	2
1.2. Definición del problema.....	3
1.3. Propuesta de solución.....	3
1.4. Objetivo general	3
1.5. Objetivos específicos.....	4
1.6. Justificación social	4
1.7. Justificación económica.	5
1.8. Justificación tecnológica	5
1.9. Límites.....	5
1.10. Alcances	6
1.11. Área de créditos de la caja de ahorro	8
1.11.1. Créditos	8
1.11.2. Tipos de créditos	9
1.11.3. Garantía.....	9
1.11.4. Tipos de garantías.....	9
1.11.5. Procedimiento de colocación de créditos	10
1.11.6. Etapa de proceso de análisis.....	10
1.11.6.1. Diagrama de flujo del proceso de análisis de la solicitud	11
1.11.7. Etapa de proceso de aprobación	11
1.11.7.1. Diagrama de flujo de proceso de aprobación	12
CAPÍTULO II	13
MARCO TEÓRICO.....	13
2. MARCO TEÓRICO.....	14
2.1. Definición de software	14
2.1.1. Definición de software libre.....	14

2.2.	Definición de entorno de desarrollo	15
2.3.	Definición de lenguaje de programación.....	16
2.3.1.	Lenguaje de programación java	16
2.3.1.1.	JAVA SERVER FACES	17
2.4.	Definición de servidores de aplicaciones.....	24
2.4.1.	Servidor de aplicaciones JBOSS	25
2.5.	Definición de la arquitectura de la aplicación.....	25
2.5.1.	Capa de presentación.....	26
2.5.2.	Capa de lógica de negocio.....	26
2.5.2.1.	EJB	26
2.5.3.	Capa de acceso a datos	29
2.5.3.1.	JPA	30
CAPÍTULO III.....		32
DESARROLLO DE LA APLICACIÓN		32
3.	DESARROLLO DE LA APLICACIÓN	33
3.1.	Análisis de requerimientos	33
3.2.	Análisis de las herramientas de desarrollo de software.....	34
3.3.	Metodología RUP.....	34
3.4.	Fase de inepción.....	37
3.4.1.	Plan de desarrollo de software.....	37
3.4.2.	Elementos del modelo de casos de uso	38
3.4.3.	Diagrama del modelo de caso de uso	39
3.4.4.	Actores	40
3.5.	Casos de uso.....	41
3.5.1.	Especificación de caso de uso: Registrar usuarios y accesos	42
3.5.2.	Especificación de caso de uso: Modificar usuarios y accesos.....	44
3.5.3.	Especificación de caso de uso: Registrar cuenta de ahorros	47
3.5.4.	Especificación de caso de uso: Modificar cuenta de ahorros	50
3.5.5.	Especificación de caso de uso: Administración de ciclos	52
3.5.6.	Especificación de caso de uso: Administración de cajas de ahorros	54
3.5.7.	Especificación de caso de uso: Registrar transacción caja de ahorros	57
3.5.8.	Especificación de caso de uso: Administración de la auditoría del sistema.....	60
3.6.	Fase de elaboración	64
3.6.1.	Arquitectura de la aplicación.....	64

3.6.2.	Estructura de la aplicación	67
3.6.3.	Almacenamiento de datos	67
3.7.	Fase de construcción	68
3.7.1.	Modelo físico.....	69
3.7.2.	Modelo entidad relación.....	70
3.7.3.	Diagrama de paquetes	71
3.7.4.	Diagrama de actividades	72
3.7.4.1.	Diagrama de actividades registro de usuarios y accesos	72
3.7.4.2.	Diagrama de actividades registro cuenta de ahorros	73
3.7.4.3.	Diagrama de actividades administración de ciclos.....	74
3.7.4.4.	Diagrama de actividades Administración de cajas de ahorro.....	75
3.7.4.5.	Diagrama de actividades Registrar transacción caja de ahorro.	76
3.7.5.	Diagrama de componentes	77
3.7.6.	Diagrama de componentes y despliegue	78
3.8.	Fase de transición.....	79
3.8.1.	Caso de prueba registro de usuarios y accesos.....	79
3.8.2.	Caso de prueba registrar cuenta de ahorros.....	80
3.8.3.	Caso de prueba administración de ciclos	82
3.8.4.	Caso de pruebas administración de cajas de ahorros.....	83
3.8.5.	Caso de prueba registrar transacción caja de ahorro	84
CAPÍTULO IV.....		88
CONCLUSIONES Y RECOMENDACIONES.....		88
4.	CONCLUSIONES Y RECOMENDACIONES.....	89
4.1.	CONCLUSIONES	89
4.2.	RECOMENDACIONES	90
4.3.	BIBLIOGRAFÍA.....	91
4.4.	GLOSARIO DE TÉRMINOS.....	93
4.5.	ANEXOS.....	94
4.5.1.	MANUAL DE INSTALACIÓN	94
4.5.1.1.	PREREQUISITOS	94
4.5.1.2.	INSTALACIÓN DE HERRAMIENTAS	94
4.5.2.	MANUAL DE USUARIO	108

ÍNDICE DE FIGURAS

FIGURA 1: BLOQUES MODULARES DEL SISTEMA CAJA DE AHORROS COMUNITARIO	7
FIGURA 2: PROCESO DE ANÁLISIS DE SOLICITUD DE CRÉDITO	11
FIGURA 3: PROCESO DE APROBACIÓN DE CRÉDITO	12
FIGURA 4: PATRÓN DE DISEÑO MVC	19
FIGURA 5: CICLO DE VIDA DEL PROCESAMIENTO DE UNA SOLICITUD EN JSF.....	20
FIGURA 6: ÁRBOL DE COMPONENTES DEL LADO DEL SERVIDOR UI	21
FIGURA 7: APLICACIÓN DE LOS VALORES DE UNA SOLICITUD (HTTP REQUEST)	22
FIGURA 8: ACTUALIZACIÓN DE LAS PROPIEDADES DEL MODEL.....	23
FIGURA 9: INVOCAR LA APLICACIÓN.....	23
FIGURA 10: PRODUCIR RESPUESTA JSF.....	24
FIGURA 11: FUNCIONAMIENTO DE COMPONENTE EJB	27
FIGURA 12: TIPOS DE EJBS	28
FIGURA 13: ARQUITECTURA JPA	30
FIGURA 14: DIAGRAMA GENERAL DE RUP	35
FIGURA 15: DIAGRAMA GANTT.....	38
FIGURA 16: CRONOGRAMA DEL PROYECTO.....	38
FIGURA 17: MODELO GENERAL DE CASOS DE USO Y ACTORES DEL NEGOCIO.....	40
FIGURA 18: CASO DE USO REGISTRAR USUARIOS Y ACCESOS.....	42
FIGURA 19 CASO DE USO MODIFICAR USUARIOS Y ACCESOS.....	44
FIGURA 20: CASO DE USO REGISTRAR CUENTA DE AHORROS.....	47
FIGURA 21: CASO DE USO MODIFICAR CUENTA DE AHORROS	50
FIGURA 22: CASO DE USO ADMINISTRACIÓN DE CICLOS	52
FIGURA 23: CASO DE USO ADMINISTRACIÓN CAJAS DE AHORROS	54
FIGURA 24: CASO DE USO REGISTRAR TRANSACCIÓN CAJA DE AHORRO.....	57
FIGURA 25: CASO DE USO ADMINISTRACIÓN DE LA AUDITORÍA DEL SISTEMA.....	60
FIGURA 26: VISTA LÓGICA DE LOS PATRONES DE LA ARQUITECTURA	65
FIGURA 27: ARQUITECTURA DE LA APLICACIÓN.....	66
FIGURA 28: MODELO FÍSICO	69
FIGURA 29: MODELO ENTIDAD RELACIÓN	70
FIGURA 30: DIAGRAMA DE PAQUETES MODELO	71
FIGURA 31: DIAGRAMA DE PAQUETES WEB.....	71
FIGURA 32: DIAGRAMA DE ACTIVIDADES REGISTRO USUARIOS Y ACCESOS	72
FIGURA 33: REGISTRO CUENTA AHORROS.....	73
FIGURA 34: DIAGRAMA DE ACTIVIDADES ADMINISTRACIÓN DE CICLOS	74
FIGURA 35: DIAGRAMA DE ACTIVIDADES ADMINISTRACIÓN DE CAJAS DE AHORRO.....	75
FIGURA 36 : DIAGRAMA DE ACTIVIDADES REGISTRAR TRANSACCIÓN CAJA DE AHORROS	76
FIGURA 37 : DIAGRAMA DE COMPONENTES.....	77
FIGURA 38: DIAGRAMA DE COMPONENTES DE DESPLIEGUE	78

ÍNDICES DE TABLAS

TABLA 1: TIPOS DE CRÉDITOS	9
TABLA 2: PLAN DE DESARROLLO DE SOFTWARE.....	37
TABLA 3: ACTORES CASO USO REGISTRAR USUARIOS	43
TABLA 4: ACTORES CASO USO MODIFICAR USUARIOS	45
TABLA 5: ACTORES CASO USO REGISTRAR CUENTAS AHORROS	48
TABLA 6 : ACTORES CASO USO MODIFICAR CUENTAS DE AHORROS	51
TABLA 7 : ACTORES CASO USO ADMINISTRACIÓN DE CICLOS.....	53
TABLA 8: ACTORES CASO USO ADMINISTRACIÓN DE CAJAS DE AHORROS.....	55
TABLA 9: ACTORES CASO USO REGISTRAR TRANSACCIÓN CAJA DE AHORROS	58
TABLA 10: ACTORES CASO USO ADMINISTRACIÓN DE LA AUDITORÍA DEL SISTEMA	61

RESUMEN

Las cajas de ahorro comunitarias nacen con el único fin de ayudar a las personas de escasos recursos debido a que en la banca tradicional se los considera personas sin capacidad de pago, con lo cual no pueden acceder a créditos rápidos y oportunos.

En los tiempos actuales es elemental contar con un crédito para solventar una idea de emprendimiento que se tenga en mente.

La mayoría de personas que conforman las cajas de ahorros en la parroquia de Ambuquí se dedican a la elaboración de artesanías y a la producción de de varios productos agrícolas por lo es indispensable contar con un crédito para la comprar insumos agrícolas y materias primas.

El objetivo de este proyecto es contar con un sistema informático que permita atender todos los requerimientos de cada una de las cajas de ahorros existentes en la parroquia de Ambuquí en los servicios que brinda como es el ahorro, el ahorro en los fondos de reserva, los créditos, siendo los más beneficiados los socios que conforman cada una de las cajas de ahorros de la parroquia de Ambuquí.

La investigación de nuevas tecnologías y uso del conocimiento que se adquiere en las aulas de la Universidad hacen que tengamos la capacidad para dar soluciones prácticas y reales a problemáticas de la vida cotidiana, ayudando a si a un conglomerado de personas a crecer económicamente y socialmente, nosotros como estudiantes ponemos en práctica lo aprendido y obtenemos mucha más experiencia para el desenvolvimiento en el campo profesional.

El sistema de caja de ahorro comunitario es una aplicación que permitirá agilizar los procesos que se realizan en cada una de las cajas de ahorros existentes permitiéndoles a los socios realizar el trabajo más rápido y eficiente.

SUMMARY

The communities savings banks are arise with the only finish of helping to the persons of poor resources by the fact that the traditional banking it considered to be persons without capacity of payment, what they can't accede to quick and opportune credits.

In Present times the process a credit is elementary to solve an idea of beginning that is gets in mind.

Most of people who shape the savings banks in Ambuquí dedicate their time to the production of crafts and to the production of numerous agricultural products for this reason is indispensable take a credit to buy agricultural and principal materials.

The aim of this project is poses with computer system that allows attending all the requirements of each one of the existing savings banks in Ambuquí town in the services that it has the savings banks, the saving in the reserve fund and credits, being more beneficiaries the associates who form of each saving bank of Ambuquí Town.

The investigation of new technologies and use of the knowledge that is learned in the classrooms of the University promote the aptitude to give practical and real solutions to problematic of the daily living, It's helping so to a people's conglomerate to growing economically and socially, we as students make practice what we learn and we get much more experience for the development in the professional field.

The savings bank system community is an application which will allow speeding the processes that are realized at each of the existing savings banks, in order to Associates to make fast and efficient work.

CAPÍTULO I

ANTECEDENTES GENERALES



1. ANTECEDENTES GENERALES

1.1. Introducción

El presente proyecto de tesis tiene la finalidad de solucionar un problema en la parroquia de Ambuquí el mismo que ayudará a la parroquia a automatizar el trabajo que se realiza en cada una de las cajas de ahorro existentes en la parroquia, permitiendo realizar los procesos de forma automática, como un aporte de parte de la Universidad Técnica del Norte para la comunidad.

Para dar solución a esta problemática, conjuntamente con los socios de las cajas de ahorros se ve la necesidad de realizar una aplicación web la misma que beneficiará a cerca de trece cajas de ahorros existentes en la parroquia.

Las cajas de ahorro nacen con la única idea de ser una ayuda para la comunidad, debido a que las personas de los sectores más pobres tienen un acceso restringido para la obtención de un crédito en la banca tradicional ya que se les considera sin capacidad de pago, hoy en día obtener un crédito en una institución financiera se ha convertido en algo muy necesario para realizar un emprendimiento de algún proyecto personal o en sociedad, es imprescindible contar con oportuno y rápido acceso a un crédito que pueda viabilizar un proyecto de emprendimiento ayudando a las personas crecer económicamente.

En la parroquia de Ambuquí, las cajas de ahorro se establecen hace doce años.

Para las personas de la parroquia de Ambuquí, que necesitan ahorrar y obtener créditos de forma rápida y oportuna, esta modalidad de ahorro y crédito se ha convertido en algo muy necesario en el diario vivir.

Las cajas de ahorro son asociaciones creadas con un determinado número de socios los mismos que obligatoriamente deben asistir cada semana del año a reunirse en un lugar determinado para depositar el ahorro correspondiente a la semana según el valor que se fija semanalmente, esto es de acuerdo al número de acciones que se adquiere a principio de cada ciclo.

El proyecto se denomina Sistema de Caja de Ahorro Comunitaria (SCAC) el cual permitirá realizar procesos automáticos sobre las cajas de ahorro con lo cual los socios ya no realizarán los procesos de forma manual y obsoleta, para realizar el proyecto se utilizará herramientas

de software libre con la finalidad de que a la parroquia no le cueste ningún valor por cuestiones de licenciamiento.

Esperando que el presente proyecto sea de gran aporte para las personas que van a ser beneficiadas y también para los lectores sobre la tecnología utilizada para este proyecto aplicaremos todos los conocimientos de lo que se aprendió en las aulas y las más actuales herramientas tecnológicas para elaborar esta clase de proyectos.

1.2. Definición del problema

La parroquia de Ambuquí desde varios años ha venido realizando esta modalidad de ahorro, se lo ha venido realizando de una manera no automática, con lo cual se hace tedioso al momento de realizar las transacciones en las cuentas de cada uno de los socios, se ha venido manejando de forma manual realizando cálculos con herramientas obsoletas, con lo cual existe un alto riesgo de equivocarse por lo que se maneja muchos datos en cajas de ahorro.

El problema es que la parroquia no cuenta o no posee una herramienta computarizada que gestione automáticamente los procesos que se realizan en las cajas de ahorro.

1.3. Propuesta de solución

Nosotros como parte de la Universidad Técnica del Norte, con el propósito de retribuir a la sociedad lo que se aprendió en las aulas proponemos la solución de realizar un sistema informático que se utilice herramientas de última tecnología y que sean de software libre para que a la parroquia no le cueste los licenciamientos de las herramientas que se va utilizar en el desarrollo y que ayudará el cien por ciento en la forma obsoleta que se venía realizando procesos de las cajas de ahorro comunitarias en la parroquia beneficiando social y económicamente a la parroquia, con el fin de que se pueda percibir un progreso en las personas que acuden a ahorrar.

1.4. Objetivo general

Implementar un Sistema Web utilizando para el desarrollo herramientas de software libre que permita automatizar los procesos que se realizan en las cajas de ahorros de Ambuquí para que el trabajo que se realiza sea mucho más eficaz y efectivo para los socios de la caja de ahorros.

1.5. Objetivos específicos

- Realizar una adecuada ordenación de los procesos que se realizan en las cajas de ahorros.
- Realizar un estudio de los requerimientos necesarios para que la aplicación web funcione correctamente de acuerdo a la caja de ahorros.
- Reducir los tiempos y costos que implica el proceso de solicitud de crédito de los socios.
- Estudiar las herramientas apropiadas a utilizar en el desarrollo del proyecto con la creación previa de un demo de la aplicación para la posterior revisión de fallas con sus respectivos manuales para el correcto uso de la aplicación tanto para el administrador como para los demás usuarios.
- Realizar reportes generales de la caja de ahorros.
- Realizar la aplicación lo suficientemente genérica para ser aplicada en las diferentes parroquias de Imbabura.

1.6. Justificación social

De acuerdo a lo que persigue la Universidad Técnica del Norte aportando a la comunidad profesionales de excelencia y que dichos profesionales sean de aporte para la comunidad, se ve la oportunidad de ayudar a la parroquia de Ambuquí, en especial a las personas que conforman las cajas de ahorro que son personas que van a ayudar al crecimiento social y económico de la parroquia, ya que son personas que tienen gran incentivo para el ahorro y de gran entusiasmo para realizar cualquier tipo de emprendimiento, debido a que la parroquia es rica en métodos productivos ya que los habitantes de la parroquia y sus alrededores son principalmente pequeños agricultores que se dedican a la producción de varios productos agrícolas como fréjol, caña, tomate, pepino o pimiento. Otra parte de la población se dedica en menor escala al comercio y las artesanías. Todos y todas requieren de créditos ya sea para comprar insumos agrícolas o materias primas.

Con lo cual este proyecto es de gran ayuda para el crecimiento económico y social para la parroquia en general.

1.7. Justificación económica.

El presente proyecto es factible de ser realizado, por el motivo que no representa un gasto para los socios de las cajas de ahorro tanto en su elaboración como en su implantación, además que, en la época en que nos encontramos, donde el conocimiento y la información, puede mejorar nuestros ingresos económicos, estamos seguros que será de mucha importancia y trascendencia para los socios que se beneficiarán con el proyecto.

Debido a que las personas que conforman las cajas de ahorro son de escasos recursos y están buscando crecer económicamente el sistema realizado no tiene ningún costo, puesto que se ha desarrollado con herramientas de software libre, con lo cual será un aporte esencial para todos los socios que conforman las cajas de ahorro.

1.8. Justificación tecnológica

En tiempos actuales es imprescindible el uso de la tecnología en la vida diaria y no se diga para realizar procesos que necesitan obligatoriamente hacer uso de cambios tecnológicos para poder ser automatizados, es primordial el uso de la tecnología para realizar este proyecto debido a que la utilización de la tecnología mejora consistentemente la calidad del servicio para los usuarios, se mejora en la productividad y la eficiencia y lo que es más se reduce costos y gastos.

El proyecto conlleva a utilizar los últimos avances tecnológicos ya que de esta forma como estudiantes investigarán y aplicarán los conocimientos para beneficios de las personas que van a ser uso del sistema, y para beneficio propio del estudiante ya que con el desarrollo del proyecto obtendrán mucha más experiencia en la utilización de las herramientas que se investiga para realizar el proyecto con lo cual servirá para la carrera profesional.

El sistema realizado hace uso de herramientas de software libre para que a los socios de las cajas de ahorro no les represente ningún valor su elaboración siendo un aporte primera mente de la Universidad y el estudiante que aporta un granito de arena para que se brinde a la comunidad lo que se aprendió en las aulas y todos salgamos beneficiados.

1.9. Limites

La implementación del sistema en la parroquia de Ambuquí presentara algunas limitaciones que con el avanzar y crecimiento de las cajas de ahorro se irán solucionando.

- La falta de conocimiento del sistema y de la tecnología de los usuarios al inicio será un poco difícil de adaptarse a este cambio pero con el manejo y la utilización de los manuales de usuario será mucho más fácil comprenderlo.
- El modelo únicamente contará con información de los procesos que se realizan en las cajas de ahorro de la parroquia.
- El Sistema será implementado y comprobado en la Parroquia de Ambuquí con una de las cajas de ahorro existentes.
- La falta de infraestructura en la parroquia no permite darle mayor alcance al sistema ya que el sistema será instalado desde su inicio.

1.10. Alcances

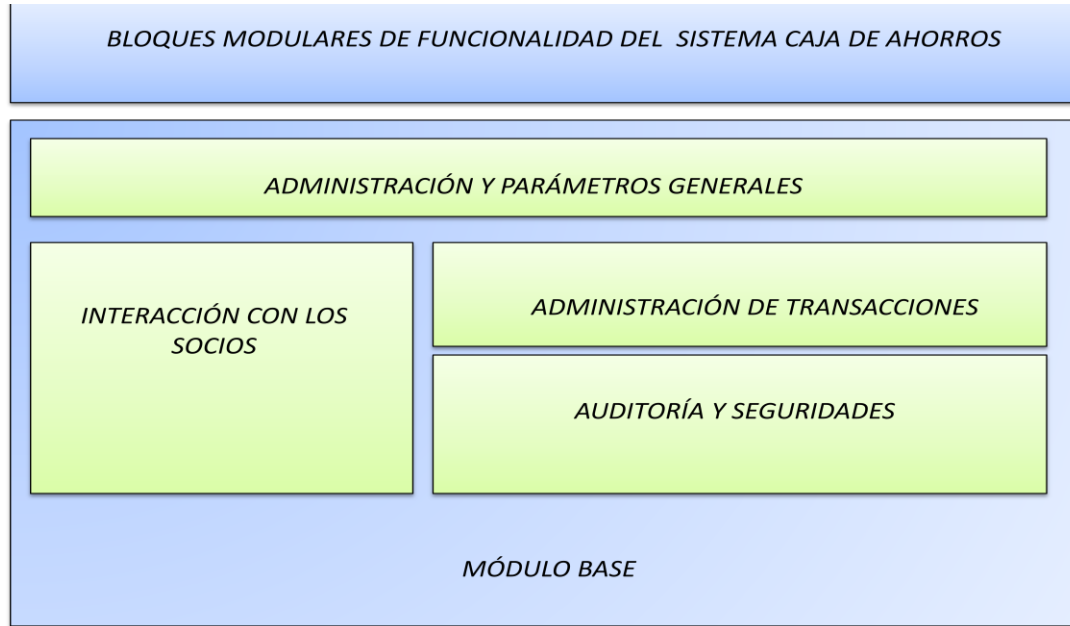
El sistema de caja de ahorro comunitario será una aplicación web desarrollada en herramientas de software libre, a la que podrán acceder todos los usuarios que se encuentren registrados y que tengan privilegios para ingresar, pudiendo organizar y llevar un adecuado control de los procesos que se realizan en las caja de ahorros con la finalidad de colaborar en la realización de emprendimientos de proyectos de microempresa que se realiza en la parroquia.

El sistema facilitará las solicitudes de créditos, las búsquedas de los socios, agilizará la apertura de cuentas de ahorro para que los socios realicen sus ahorros, como de los diferentes procesos que se realiza en las cajas de ahorros, gestionando de una manera rápida y directa entre caja de ahorros-socio usando únicamente herramientas de desarrollo de software libre.

Dicho sistema será desarrollado como proyecto creativo innovador para la culminación de estudios de la carrera de Ingeniería en Sistemas

La aplicación tiene como finalidad sistematizar los procesos que se realizan en las cajas de ahorros permitiendo ser más eficientes al momento gestionar los procesos que se realizan en las cajas de ahorros de la parroquia de Ambuquí.

El sistema de caja de ahorros consta de los siguientes bloques modulares sobre los cuales se establecerá su funcionamiento y alcance principal:



Fuente: Propia

Figura 1: Bloques modulares del sistema caja de ahorros comunitario

- **Administración y Parámetros Generales**

Módulo diseñado para administrar, modificar las tablas de parámetros del sistema.

- **Interacción con los Usuarios y Socios**

Módulo diseñado para que los socios ingresen al sistema y puedan realizar consultas del estado de cuenta y poder descargar en un archivo pdf.

- **Auditoría y Seguridades**

Módulo para tener pistas de quien modifico realizó un cambio en algún registro del sistema.

- **Administración de Transacciones**

Módulo para realizar los procesos de ahorros, créditos, fondos de reserva.

- **Módulo Base**

Módulo diseñado para asignar roles, presentar menús a los usuarios que ingresan al sistema.

1.11. Área de créditos de la caja de ahorro

El área de créditos en las cajas de ahorro es una de las funciones principales que tiene las cajas de ahorro debido a que los socios son principalmente pequeños agricultores que se dedican a la producción de varios productos agrícolas.

Todos y todas requieren de créditos inmediatos ya sea para comprar insumos agrícolas o materias primas, con lo cual se necesita que los créditos sean lo más rápido y oportunos.

Los créditos se realizan a todos los socios que integran la caja de ahorros, sin distinción de raza, sexo, religión, situación económica y social.

Los créditos que se otorgan a los socios son el cincuenta por ciento más el monto total del ahorro del socio en el ciclo, con lo que cual el socio paga un interés del cinco por ciento mensual, otorgando a un plazo de tres hasta seis meses si el monto del crédito es de seiscientos dólares, el pago de los intereses se lo realiza cada mes.

La función de una caja de ahorros es otorgar créditos a sus socios con lo cual generar utilidades para al final de cada ciclo ser repartido a todos quienes conforman las cajas de ahorros.

Una de las fortalezas de las cajas de ahorro es el otorgamiento de créditos a todos los socios sin tener que realizar trámites a diferencia de las entidades bancarias quienes a personas de escasos recursos los consideran como sin capacidad de pago por tal razón las personas acudían a realizar prestamos a usureros quienes los prestaban el dinero pero con altos intereses, la visión de las cajas de ahorro en un futuro inmediato es ser cooperativa de ahorros por lo que se están realizando las gestiones para que las cajas de ahorro sean jurídicamente legales

1.11.1. Créditos

Para conocer a lo que se refiere un crédito definamos que es un crédito con lo cual vamos a conocer y entender de mejor manera lo que se refiere el crédito en las cajas de ahorros.

El crédito es una operación financiera en la que se pone a nuestra disposición una cantidad de dinero hasta un límite especificado y durante un período de tiempo determinado.

La cantidad y el plazo otorgado a cada uno de los socios de la caja de ahorro dependen de la capacidad de pago del socio, por ejemplo si un socio tiene diez acciones el ahorra dos dólares con cincuenta cada semana con lo cual se le hace una proyección al año de ciento treinta dólares lo que significa que al socio se le otorgaría un crédito de la cantidad de su capacidad de pago.

1.11.2. Tipos de créditos

Debido a que las cajas de ahorros es limitado el número de socios que las conforman, los créditos que se otorgan son personales. La Caja de Ahorro tiene a disposición de sus socios los siguientes productos de crédito.

PRODUCTOS	MONTO		PLAZO MAX	INTERÉS	GARANTÍA
	Mínimo	Máximo			
CRÉDITO PERSONAL	300	600	3-6 meses	5%	Letra cambio

Fuente: Propia

Tabla 1: Tipos de créditos

1.11.3. Garantía

Una garantía es un negocio jurídico mediante el cual se pretende dotar de una mayor seguridad al cumplimiento de una obligación o pago de una deuda.

1.11.4. Tipos de garantías

En las cajas de ahorros de la parroquia de Ambuquí para que los créditos sean lo más rápido y oportuno para el socio acreedor deberá firmar una letra de cambio como garantías del dinero prestado con el objetivo de tener un respaldo para recuperar el dinero del crédito otorgado al socio en el caso de que el socio no cumpla con la cancelación del crédito otorgado y con su respectivo interés.

Por el momento se está optando otra política para cobrar los créditos vencidos que es no como directiva si no como una sola persona, esta persona puede ser parte de la directiva , esto se lo realiza hasta que las cajas de ahorro sean jurídicamente legales .

1.11.5. Procedimiento de colocación de créditos

A continuación detallaremos el procedimiento que se sigue para la colocación del crédito para los socios que han realizado la solicitud del crédito

1.11.6. Etapa de proceso de análisis

En este proceso es el inicio para obtener los datos iniciales para conocer la viabilidad de la propuesta de crédito para los socios.

- **Capacidad de Pago**

Es conocer si el socio está en condiciones económicas que le permitan reembolsar la obligación del crédito que adquirió con la caja de ahorros.

- **Voluntad de Pago**

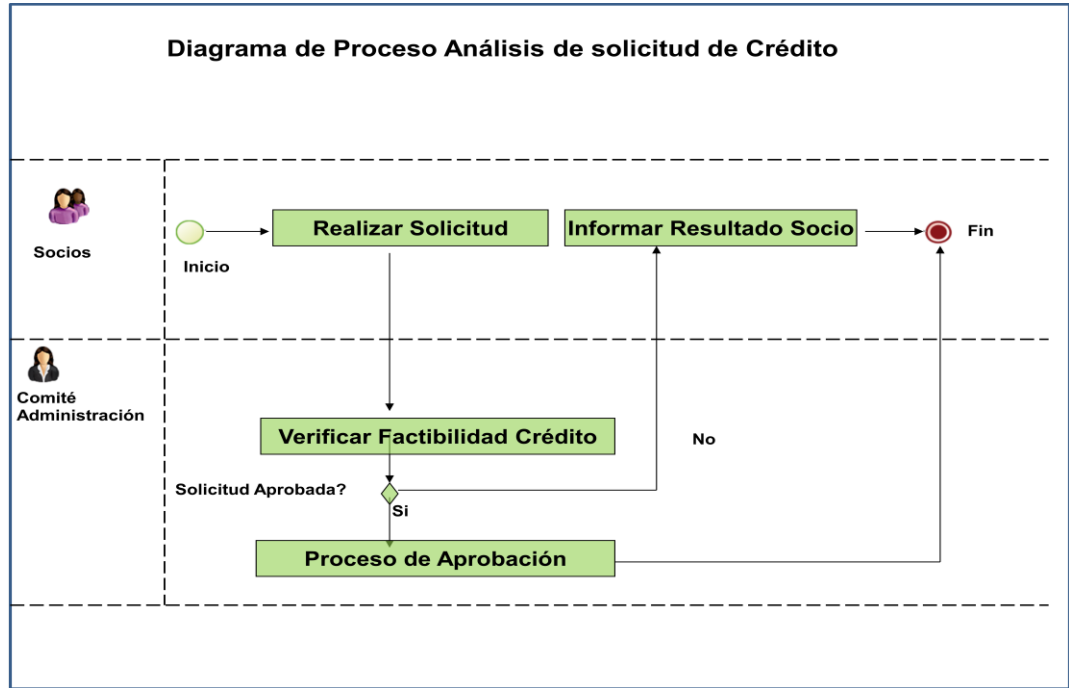
Se define como la capacidad moral del solicitante, que se mide por las condiciones de responsabilidad personal para afrontar la obligación que adquirió al obtener el crédito en la caja de ahorros.

Con los resultados obtenidos en el proceso de análisis es posible establecer:

- Si es factible conceder el crédito al socio.
- Condiciones del crédito para el socio (monto, plazo, tasa).

Es decir se puede dar una opinión que permita tomar la decisión de aprobar o rechazar la solicitud de crédito.

1.11.6.1. Diagrama de flujo del proceso de análisis de la solicitud



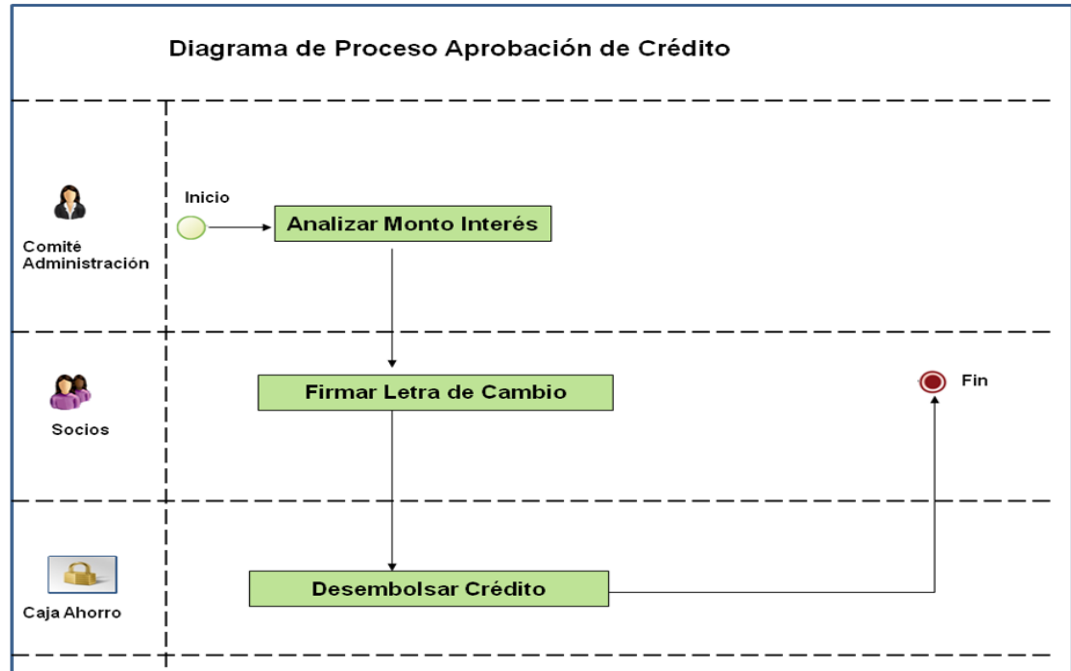
Fuente: Propia

Figura 2: Proceso de análisis de solicitud de crédito

1.11.7. Etapa de proceso de aprobación

Este proceso es la continuación que tienen que realizar la directiva para otorgar al socio un crédito, se llega a este proceso después de haber cumplido con el proceso de análisis de la solicitud

1.11.7.1. Diagrama de flujo de proceso de aprobación



Fuente: Propia

Figura 3: Proceso de aprobación de crédito

- El socio realiza la solicitud del crédito.
- Se pone a consideración la cantidad que se tiene en el momento para otorgar los créditos.
- La directiva y los socios verifican el monto a ser otorgado al socio que solicitó el crédito
- Se realiza el desembolso de la cantidad por parte de la caja de ahorros.

CAPÍTULO II

MARCO TEÓRICO



2. MARCO TEÓRICO

2.1. Definición de software

Software es un conjunto de programas, instrucciones y reglas informáticas que permiten ejecutar distintas tareas en un ordenador. Se considera que el software es el equipamiento lógico e intangible de un ordenador.

El software permite poner en relación al ser humano y al ordenador y también a los ordenadores entre sí, sin el software los ordenadores serían objetos inertes sin la posibilidad de realizar ninguna actividad.

Clasificación del software

El software puede dividirse en dos categorías básicas:

- **Software de sistema**

Es el software básico o sistema operativo. Es un conjunto de programas que controlan los trabajos del ordenador o computadora. Se encarga de administrar y asignar los recursos de hardware.

- **Software de aplicación**

Son los programas que controlan y dirige las distintas tareas que se realizan los ordenadores. Creando un ambiente amigable entre el ordenador y el usuario. Llevan a cabo tareas de tratamiento de textos, gestión de bases de datos etc.

2.1.1. Definición de software libre

Para la elaboración del proyecto se utilizan herramientas de software libre, estas herramientas se las obtiene desde sus respectivos sitios web, se utilizan estas herramientas con la finalidad de usar y ejecutar sin ninguna restricción de pagar o pedir permisos con lo cual a la parroquia no le cuesta ningún valor para la obtención de las herramientas utilizadas para la elaboración de la aplicación, por ejemplo servidor de aplicaciones, IDE de desarrollo, servidor de base de datos etc.

Otras de las razones de utilizar software libre es porque en el Ecuador el Presidente constitucional de la República Rafael Correa firmó el 10 de abril del 2008 el Decreto 1014, en el cual el software libre pasa a ser Política de Estado para ser adoptado por

todas las entidades públicas, de este modo en el Ecuador las entidades públicas están obligadas a utilizar software libre para ser independientes tecnológicamente.

El software libre no es solo un concepto tecnológico, es también un concepto social, la apuesta de un gobierno por el software libre, significa apostar por su ciudadanía, significa apostar por una soberanía tecnológica, por independencia tecnológica.

La utilización de software libre hace que podamos utilizar estas herramientas libremente obteniendo las versiones más actuales de sus respectivos sitios web para utilizarlas, estudiarlas, modificarlas con la finalidad de las necesidades que se presente en la elaboración del proyecto.

En definitiva software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software según (Lizárraga Celaya & Díaz Martínez, 2007). El software libre se refiere a cuatro libertades de los usuarios del software:

- La libertad de usar el programa, con cualquier propósito.
- La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades. El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias, con lo que puedes ayudar a tu vecino.
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. El acceso al código fuente es un requisito previo para esto.
- Un programa es de software libre si cumple con las anteriores libertades mencionadas.

2.2. Definición de entorno de desarrollo

Para la elaboración del proyecto se utiliza el entorno de desarrollo llamado Eclipse¹ el mismo que permite utilizar herramientas adicionales (plugins) que fácilmente se las puede añadir según las necesidades que se presentan en el desarrollo de las aplicaciones, este entorno de desarrollo permite añadir fácilmente controlador de versiones (SVN)², con lo cual

¹ **Eclipse:** es un entorno de desarrollo integrado de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido"

² **SVN:** Sistema de control de versiones centralizado que permite versionar archivos para poder trabajar en equipo

ayuda a tener un historial de cambios sobre los archivos que se están realizando y también permite trabajar en grupos de trabajo sobre el mismo proyecto.

Los Entornos de desarrollo se definen como programas informáticos compuesto por un conjunto de herramientas de programación.

Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI³).

- **Eclipse:** es un entorno de desarrollo integrado, de código abierto y multiplataforma. Mayoritariamente se utiliza para desarrollar lo que se conoce como "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores. Es una potente y completa plataforma de programación, desarrollo y compilación de elementos tan variados como sitios web, programas en C++ o aplicaciones Java. No es más que un entorno de desarrollo integrado (IDE⁴) en el que encontrarás todas las herramientas y funciones necesarias para la elaboración de proyectos.
- **NetBeans:** es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java - pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el IDE NetBeans. El IDE NetBeans es un producto libre y gratuito sin restricciones de uso.

2.3. Definición de lenguaje de programación

El lenguaje de programación es un lenguaje diseñado para describir el conjunto de acciones consecutivas que un equipo debe ejecutar. Por lo tanto, un lenguaje de programación es un modo práctico para que los seres humanos puedan dar instrucciones a un equipo y hacer que su funcionamiento sea de acuerdo a lo programado.

2.3.1. Lenguaje de programación java

³ **GUI:** Interfaz grafica de usuario , que utiliza un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz

⁴ **IDE:** Es un programa informático compuesto por un conjunto de herramientas de programación.

Java es un lenguaje de programación de alto nivel orientado a objetos creado por Sun Microsystems, Inc. que permite crear programas que funcionan en cualquier tipo de ordenador y sistema operativo, es un lenguaje fácil de aprender y utilizar entre sus principales características podemos citar las siguientes:

- Lenguaje simple
- Orientado a objetos
- Distribuido
- Interpretado y compilado a la vez
- Robusto
- Indiferente a la arquitectura
- Portable
- Alto rendimiento

2.3.1.1. JAVA SERVER FACES

Es una tecnología y framework que utiliza el lenguaje de programación java para su implementación.

Es un framework⁵ de desarrollo que sigue el patrón de diseño MVC⁶ (Model – View -Controller) con lo cual al crear una aplicación con este framework la aplicación se separa adecuadamente en sus capas que la conforman simplificando el desarrollo de interfaces de usuario del lado del servidor en aplicaciones Java EE.

Una de las características de JSF es ser orientado a componentes similar a como se desarrolla aplicaciones de escritorio donde la interfaz se realiza mediante componentes y basada en eventos.

Patrón de diseño MVC:

- **Vista**

⁵ **Framework:** Es la implementación de un grupo de componentes y servicios basados en patrones, estándares, y prácticas disponibles para web.

⁶ **MVC:** es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de negocio en tres componentes distintos

Se encarga de la presentación de los elementos que se visualizan en pantalla, lo integran un conjunto de ficheros JSP con las tag libraries de JSF, facelets (ficheros xhtml) ,describen la jerarquía de componentes JSF que conforman cada una de las páginas de la interfaz de usuario de la aplicación.

Vinculan los componentes JSF con los Managed Beans, hace uso de la sintaxis del Unified Expression Language⁷ para referenciar los Managed Beans y sus atributos.

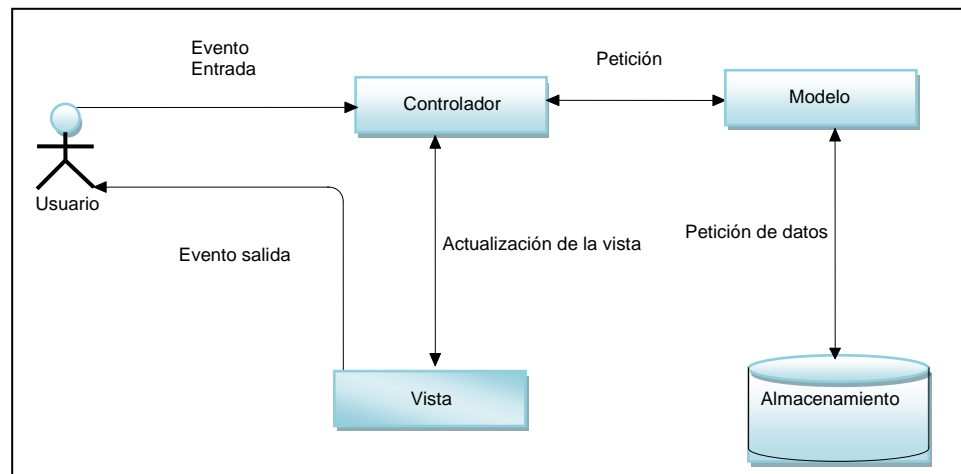
- **Modelo**

En la arquitectura MVC la lógica de negocio de la aplicación, incluyendo el acceso a los datos y su manipulación, está encapsulada dentro del modelo. El Modelo lo forman una serie de componentes de negocio independientes del Controlador y la Vista, permitiendo así su reutilización y el desacoplamiento entre las capas, el modelo puede ser implementado mediante clases estándar Java o través de Enterprise JavaBeans.

- **Controlador**

El controlador es el que dirige la aplicación. Todas las peticiones a la capa intermedia que se realicen desde el cliente son dirigidas al controlador, cuya misión es determinar las acciones a realizar para cada una de estas peticiones e invocar al resto de los componentes de la aplicación (Modelo y Vista) para que realicen las acciones requeridas en cada caso, encargándose también de la coordinación de todo el proceso.

⁷ **Unified Expression Language:** El lenguaje de expresiones permite acceder de manera muy sencilla a las propiedades de los backbeans.



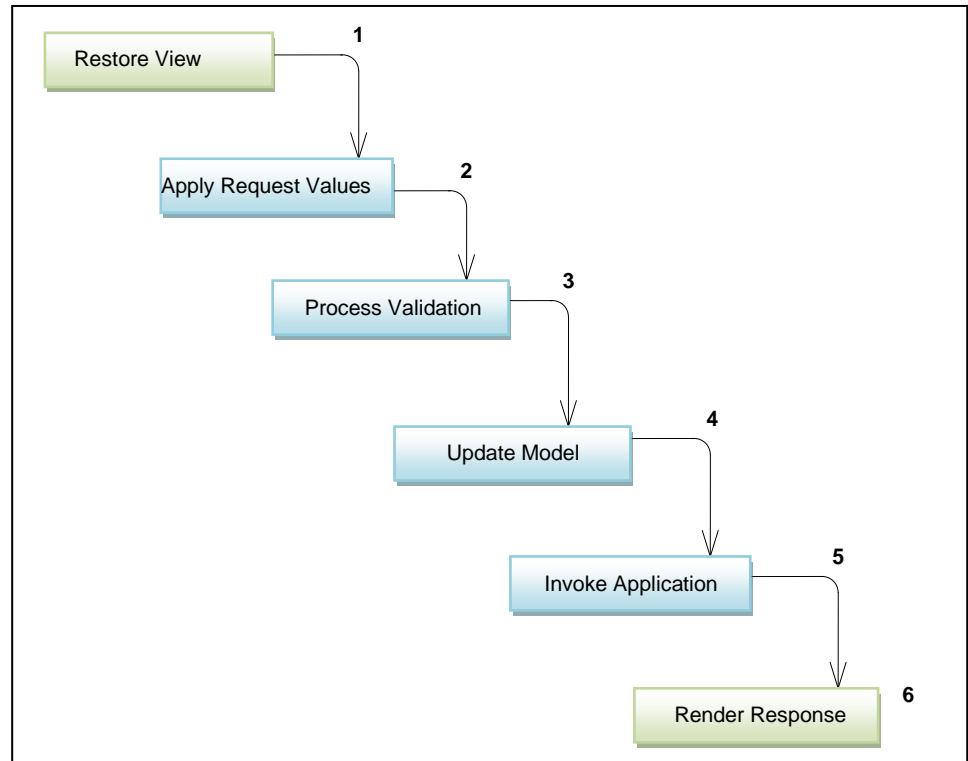
Fuente: Propia

Figura 4: Patrón de diseño MVC

- **Flujo MVC (Meroño Sánchez, 2009)**

1. El usuario interactúa con la interfaz de usuario de alguna forma (por ejemplo, el usuario pulsa un botón, enlace).
2. El controlador recibe (por parte de los objetos de la interfaz-vista) la notificación de la acción solicitada por el usuario. El controlador gestiona el evento que llega, frecuentemente a través de un gestor de eventos (handler) o callback.
3. El controlador accede al modelo, actualizándolo, posiblemente modificándolo de forma adecuada a la acción solicitada por el usuario (por ejemplo, el controlador actualiza el carro de la compra del usuario). Los controladores complejos están a menudo estructurados usando un patrón de comando que encapsula las acciones y simplifica su extensión.
4. El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz apropiada para el usuario donde se reflejan los cambios en el modelo (por ejemplo, produce un listado del contenido del carro de la compra). El modelo no debe tener conocimiento directo sobre la vista.
5. La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente.

Ciclo de vida JSF



Fuente: (Plauchu, 2011)

Figura 5: Ciclo de vida del procesamiento de una solicitud en JSF

1. Restaurar vista (Restore View)

En esta etapa el Faces Servlet localiza la vista correspondiente a la URL⁸ de la petición recibida.

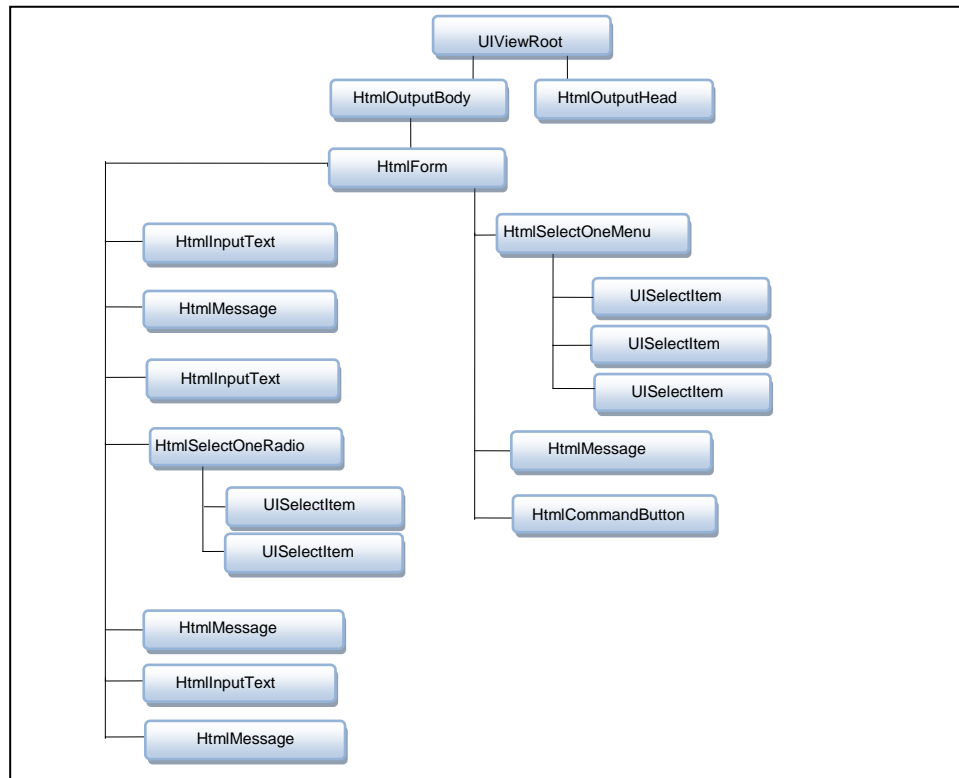
Si es la primera visita genera el árbol de componentes de esa vista (UIViewRoot) a partir de su fichero de definición (JSP, Facelets⁹).

Si la vista ya fue generada recupera el correspondiente (UIViewRoot) previo.

Si no encuentra datos enviados por método POST, se salta directamente a la etapa de producir respuesta (Render Response).

⁸ **URL:** son las siglas de Localizador de Recurso Uniforme (Uniform Resource Locator), la dirección global de documentos y de otros recursos en la World Wide Web.

⁹ **Facelets:** es un framework ligero que permite el uso de plantillas en aplicaciones JSF.



Fuente: (Plauchu, 2011)

Figura 6: Árbol de componentes del lado del servidor UI

2. Aplicar valores a petición (Apply Request Values)

Una vez que la vista ha sido restaurada, la fase siguiente (Apply Request Values), realizara el trabajo de procesar la petición HTTP. Cada componente de la interface de usuario (UI components) en la vista es capaz de tomar valores enviados por el cliente.

A partir de los valores de entrada que llegan en la petición HTTP (campos de entrada, parámetros HTTP, cookies, campos cabecera HTTP, etc) se actualizan los valores vinculados a los nodos del árbol de componentes (UIViewRoot) que representa el interfaz de usuario.

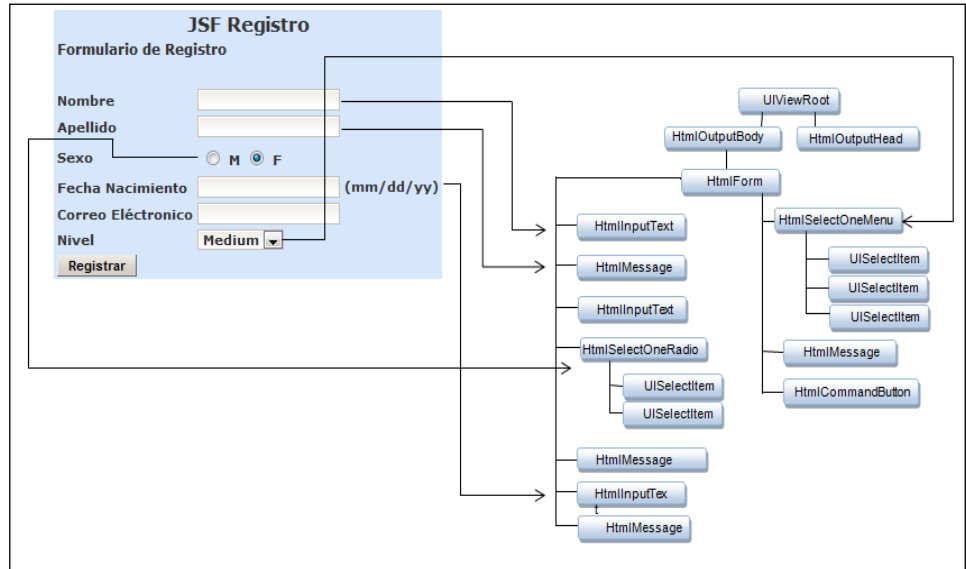
Sólo se actualiza el estado de los componentes, no los valores de los Managed Beans

Se realizan las conversiones String-Objeto necesarias (actúan los Converters)

Si hay errores de conversión se salta a la fase de Renderizado para informar del error.

- Se renderiza la misma página.

- Se incluye componentes para mensajes de error (<h:message>,<h:messages>) se hacen visibles con los errores de conversión de los componentes implicados.



Fuente: (Plauchu, 2011)

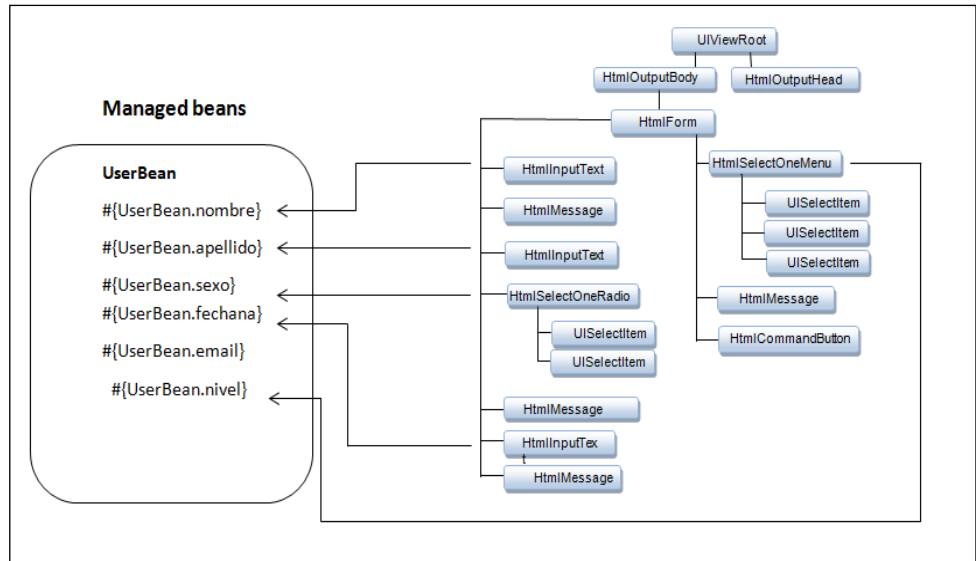
Figura 7: Aplicación de los valores de una solicitud (http request)

3. Procesar validaciones (Process Validation)

JSF recorre recursivamente el árbol de componentes (UIViewRoot) para verificar que los valores recibidos por cada componente sean aceptables, en esta fase actúan los Validators asociados a cada componente, si existen errores de conversión se salta a la fase de renderizado para informar del error.

4. Actualizar modelo (Update Model)

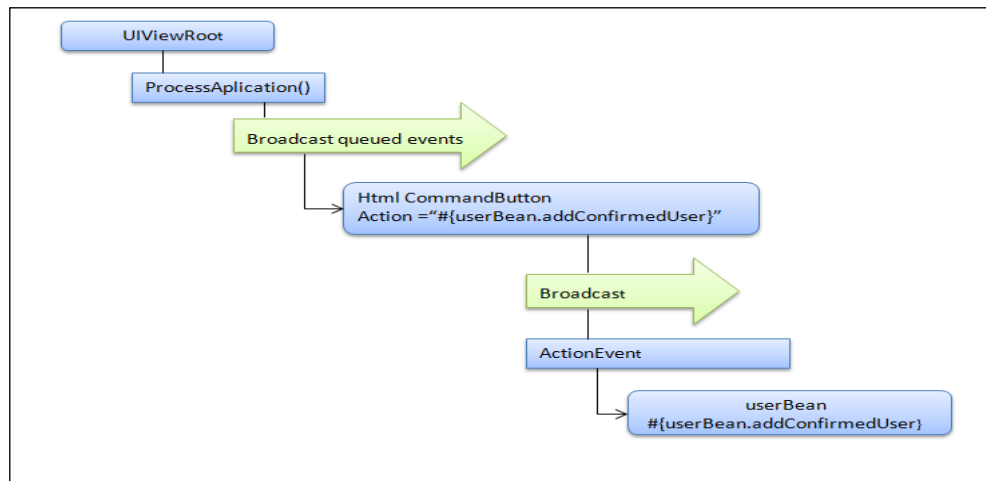
Actualiza los beans (mediante los setters del ManagedBean) del modelo asociados al componente. Esta asociación es generalmente usada con el atributo value del componente jsf.



Fuente: (Plauchu, 2011)

Figura 8: Actualización de las propiedades del model

5. Invocar la aplicación (Invoke Application)



Fuente: (Plauchu, 2011)

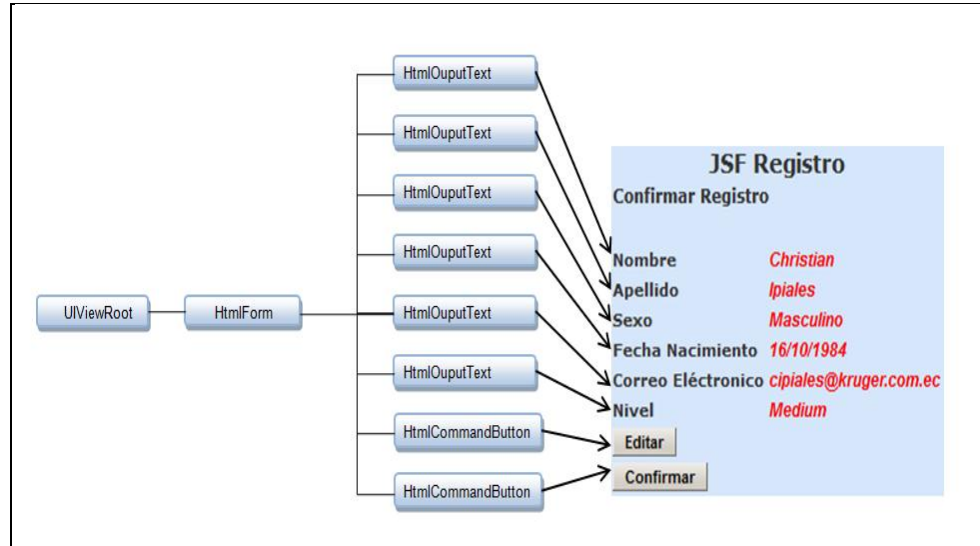
Figura 9: Invocar la aplicación

Se ejecutan los eventos encolados y pendientes (actionEvent).

En esta fase se ejecuta la lógica de negocio y es recomendable implementar aquí las validaciones del modelo de domino o procesos de negocio necesarios.

Como resultado de estas validaciones o procesos de negocio, se manipula la navegación entre las pantallas.

6. Producir respuesta (final) (Render Response)



Fuente: (Plauchu, 2011)

Figura 10: Producir respuesta JSF

Se genera la respuesta HTML, cada componente pregunta al modelo de presentación para recuperar los datos que necesita mostrar (getters del UIComponent).

Utiliza un kit de rendero (o renderkit) que puede ser modificado para presentar la respuesta en xml, por ejemplo para un browser tipo wap.

Este ciclo de fases se repite continuamente en cada petición que se hace al servidor. Si se realizan peticiones parciales como en ICEfaces, también se produce el mismo ciclo con leves modificaciones para evitar ejecutar las conversiones y validaciones en campos que aún no han sido llenados por ejemplo.

2.4. Definición de servidores de aplicaciones.

Un servidor de aplicaciones (Lopez Franco, 2001) no es más que un cambio de nombre, para algunos servidores Web de nueva generación que proporcionan la lógica de negocio sobre la cual construir y desplegar aplicaciones. Suelen asociarse con servidores de alto rendimiento pensados para proporcionar servicios a sitios Web (Web sites) con grandes necesidades: como afluencia de visitas, movimiento de datos, atención de transacciones hacia bases de datos, etc. Generalmente los fabricantes del sector tienen a disposición del público un

servidor Web básico y otro con multitud de extensiones fuertemente integradas al que llaman servidor de aplicaciones.

Un servidor de aplicaciones el corazón de un gran sistema distribuido.

2.4.1. Servidor de aplicaciones JBOSS

JBOSS SA (Alferez Sánchez, 2010) fue implementado en java puro y elaborado siguiendo el estándar J2EE con lo cual puede ser utilizado en cualquier sistema operativo que la soporte, este servidor de aplicaciones es el primero de código abierto, ofrece una plataforma de alto rendimiento para aplicaciones de e-business¹⁰.

Entre sus principales características podemos citar las siguientes:

- Cumple los estándares
- Producto de Licencia de código abierto sin coste adicional.
- Confiable a nivel de empresa
- Soporte completo para JMX¹¹
- Orientado a arquitectura de Servicios.

2.5. Definición de la arquitectura de la aplicación.

La arquitectura de la aplicación se refiere a como está diseñada tanto físicamente y lógicamente la aplicación.

Para el desarrollo de la aplicación se utilizó una arquitectura de tres capas basado en la arquitectura de JEE (Dos ideas, 2011) formado por tres capas principales:

- Capa de presentación
- Capa de lógica de negocio
- Capa de acceso a datos

¹⁰ **E-business:** es la integración del negocio de una empresa incluyendo productos, procesos y servicios por medio del Internet.

¹¹ **JMX:** es un estándar de Sun para la gestión y monitorización remota de recursos java como son Aplicaciones y dispositivos, Servidores y servicios, Java Virtual Machine.

2.5.1. Capa de presentación

Esta capa resuelve la presentación de datos al usuario. Esta capa se encarga entonces de "dibujar" las pantallas de la aplicación al usuario, y tomar los eventos que el cliente genere (por ejemplo, el hacer click en un botón).

Existen numerosas tecnologías para esta capa (JSP, Struts, JSF, Flex, Velocity, Wicket, GWT, etc), cada una con sus ventajas y desventajas. Para realizar la aplicación se utiliza JSF.

2.5.2. Capa de lógica de negocio

Esta capa resuelve la lógica de la aplicación, contiene los algoritmos, validaciones y coordinación necesaria para resolver los procesos que se realizan en la aplicación.

Los elementos fundamentales de esta capa son los objetos de dominio. Estos objetos representan los objetos principales del negocio. Son simples objetos que sólo contienen los datos que representan (por ejemplo, una Cuenta, un Usuario, un Socio, un Ahorro).

La lógica para manipularlos se encuentra en los llamados objetos de negocio (Service Object, o también a veces Business Object). Estos objetos contienen la lógica del negocio, y manipulan a los objetos de dominio.

Los Service Object constan de una interfaz (dónde se expone la lógica del negocio) y una clase que la implementa. Para esta capa se utiliza EJBs que gestionan la lógica de toda la aplicación.

2.5.2.1. EJB

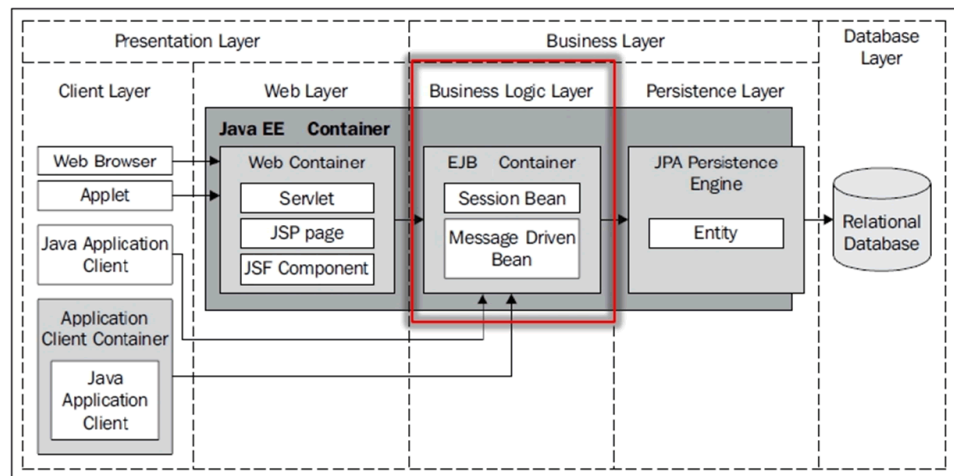
Los EJB (Enterprise JavaBean) son componentes de software que encapsulan la lógica de negocio de una aplicación y se ejecutan del lado del servidor, en cada uno de ellos se encapsula parte de la lógica de negocio de la aplicación.

Los EJB son Beans gestionados por un contenedor del servidor de aplicaciones para gestionar el acceso a recursos (bases de datos, colas de mensajes, ficheros, etc) y proporcionar servicios (seguridad, transacciones, mensajería, nombres) de forma sistemática y optimizada, permitiendo construir aplicaciones de negocio portables, reutilizables y escalables usando el lenguaje de programación Java.

Entre las principales ventajas que aportan los EJB podemos citar las siguientes:

- Permiten la distribución de la lógica de negocio entre distintas maquinas manteniendo la transparencia.
- Manejo de transacciones para controlar accesos concurrentes asegurando la integridad de los datos.
- Simplifican el desarrollo de aplicaciones que atiendan una gran variedad de clientes (clientes web o "de escritorio" (stand-alone), cliente locales o remotos, etc.)
- Diseño orientado a componentes (reusabilidad y abstracción), el desarrollador se concentra en la lógica de negocio y utiliza los servicios proporcionados por el contenedor

Los EJBs según (Meroño Sánchez, 2009) son componentes, desde el punto de vista que encapsulan comportamiento y permite reutilizar porciones de código, pero también pueden ser vistos como un framework, ya que, desplegados en un contenedor, proveen servicios para el desarrollo de aplicaciones empresariales, servicios que son invisibles para el programador y no ensucian la lógica de negocio con funcionalidades transversales al modelo de dominio (a menudo requisitos no funcionales o aspectos). En la especificación 3.0, los EJB no son más que POJOs (clases planas comunes y corrientes de Java) con algunos poderes especiales implícitos, que se activan en tiempo de ejecución (runtime) cuando son ejecutados en un contenedor de EJBs.



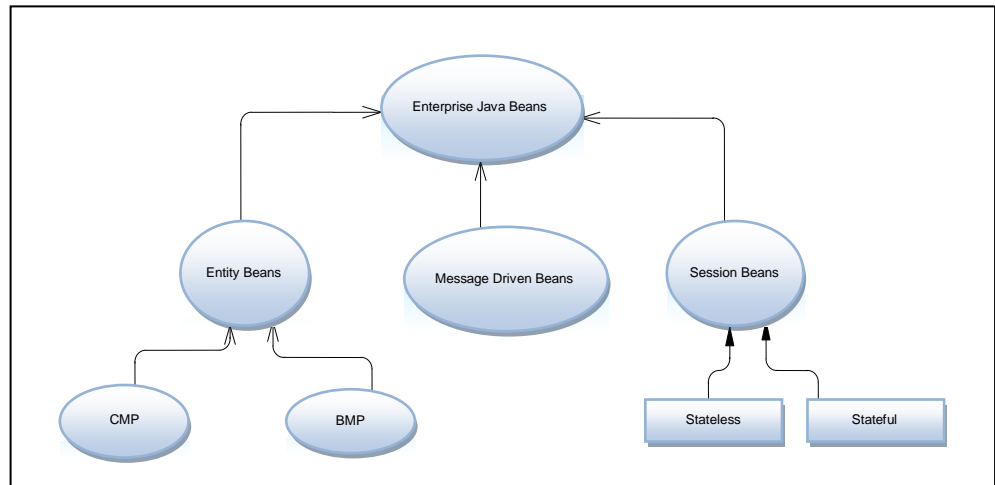
Fuente: (Meroño Sánchez, 2009)

Figura 11: Funcionamiento de Componente EJB

El funcionamiento de los componentes EJB se basa fundamentalmente en el trabajo del contenedor EJB. El contenedor EJB es un programa Java que corre en el servidor y que contiene todas las clases y objetos necesarios para el correcto funcionamiento de los EJBs.

Tipos de EJBs

Existen tres tipos de EJBs:



Fuente: (Meroño Sánchez, 2009)

Figura 12: Tipos de EJBs

1. EJBs dirigidos por mensajes (Message-driven EJBs)

También se localizan la lógica de negocio y los servicios que proveen son parecidos a los Beans de Sesión, con la diferencia de que los MDBs son usados para invocar métodos de forma asincrónica. Cuando se produce la invocación de un método de un MDB desde un cliente, la llamada no bloquea el código del cliente y el mismo puede seguir con su ejecución, sin tener que esperar indefinidamente por la respuesta del servidor. Los MDBs encapsulan el popular servicio de mensajería de Java, JMS.

2. Session EJBs (EJBs de sesión)

Las EJBs de sesión ejecutan métodos de forma síncrona, pudiendo devolver valores.

En una aplicación típica, dividida en cuatro grandes capas (presentación, lógica de negocio, persistencia y base de datos), los Beans se localizan en la

lógica de negocio. Hay dos grandes tipos de Beans de Sesión: Stateless (sin estado) y Stateful (con estado). El primero no conserva el estado de ninguno de sus atributos de la invocación de un método a otro y el segundo conserva el estado a lo largo de toda una sesión. Los Beans de Sesión Stateless son los únicos que pueden exponerse como servicios web.

Los EJBs de sesión pueden ser con estado, sin estado o únicas (singleton):

- Las EJBs **Stateful** con estado las utiliza un único cliente para ejecutar sus métodos y mantienen sus atributos entre llamadas
- Las EJBs **Stateless** sin estado pueden ser utilizadas consecutivamente por clientes arbitrarios cada vez que ejecutan un método. Pueden mantener sus atributos entre llamadas
- Las EJBs **Singleton** únicas tienen una sola instancia para toda la aplicación

3. Entity Beans

Representan entidades de negocio con mapeo objeto/relacional, son empleados en la capa de persistencia para representar datos del modelo.

2.5.3. Capa de acceso a datos

Esta capa resuelve el acceso a datos, abstrayendo a su capa superior de la complejidad del acceso e interacción con los diferentes orígenes de datos, se encarga de proveer un API¹² simple de usar, orientado al negocio, sin exponer complejidades propias de un repositorio de datos.

En esta capa se resuelven:

- cualquier acceso a la base de datos
- cualquier acceso a filesystem
- cualquier acceso a otros sistemas
- cualquier acceso a un repositorio de datos en cualquier forma.

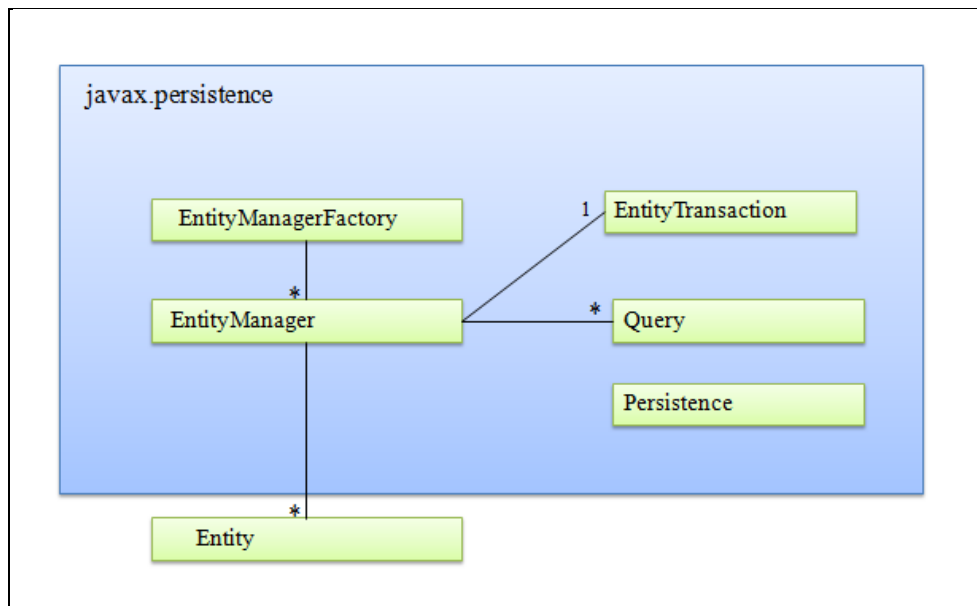
¹² **API**: es una Interfaz de Programación de Aplicaciones provista por los creadores del lenguaje Java, y que da a los programadores los medios para desarrollar aplicaciones Java

Estos objetos se construyen a partir del patrón Data Access Object (DAO). Los DAO constan de una interfaz (dónde se expone la lógica de acceso a datos) y una clase que la implementa.

2.5.3.1. JPA

JPA es el estándar para gestionar persistencia de Java, realiza por nosotros toda la conversión entre nuestros objetos y las tablas de una base de datos. Esta conversión se llama ORM (Object Relational Mapping - Mapeo Relacional de Objetos), y puede configurarse a través de metadatos (mediante archivos xml o anotaciones). JPA también permite seguir el sentido inverso, creando objetos a partir de las tablas de una base de datos, A estos objetos los llamaremos desde ahora *entidades* (entities).

Arquitectura JPA



Fuente: (Meroño Sánchez, 2009)

Figura 13: Arquitectura JPA

El componente principal de la arquitectura es el “EntityManager”.

La clase Persistence dispone de métodos estáticos que permiten obtener una instancia de EntityManagerFactory de forma independiente al proveedor de JPA y a partir del que se puede obtener el EntityManager necesario.

Mediante EntityManager podemos obtener objetos “Query” para cargar objetos persistidos con determinados criterios.

JPA emplea el lenguaje de consulta JPQL (Java Persistence Query Language) ó SQL (Structured Query Language). También podemos realizar operaciones CRUD (Create, Read, Update, Delete) sobre los objetos persistentes

“Entity”. Cada instancia de EntityManager tiene una relación uno a uno con una EntityTransaction, permitiendo el manejo de operaciones sobre objetos persistentes con la idea global de transacción, de forma que se ejecuten todas las operaciones o ninguna.

Por otro lado la unidad de persistencia es lo que aporta la información necesaria para enlazar con la base de datos relacional adecuada.

CAPÍTULO III

DESARROLLO DE LA APLICACIÓN



3. DESARROLLO DE LA APLICACIÓN

3.1. Análisis de requerimientos

Para la obtención de los requerimientos necesarios para realizar la aplicación se tuvo que asistir a los talleres que se realizan todos los días sábados conjuntamente con los socios de una de las cajas de ahorros, pudiendo dinamizar la obtención de requerimientos realizando ejemplos prácticos de los procesos que se realizan en las cajas de ahorros, logrando obtener los siguientes requerimientos que se describirán de forma general en las siguientes líneas.

Las cajas de ahorros las componen los socios los mismos que al iniciar el ciclo de un año compran acciones las mismas que tienen un determinado valor, siendo este valor nominal para obtener un capital inicial, esto se realiza solo la primera semana de inicio del ciclo, las siguientes semanas del ciclo se debe ahorrar un valor por cada acción que se compro la primera semana del ciclo.

Los socios pueden ser parte de la directiva de la caja de ahorros con lo cual cada año se reeligen directivos para la caja de ahorros.

En la caja de ahorros se realizan tres procesos principales el proceso de los ahorros, el proceso de los créditos y el proceso de los fondos de reserva.

En el proceso de ahorros se realiza los ahorros cada semana de acuerdo a las acciones que compró al inicio del ciclo y un valor de ahorro semanal, se debe realizar los ahorros de manera puntual, si el socio se atrasa con los ahorros deberá pagar su respectivo interés.

En el proceso de fondos de reserva cada semana el socio debe ahorrar una cuota que se fija al inicio del ciclo estos ahorros son destinados para integración y se lo utiliza al final de cada ciclo.

En el proceso de créditos se otorga crédito a todos los socios, se realiza el crédito según la capacidad de ahorro.

En una caja de ahorros existe como máximo treinta y cinco socios.

Un socio puede pertenecer a varias cajas de ahorros.

Si se desea ingresar un nuevo socio el socio deberá ahorrar en el fondo de reserva una determinada cantidad, mas el ahorro desde el inicio de la organización hasta la fecha de ingreso.

El cierre de la caja de ahorro se realiza cada año que corresponde a un ciclo con lo cual a partir de este punto la información no puede ser modificada, borrada y creada originando un nuevo periodo o ciclo para la nueva información.

3.2. Análisis de las herramientas de desarrollo de software

Para la elaboración de la aplicación se toma en cuenta las necesidades que tienen las cajas de ahorro en su situación actual y la proyección en un futuro inmediato, para la situación actual existen en la parroquia quince cajas de ahorros lo que va existir muchos usuarios que tendrán que usar el sistema y la proyección de las cajas de ahorro en un futuro inmediato es crecer y convertirse en cooperativa de ahorros con lo cual nos hace prever que debe ser un sistema que va ha ser escalable y también se toma en cuenta que las herramientas para la elaboración de la aplicación no debe tener ningún costo para la parroquia.

Entonces JAVA provee una plataforma empresarial denominada JEE la versión que se utiliza para la elaboración del proyecto es la versión 6.

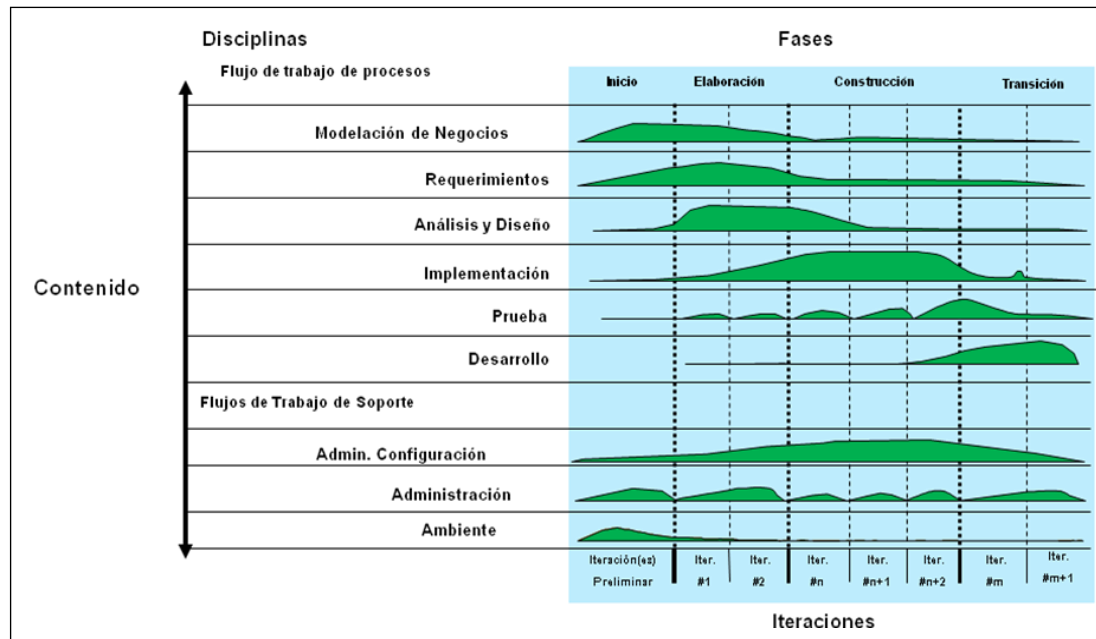
3.3. Metodología RUP

La metodología que se utiliza para la elaboración del sistema es RUP (Rational Unified Process) esta metodología permite la elaboración del sistema de forma ordenada y resolverá las necesidades de los usuarios permitiendo obtener una aplicación de alta calidad.

Esta metodología permite realizar la aplicación siguiendo las fases que nos indica RUP para la elaboración de desarrollo de software de calidad.

RUP es un marco de referencia de ingeniería del software para definir implementar y distribuir aplicaciones del software, sus características son:

- Utiliza UML
- Gramática bien definida
- Terminología utilizada para la definición de los procesos



Fuente: (Mejía Álvarez, 2002)

Figura 14: Diagrama general de RUP

La metodología RUP define disciplinas y fases, existen nueve disciplinas asociadas a la entrega del software las disciplinas son:

1. Modelado del Negocio

Consiste en entender el negocio

2. Requerimientos

Define que debe realizar el sistema

3. Análisis y Diseño

Como se realizaran los requerimientos

4. Implementación

Consiste en construir el sistema

5. Pruebas

Lleva a cabo verificación, validación y retroalimentación

6. Distribución

Consiste en la entrega del producto

7. Administración del proyecto

Define actividades para planear, presupuestar, asignar recursos y dar seguimiento al proyecto

8. Administración de cambios y configuración

Consiste en realizar nuevas versiones y actualizaciones.

9. Ambiente

En esta disciplina se realiza el mantenimiento del software.

Las fases que componen el ciclo de vida de RUP son cuatro las mismas que cada cual concluye con un producto intermedio, al término de cada fase se realiza una evaluación para determinar si se ha cumplido o no con los objetivos de las misma.

Las fases que comprenden el ciclo de vida de RUP son:

- **Fase Incepción**

En esta fase se certifican los requerimientos se definen los caso de negocio y se empieza a describir el alcance, actores primarios que interactuaran con el software, factores críticos con la entrega del proyecto.

- **Fase de Elaboración**

En esta fase se analiza el dominio de diseño de la aplicación y establecemos las bases de la arquitectura del software, es la más importante en la entrega de un proyecto, es la de más alto riesgo por lo crítico en la toma de decisiones de diseño.

- **Fase de Construcción**

Se compone de desarrollo e integración de cada componente de la aplicación

- **Fase de Transición**

Se centra en las actividades requeridas para colocar el software en las manos de los usuarios finales, incluye varias iteraciones de entregas beta, corrección de errores y mejoras

- **Fase Evolución**

Comprende las actividades asociadas con la implementación y la elaboración de nuevas versiones de un producto más maduro, cada ciclo de evolución comprende las fases de Incepción, elaboración, construcción, transición.

Al utilizar este marco de referencia se tiene las siguientes ventajas:

RUP ha venido madurando con el tiempo, el uso de UML hace que se apegue a estándares de la industria, las fases se adaptan a la organización, existen herramientas que ayudan a implementar RUP en todas las áreas de la empresa, define actividades roles y responsabilidades desde jefe de proyectos hasta analistas y desde los desarrolladores hasta equipo de pruebas.

3.4. Fase de inyección.

En esta fase se desarrolla los requisitos del proyecto desde la perspectiva del usuario, los cuales fueron establecidos en el alcance del proyecto. Los principales casos de uso serán identificados. Los casos de uso describen los límites y interacción entre el sistema y los usuarios mediante una representación gráfica del entorno del sistema y su funcionalidad principal (casos de uso), la aceptación del usuario del plan de desarrollo de software marca el final de esta fase.

El Plan de Desarrollo de Software es recoger toda la información necesaria para el control del proyecto. Describe el enfoque al desarrollo de software y el plan de alto nivel generado y usado por desarrolladores para gestionar el proyecto.

Además define las actividades de desarrollo enfocadas en FASES e ITERACIONES requeridas para la implementación del Sistema de Caja de Ahorro Comunitario.

3.4.1. Plan de desarrollo de software

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una la distribución de tiempos y el número de iteraciones de cada fase.

Fase	Iteraciones	Comienzo	Finalización	días/iteración
Inyección	1	2011-11-01	2011-11-22	20
Elaboración	1	2011-11-23	2011-12-20	29
Construcción	1	2011-12-21	2012-04-23	135
Transición	1	2012-04-24	2012-05-04	20

Fuente: Propia

Tabla 2: Plan de desarrollo de software

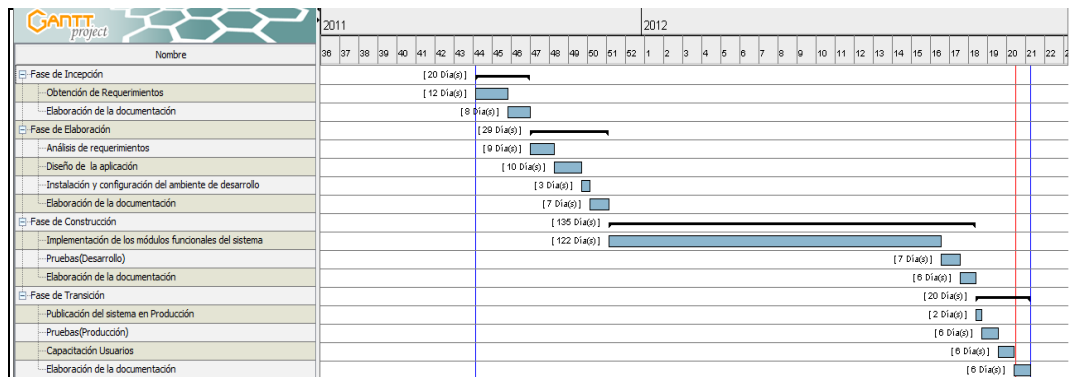
En el siguiente Diagrama de Gantt se aprecia gráficamente la distribución temporal de las distintas fases e iteraciones.

Tipo	Prioridad	Nombre	Duración	Fecha de inicio	Fecha de fin
☐	●	Fase de Incepción	20	1/11/11	21/11/11
☐	●	Obtención de Requerimientos	12	1/11/11	13/11/11
☐	●	Elaboración de la documentación	8	13/11/11	21/11/11
☐	●	Fase de Elaboración	29	21/11/11	20/12/11
☐	●	Análisis de requerimientos	9	21/11/11	30/11/11
☐	●	Diseño de la aplicación	10	30/11/11	10/12/11
☐	●	Instalación y configuración del ambiente de desarrollo	3	10/12/11	13/12/11
☐	●	Elaboración de la documentación	7	13/12/11	20/12/11
☐	●	Fase de Construcción	135	20/12/11	3/05/12
☐	●	Implementación de los módulos funcionales del sistema	122	20/12/11	20/04/12
☐	●	Pruebas(Desarrollo)	7	20/04/12	27/04/12
☐	●	Elaboración de la documentación	6	27/04/12	3/05/12
☐	●	Fase de Transición	20	3/05/12	23/05/12
☐	●	Publicación del sistema en Producción	2	3/05/12	5/05/12
☐	●	Pruebas(Producción)	6	5/05/12	11/05/12
☐	●	Capacitación Usuarios	6	11/05/12	17/05/12
☐	●	Elaboración de la documentación	6	17/05/12	23/05/12

Fuente: Propia

Figura 15: Diagrama Gantt

A partir del diagrama podemos obtener el cronograma del proyecto



Fuente: Propia

Figura 16: Cronograma del Proyecto

3.4.2.Elementos del modelo de casos de uso

- Caso de uso

Un Caso de uso es una representación de una unidad discreta de trabajo realizada por un usuario (u otro sistema) usando el sistema en operación. Se ejecuta en su totalidad o no se ejecuta nada, devolviendo algo de valor al usuario

- **Actor**

Un actor es un usuario del sistema. Esto incluye usuarios humanos y otros sistemas computacionales. Un actor usa un Caso de Uso para ejecutar una porción de trabajo de valor para el negocio. El conjunto de casos de uso al que un actor tiene acceso define rol en el sistema y el alcance de su acción

- **Asociaciones**

Las asociaciones son usadas para describir las relaciones entre actores y los casos de uso en los que ellos participan. Estas relaciones son conocidas comúnmente como una "asociación comunicante".

- **Relación de extensión**

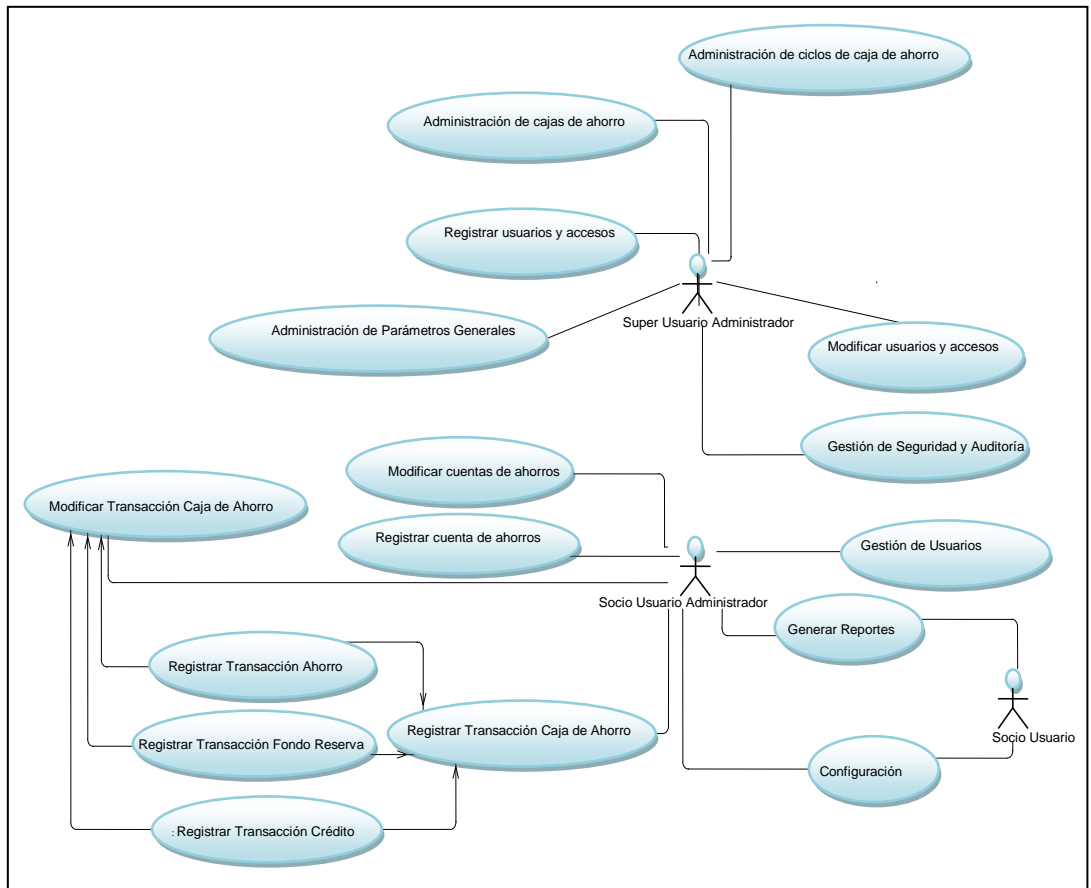
Un caso de uso extiende a otro cuando sin alterar a este, se incorpora su funcionalidad como parte integral del primero. Se denota con una relación que apunta del caso extendido al caso base y la conexión se hace o bien al principio del flujo de eventos principal del caso base o en alguno de los puntos de extensión que este haya definido

- **Relación de inclusión**

Esta relación es usada para incluir comportamiento común desde un caso de uso incluido dentro de un caso de uso base con el propósito de soportar la reutilización de comportamientos comunes.

3.4.3. Diagrama del modelo de caso de uso

El siguiente diagrama de casos de uso muestra los distintos requisitos funcionales del sistema identificados durante la fase de Inicio, y cómo se relacionan con su entorno (usuarios u otras aplicaciones)



Fuente: Propia

Figura 17: Modelo General de Casos de Uso y Actores del Negocio

3.4.4. Actores

- **Socio Usuario**

Es la persona que invierte activamente dinero en la caja de ahorros es decir ingresa ahorros en la caja de ahorros semanalmente.

- **Super Usuario Administrador**

Es el usuario que administra todas las cajas de ahorro.

- **Socio Usuario Administrador**

Es la persona que además de ser socio es el que dirige a toda la caja de ahorro.

- **Comité de administración**

Es el grupo de personas que dirige la caja de ahorros, son responsables de la custodia y seguridad de la caja de ahorros.

- **Auditor**

Es la persona que registra los ahorros y los créditos en las libretas.

- **Secretario**

Es la persona que registra los ahorros y controla la asistencia.

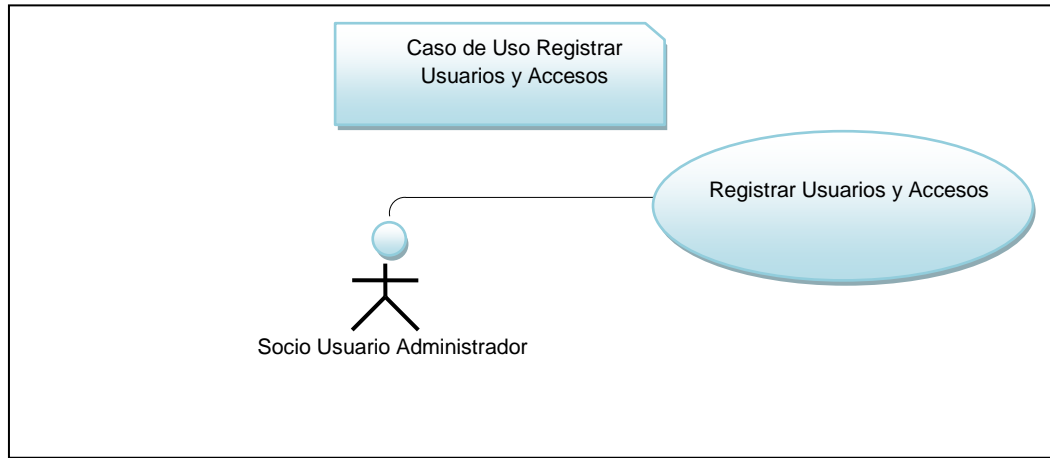
3.5. Casos de uso

Los siguientes casos de uso fueron identificados para el desarrollo del sistema de caja de ahorro comunitario.

- Caso de uso: Registrar usuarios y accesos
- Caso de uso: Modificar usuarios y accesos
- Caso de uso: Registrar cuentas de ahorros
- Caso de uso: Modificar cuentas de ahorros
- Caso de uso: Administración de ciclos de caja de ahorro
- Caso de uso: Administración de cajas de ahorro
- Caso de uso: Registrar Transacción Caja de Ahorro
- Caso de uso: Modificar Transacción Caja de Ahorro
- Caso de uso: Registrar Transacción Ahorro
- Caso de uso: Registrar Transacción Fondo Reserva
- Caso de uso: Registrar Transacción Crédito
- Caso de uso: Generar registros de auditoría
- Caso de uso: Descargar información

En la siguiente sección se describirá detalladamente cada uno de los casos de uso que fueron identificados en el sistema de caja de ahorro comunitario.

3.5.1. Especificación de caso de uso: Registrar usuarios y accesos



Fuente: Propia

Figura 18: Caso de uso Registrar usuarios y accesos

1. Nombre de Caso de Uso

Registrar usuarios y accesos

1.1. Descripción breve

El caso de uso describe el proceso de registro de usuarios que podrán ingresar al sistema para realizar cualquier proceso según el rol que tengan asignado en el sistema.

Este proceso lo realiza un socio usuario administrador, mediante el uso de una interfaz gráfica donde se registran los datos del usuario.

1.2. Objetivo

Registrar los usuarios que podrán ingresar al sistema.

1.3. Precondiciones

El socio usuario administrador debe estar ingresado y autenticado en el sistema.

1.4. Poscondiciones de éxito

El registro del usuario fue realizado satisfactoriamente.

1.5. Poscondiciones de falla

El usuario no fue registrado correctamente, se indicará el motivo por el cual no se pudo registrar al usuario en el sistema.

1.6. Actor principal

<i>Actor</i>	<i>Evento</i>
<i>Socio Usuario Administrador</i>	<i>Es el funcionario encargado del proceso de registro de nuevos usuarios.</i>

Fuente: Propia

Tabla 3: Actores caso uso registrar usuarios

2. Flujo de Eventos

2.1. Flujo básico

1. El usuario selecciona la Opción de menú Administración de usuarios.
2. El usuario selecciona la opción “Nuevo”
3. El sistema despliega el formulario para el ingreso de los datos requeridos para el registro de un nuevo usuario.
4. El usuario ingresa los datos solicitados en el formulario.
5. El usuario podrá seleccionar los roles que se desea asignar al nuevo usuario.
6. El usuario presiona el botón “Guardar”.
7. El sistema valida que los datos se encuentren correctamente ingresados.
8. El sistema registra la información en las correspondientes a los datos ingresados del usuario.
9. Termina el CU.

2.2. Flujos alternativos

En el paso 4 del flujo básico

El usuario ingresa en el formulario la siguiente información necesaria para el registro del usuario:

- Nombre usuario
- Password

- Correo electrónico

En el paso 5 del flujo básico

El usuario selecciona los roles que desea asignarles a el usuario que esta creado, podrá asignarle o quitarle si lo desea.

2.3. Flujos de excepción

Información mal ingresada

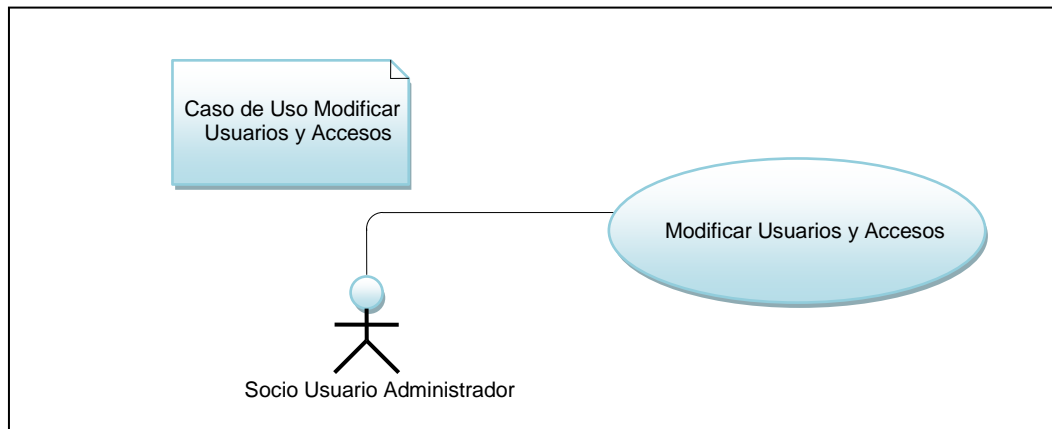
Si en el paso 4 del flujo principal, el sistema detecta errores de ingreso de la información, el sistema desplegará el respectivo mensaje en base al tipo de error detectado.

Si en el paso 4 del flujo principal el sistema detecta que ya existe un nombre de usuario el sistema desplegará el respectivo mensaje indicando que ya existe un nombre de usuario en el sistema.

3. Requerimientos especiales

El correo electrónico deberá ser ingresado en el formato correspondiente caso contrario se desplegará el respectivo mensaje de error indicando que el formato del correo electrónico no es el correcto.

3.5.2. Especificación de caso de uso: Modificar usuarios y accesos



Fuente: Propia

Figura 19 Caso de Uso Modificar Usuarios y Accesos

1. Nombre de Caso de Uso

Modificar usuarios y accesos

1.1. Descripción breve

Este caso de uso describe el proceso de la modificación de usuarios que podrán ingresar al sistema para realizar cualquier proceso según el rol que tengan asignado en el sistema.

Este proceso lo realiza un socio usuario administrador, mediante el uso de una interfaz gráfica donde se modifican los datos del usuario.

1.2. Objetivo

Modificar los datos de los usuarios que ingresan al sistema.

1.3. Precondiciones

El socio usuario administrador debe estar ingresado y autenticado en el Sistema.

1.4. Poscondiciones de éxito

La modificación del usuario fue realizada satisfactoriamente.

1.5. Poscondiciones de falla

El usuario no fue modificado correctamente, se indicará el motivo por el cual no se pudo modificar al usuario en el sistema.

1.6. Actor principal

<i>Actor</i>	<i>Evento</i>
<i>Socio Usuario Administrador</i>	<i>Es el funcionario encargado del proceso de modificación de los usuarios en el sistema</i>

Fuente: Propia

Tabla 4: Actores caso uso modificar usuarios

2. Flujo de Eventos

2.1. Flujo básico

1. El usuario selecciona la Opción del menú Administración de usuarios.
2. El usuario selecciona al usuario que desea modificarlo.

3. El sistema despliega el formulario con los datos del usuario seleccionado.
4. El usuario modifica los datos que desea cambiar.
5. El usuario podrá añadir roles al usuario que lo está modificando.
6. El usuario presiona el botón “Guardar”.
7. El sistema válida que los datos se encuentren correctamente ingresados.
8. El sistema modificará la información correspondiente a los datos modificados del usuario.
9. Termina el CU

2.2. *Flujos alternativos*

En el paso 4 del flujo básico

El usuario podrá modificar en el formulario la siguiente información:

- Nombre usuario
- Password
- Correo electrónico

En el paso 5 del flujo básico

El usuario selecciona los roles que desea asignarles a el usuario que está modificando, podrá asignarle o quitarle si lo desea.

2.3. *Flujos de excepción*

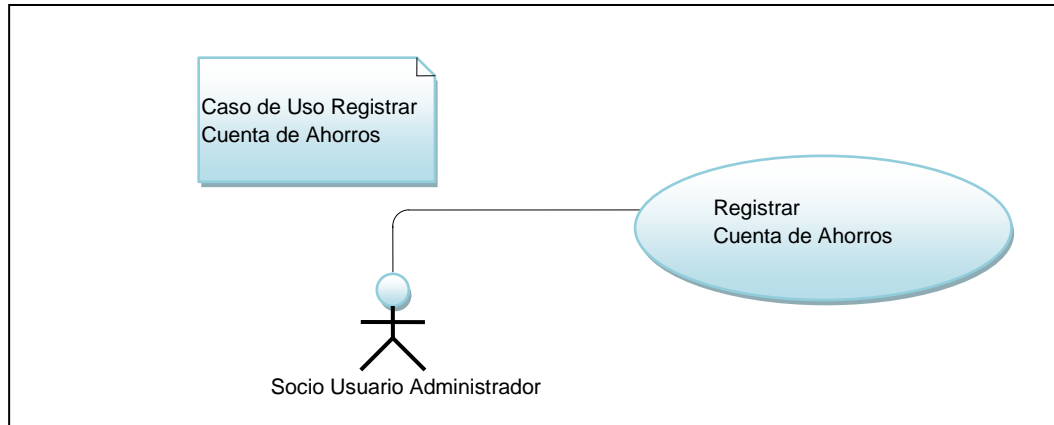
Información mal ingresada

Si en el paso 4 del flujo principal, el sistema detecta errores de ingreso de la información, el sistema desplegará el respectivo mensaje en base al tipo de error detectado.

3. *Requerimientos especiales*

El correo electrónico deberá ser ingresado en el formato correcto caso contrario se desplegara un error indicando que el formato del correo electrónico no es el correcto.

3.5.3. Especificación de caso de uso: Registrar cuenta de ahorros



Fuente: Propia

Figura 20: Caso de uso Registrar Cuenta de Ahorros

1. Nombre de Caso de Uso

Registrar Cuenta de Ahorros

1.1. Descripción breve

Este caso de uso describe el proceso del registro de las cuentas de ahorros en las cajas de ahorro, cada cuenta pertenecerá a un socio.

Este proceso lo realiza un socio usuario administrador de la caja de ahorros, mediante el uso de una interfaz gráfica donde se registran los datos de la cuenta, del socio, y del usuario para el socio.

1.2. Objetivo

Registrar las cuentas de ahorros de un nuevo socio.

1.3. Precondiciones

El socio usuario administrador de la caja de ahorros debe estar ingresado y autenticado en el Sistema.

1.4. Poscondiciones de éxito

El registro de la cuenta, del socio, y el usuario del socio fueron realizados satisfactoriamente.

1.5. Poscondiciones de falla

La cuenta, socio, usuario no fue registrado correctamente.

Se indicará el motivo por el cual no se pudo registrar la cuenta en el sistema.

1.6. Actor principal

<i>Actor</i>	<i>Evento</i>
<i>Socio Usuario Administrador</i>	<i>Es el funcionario encargado del proceso de registro de nuevas cuentas para la caja de ahorros.</i>

Fuente: Propia

Tabla 5: Actores caso uso registrar cuentas ahorros

2. Flujo de Eventos

2.1. Flujo básico

1. El usuario selecciona la Opción del menú Administración de Cuentas.
2. El usuario selecciona la opción “Nuevo”
3. El sistema despliega el formulario para el ingreso de los datos requeridos para el registro de una nueva cuenta.
4. El usuario ingresa en el formulario un número de cédula válido.
5. El usuario ingresa los restantes datos solicitados en el formulario después de ingresar un número de cédula válido.
6. El usuario presiona el botón “Guardar”
7. El sistema valida que los datos se encuentren correctamente ingresados
8. El sistema registra la información correspondiente a los datos ingresados de la cuenta, socio, usuario.
9. Termina el CU

2.2. Flujos alternativos

En el paso 4 del flujo básico

El usuario ingresa en el formulario la siguiente información necesaria para el registro de la cuenta:

- Cédula de identidad

En el paso 5 del flujo básico

Ingresa la faltante información para registrar al socio y al usuario como son:

- Acciones
- Primer Nombre
- Segundo nombre
- Primer Apellido
- Segundo Apellido
- Dirección
- Teléfono celular
- Teléfono convencional
- Sexo
- Cuenta de usuario
- Password
- Email

2.3. Flujos de excepción

Información mal ingresada

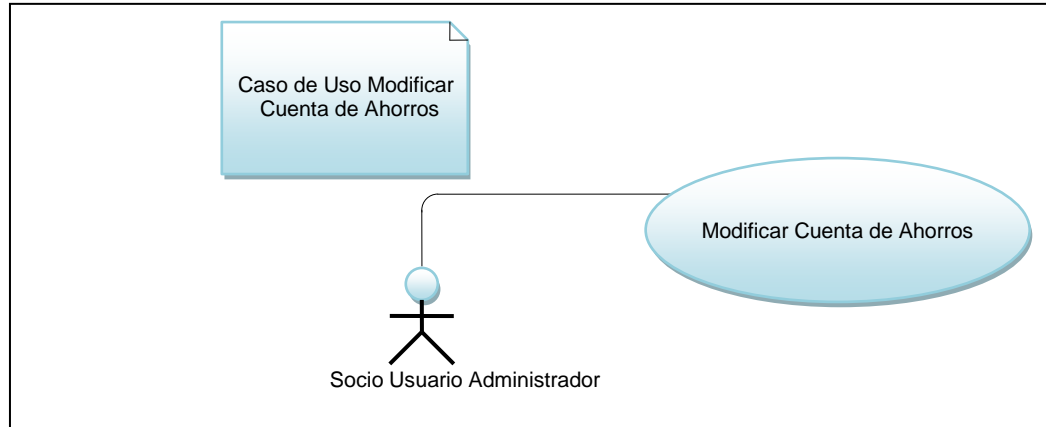
Si en el paso 4 del flujo principal, el sistema detecta que no se ha ingresado un número de cédula válido, el sistema desplegará el respectivo mensaje de error.

Si en el paso 5 del flujo principal, el sistema detecta que no se ha ingresado algún campo obligatorio el sistema desplegara el respectivo mensaje de error.

3. Requerimientos especiales

El email deberá ser en el formato correspondiente caso contrario se desplegará un error indicando que el formato del correo electrónico no es el correcto.

3.5.4. Especificación de caso de uso: Modificar cuenta de ahorros



Fuente: Propia

Figura 21: Caso de uso modificar cuenta de ahorros

1. Nombre de Caso de Uso

Modificar Cuenta de Ahorros

1.1. Descripción breve

Este caso de uso describe el proceso de modificación de las cuentas de ahorros en las cajas de ahorro. Este proceso lo realiza un socio usuario administrador de la caja de ahorros, mediante el uso de una interfaz gráfica donde se podrá modificar los datos de la cuenta, del socio, y del usuario que pertenece al socio.

1.2. Objetivo

Modificar las cuentas de ahorros.

1.3. Precondiciones

El socio usuario administrador de la caja de ahorros debe estar ingresado y autenticado en el Sistema.

1.4. Poscondiciones de éxito

La modificación de la cuenta, del socio, y el usuario fueron realizados satisfactoriamente.

1.5. Poscondiciones de falla

La cuenta, socio, usuario no fueron modificados correctamente.

Se indicará el motivo por el cual no se pudo modificar la cuenta en el sistema.

1.6. Actor principal

<i>Actor</i>	<i>Evento</i>
<i>Socio Usuario Administrador</i>	<i>Es el funcionario encargado del proceso de modificación de las cuentas de ahorros.</i>

Fuente: Propia

Tabla 6 : Actores caso uso modificar cuentas de ahorros

2. Flujo de Eventos

2.1. Flujo básico

1. El usuario selecciona la Opción de menú Administración de Cuentas.
2. Aparecerá el listado de todas las cuentas que pertenecen a la caja de ahorros.
3. El usuario selecciona el número de cédula del socio a quien le pertenece la cuenta que desea modificar.
4. El sistema mostrará el formulario para el ingreso de los datos que se pueden modificar.
5. El usuario presiona el botón “Guardar”
6. El sistema valida que los datos se encuentren correctamente ingresados
7. El sistema registra la información correspondiente a los datos modificados de la cuenta, socio, usuario.
8. Termina el CU

2.2. Flujos alternativos

En el paso 4 del flujo básico

Se podrá modificar los siguientes campos:

- Acciones
- Primer Nombre
- Segundo nombre
- Primer Apellido
- Segundo Apellido
- Dirección

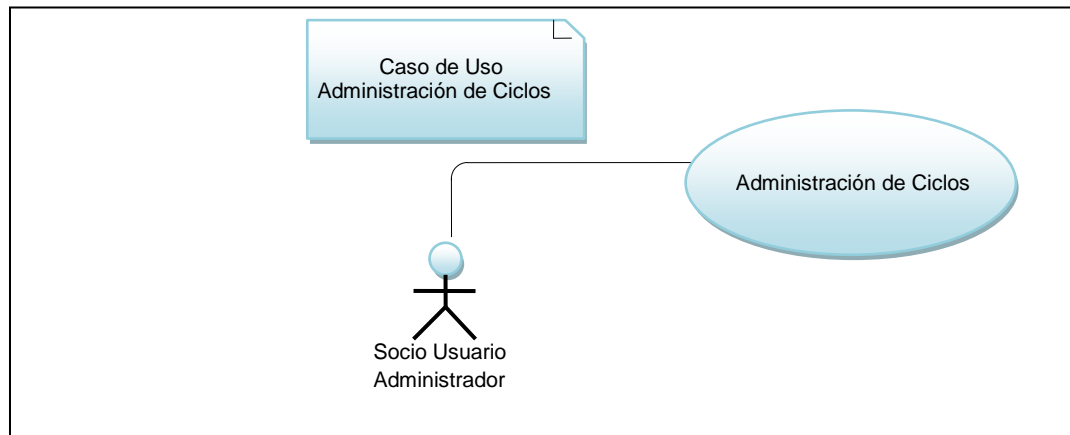
- Teléfono celular
- Teléfono convencional
- Sexo
- Cuenta de usuario
- Password
- Email

2.3. Flujos de excepción

Información mal ingresada

Si en el paso 4 del flujo principal, el sistema detecta que no se ha ingresado algún campo obligatorio el sistema desplegara el respectivo mensaje de error.

3.5.5. Especificación de caso de uso: Administración de ciclos



Fuente: Propia

Figura 22: Caso de uso Administración de Ciclos

1. Nombre de Caso de Uso

Administración de Ciclos

1.1. Descripción breve

Este caso de uso describe el proceso del registro y modificación de los ciclos correspondientes a un periodo de tiempo para las cajas de ahorros.

Este proceso lo realiza un socio usuario administrador, mediante el uso de una interfaz gráfica donde se registran y se modifican los datos del ciclo.

1.2. Objetivo

Registrar el ciclo de la caja de ahorro.

1.3. Precondiciones

El usuario administrador debe estar ingresado y autenticado en el sistema.

1.4. Poscondiciones de éxito

El registro o modificación del ciclo fueron realizados satisfactoriamente.

1.5. Poscondiciones de falla

El ciclo o modificación no fueron realizados correctamente.

Se indicará el motivo por el cual no se pudo registrar o modificar el ciclo en el sistema.

1.6. Actor principal

<i>Actor</i>	<i>Evento</i>
<i>Usuario Administrador</i>	<i>Es el funcionario encargado del proceso de registro o modificación del ciclo.</i>

Fuente: Propia

Tabla 7 : Actores caso uso administración de ciclos

2. Flujo de Eventos

2.1. Flujo básico

1. El usuario selecciona la Opción de menú Administración de Ciclos.
2. El usuario selecciona la opción “Nuevo”
3. El sistema despliega el formulario para el ingreso de los datos requeridos para el registro de un nuevo ciclo.
4. El usuario presiona el botón “Guardar”
5. El sistema valida que los datos se encuentren correctamente ingresados
6. El sistema registra la información correspondiente a los datos ingresados del ciclo.
7. Termina el CU

2.2. Flujos alternativos

Modificar

1. El usuario selecciona la Opción de menú Administración de ciclos.
2. El usuario selecciona el ciclo que desea modificar
3. El sistema mostrara los datos que pueden ser modificados
4. El usuario modificará los datos en el formulario
5. El usuario presionara el botón guardar
6. El sistema validará que los datos que se van a modificar estén correctamente ingresados.
7. El sistema registra los nuevos cambios que se realizaron en el ciclo.
8. Termina el CU.

En el paso tres del flujo básico el usuario ingresa la siguiente información:

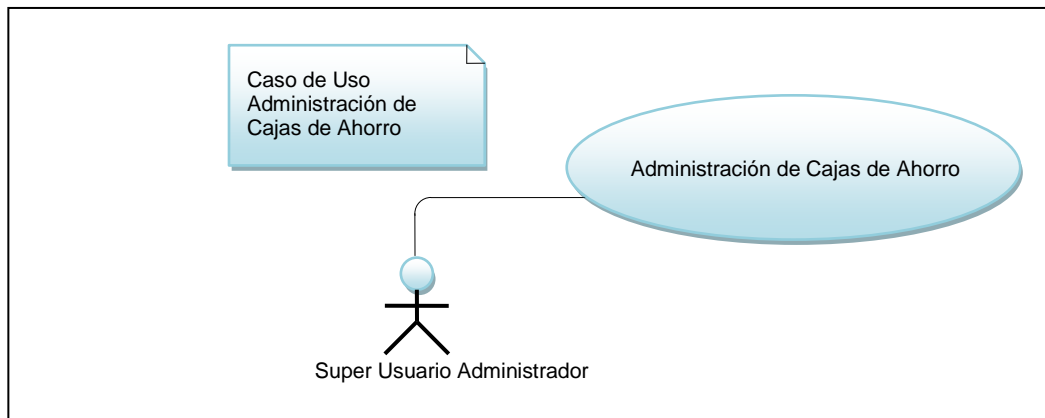
- Fecha inicio de ciclo
- Fecha de fin de ciclo
- Costo de la acción

2.3. Flujos de excepción

Información mal ingresada

Si en el paso tres del flujo básico, el sistema detecta que no se ha ingresado los datos requeridos, el sistema desplegará el respectivo mensaje de error.

3.5.6. Especificación de caso de uso: Administración de cajas de ahorros



Fuente: Propia

Figura 23: Caso de uso administración cajas de ahorros

1. Nombre de Caso de Uso

Administración de Cajas de Ahorros

1.1. Descripción breve

Este caso de uso describe el proceso del registro y modificación de las cajas de ahorros.

Este proceso lo realiza un super usuario administrador, mediante el uso de una interfaz gráfica donde se registran y se modifican los datos de la caja de ahorros.

1.2. Objetivo

Registrar la caja de ahorros del sistema.

1.3. Precondiciones

El super usuario administrador debe estar ingresado y autenticado en el Sistema.

1.4. Poscondiciones de éxito

El registro o modificación de la caja de ahorros fueron realizados satisfactoriamente.

1.5. Poscondiciones de falla

La caja de ahorros no fue registrada o modificada correctamente.

Se indicará el motivo por el cual no se pudo registrar o modificar la caja de ahorros en el sistema.

1.6. Actor principal

<i>Actor</i>	<i>Evento</i>
<i>Super Usuario Administrador</i>	<i>Es el funcionario encargado del proceso de registro o modificación de la caja de ahorros.</i>

Fuente: Propia

Tabla 8: Actores caso uso administración de cajas de ahorros

2. Flujo de Eventos

2.1. Flujo básico

1. El usuario selecciona la opción del menú Administración de Cajas de Ahorro.
2. El usuario selecciona la opción “Nuevo”.
3. El sistema despliega el formulario para el ingreso de los datos requeridos para el registro de una nueva caja de ahorros.
4. El usuario presiona el botón “Guardar”.
5. El sistema verifica que los datos ingresados se encuentren correctamente.
6. El sistema registra la información correspondiente a los datos ingresados de la nueva caja de ahorros.
7. Termina el CU

2.2. *Flujos alternativos*

Modificar

1. El usuario selecciona la opción de menú Administración de Cajas de Ahorro.
2. El usuario selecciona del listado la caja de ahorro que desea modificar.
3. El sistema mostrará los datos de la caja de ahorro que pueden ser modificados.
4. El usuario modificará los datos que desea cambiar.
5. El usuario presionará el botón guardar.
6. El sistema validará que los datos que se van a modificar estén correctamente ingresados.
7. El sistema registra los nuevos cambios que se realizaron en la caja de ahorros.
8. Termina el CU.

En el paso tres del flujo básico el usuario ingresa la siguiente información para registrar la caja de ahorros:

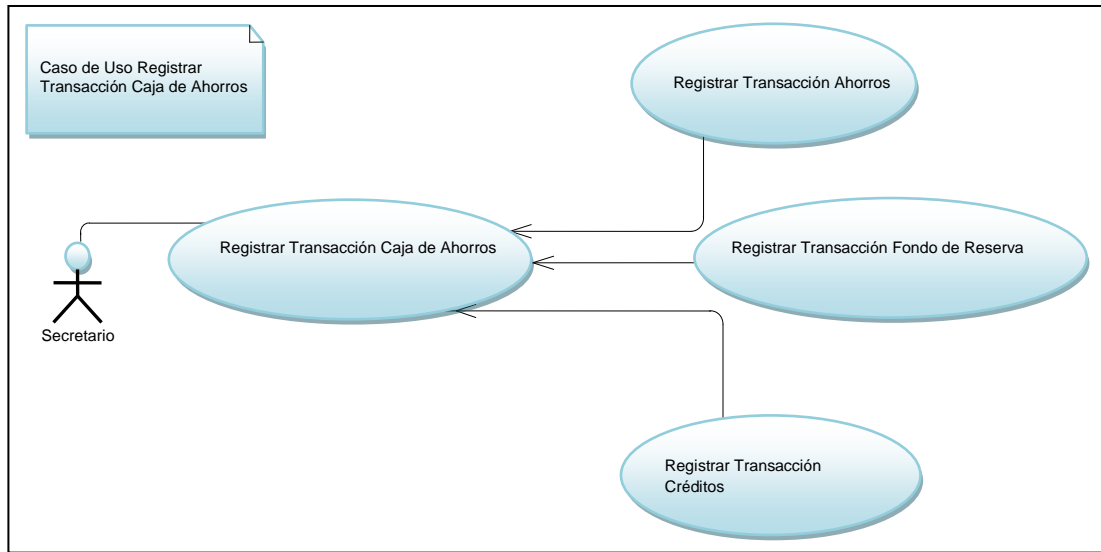
- Nombre de Caja de Ahorros

2.3. *Flujos de excepción*

Información mal ingresada

Si en el paso tres del flujo básico, el sistema detecta que no se ha ingresado los datos requeridos, el sistema desplegará el respectivo mensaje de error.

3.5.7. Especificación de caso de uso: Registrar transacción caja de ahorros



Fuente: Propia

Figura 24: Caso de Uso Registrar Transacción Caja de Ahorro

1. Nombre de Caso de Uso

Registrar Transacción Caja de Ahorros

1.1. Descripción breve

Este caso de uso describe el proceso del registro de la transacción de la caja de ahorros.

Este proceso lo realiza un usuario con el rol de secretario, mediante el uso de una interfaz gráfica donde se registran y se modifican los datos de la transacción caja de ahorros.

1.2. Objetivo

Registrar la transacción caja de ahorros del sistema.

1.3. Precondiciones

El usuario secretario debe estar ingresado y autenticado en el sistema.

1.4. Poscondiciones de éxito

El registro o modificación de la transacción caja de ahorro fueron realizados satisfactoriamente.

1.5. Poscondiciones de falla

La transacción caja de ahorro no fue registrada o modificada correctamente. Se indicará el motivo por el cual no se pudo registrar o modificar la transacción caja de ahorro en el sistema.

1.1. Actor principal

<i>Actor</i>	<i>Evento</i>
<i>Usuario Secretario</i>	<i>Es el funcionario encargado del proceso de registro o modificación de la transacción caja de ahorros.</i>

Fuente: Propia

Tabla 9: Actores caso uso registrar transacción caja de ahorros

2. Flujo de Eventos

2.1. Flujo básico

1. El usuario selecciona la opción del menú Transacciones Caja de Ahorro.
2. El usuario selecciona la opción “Nuevo”.
3. El sistema despliega el formulario para el ingreso de los datos requeridos para el registro de una nueva transacción caja de ahorros.
4. El usuario presiona el botón “Aceptar”.
5. El sistema verifica que los datos se encuentren correctamente ingresados.
6. El sistema registra la información correspondiente a los datos ingresados de la nueva transacción caja de ahorros.
7. Termina el CU.

2.2. Flujos alternativos

Registrar el detalle de la transacción caja de ahorros.

1. El usuario selecciona el detalle caja de ahorro que desea modificar presionando el botón editar que se encuentra en cada uno de los detalles de la transacción caja de ahorros.

2. El usuario registra los datos correspondientes al detalle caja de ahorros, los cuales son: ahorro, ahorro de fondo de reserva, pago de créditos, pago de interés, pago de acciones, asistencia, solicitud de créditos.
3. El usuario presiona el botón guardar que se encuentra activado en el final de cada detalle transacción caja de ahorro.
4. El sistema validará que los datos ingresados sean correctos.
5. El sistema registra los datos en las cuentas correspondientes de cada detalle transacción caja de ahorros.
6. Termina el CU.

Registrar Transacción Créditos

1. El usuario selecciona del detalle caja de ahorro el check correspondiente a la columna de solicitud de crédito con lo que se indica que el socio está solicitando un crédito.
2. El sistema registrará la solicitud de crédito.
3. El usuario selecciona la solicitud de crédito presionando el botón verificar crédito.
4. El sistema mostrará los datos correspondientes a la solicitud de crédito.
5. El usuario deberá ingresar los datos correspondientes al nuevo crédito.
6. El usuario presiona el botón aceptar.
7. El sistema validará que los datos ingresados sean correctos.
8. El sistema registra los datos del crédito.
9. Termina el CU.

Registrar transacción Ahorro

1. El usuario selecciona el detalle caja de ahorro que desea modificar presionando el botón editar que se encuentra en cada uno de los detalles.
2. El usuario registra los datos correspondientes al ahorro.
3. El usuario presiona el botón guardar que se encuentra activado en el final de cada detalle transacción caja de ahorro.
4. El sistema validará que los datos ingresados sean correctos.

5. El sistema registra los datos en las cuentas correspondientes de cada detalle transacción caja de ahorro.
Termina el CU.

Registrar transacción Ahorro Fondo Reserva

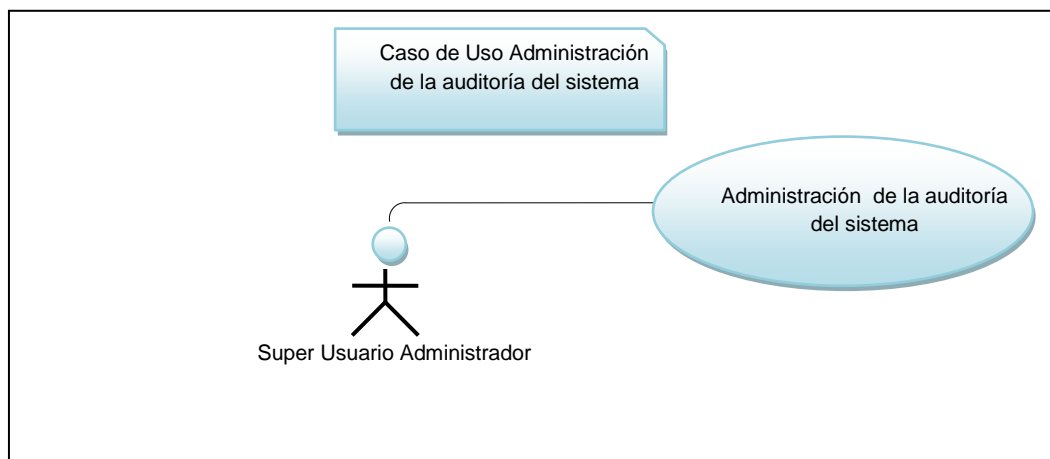
1. El usuario selecciona el detalle caja de ahorro que desea modificar presionando el botón editar que se encuentra en cada uno de los detalles.
2. El usuario registra los datos correspondientes al ahorro fondo reserva.
3. El usuario presiona el botón guardar que se encuentra activado en el final de cada detalle transacción caja de ahorro.
4. El sistema validará que los datos ingresados sean correctos.
5. El sistema registra los datos en las cuentas correspondientes de cada detalle transacción caja de ahorro.
Termina el CU.

2.3. Flujos de excepción

Información mal ingresada

Si el sistema detecta que no se ha ingresado los datos requeridos, el sistema desplegará el respectivo mensaje de error.

3.5.8. Especificación de caso de uso: Administración de la auditoría del sistema



Fuente: Propia

Figura 25: Caso de uso Administración de la auditoría del sistema

4. Nombre de Caso de Uso

Administración de la auditoría del sistema.

4.1. Descripción breve

El caso de uso describe el proceso de la administración de las auditorías que se realizan al sistema permitiendo conocer cambios que se realizan a los registros del sistema.

Este proceso lo realiza un usuario con el rol de super usuario, mediante el uso de una interfaz gráfica donde se administran los datos de las auditorías del sistema.

4.2. Objetivo

Administrar la auditoría del sistema.

4.3. Precondiciones

El super usuario administrador debe estar ingresado y autenticado en el sistema.

4.4. Poscondiciones de éxito

Los registros para la administración de la auditoría fueron registrados satisfactoriamente.

4.5. Poscondiciones de falla

Los registros para la administración de la auditoría no fueron registrados satisfactoriamente.

4.6. Actor principal

<i>Actor</i>	<i>Evento</i>
<i>Super Usuario Administrador</i>	<i>Es el funcionario encargado del proceso de registrar datos para la administración de la auditoría del sistema.</i>

Fuente: Propia

Tabla 10: Actores caso uso administración de la auditoría del sistema

5. Flujo de Eventos

5.1. Flujo básico

1. El usuario selecciona la opción de menú Administración auditoría sistema.
2. El usuario selecciona la opción “Nuevo”
3. El sistema despliega el formulario para el ingreso de los datos requeridos para el registro de una nueva tabla para ser auditada.
4. El usuario ingresa los datos solicitados en el formulario
5. El usuario podrá seleccionar los eventos que deben ejecutarse para realizar la auditoría sobre la tabla.
6. El usuario presiona el botón “Guardar”
7. El sistema valida que los datos se encuentren correctamente ingresados
8. El sistema registra la información correspondiente a la administración de la auditoría.
9. Termina el CU.

5.2. Flujos alternativos

En el paso cuatro del flujo básico

El usuario ingresa en el formulario la siguiente información necesaria para el registro de la auditoría:

- Nombre de la tabla
- Eventos a ejecutarse (Select ,Insert ,Update,Delete)

En el paso 5 del flujo básico

El usuario selecciona con las opciones de Si o No sobre los eventos que desea que se ejecuten para que se registre la auditoría.

Búsqueda de auditorías del sistema.

1. El usuario selecciona el nombre de la tabla que desea buscar las auditorías que se generaron sobre la tabla.
2. El usuario selecciona la fecha inicial y la fecha final correspondientes a rango de fechas que desea que se muestre los registros de auditoría que se generaron.
3. El usuario presiona el botón buscar que se encuentra al final de las fechas inicial y final.
4. El sistema validará que los datos ingresados sean correctos.

5. El sistema busca los datos que coincidan con el rango de fechas que se ingreso.
6. El sistema muestra los resultados que se generaron en la búsqueda
7. Termina el CU.

Ver campos de auditoría.

1. El usuario selecciona el link ver campos que se encuentra al final de cada tabla auditada
2. El sistema muestra todos los campos que son auditados sobre la tabla.
3. El usuario selecciona la opción “Nuevo”.
4. El sistema despliega el formulario para el ingreso de los datos requeridos para el registro de un nuevo campo de tabla para ser auditado.
5. El usuario selecciona los eventos que desea que se ejecuten para que se generen los registros de auditoría sobre el campo nuevo.
6. El usuario presiona el botón Guardar.
7. El sistema válida que los datos se encuentren correctamente ingresados.
8. El sistema registra la información correspondiente al nuevo campo de la tabla para ser auditado.

5.3. *Flujos de excepción*

Información mal ingresada

Si en el paso cuatro del flujo principal, el sistema detecta errores de ingreso de la información, el sistema desplegará el respectivo mensaje en base al tipo de error detectado.

Si en el paso cuatro del flujo principal el sistema detecta que ya existe un nombre de tabla el sistema desplegará el respectivo mensaje indicando que ya existe un nombre de tabla en el sistema.

3.6. Fase de elaboración

En la fase de elaboración se analizan los requisitos y se desarrolla un prototipo de la arquitectura (incluyendo las partes más relevantes o críticas del sistema la revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase.

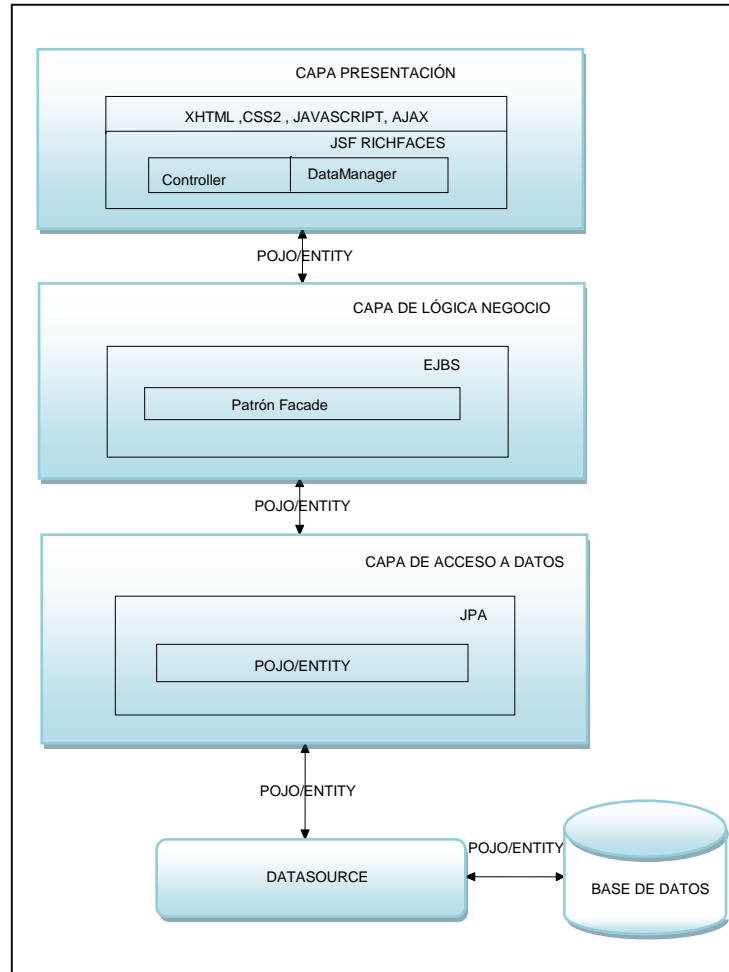
La arquitectura de la aplicación permite definir la organización interna de la misma estableciendo la funcionalidad en capas independientes y bien definidas que permiten tener mejor control de la ubicación de las reglas de negocio y por lo tanto facilita las tareas de mantenimiento de la misma.

3.6.1.Arquitectura de la aplicación

La arquitectura de la aplicación está basada en la arquitectura JEE y en patrones de diseño ampliamente usados, así como estándares abiertos.

La arquitectura que se implementa es de tres capas la misma que permite una separación de la presentación con la lógica del negocio y el acceso a los datos permitiendo tener mejor control de la ubicación de la lógica de negocio y por lo tanto facilita las tareas de mantenimiento de las mismas, por lo cual esto significa que si se modifica alguna capa no afectara a las demás capas:

- Capa de presentación
(JSF)
- Capa de lógica de negocio
(EJB)
- Capa de acceso a datos
(JPA)



Fuente: Propia

Figura 26: Vista lógica de los patrones de la Arquitectura

Para la implementación de la capa de presentación se utiliza JSF en la versión 2.0 la que es un Framework web MVC (model-view-controller) incluido como estándar para la capa web en Java EE .

Para la implementación de la capa de lógica de negocio se utiliza EJBs en la versión 3.0 que son componentes gestionados por el contenedor de EJB que implementan la lógica de la aplicación. Gestionan el tratamiento y flujo de datos en la parte del servidor.

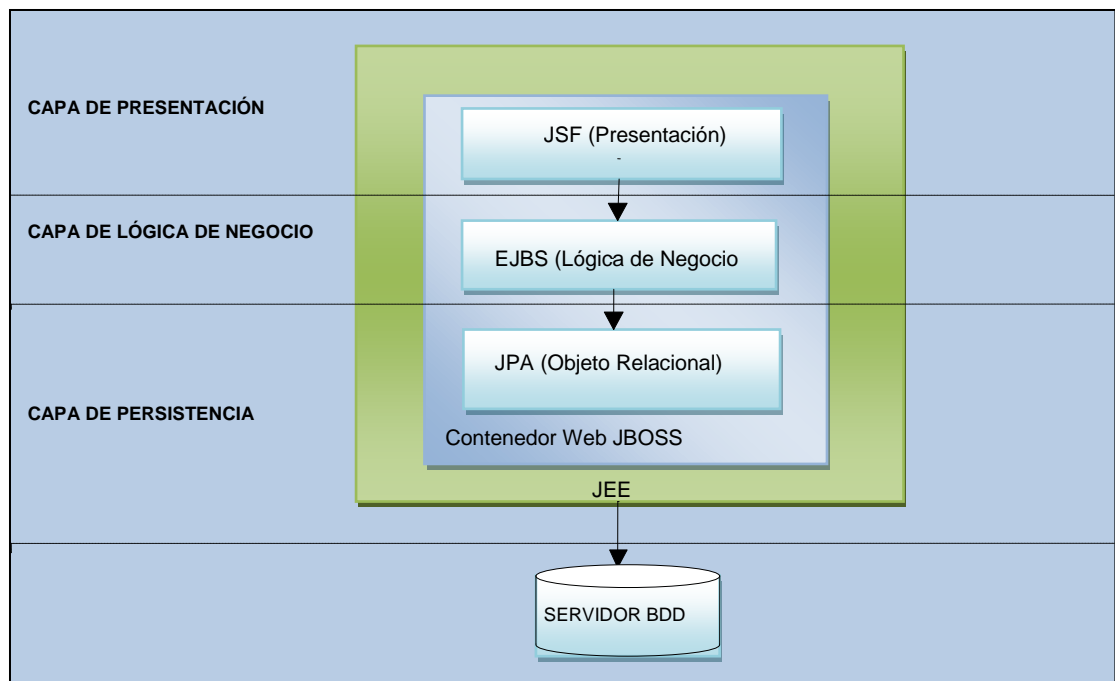
Para la implementación de la capa de acceso a datos se utilizó JPA en la versión 2.0

(Java Persistence API) como mecanismo de mapeo Objeto/Relacional para proveer de una capa de acceso a la base de datos.

Para implementar la arquitectura JEE se utiliza el servidor JBOSS con soporte para JEE.

Para implementar la parte del View-Controller del patrón MVC se utiliza una combinación entre el Framework Java Server Faces JSF, desarrollado por Java Community Process como JSR 127, que definía JSF 1.0 y 1.1, y JSR13 252 que definía JSF 1.2 y JSR 314 que define JSF 2.0 y componentes RichFaces que permiten crear un ambiente Ajax de manera fácil, rápida y limpia.

A nivel de la capa de lógica de negocio se utilizara EJBS debido a que son componentes que son responsables de implementar la lógica de la aplicación, y a nivel de la capa de Persistencia se utilizará JPA como herramienta de mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante anotaciones que permiten establecer estas relaciones.



Fuente: Propia

Figura 27: Arquitectura de la aplicación

¹³ JSR: Java Specification Request documentos formales que describen las especificaciones y tecnologías propuestas para que sean añadidas a la plataforma Java.

3.6.2. Estructura de la aplicación

La aplicación está construida como un JEE Enterprise Application, siendo empaquetada en un fichero EAR para ser desplegada en el servidor de JBOSS.

La aplicación está desarrollada de forma separada en dos proyectos:

prjCajaAhorroWeb: Proyecto Web de la aplicación, contiene la implementación de la vista y el controlador.

Dentro del proyecto encontramos la lógica de navegación y control de eventos de la interfaz, páginas del sistema, validación de entrada de datos entre otros.

prjModeloCajaAhorro: Proyecto de implementación del modelo del negocio.

El proyecto contiene:

- EJBS: Constituye el punto de entrada a la implementación interna del sistema donde se exponen los métodos de lógica del negocio.
- Componentes Empresariales: Implementación en software de conceptos de negocios (Lógica de dominio)
- Entidades empresariales: Contenedores de datos (DTO) que encapsulan y ocultan los detalles de su representación.
- Componentes lógicos de acceso a datos (Capa de Persistencia): Constituye los objetos que permiten un único acceso a los datos. Esta capa realiza todas las conversiones y validaciones necesarias que estén relacionadas con el modelo de base de datos y permite realizar el CRUD (Create, Read, Update y Delete) de las entidades de negocio

3.6.3. Almacenamiento de datos

El almacenamiento de datos del Sistema Caja de Ahorros Comunitario son almacenados en base de datos relacional, el servidor de base de datos que se utiliza es PostgreSQL 9.1.

PostgreSQL (Pecos Martínez, 2009) es un sistema de gestión de base de datos objeto-relacional, ya que incluye características de la orientación a objetos, como puede ser la

herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional etc.

A continuación se enumeran las principales características de este gestor de bases de datos:

- Implementación del estándar SQL92/SQL99¹⁴.
- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC¹⁵, IP), cadenas de bits, etc. También permite la creación de tipos propios.
- Incorpora una estructura de datos array.
- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, etc.
- Permite la declaración de funciones propias, así como la definición de disparadores.
- Soporta el uso de índices, reglas y vistas.
- Incluye herencia entre tablas por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.

PostgreSQL es un magnífico gestor de bases de datos, capaz de competir con muchos gestores comerciales, por otro lado, la velocidad de respuesta que ofrece este gestor con bases de datos relativamente pequeñas puede parecer un poco deficiente, aunque esta misma velocidad la mantiene al gestionar bases de datos realmente grandes, cosa que resulta excelente.

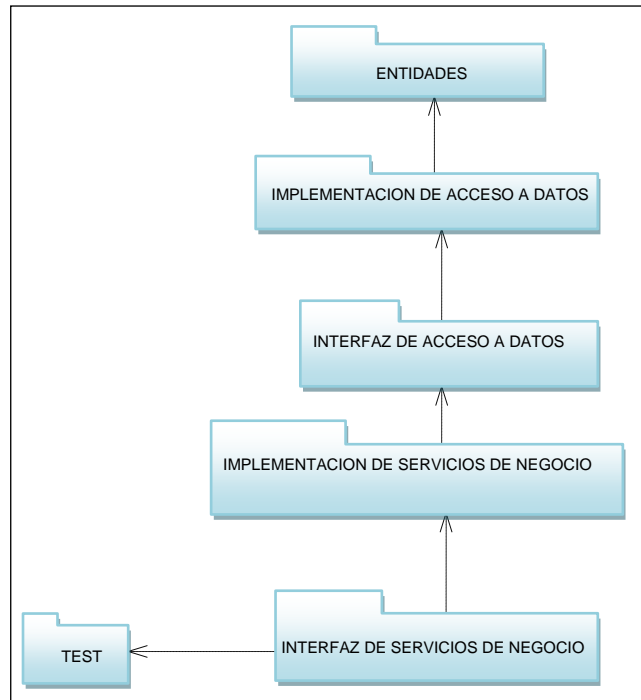
3.7. Fase de construcción

En la fase de construcción se implementan todos los casos de uso que forman parte de la fase de inyección. El sistema de Caja de Ahorro Comunitario se construye en base a las iteraciones, a la cual se le aplican las pruebas correspondientes y se validan con el usuario.

¹⁴ **SQL92/SQL99**: Son estándares del lenguaje SQL

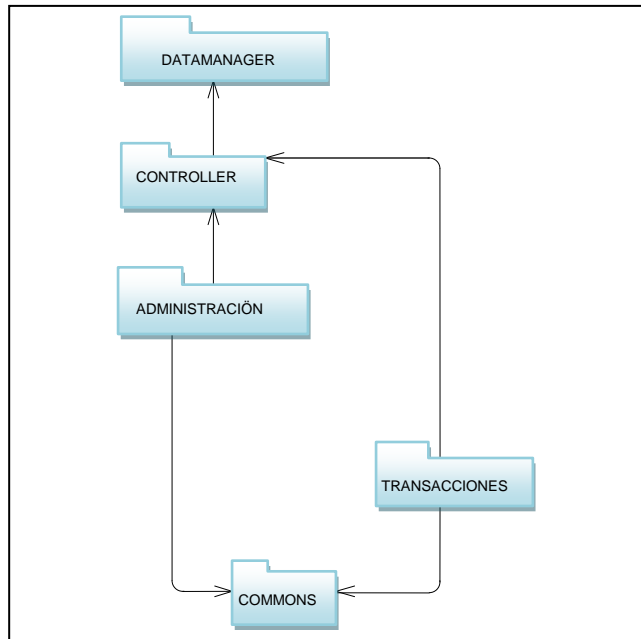
¹⁵ **MAC**: (*Media Access Control*) es un identificador de 48 bits (6 bloques hexadecimales) que corresponde de forma única a una tarjeta o dispositivo de red.

3.7.3. Diagrama de paquetes



Fuente: Propia

Figura 30: Diagrama de paquetes Modelo

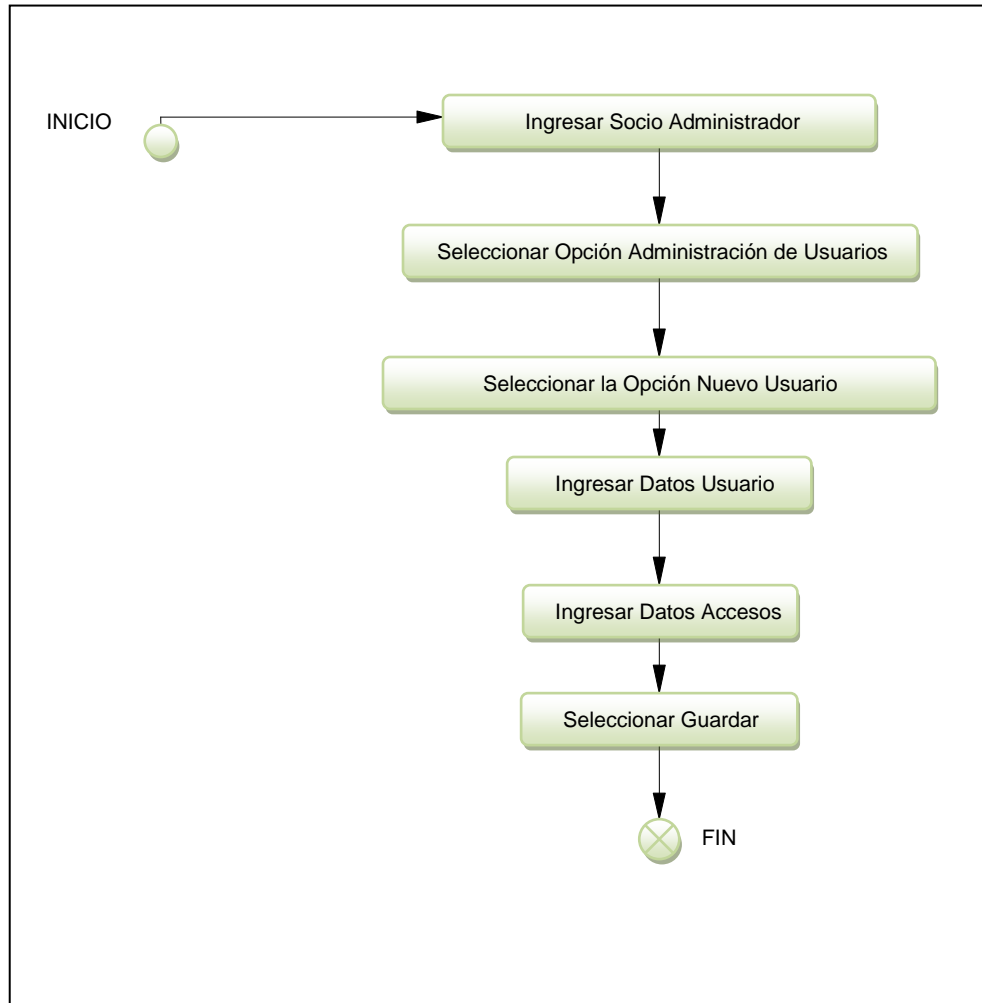


Fuente: Propia

Figura 31: Diagrama de paquetes Web

3.7.4. Diagrama de actividades

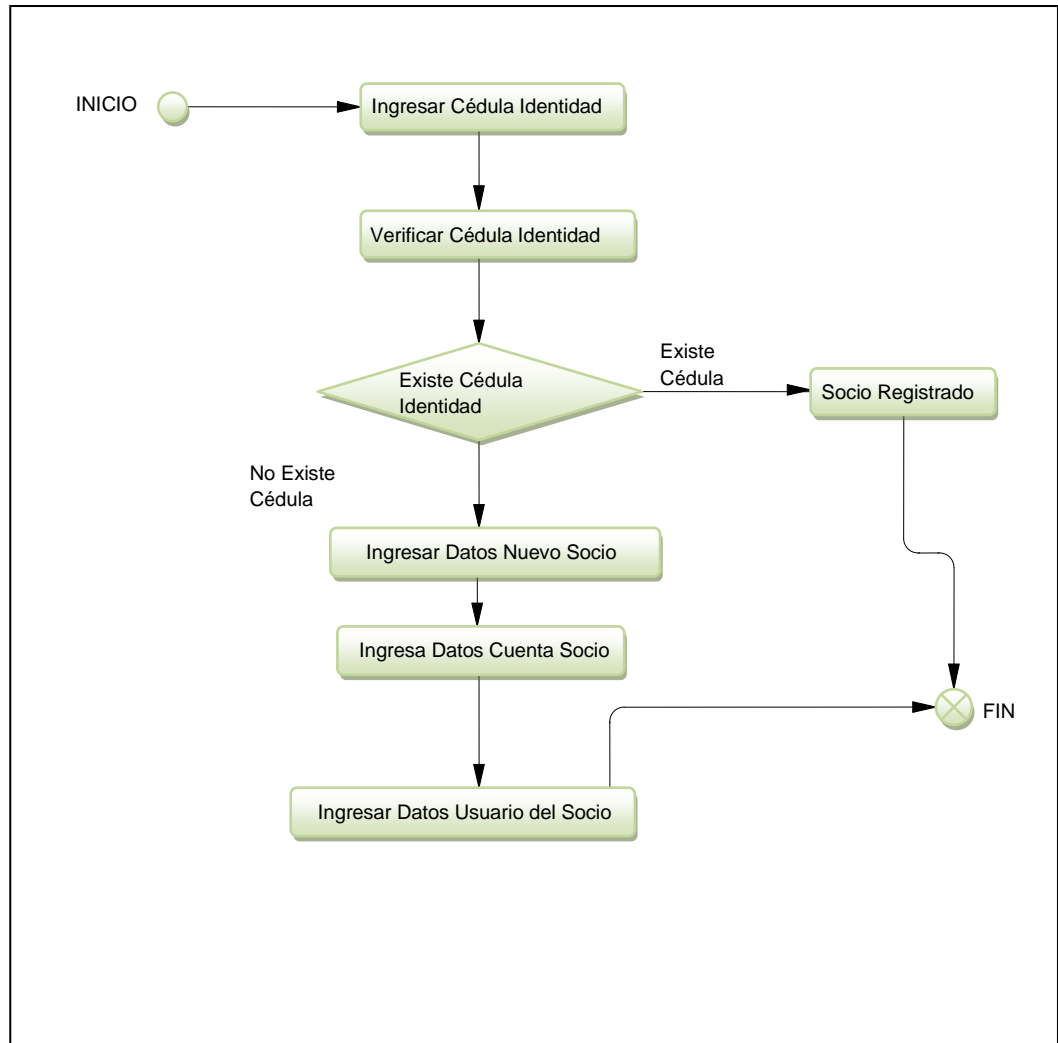
3.7.4.1. Diagrama de actividades registro de usuarios y accesos



Fuente: Propia

Figura 32: Diagrama de Actividades Registro Usuarios y Accesos

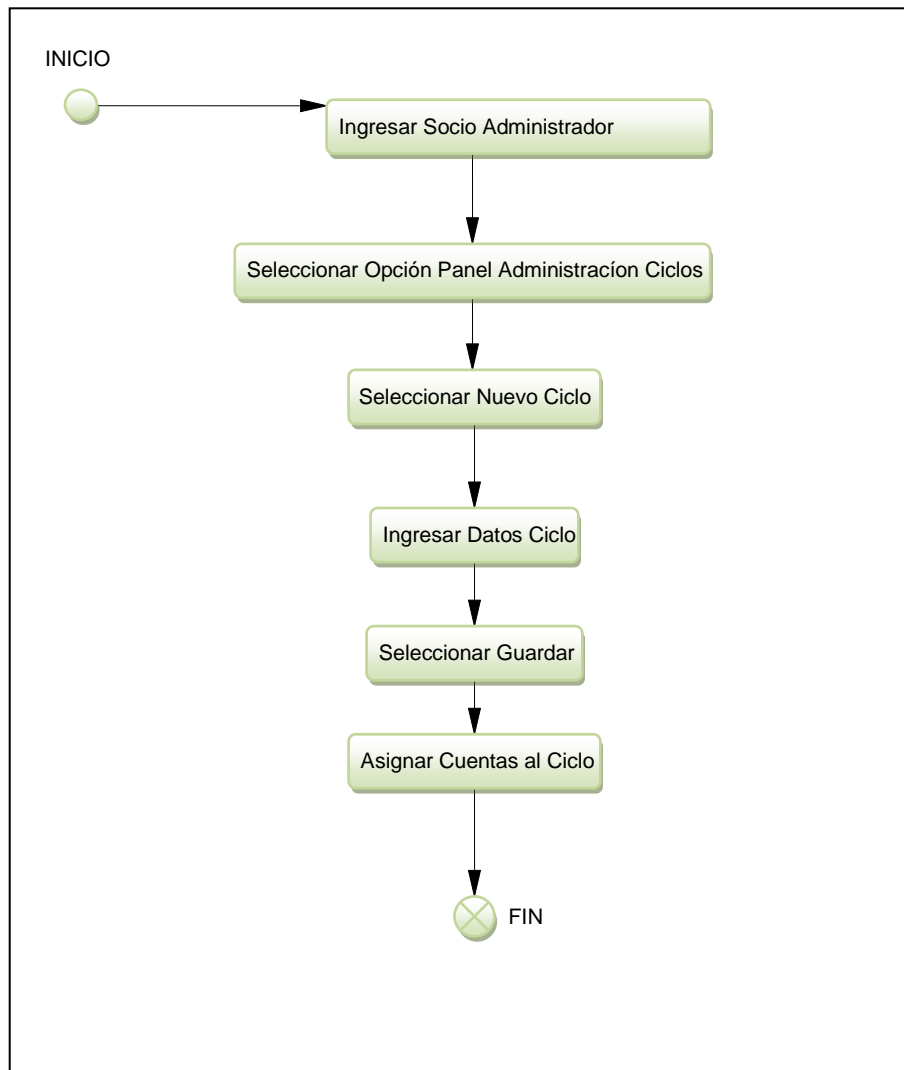
3.7.4.2. Diagrama de actividades registro cuenta de ahorros



Fuente: Propia

Figura 33: Registro Cuenta Ahorros

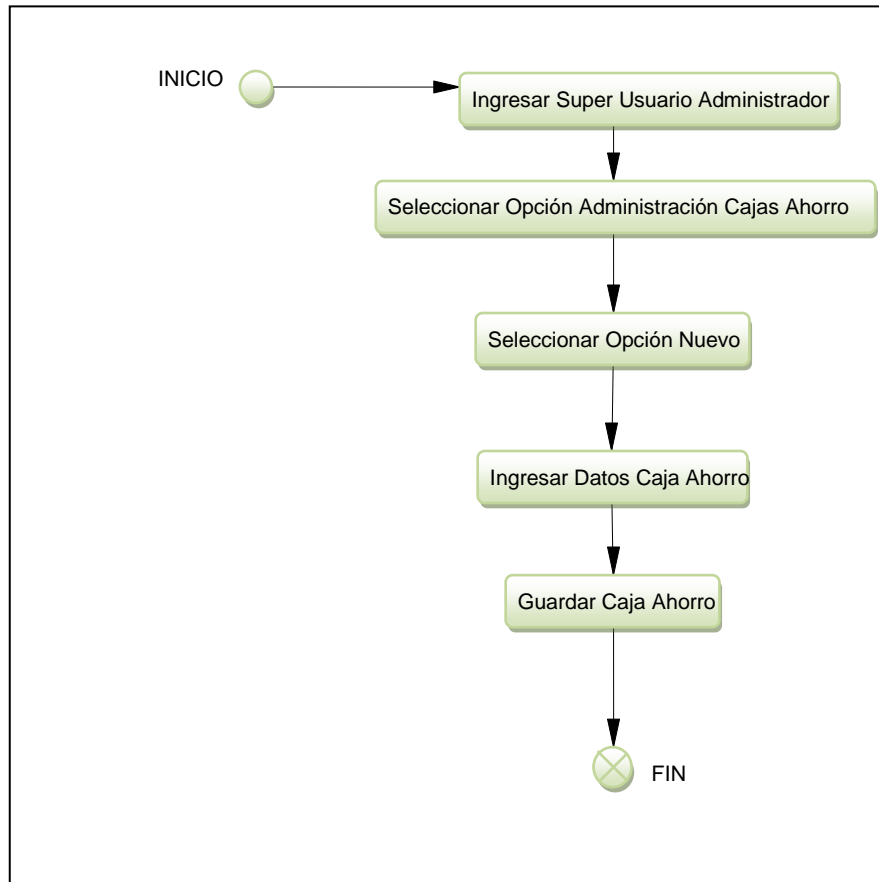
3.7.4.3. Diagrama de actividades administración de ciclos.



Fuente: Propia

Figura 34: Diagrama de actividades Administración de ciclos

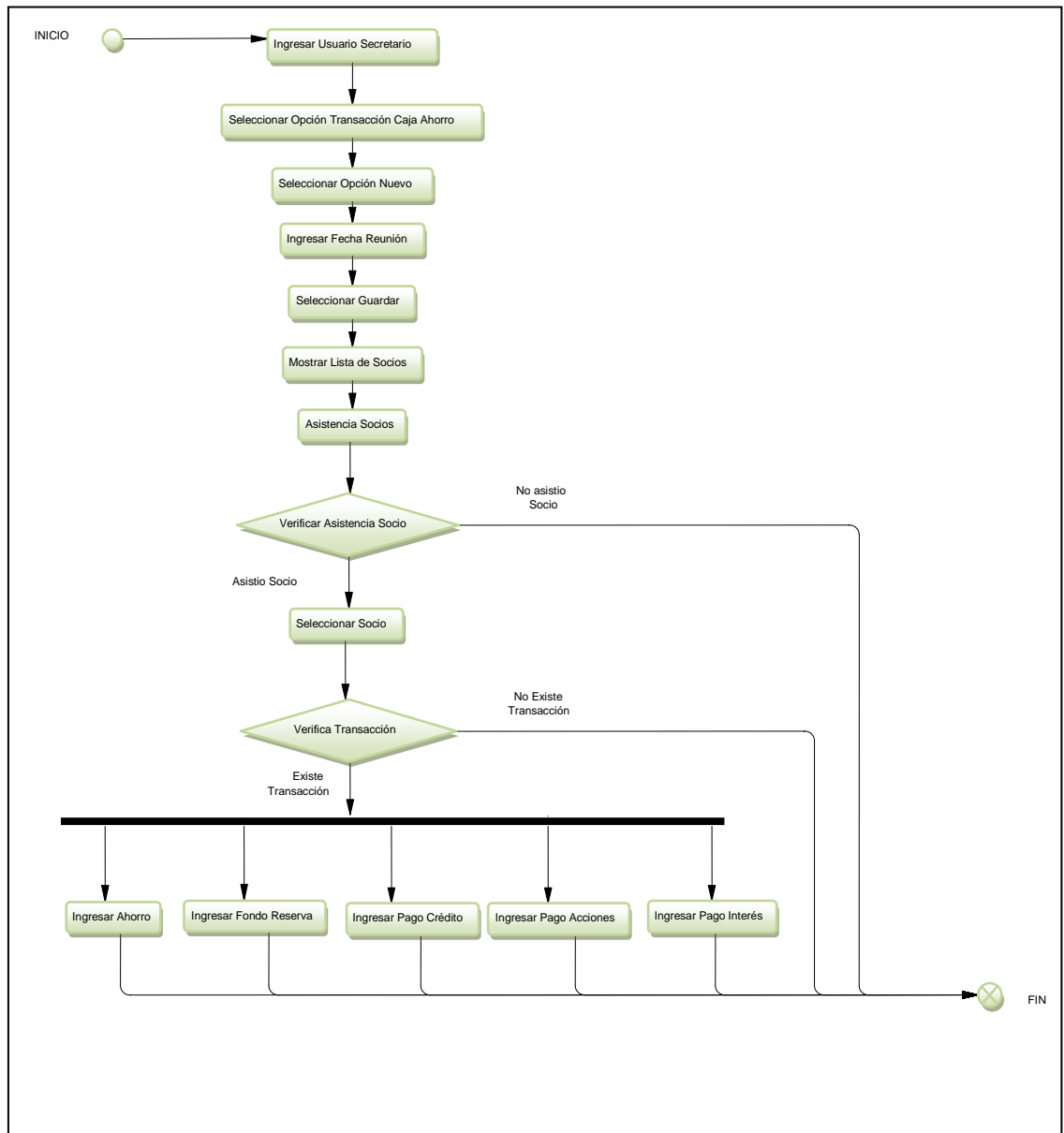
3.7.4.4. Diagrama de actividades Administración de cajas de ahorro.



Fuente: Propia

Figura 35: Diagrama de actividades Administración de Cajas de Ahorro

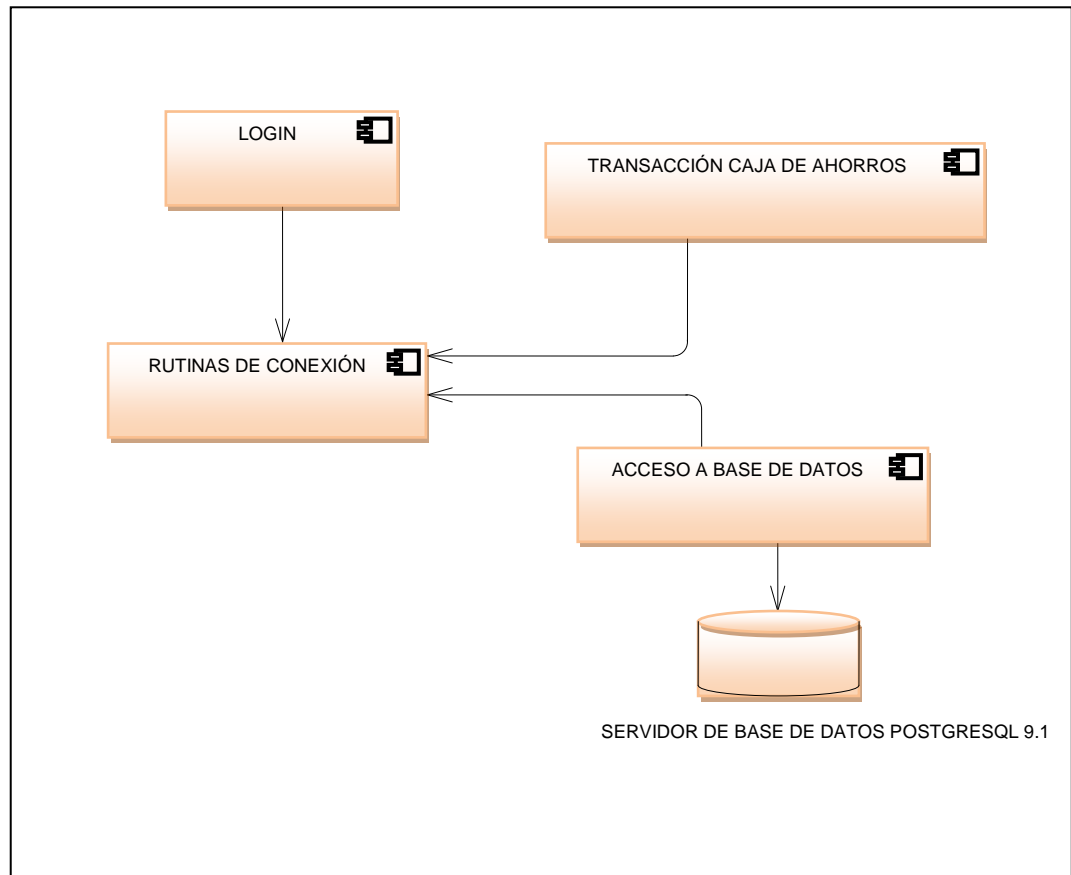
3.7.4.5. Diagrama de actividades Registrar transacción caja de ahorro.



Fuente: Propia

Figura 36 : Diagrama de actividades Registrar transacción caja de ahorros

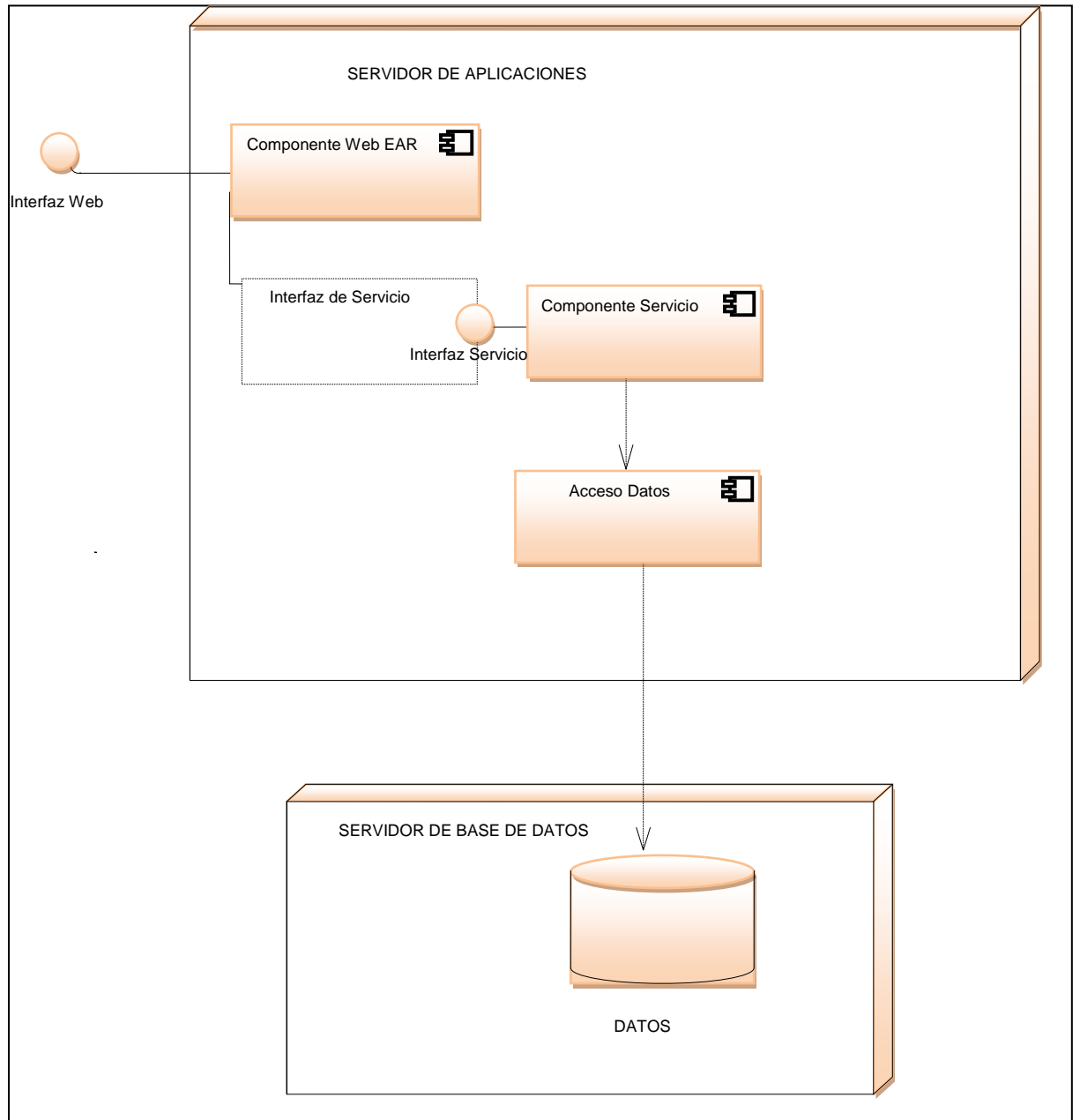
3.7.5. Diagrama de componentes



Fuente: Propia

Figura 37 : Diagrama de Componentes

3.7.6. Diagrama de componentes y despliegue



Fuente: Propia

Figura 38: Diagrama de componentes de despliegue

3.8. Fase de transición

En la fase de transición se especifican los casos de pruebas.

3.8.1.Caso de prueba registro de usuarios y accesos

1. Descripción

Este artefacto cubre el conjunto de pruebas realizadas sobre el caso de uso “Registro Usuarios y Accesos”

Las pruebas realizadas en este caso son:

- Registrar usuarios.

El entorno del cual partiremos para realizar la prueba será el formulario de entrada de la aplicación.

2. Registrar usuarios y accesos

2.1. Descripción

Ingresar al sistema con el usuario que tenga permisos para registrar usuarios.

2.2. Condiciones de Ejecución

El Socio Usuario Administrador debe estar ingresado y autenticado en el sistema.

2.3. Entrada

- Introducir el usuario en el campo usuario
- Introducir la contraseña en el campo contraseña
- Pulsar el botón ingresar de la aplicación.
- Se muestra el menú con las opciones que puede realizar el usuario actual que se encuentra autenticado en el sistema.
- Seleccionar del menú administración escoger la opción administración de usuarios
- Escoger la opción Nuevo aparece la interfaz para registrar al nuevo usuario.
- El usuario ingresa el nombre del usuario.
- El usuario ingresa el password del usuario.
- El usuario selecciona los roles que desea asignarle al usuario que está creando
- Después de ingresar los datos del usuario presionar el botón guardar.

- El sistema muestra el mensaje indicando que el usuario se guardo correctamente.

2.4. Resultado esperado

El sistema registra al usuario con sus accesos.

2.5. Evaluación de la prueba

Prueba superada con éxito.

3.8.2.Caso de prueba registrar cuenta de ahorros

1. Descripción

Este artefacto cubre el conjunto de pruebas realizadas sobre el caso de uso “Registrar cuenta de ahorros”

Las pruebas realizadas en este caso son:

- Registrar cuentas de ahorros.

El entorno del cual partiremos para realizar la prueba será el formulario de entrada de la aplicación.

2. Registrar cuenta de ahorros

2.1. Descripción

Ingresar al sistema con el usuario que tenga permisos para registrar cuenta de ahorros.

2.2. Condiciones de Ejecución

El Socio Usuario Administrador debe estar ingresado y autenticado en el sistema.

2.3. Entrada

- Introducir el usuario en el campo usuario
- Introducir la contraseña en el campo contraseña
- Pulsar el botón ingresar de la aplicación.

- Se muestra el menú con las opciones que puede realizar el usuario actual que se encuentra autenticado en el sistema.
- Seleccionar del menú administración escoger la opción administración de cuentas.
- Escoger la opción nuevo aparece la interfaz para registrar la nueva cuenta de ahorros.
- El usuario ingresa la cédula de identidad.
- El sistema verifica si la cédula de identidad es correcta.
- El sistema verifica si ya se encuentra registrado el socio.
- Habrá dos opciones si ya se encuentra registrado el socio
- Verificar si ya tiene cuenta de ahorros en la caja de ahorros del usuario administrador logueado.
- Si ya tiene no podrá registrar la cuenta de ahorros y el sistema mostrara un mensaje indicándole que ya tiene una cuenta de ahorros.
- Si no la tiene y si ya se encuentra registrado podrá ingresar los datos para registrar la cuenta de ahorros.
- El usuario ingresa el primer nombre del socio.
- El usuario ingresa el segundo nombre del socio.
- El usuario ingresa el primer apellido del socio.
- El usuario ingresa el segundo apellido del socio.
- El usuario ingresa el número de acciones del socio.
- El usuario ingresa la dirección del socio.
- El usuario ingresa el teléfono celular del socio.
- El usuario ingresa el teléfono convencional del socio.
- El usuario selecciona el sexo del socio.
- El usuario ingresa el nombre de usuario para el socio.
- El usuario ingresa el email del usuario para el socio.
- El usuario ingresa el password para el usuario del socio.
- Después de ingresar los datos para registrar la cuenta de ahorros el usuario presionar el botón guardar.
- El sistema muestra el mensaje indicando que la cuenta se guardo correctamente.

2.4. Resultado esperado

El sistema registra la cuenta de ahorros

2.5. Evaluación de la prueba

Prueba superada con éxito

3.8.3.Caso de prueba administración de ciclos

1. Descripción

Este artefacto cubre el conjunto de pruebas realizadas sobre el caso de uso “Administración de Ciclos”

Las pruebas realizadas en este caso son:

- Registrar Ciclos.

El entorno del cual partiremos para realizar la prueba será el formulario de entrada de la aplicación.

2. Registrar ciclos

2.1. Descripción

Ingresar al sistema con el usuario que tenga permisos para registrar ciclos.

2.2. Condiciones de Ejecución

El Socio Usuario Administrador debe estar ingresado y autenticado en el sistema.

2.3. Entrada

- Introducir el usuario en el campo usuario
- Introducir la contraseña en el campo contraseña
- Pulsar el botón ingresar de la aplicación.
- Se muestra el menú con las opciones que puede realizar el usuario actual que se encuentra autenticado en el sistema.
- Seleccionar del menú administración escoger la opción administración de ciclos.
- Escoger la opción Nuevo aparece la interfaz para registrar al nuevo ciclo.
- El usuario ingresa la fecha inicial.

- El usuario ingresa la fecha final.
- El usuario ingresa la cantidad de ahorro para los fondos de reserva del ciclo.
- El usuario ingresa la tasa de interés para el ciclo.
- El usuario selecciona el tipo de acción que se va a aplicar para el nuevo ciclo.
- Después de ingresar los datos del ciclo presionar el botón guardar.
- El sistema muestra un mensaje indicando que el ciclo se guardo correctamente.

2.4. Resultado esperado

El sistema registra el nuevo ciclo.

2.5. Evaluación de la prueba

Prueba superada con éxito

3.8.4.Caso de pruebas administración de cajas de ahorros

1. Descripción

Este artefacto cubre el conjunto de pruebas realizadas sobre el caso de uso “Administración de Cajas de Ahorros”

Las pruebas realizadas en este caso son:

- Registrar Cajas de Ahorros.

El entorno del cual partiremos para realizar la prueba será el formulario de entrada de la aplicación.

2. Registrar cajas de ahorros

2.1. Descripción

Ingresar al sistema con el usuario que tenga permisos para registrar cajas de ahorros.

2.2. Condiciones de Ejecución

El Super Usuario Administrador debe estar ingresado y autenticado en el sistema.

2.3. Entrada

- Introducir el usuario en el campo usuario
- Introducir la contraseña en el campo contraseña
- Pulsar el botón ingresar de la aplicación.
- Se muestra el menú con las opciones que puede realizar el usuario actual que se encuentra autenticado en el sistema.
- Seleccionar del menú administración escoger la opción administración de cajas de ahorros
- Escoger la opción Nuevo aparece la interfaz para registrar a la nueva caja de ahorros.
- El usuario ingresa el nombre de la caja de ahorros
- El usuario ingresa el número de socios para la caja de ahorros.
- Después de ingresar los datos de la caja de ahorros presionar el botón guardar.
- El sistema muestra el mensaje indicando que la caja de ahorros se guardo correctamente.

2.4. Resultado esperado

El sistema registra la caja de ahorros.

2.5. Evaluación de la prueba

Prueba superada con éxito

3.8.5.Caso de prueba registrar transacción caja de ahorro

1. Descripción

Este artefacto cubre el conjunto de pruebas realizadas sobre el caso de uso “Transacción Caja de Ahorro”

Las pruebas realizadas en este caso son:

- Registrar transacción caja de ahorro.
- Registrar detalles transacción caja de ahorros.
- Ver detalle transacción caja de ahorros.

El entorno del cual partiremos para realizar la prueba será el formulario de entrada de la aplicación.

2. Registrar transacción caja de ahorros

2.1. Descripción

Ingresar al sistema con el usuario que tenga permisos para registrar la transacción caja de ahorro.

2.2. Condiciones de Ejecución

El usuario secretario debe estar ingresado y autenticado en el sistema.

2.3. Entrada

- Introducir el usuario en el campo usuario
- Introducir la contraseña en el campo contraseña
- Pulsar el botón ingresar de la aplicación.
- Se muestra el menú con las opciones que puede realizar el usuario actual que se encuentra autenticado en el sistema.
- Seleccionar del menú transacciones escoger la opción transacción caja de ahorro.
- Escoger la opción Nuevo aparecerá la interfaz para registrar la nueva transacción caja de ahorro.
- El usuario selecciona la fecha para la nueva transacción caja de ahorro.
- El usuario presiona el botón guardar.
- El sistema muestra el mensaje indicando que la nueva transacción caja de ahorro se guardo correctamente.

2.4. Resultado esperado

El sistema registra la transacción correctamente.

2.5. Evaluación de la prueba

Prueba superada con éxito

3. Registrar detalle transacción caja de ahorro

3.1. Descripción

Ingresar al sistema con el usuario que tenga permisos para registrar la transacción caja de ahorro.

3.2. Condiciones de ejecución

El usuario secretario debe estar ingresado y autenticado en el sistema.

3.3. Entrada

- Introducir el usuario y la contraseña en sus respectivos campos
- Pulsar el botón ingresar de la aplicación.
- Se muestra el menú con las opciones que puede realizar el usuario actual que se encuentra autenticado en el sistema.
- Seleccionar del menú transacciones escoger la opción transacción caja de ahorro.
- Seleccionar en la lista de transacciones caja de ahorro la transacción actual para registrar los detalles transacción caja de ahorro.
- Aparecerá el detalle de la transacción caja de ahorro el mismo que aparece como el listado de los socios.
- El usuario podrá registrar la asistencia de cada socio seleccionando el check de asistencia que aparece en cada socio.
- En cada detalle se podrá realizar ahorro, ahorro de fondo de reserva, pago de créditos, pago de interés, pago de acciones.
- El usuario selecciona a un socio presionando editar sobre la fila correspondiente a cada socio.
- El sistema muestra la fila seleccionada como habilitada para ingresar el ahorro, ahorro fondo reserva, pago interés, pago acciones, pago crédito.
- El usuario selecciona en la fila seleccionada el botón guardar para registrar el detalle transacción caja de ahorro.
- El sistema muestra un mensaje indicando que el registro de la transacción caja de ahorro se guardó correctamente.

3.4. Resultado esperado

El sistema registra el detalle transacción caja de ahorro correctamente.

3.5. Evaluación de la prueba

Prueba superada con éxito

4. Ver detalle transacción caja de ahorro

4.1. Descripción

Ingresar al sistema con el usuario que tenga permisos para registrar la transacción caja de ahorro.

4.2. Condiciones de ejecución

El usuario secretario debe estar ingresado y autenticado en el sistema.

4.3. Entrada

- Introducir el usuario y la contraseña en sus respectivos campos
- Pulsar el botón ingresar de la aplicación.
- Se muestra el menú con las opciones que puede realizar el usuario actual que se encuentra autenticado en el sistema.
- Seleccionar del menú transacciones escoger la opción transacción caja de ahorro.
- Seleccionar en la lista de transacciones caja de ahorro la transacción actual para registrar los detalles transacción caja de ahorro.
- Aparecerá el detalle de la transacción caja de ahorro el mismo que aparece como el listado de los socios.
- En cada detalle se podrá seleccionar el número de cuenta del socio.
- El usuario selecciona a un número de cuenta presionando sobre la fila correspondiente a cada socio.
- El sistema mostrara el estado de la cuenta correspondiente a ahorros, fondos de reserva, créditos.

4.4. Resultado esperado

El sistema muestra el detalle transacción caja de ahorro correctamente.

4.5. Evaluación de la prueba

Prueba superada con éxito.

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES



4. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

- Con la implementación del sistema de caja de ahorro comunitario se alcanzó los objetivos esperados debido a que el sistema cumple los requerimientos especificados en las transacciones generadas por los socios como son el ahorro en sus cuentas, el ahorro en sus fondos de reserva, la obtención de créditos etc.
- El sistema de caja de ahorro comunitario está desarrollado de forma genérica, para poder ser implantado en cualquier caja de ahorro de la provincia de Imbabura ya que los procesos que se pudo investigar en otras parroquias son similares a los implantados en el sistema.
- La utilización e investigación de herramientas de software libre hace que podamos aplicar las tecnologías más actuales sin restricción alguna para la elaboración de proyectos que beneficien a un conglomerado de personas.
- Se utilizó y se puso en práctica los conocimientos que se adquiere en la Universidad para dar solución a la problemática de no contar en tiempos actuales con un sistema que automatice los procesos de las cajas de ahorros en la parroquia de Ambuquí.
- La investigación de nuevas tecnologías hace que podamos comparar herramientas que ya hemos utilizado anteriormente, teniendo la posibilidad de estar en capacidad de hacer comparaciones entre cuales son más aplicables en proyectos como el que elaboramos, llegando a concluir que la utilización de JSF, EJBS, JPA hace que sea más fácil la elaboración y desarrollo en comparación con otras herramientas de desarrollo.
- La implementación del sistema informático se pudo apreciar una reducción considerable del tiempo que se emplea en los servicios que presta la caja de ahorros en la atención a cada uno de sus socios.
- La implementación del sistema de caja de ahorro comunitario da cabalidad para que todos salgamos beneficiados, la Universidad cumpliendo su objetivo que es formar profesionales altamente capacitados con capacidad de dar soluciones a los problemas y necesidades de la colectividad, el estudiante que pone en práctica lo que investiga y los conocimientos que adquiere en las aulas para tener una experiencia excepcional para el

desenvolvimiento en la vida profesional y la sociedad que se sirve de todos estos elementos para beneficio de un gran parte de sus integrantes pudiendo evidenciar el crecimiento social y económico.

4.2. RECOMENDACIONES

Para el correcto funcionamiento del sistema de caja de ahorro comunitario es recomendable:

- Se recomienda la capacitación a los usuarios que van a utilizar el sistema ya que esto ayudará a utilizar con eficiencia y eficacia el sistema.
- La documentación del sistema de caja de ahorro comunitario es de vital importancia y se debe estar en constantes actualizaciones, con el único propósito de dar un mejor soporte a los usuarios.
- Se recomienda dar un seguimiento del proyecto para que se implementen módulos que permitirán dar un mayor alcance al proyecto.
- Se recomienda el análisis y estudio de metodologías ágiles para la elaboración de proyectos empresariales.
- Con la utilización de herramientas de software libre pudimos comprobar que son totalmente eficientes para el desarrollo de proyectos empresariales por lo que se recomienda seguir utilizando y estudiando herramientas de software libre.

4.3. BIBLIOGRAFÍA

- Alférez Sánchez, J. (25 de Octubre de 2010). *INSTALACIÓN, CONFIGURACIÓN Y ADMINISTRADOR DE APLICACIONES JBOSS*. Obtenido de <http://www.alferez.es/documentos/Jboss.pdf>
- Dosideas. (2 de Septiembre de 2011). *Dos ideas*. Obtenido de <http://www.dosideas.com/cursos/mod/resource/view.php?id=10>
- Lizárraga Celaya, C., & Díaz Martínez, S. L. (1 de Octubre de 2007). *USO DE SOFTWARE LIBRE Y DE INTERNET COMO HERRAMIENTAS DE APOYO PARA EL APRENDIZAJE*. Obtenido de <http://e-spacio.uned.es/fez/eserv.php?pid=bibliuned:20449&dsID=usosoftwarelibre.pdf>
- Lopez Franco, J. M. (15 de Octubre de 2001). *Servidor de Aplicaciones*. Obtenido de <http://trevinca.ei.uvigo.es/~txapi/espanol/proyecto/superior/memoria/node21.html>
- Mejia Alvarez, P. (6 de Noviembre de 2002). *Rational Unified Process*. Obtenido de <http://delta.cs.cinvestav.mx/~pmejia/softeng/rup.ppt>
- Meroño Sánchez, J. J. (3 de Octubre de 2009). *Curso Java y Tecnologías JEE*. Obtenido de <http://www.bibliocomunidad.com/web/libros/CursoJavayJ2EE.pdf>
- Pecos Martínez, D. (24 de Noviembre de 2009). *Comparación Posgresql y MySQL*. Obtenido de http://www.danielpecos.com/docs/mysql_postgres/x15.html
- Plauchu, E. (7 de Septiembre de 2011). *Ciclo de vida del procesamiento de una solicitud en JSF2*. Obtenido de <http://gulsin.org/2011/09/07/ciclo-de-vida-del-procesamiento-de-una-solicitud-en-java-server-faces-20/>
- Keith and Schincariol ,Apress.,(2006), *Pro EJB 3 Java Persistence API*. Obtenido de <http://www.apress.com>.
- Hennebrueder,SH.,(2006), *First EJB 3 Tutorial*.Obtenido de <http://www.laliluna.de/download/first-ejb3-tutorial-en.pdf>
- Mena,MM.,(2009) *CREACIÓN DE CAJA DE AHORRO Y CRÉDITO PARA LOS PRODUCTORES DEL NOROCCIDENTE DE LA PROVINCIA DE PICHINCHA*. Obtenido de <http://repositorio.utpl.edu.ec/bitstream/123456789/3462/1/330X1266.pdf>

- JBoss, by Red Hat. (2011).JBoss, by Red Hat .Obtenido de <http://www.jboss.com>
- Grupo Investigación Sicuma.SICUMA(2011) Tutorial de Java Server Faces. Obtenido de <http://www.sicuma.uma.es/sicuma/Formacion/documentacion/JSF.pdf>
- PostgreSQL Global Development Group. (1996 – 2011) PostgreSQL. Obtenido de <http://www.postgresql.org>
- Sistema Operativo GNU. Obtenido de <http://www.gnu.org/philosophy/free-sw.es.html>
- Ecuador software libre , Free Software.Obtenido de http://www.softwarelibre.ec/site/index.php?option=com_content&view=article&id=112&Itemid=141

4.4. GLOSARIO DE TÉRMINOS

- **CICLO:** Es un periodo con fecha inicial y fecha final para las cajas de ahorros.
- **ACCIÓN:** Parte proporcional del capital social de una corporación. Esto significa que al comprar acciones, usted se convierte en accionista o propietario de una parte de una corporación
- **RUP:** Son las siglas de Rational Unified Process. Se trata de una metodología para describir el proceso de desarrollo de software
- **UML:** Por sus siglas en inglés, Unified Modeling Language - Lenguaje Unificado de Modelado- es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software
- **SOCIO:** Es la persona que invierte activamente dinero en la caja de ahorros es decir ingresa ahorros en la caja de ahorros semanalmente.
- **CAJA DE AHORRO:** Una caja de ahorros es una entidad de ahorro y crédito similar a un banco, se diferencia por su carácter legislativo con el banco, se rigen por la ley de sociedades limitadas de carácter fundacional mientras un banco son sociedades anónimas, razón por la cual tienen que destinar parte de sus utilidades a fines sociales y culturales
- **CRÉDITO:** El crédito es una operación financiera en la que se pone a disposición de los socios una cantidad de dinero hasta un límite especificado y durante un período de tiempo determinado.
- **GARANTIA:** Una garantía es un negocio jurídico mediante el cual se pretende dotar de una mayor seguridad al cumplimiento de una obligación o pago de una deuda.
- **JEE:** Es una plataforma de programación para desarrollar y ejecutar software de aplicaciones en Lenguaje de programación Java con arquitectura de n capas distribuidas y que se apoya ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones. La plataforma Java EE está definida por una especificación. Similar a otras especificaciones del Java Community Process.

4.5. ANEXOS

4.5.1. MANUAL DE INSTALACIÓN

4.5.1.1. PREREQUISITOS

- **Hardware**

Equipo Dual Core

4 Gb en RAM

100 Gb en HD

- **Sistema Operativo**

Sistema

Windows, Linux

- **Servidor de Aplicaciones**

JBoss

jboss-6.0.GA

- **Base de datos**

PostgreSQL

PostgreSQL 9.1

- **Java Developer Kit**

jdk-6u20-windows-i586

4.5.1.2. INSTALACIÓN DE HERRAMIENTAS

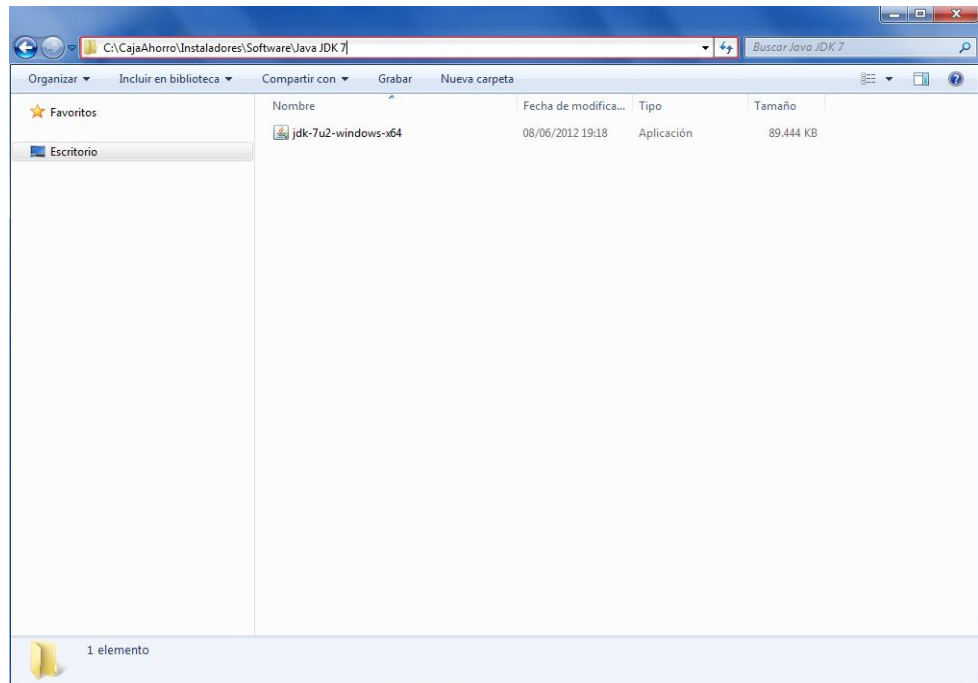
1. *Instalación de JAVA*

1.1. *Instalación de JDK Java Developer Kit*

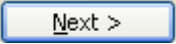
Paso 1:

Primero ubique el directorio donde se encuentra el programa de instalación que se encuentra en la carpeta de instaladores de caja de ahorro:

C:\CajaAhorro\Instaladores\Software\Java JDK 7\jdk-7u2-windows-x64

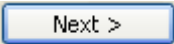


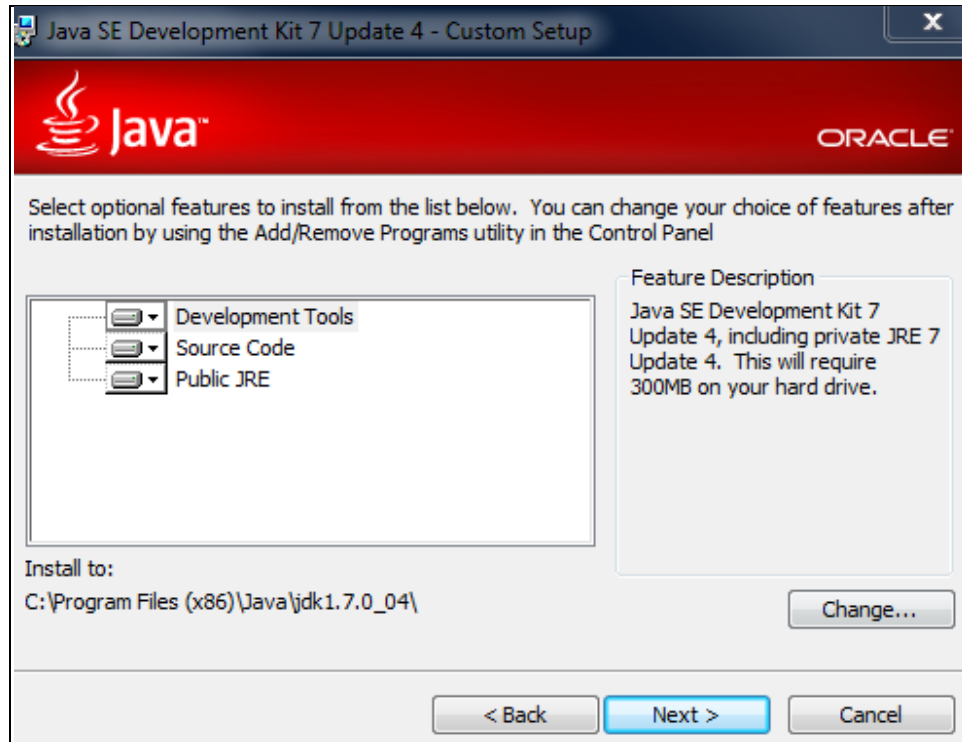
Paso 2:

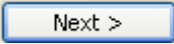
Luego de ubicar el software, ejecute el programa, esperar que inicie el proceso de instalación y en la ventana pulsar el botón .

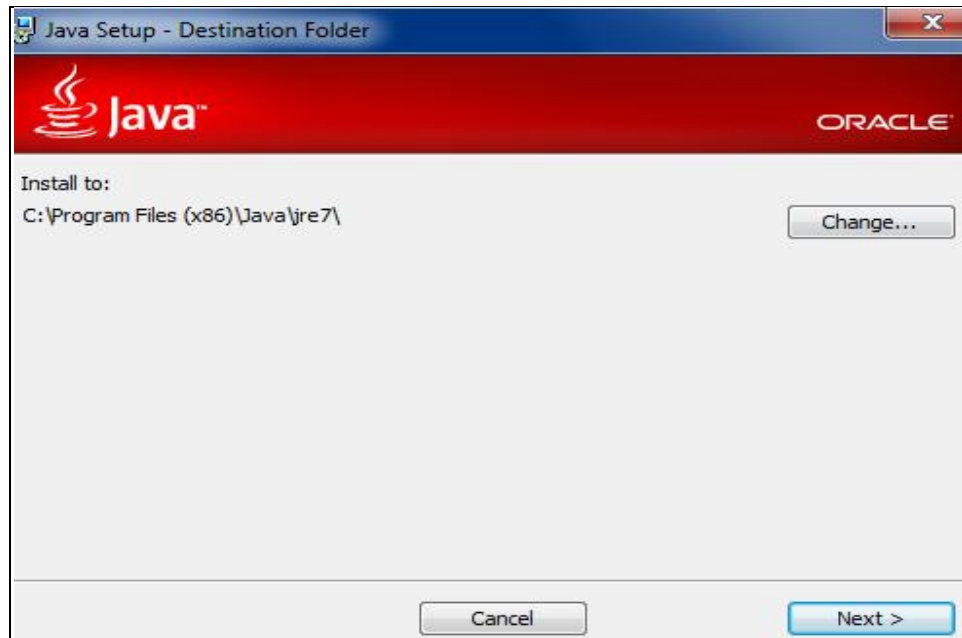


Paso 3:

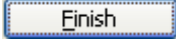
En la siguiente ventana puede ver las opciones Development Tools, Demos, Source Code, Public JRE que son las herramientas de desarrollo que se van a instalar, clic en el botón 

**Paso 4:**

En la siguiente ventana se indica que se va a instalar la opción jre7 Java Runtime Environment, clic en el botón 



Paso 5:

En la siguiente ventana clic en el botón  y la instalación de Java Developer Kit a terminado correctamente.

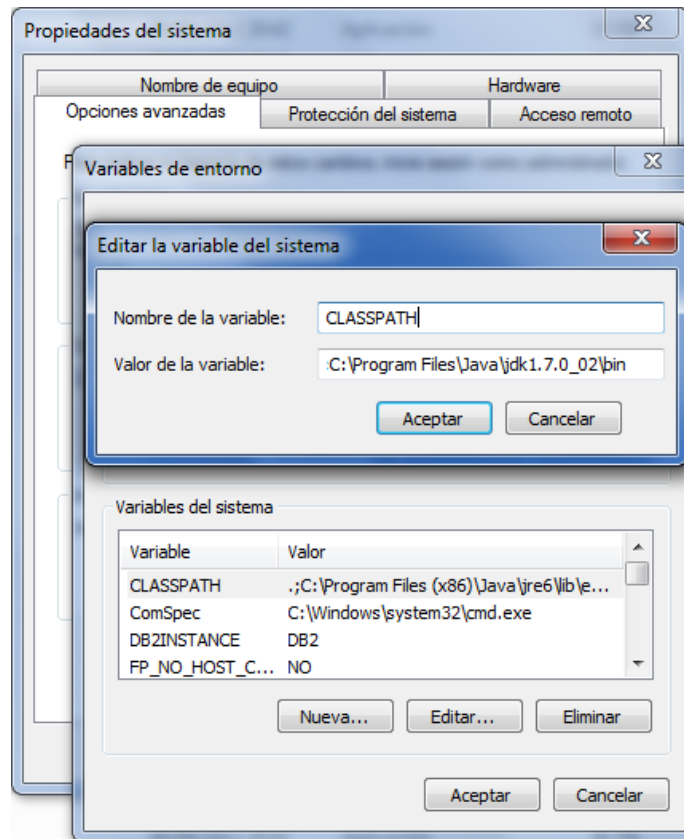


2. Configuración de las variables de entorno CLASSPATH, PATH, JAVA_HOME

2.1. CLASSPATH

Java utiliza una variable de entorno denominada **CLASSPATH**, la cual determina dónde buscar tanto las clases o librerías de Java (el API de Java) como otras clases de usuario.

Debe tener creada la variable de entorno **CLASSPATH** con el siguiente valor:
C:\Program Files\Java\jdk1.7.0_02\bin;

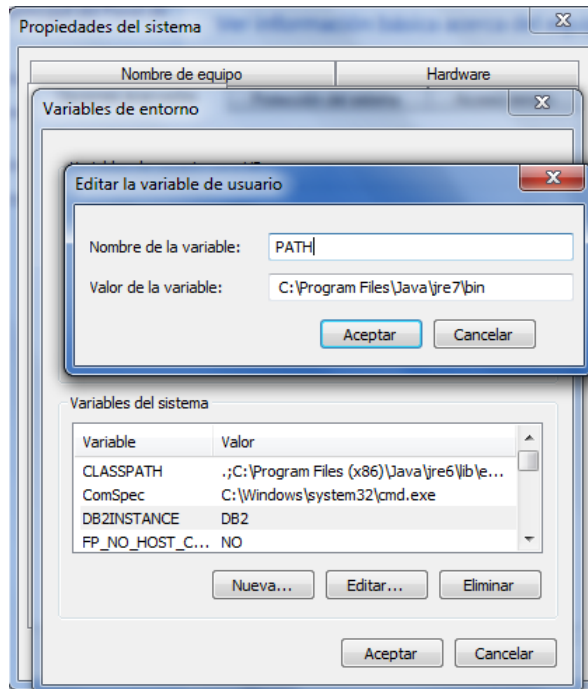


2.2. PATH

PATH es una variable que se utiliza para que se puedan correr los programas ejecutables desde cualquier directorio del disco duro.

Debe tener creada la variable de entorno PATH con el siguiente valor:

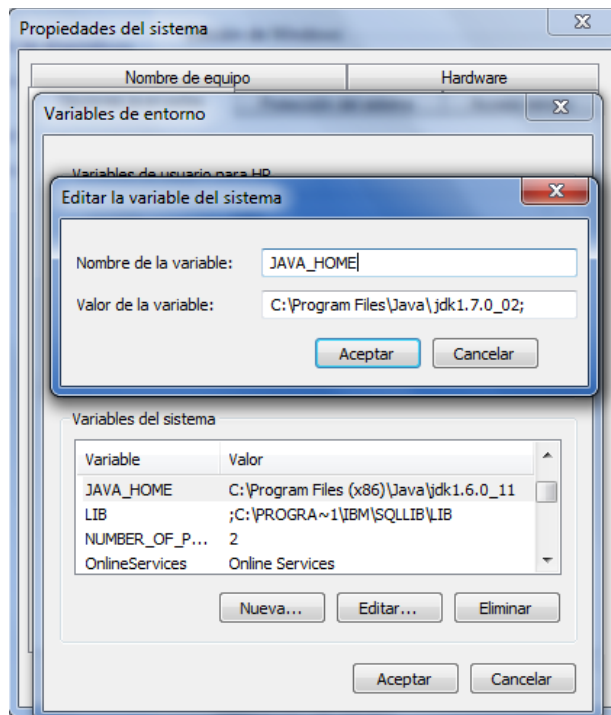
C:\Program Files\Java\jre7\bin;



2.3. JAVA HOME

Debe tener creada la variable de entorno JAVA_HOME con el siguiente valor:

C:\Program Files\Java\jdk1.7.0_02;



3. Instalación del servidor de aplicaciones jboss

3.1. Pre – Requisitos

Debe tener instalado el entorno Java Developer Kit de JDK 1.7 antes de instalar JBoss Application Server.

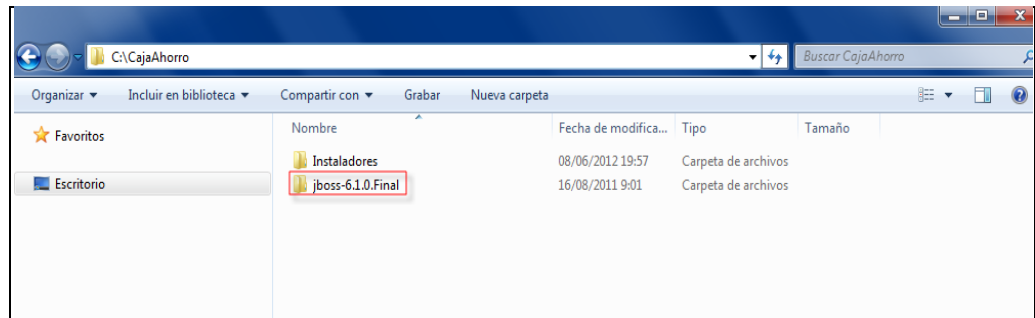
Debe tener creada la variable de entorno llamada JAVA_HOME que apunta al directorio de instalación de JDK, por ejemplo:

C:\ProgramFiles(x86)\Java\jdk1.7.0_04.

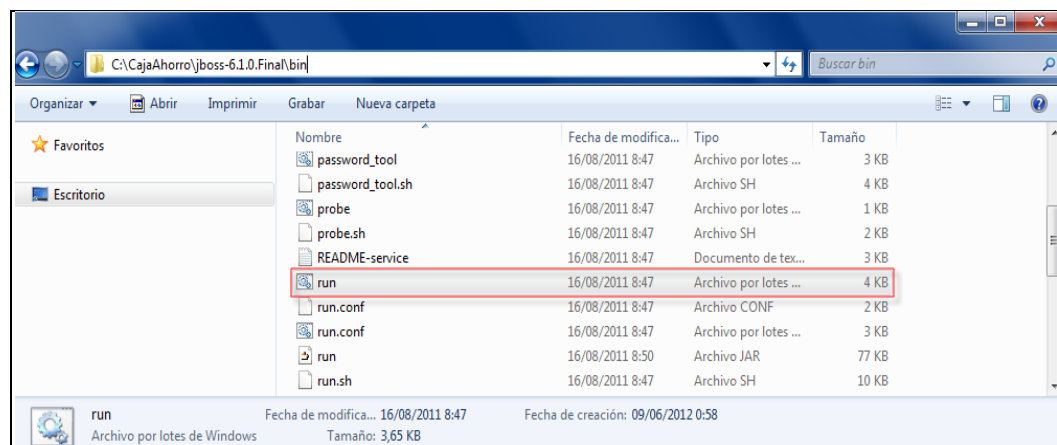
3.2. Instalación JBoss

Simplemente descomprimir el archivo *jboss-as-distribution-6.1.0.Final.zip* que se encuentra en la carpeta de instaladores de CajaAhorro.

C:\CajaAhorro\Instaladores\Software\Servidores\JBoss\jboss-as-distribution-6.1.0.Final.zip en el directorio C:\CajaAhorro



Comprobar que la instalación se ha realizado correctamente. Para ello ubicar el archivo *run.bat* que se encuentra dentro del directorio **C:\CajaAhorro\jboss-6.1.0.Final\bin**

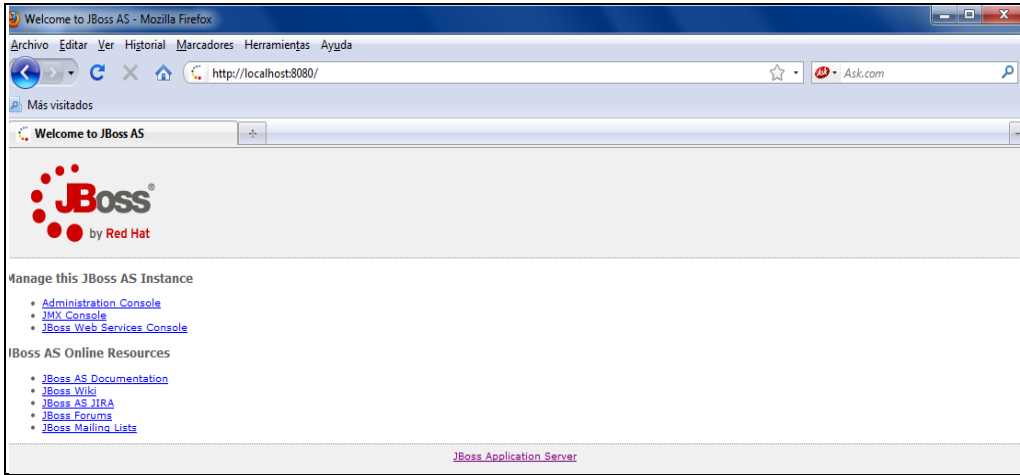


Ejecutar el archivo **run.bat** y aparece la consola de DOS y esperar que se suba el servidor de aplicaciones...

```

C:\WINDOWS\system32\cmd.exe
12:26:18,857 INFO [TomcatDeployer] deploy, ctxPath=/jbossus, warUrl=../deploy/jbossus.sar/jbossus-context.war/
12:26:19,224 INFO [TomcatDeployer] deploy, ctxPath=/jbossmq-httpil, warUrl=../deploy/jms/jbossmq-httpil.sar/jbossmq-httpil.war/
12:26:22,884 INFO [TomcatDeployer] deploy, ctxPath=/web-console, warUrl=../deploy/management/console-mgr.sar/web-console.war/
12:26:24,489 INFO [MailService] Mail Service bound to java:Mail
12:26:24,989 INFO [RARDeployment] Required license terms exist, view META-INF/ra.xml in ../deploy/jboss-ha-local-jdbc.rar
12:26:25,098 INFO [RARDeployment] Required license terms exist, view META-INF/ra.xml in ../deploy/jboss-ha-xa-jdbc.rar
12:26:25,176 INFO [RARDeployment] Required license terms exist, view META-INF/ra.xml in ../deploy/jboss-local-jdbc.rar
12:26:25,285 INFO [RARDeployment] Required license terms exist, view META-INF/ra.xml in ../deploy/jboss-xa-jdbc.rar
12:26:25,488 INFO [RARDeployment] Required license terms exist, view META-INF/ra.xml in ../deploy/jms/jms-ra.rar
12:26:25,582 INFO [RARDeployment] Required license terms exist, view META-INF/ra.xml in ../deploy/mail-ra.rar
12:26:25,691 INFO [RARDeployment] Required license terms exist, view META-INF/ra.xml in ../deploy/quartz-ra.rar
12:26:25,722 INFO [QuartzResourceAdapter] start quartz!!!
12:26:25,878 INFO [SimpleThreadPool] Job execution threads will use class loader of thread: main
12:26:25,925 INFO [QuartzScheduler] Quartz Scheduler v.1.5.2 created.
12:26:25,925 INFO [RAMJobStore] RAMJobStore initialized.
12:26:25,925 INFO [StdSchedulerFactory] Quartz scheduler 'DefaultQuartzScheduler' initialized from default resource file in Quartz pac
12:26:25,925 INFO [StdSchedulerFactory] Quartz scheduler version: 1.5.2
12:26:25,925 INFO [QuartzScheduler] Scheduler DefaultQuartzScheduler_$_NON_CLUSTERED started.
12:26:27,360 INFO [ConnectionFactoryBindingService] Bound ConnectionManager 'jboss.jca:service=DataSourceBinding,name=DefaultDS' to JN
12:26:28,374 INFO [A] Bound to JNDI name: queue/A
12:26:28,374 INFO [B] Bound to JNDI name: queue/B
12:26:28,390 INFO [C] Bound to JNDI name: queue/C
12:26:28,390 INFO [D] Bound to JNDI name: queue/D
12:26:28,390 INFO [E] Bound to JNDI name: queue/E
12:26:28,421 INFO [testTopic] Bound to JNDI name: topic/testTopic
12:26:28,437 INFO [securedTopic] Bound to JNDI name: topic/securedTopic
12:26:28,437 INFO [testDurableTopic] Bound to JNDI name: topic/testDurableTopic
12:26:28,452 INFO [testQueue] Bound to JNDI name: queue/testQueue
12:26:28,546 INFO [UIServerInfoService] JBossMQ UII service available at : /127.0.0.1:8093
12:26:28,688 INFO [IDLQ] Bound to JNDI name: queue/IDLQ
12:26:28,936 INFO [ConnectionFactoryBindingService] Bound ConnectionManager 'jboss.jca:service=ConnectionFactoryBinding,name=JmsXA' to
12:26:29,232 INFO [TomcatDeployer] deploy, ctxPath=/jmx-console, warUrl=../deploy/jmx-console.war/
12:26:30,223 INFO [HttpProtocol] Arrancando Coyote HTTP/1.1 en puerto http-127.0.0.1-8080
12:26:30,371 INFO [ajpProtocol] Starting Coyote AJP/1.3 on ajp-127.0.0.1-8089
12:26:30,418 INFO [Server] JBoss (MX MicroKernel) [4.2.2.GA (build: SUNJag-JBoss_4_2_2_GA date=200710221139)] Started in 52s:641ms
    
```

Ahora para probar introducir en el navegador la dirección <http://localhost:8080/>. Deberá obtenerse la página de inicio de JBoss.

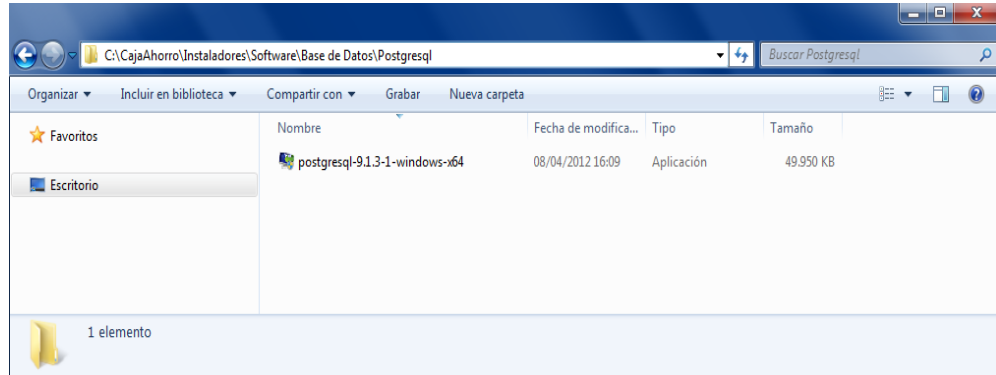


4. Instalación del servidor de base de datos postgresql

Paso 1:

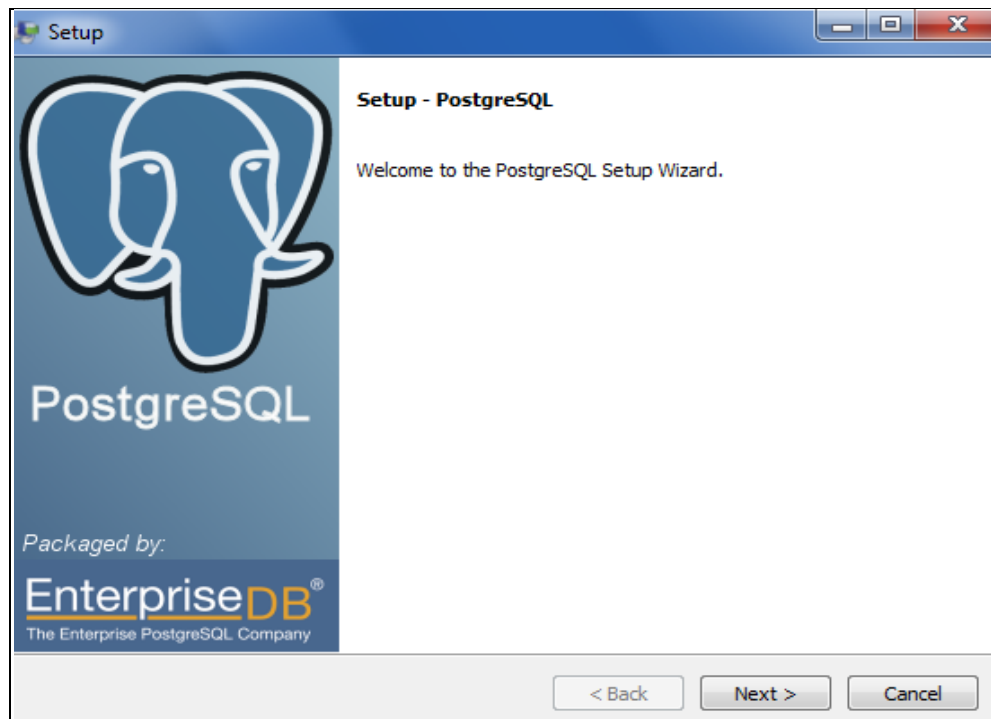
Primero ir al directorio donde se encuentra el programa de instalación que se encuentra en la carpeta de instaladores de Caja Ahorro:

C:\CajaAhorro\Instaladores\Software\Base de Datos\Postgresql\ postgresql-9.1.3-1-windows-x64

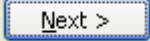


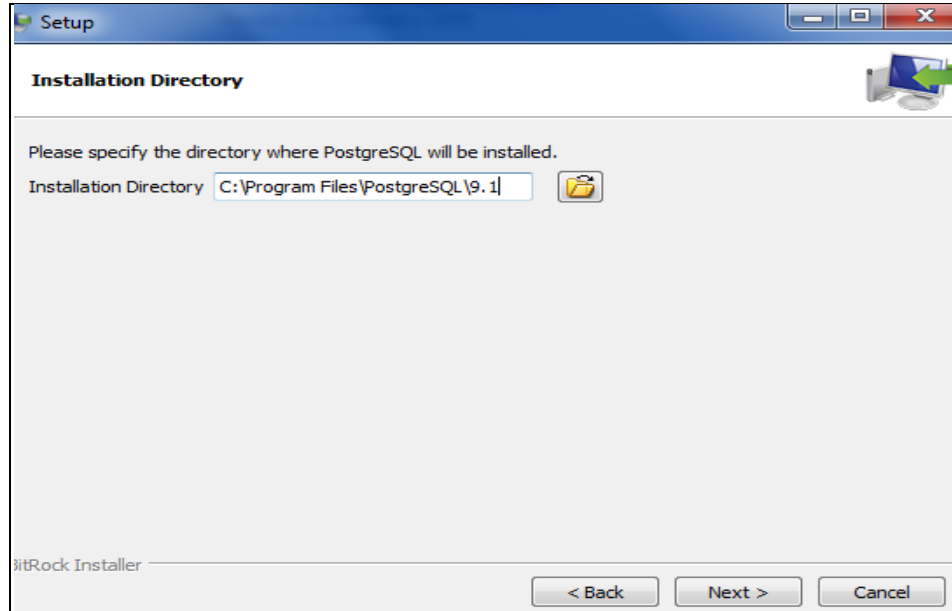
Paso 2:


Luego de ubicar el software, ejecute el programa, inicia el proceso de instalación y aparece la ventana de bienvenida:

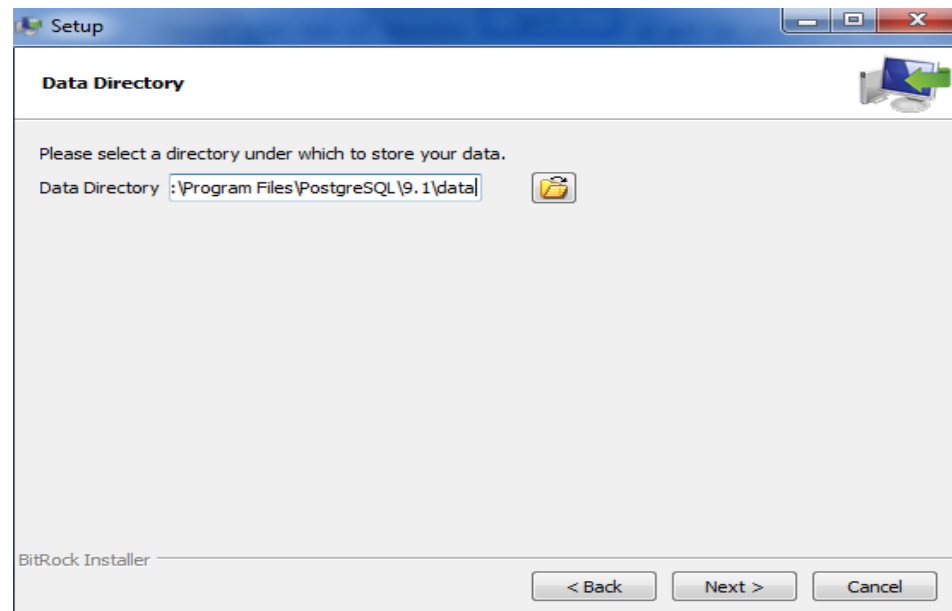


Paso 3:

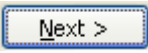
Hacer clic en el botón  y en la ventana que aparece el directorio donde se instalará postgresql.

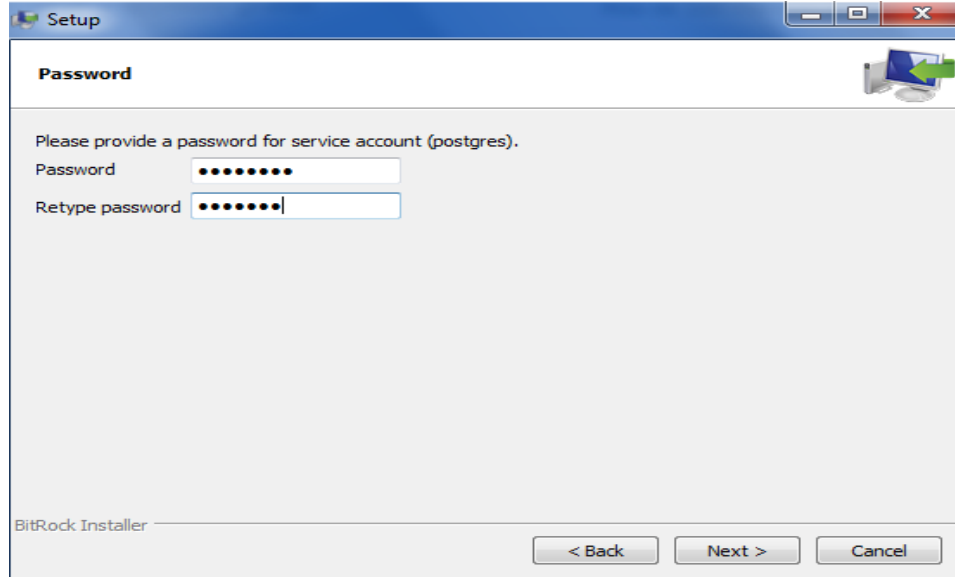
**Paso 4:**

Hacer clic en el botón  y en la ventana que aparece el directorio donde se almacenarán sus datos.



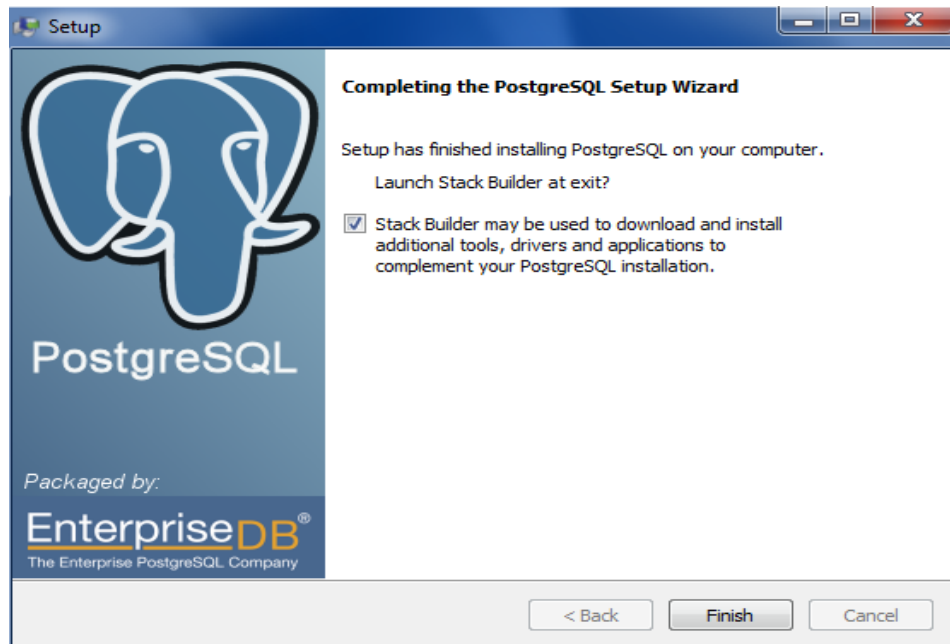
Paso 5:

Hacer clic en el botón  y en la ventana pedirá proporcionar una contraseña para la cuenta del servicio, le asignamos Postgres como contraseña.



Paso 6:

Luego hacer clic en el botón  con lo cual la instalación está completa

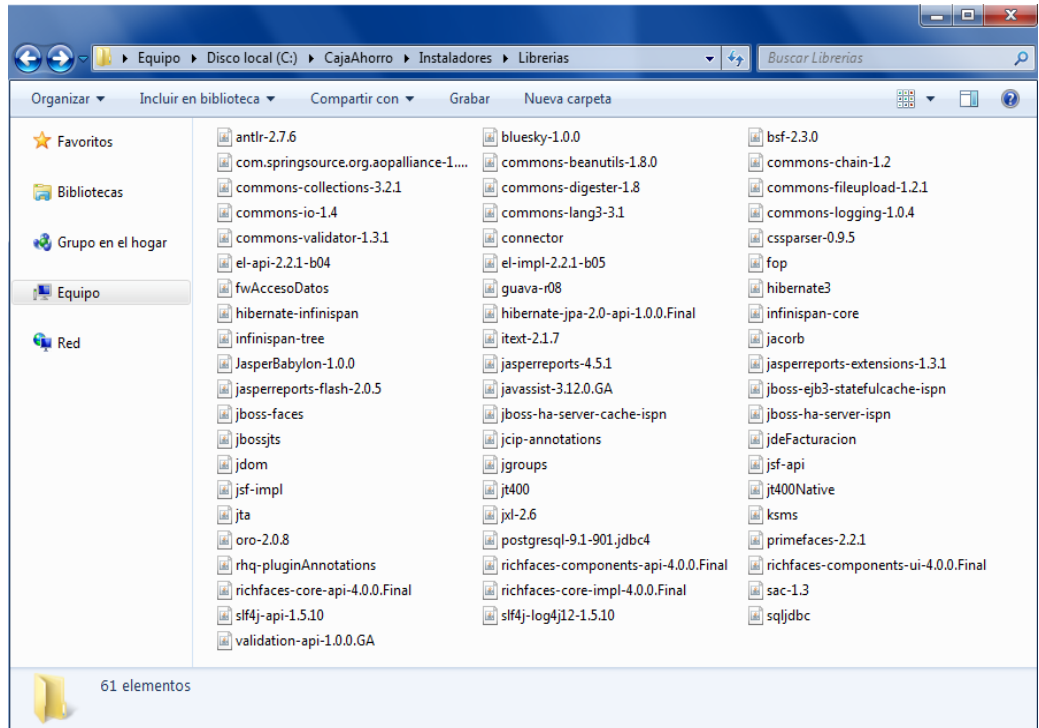


5. Librerías

Las librerías que necesita el sistema de caja de ahorro son:

Los archivos se encuentran en la carpeta librerías que está en el directorio:

C:\CajaAhorro\Instaladores\Librerías

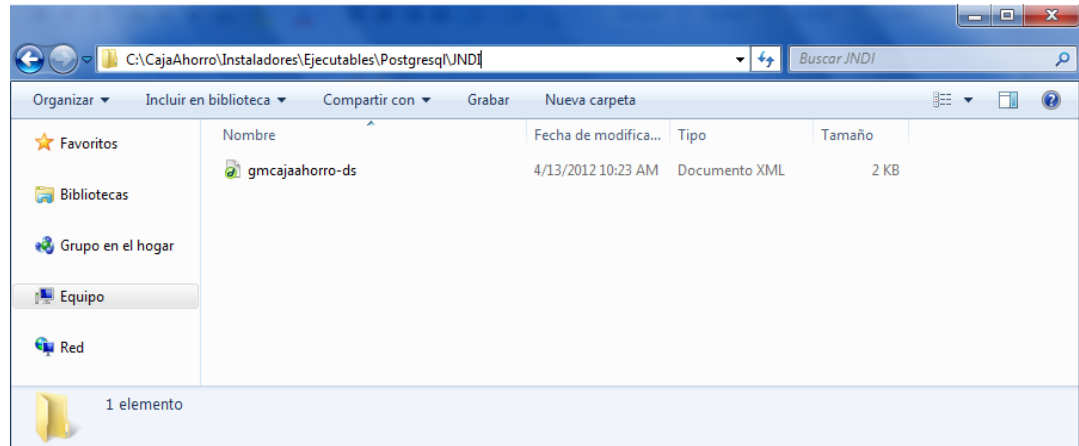


Estos archivos se deben copiar en el directorio de librerías del servidor de aplicaciones, para esto copiar los archivos en el directorio lib del servidor que se encuentra en la dirección **C:\CajaAhorro\jboss-6.1.0.Final\bin**

6. Configuración JNDI

La configuración para la conexión a la base de datos gmcajaahorro se realizar mediante el archivo xml donde se encuentra la información para la conexión a la base de datos. El archivo gmcajaahorro-ds.xml es el que contiene la información para realizar la conexión, que se encuentra en el directorio.

C:\CajaAhorro\Instaladores\Ejecutables\Postgresql\JNDI\gmcajaahorro-ds.xml



Abrir el archivo *gmcajaahorro-ds.xml* y modificar los siguientes puntos:

1.-localhost:5432

localhost Dirección donde está instalado la base de datos

5432 Puerto que utiliza el servidor de base de datos Postgresql

2.- postgres

Usuario para acceder a la base de datos gmcajaahorro

3.- Postgres

Se refiere al password del usuario para acceder a la base de datos.

```

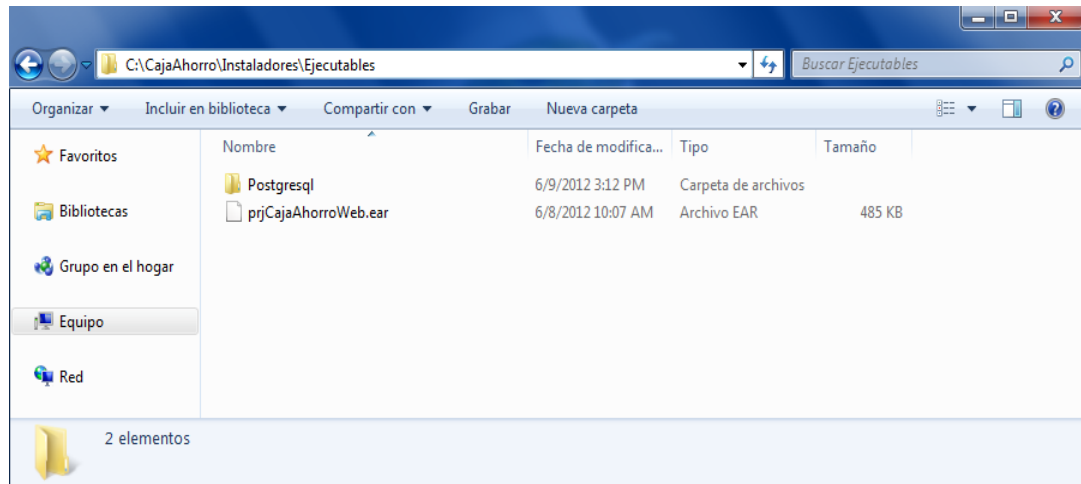
1 <?xml version="1.0" encoding="UTF-8"?>
2 <datasources>
3   <local-tx-datasource>
4     <jndi-name>caja_ahorro</jndi-name>
5     <connection-url>jdbc:postgresql://localhost:5432/gmcajaahorro</connection-url>
6     <driver-class>org.postgresql.Driver</driver-class>
7     <user-name>postgres</user-name>
8     <password>postgres</password>
9     <metadata>
10      <type-mapping>PostgreSQL 7.2</type-mapping>
11    </metadata>
12  </local-tx-datasource>
13 </datasources>
14
15

```

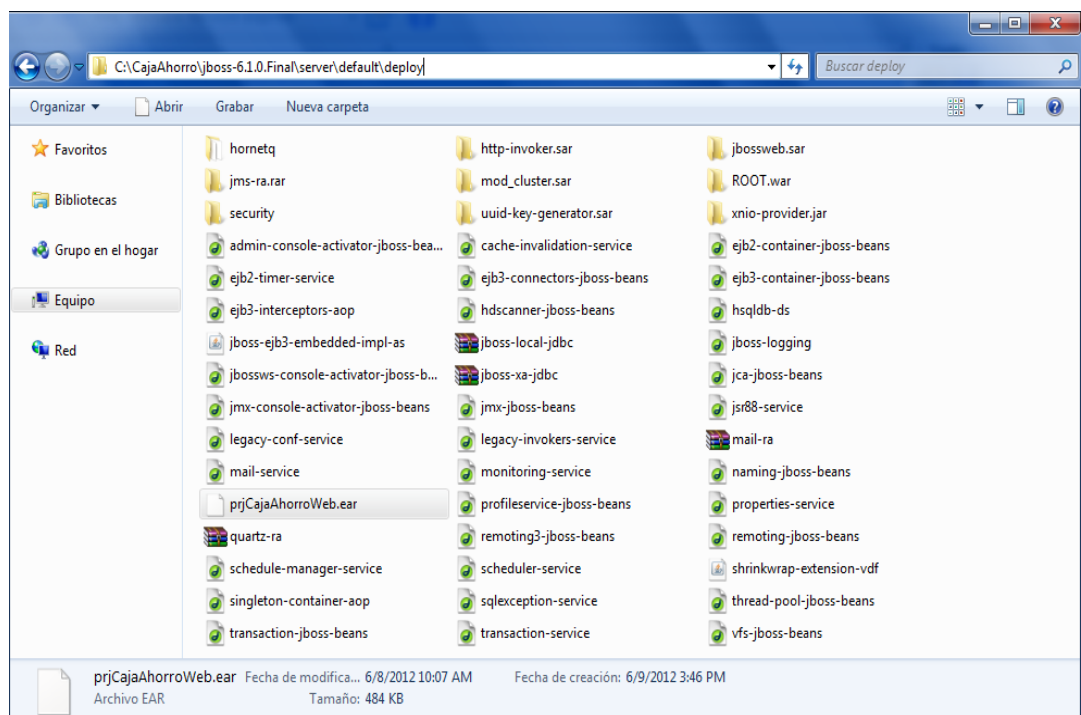
Copiar el archivo de configuración *gmcajaahorro-ds.xml* en el servidor de aplicaciones en la dirección *C:\CajaAhorro\jboss-6.1.0.Final\server\default\deploy*

7. Instalación de la aplicación

Para la instalación de la aplicación Caja de Ahorro ubicar el archivo `prjCajaAhorroWebEAR.ear` que se encuentra en la directorio `C:\CajaAhorro\Instaladores\Ejecutables\prjCajaAhorroWeb.ear`



Copiar el archivo `prjCajaAhorroWeb.ear` en el servidor de aplicaciones en la dirección `C:\CajaAhorro\jboss-6.1.0.Final\server\default\deploy`



Ejecutar el archivo `run.bat` que se encuentra dentro del directorio `C:\CajaAhorro\jboss-6.1.0.Final\bin` para iniciar el servidor de aplicaciones, y esperar hasta que inicie.

Abrir un navegador y en la barra de dirección escribir **http://localhost:8080/prjCajaAhorroWeb** y aparece la pantalla de bienvenida para ingresar al sistema de caja de ahorro comunitario

4.5.2.MANUAL DE USUARIO

El presente manual está realizado de tal manera que el usuario identifique fácilmente los funcionamientos del sistema Caja de Ahorro Comunitario y pueda manipularlo sin ningún problema.

1. Sumario

El sistema caja de ahorro comunitario es un sistema creado con el propósito de automatizar, controlar y presentar de mejor manera la información.

2. Guía de utilización

El presente manual está realizado de manera secuencial, indicando los pasos que debe seguir el usuario para el correcto funcionamiento del sistema.

2.1. Requerimientos

Para una correcta visualización de las pantallas que se presentan en el sistema es recomendable la utilización de los siguientes navegadores Web:

- **Internet Explorer 9**
- **Mozilla Firefox 5.0**

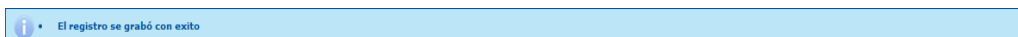
2.2. Aspectos Generales

El sistema para indicar los mensajes que se producen al realizar algún proceso mostrará los siguientes mensajes en parte inferior del las pantallas.

Mensaje de error

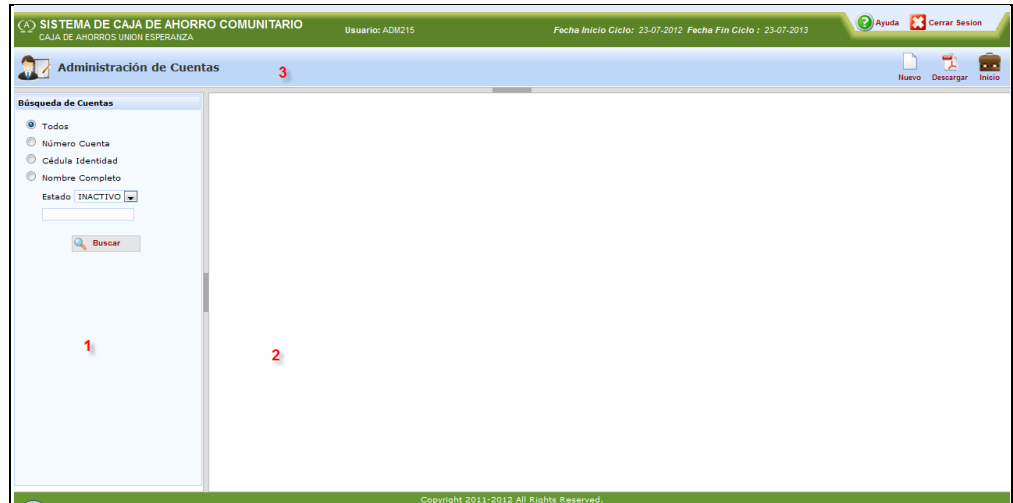


Mensaje de éxito



2.3. Sección de las pantallas del sistema de caja de ahorro

Las pantallas del sistema son estructuradas de las siguientes secciones:



1. Sección de de búsqueda

Esta sección permite realizar búsquedas de acuerdo a las opciones de búsquedas seleccionadas.

2. Sección de resultado de búsqueda

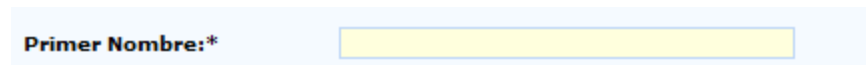
Esta sección permite mostrar los resultados de la búsqueda realizada.

3. Sección de barra de títulos

Permite mostrar en la barra el título de la pantalla a la que se refiere y los botones que permitirán realizar diferentes acciones.

2.4. Campos requeridos

Los campos que son requeridos u obligatorios en el sistema estarán identificados con el color amarillo y con un asterisco al final de cada campo como se muestra en la siguiente figura.



3. Ingreso al sistema

Para ingresar al sistema el usuario deberá ingresar el usuario y el password en la pantalla inicial del sistema.

El sistema validará si los datos son correctos, si los datos ingresados son correctos el usuario ingresará al sistema caso contrario se mostrará un mensaje indicando las razones porque no puede ingresar al sistema.

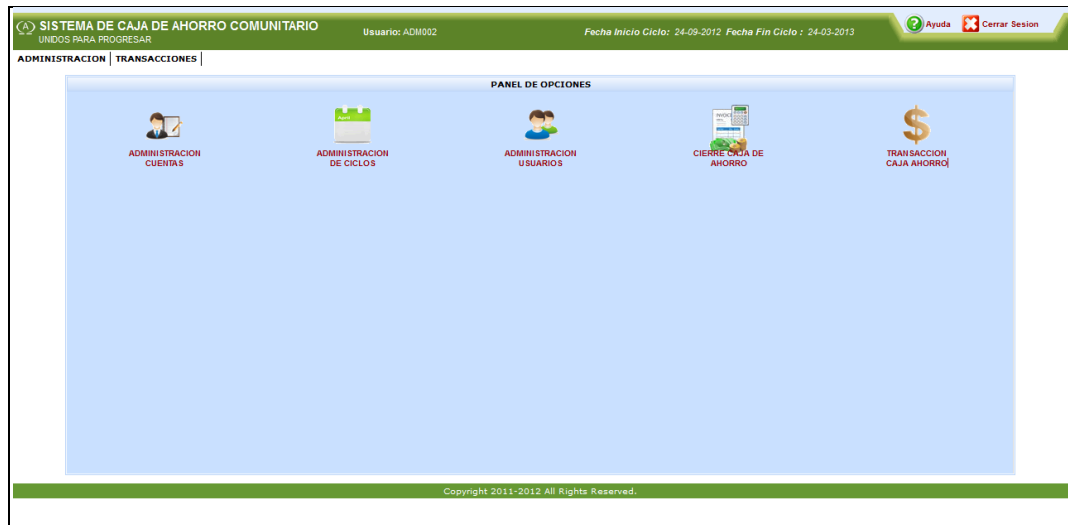
Usuario: Ingrese el usuario.

Password: Ingrese el password.

Botón Ingresar: Luego de ingresar el usuario y el password válidos, puede presionar <ENTER> o hacer clic en *Ingresar*.

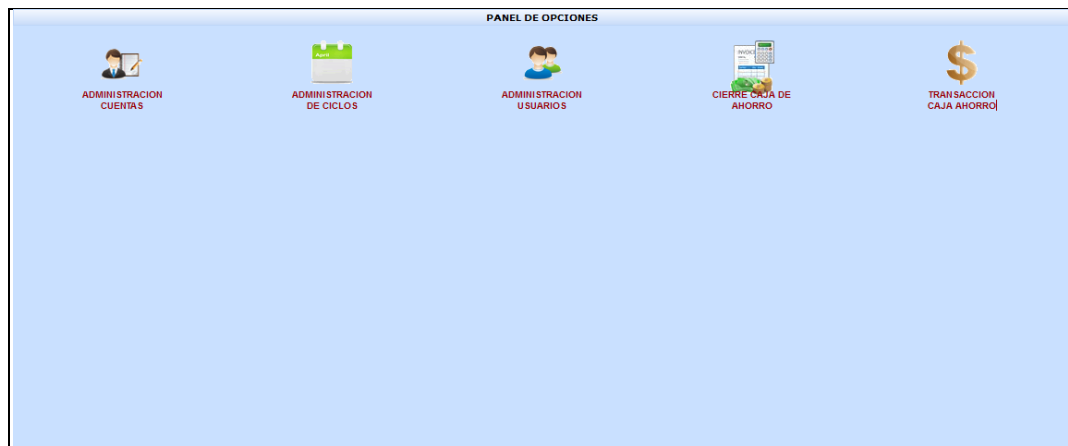
Se presentará el Panel Inicial con los servicios disponibles para el usuario ingresado.

Dependiendo del rol que sea asignado a un usuario se mostrarán las opciones del menú y los servicios del Panel.



4. Administración de Cuentas

Para la creación de una cuenta de ahorros se debe seleccionar en el panel principal la opción Administración de Cuentas.



Una vez que se ha seleccionada la opción Administración de Cuentas el sistema presentara el listado de todas las cuentas que son administradas por el usuario administrador de la caja de ahorro. La siguiente pantalla está compuesta por: sección de búsqueda, sección de resultados de búsqueda, sección de barra de títulos.

Número Cuenta	Cédula Identidad	Nombre Completo	Acciones	Caja Ahorro	Estado
0000200	1001705852	ALEXANDRA RUTH ARMAS FUENTES	45	CAJA01	Activo
0000201	1000398055	FRANCISCA MARIA RUIZ	10	CAJA01	Activo
0000202	1001045879	LUIS MARCELIANO IPIALES IPIALES	20	CAJA01	Activo
0000203	1001924511	NARCIZA CONSUELO CARDENAS RIVERA	23	CAJA01	Activo
2001	1003187851	CHRISTIAN GIOVANY IPIALES MORETA	40	CAJA01	Activo
2002	1003187852	LUIS MARCELO IPIALES MORETA	35	CAJA01	Activo
2003	1003187853	JUAN CARLOS IPIALES MORETA	20	CAJA01	Activo
2004	1003187854	IRALDA ALEXANDRA IPIALES MORETA	20	CAJA01	Activo

La opción de administración de cuentas permite realizar las siguientes acciones:

Nuevo:

Permite la creación de una nueva cuenta de ahorros.

Inicio:

Permite regresar a la pantalla inicial del Sistema caja de ahorro comunitario.

Búsqueda:

Permite realizar búsqueda cuentas de ahorros según los filtros correspondientes a las cuentas.

Editar transacción caja de ahorro

Permite realizar la edición la cuenta seleccionada.

4.1. Nueva cuenta de ahorros

Cuando seleccione la opción Nuevo, el sistema muestra la pantalla que permitirá registrar nueva cuenta de ahorro.

Se debe ingresar un número de cédula el mismo que el sistema validará si es correcto o no lo es.

Luego deberá ingresar los datos principales correspondientes a la nueva cuenta de ahorros como se muestra en la siguiente figura.

Después de ingresar los datos correspondientes se deberá presionar el botón guardar para que se registre la nueva cuenta de ahorros en el sistema.

4.2. Editar cuentas de ahorro

Cuando seleccione la opción editar de una de las cuentas de ahorros el sistema permitirá editar los datos correspondientes a las cuentas de ahorro.

Para editar la cuenta de ahorros deberá seleccionar el número de cédula correspondiente a la persona que es responsable de la cuenta de ahorros, como se muestra en la siguiente figura.

Número Cuenta	Cédula Identidad	Nombre Completo	Acciones	Caja Ahorro	Estado
0000200	1001705852	ALEXANDRA RUTH ARMAS FUENTES	45	CAJA01	Activo
0000201	1000398055	FRANCISCA MARIA RUIZ	10	CAJA01	Activo
0000202	1001045879	LUIS MARCELIANO IPIALES IPIALES	20	CAJA01	Activo
0000203	1001924511	NARCIZA CONSUELO CARDENAS RIVERA	23	CAJA01	Activo
2001	1003187851	CHRISTIAN GIOVANY IPIALES MORETA	40	CAJA01	Activo
2002	1003187852	LUIS MARCELO IPIALES MORETA	35	CAJA01	Activo
2003	1003187853	JUAN CARLOS IPIALES MORETA	20	CAJA01	Activo
2004	1003187854	IRALDA ALEXANDRA IPIALES MORETA	20	CAJA01	Activo

Presionando el botón guardar el sistema permitirá registrar los datos modificados de la cuenta de ahorros seleccionada.

4.3. Descargar archivo pdf

Presionando el botón descargar permitirá descargar un archivo en formato pdf del listado correspondiente a las cuentas de ahorros que se encuentran en pantalla.

Número Cuenta	Cédula Identidad	Nombre Completo	Acciones	Caja Ahorro	Estado
0000206	1001924511	NARCIZA CONSUELO CARDENAS RIVERA	10	CAJA02	Activo
0000207	1710666734	MARCELA ECHEVERRIA	20	CAJA02	Activo
0000205	1003187851	CHRISTIAN GIOVANY PIALES MORETA	30	CAJA02	Activo
0000209	1720313343	JOSE RODRIGUEZ	20	CAJA02	Activo

El archivo resultante se presentará de la siguiente forma:


SISTEMA DE CAJA DE AHORRO COMUNITARIO
LISTADO DE CUENTAS DE AHORROS
UNIDOS PARA PROGRESAR

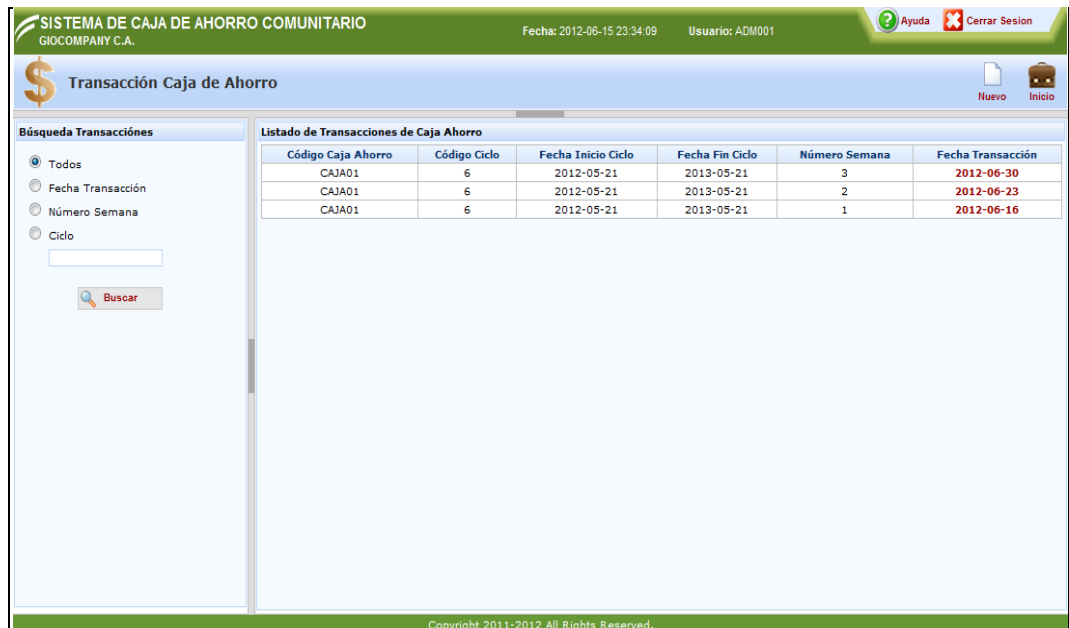
Número Cuenta	Cédula Identidad	Nombre Completo	Acciones
0000206	1001924511	NARCIZA CONSUELO CARDENAS RIVERA	10
0000207	1710666734	MARCELA ECHEVERRIA	20
0000205	1003187851	CHRISTIAN GIOVANY PIALES MORETA	30
0000209	1720313343	JOSE RODRIGUEZ	20

5. Transacción Caja de Ahorro

En la siguiente figura se muestra el panel de opciones que permitirá el ingreso a la transacción caja de ahorro.



Una vez seleccionada la opción Transacción caja de ahorro, el sistema presentará la pantalla para realizar búsquedas de transacciones cajas de ahorro que ya se han realizado anteriormente.



La opción de Transacción de caja de ahorro permite realizar las siguientes acciones:

Nuevo:

Permite la creación de una nueva transacción caja de ahorro.

Inicio:

Permite regresar a la pantalla inicial del Sistema caja de ahorro comunitario.

Búsqueda:

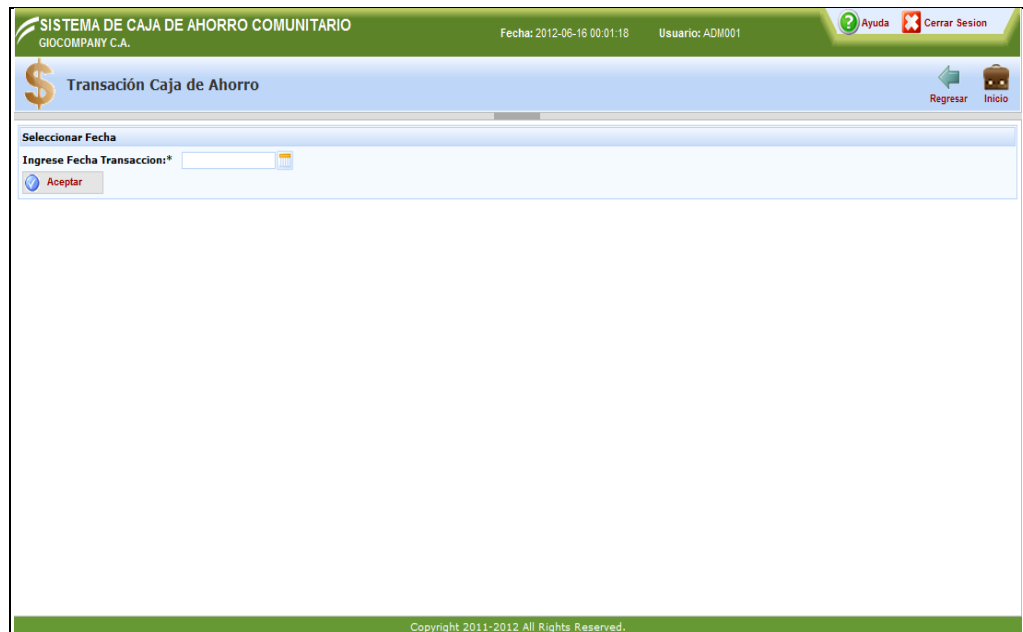
Permite realizar búsqueda de transacciones de caja de ahorro según los filtros correspondientes a las transacciones caja de ahorros.

Editar transacción caja de ahorro

Permite realizar la edición de las transacciones cajas de ahorro.

5.1. Nueva transacción caja de ahorro.

Cuando seleccione la opción Nuevo, el sistema muestra la pantalla que permitirá registrar la transacción caja de ahorro.



En la ficha de registro de Nueva Transacción caja de ahorro se deben ingresarse los siguientes datos:

- **Fecha Transacción:** Es la fecha correspondiente a la nueva transacción caja de ahorro.

Para registrar la información de la nueva transacción caja de ahorro nuevo usuario, haga clic en el botón aceptar.

Observaciones:

En caso de no ingresar datos o ingresar datos no admitidos se presentará un mensaje.
 En caso contrario se presentará un mensaje de éxito

Una vez creado la transacción caja de ahorro se pueden realizar las siguientes opciones:

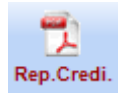
Para regresar al listado de transacciones caja de ahorros existentes. Haga clic en



Para ir a la opción de créditos. Haga clic en Créditos



Para descarga los reportes de créditos, deudas, transacciones semanales haga clic en

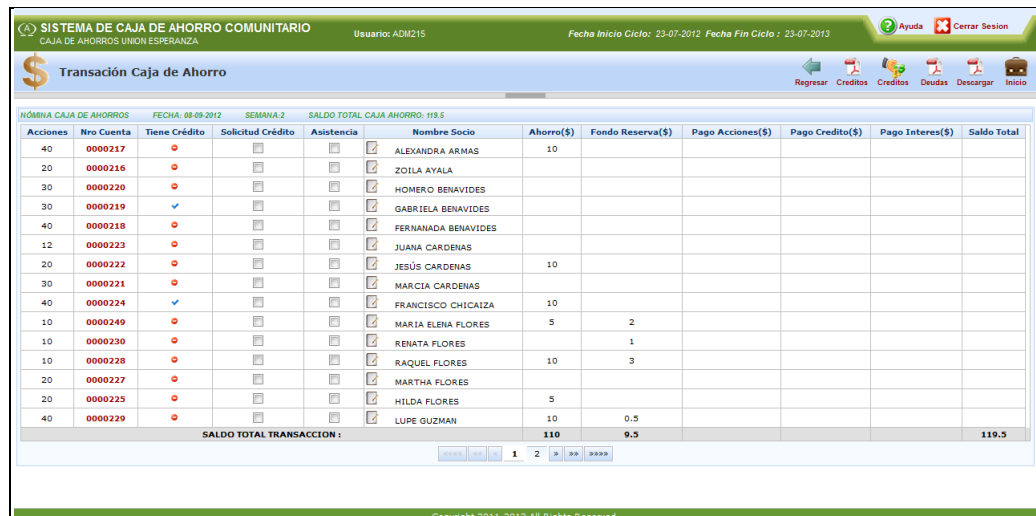


Para regresar a la pantalla inicial del Sistema Caja de ahorro, Haga clic en Inicio



5.2. Registrar detalle transacción caja de ahorro.

Permite realizar el registro del detalle de las cajas de ahorros, cuando se selecciona esta opción el sistema muestra la pantalla siguiente.



Acciones	Nro Cuenta	Tiene Crédito	Solicitud Crédito	Asistencia	Nombre Socio	Ahorro(\$)	Fondo Reserva(\$)	Pago Acciones(\$)	Pago Crédito(\$)	Pago Interes(\$)	Saldo Total
40	0000217	o	<input type="checkbox"/>	<input type="checkbox"/>	ALEXANDRA ARMAS	10					
20	0000216	o	<input type="checkbox"/>	<input type="checkbox"/>	ZOILA AYALA						
30	0000220	o	<input type="checkbox"/>	<input type="checkbox"/>	HOMERO BENAVIDES						
30	0000219	✓	<input type="checkbox"/>	<input type="checkbox"/>	GABRIELA BENAVIDES						
40	0000218	o	<input type="checkbox"/>	<input type="checkbox"/>	FERNANADA BENAVIDES						
12	0000223	o	<input type="checkbox"/>	<input type="checkbox"/>	JUANA CARDENAS						
20	0000222	o	<input type="checkbox"/>	<input type="checkbox"/>	JESÚS CARDENAS	10					
30	0000221	o	<input type="checkbox"/>	<input type="checkbox"/>	MARCIA CARDENAS						
40	0000224	✓	<input type="checkbox"/>	<input type="checkbox"/>	FRANCISCO CHICAIZA	10					
10	0000249	o	<input type="checkbox"/>	<input type="checkbox"/>	MARIA ELENA FLORES	5	2				
10	0000230	o	<input type="checkbox"/>	<input type="checkbox"/>	RENATA FLORES		1				
10	0000228	o	<input type="checkbox"/>	<input type="checkbox"/>	RAQUEL FLORES	10	3				
20	0000227	o	<input type="checkbox"/>	<input type="checkbox"/>	MARTHA FLORES						
20	0000225	o	<input type="checkbox"/>	<input type="checkbox"/>	HILDA FLORES	5					
40	0000229	o	<input type="checkbox"/>	<input type="checkbox"/>	LUPE GUZMAN	10	0.5				
SALDO TOTAL TRANSACCION :						110	9.5				119.5

Se identifican las siguientes columnas:

Acciones: Identifica el número de acciones que tiene cada cuenta.

Número de Cuenta: Identifica el número de cuenta del socio.

Tiene Crédito: Identifica si la cuenta tiene o no créditos.


Solicitud de Crédito: Identifica si el socio ha realizado la solicitud de crédito.

Asistencia: Identifica la asistencia a cada una de las reuniones de los socios.

Nombre Socio: Identifica el nombre del socio.

5.3. Editar detalle transacción caja de ahorro.

Permite registrar las transacciones correspondientes al ahorro, ahorro fondo de reserva, créditos, pago acciones pago créditos, pago interés.

Para la realización de esta acción es necesario presionar el icono  que se encuentra en la parte derecha del nombre del socio, con lo cual se habilitarán las columnas para el ahorro, el ahorro fondo de reserva, pago de acciones, pago de créditos, pago de interés.

NÚMERA CAJA DE AHORROS		FECHA: 08-09-2012	SEMANA: 2	SALDO TOTAL CAJA AHORRO: 119.5							
Acciones	Nro Cuenta	Tiene Crédito	Solicitud Crédito	Asistencia	Nombre Socio	Ahorro(\$)	Fondo Reserva(\$)	Pago Acciones(\$)	Pago Credito(\$)	Pago Interes(\$)	Saldo Total
40	0000217	o	<input type="checkbox"/>	<input type="checkbox"/>	ALEXANDRA ARMAS	10					
20	0000216	o	<input type="checkbox"/>	<input type="checkbox"/>	ZOILA AYALA						
30	0000220	o	<input type="checkbox"/>	<input type="checkbox"/>	HOMERO BENAVIDES	1					
30	0000219	v	<input type="checkbox"/>	<input type="checkbox"/>	GABRIELA BENAVIDES	7.5	9.5	10			
40	0000218	o	<input type="checkbox"/>	<input type="checkbox"/>	FERNANDA BENAVIDES						
12	0000223	o	<input type="checkbox"/>	<input type="checkbox"/>	JUANA CARDENAS						
20	0000222	o	<input type="checkbox"/>	<input type="checkbox"/>	JESÚS CARDENAS	10					
30	0000221	o	<input type="checkbox"/>	<input type="checkbox"/>	MARCIA CARDENAS						
40	0000224	v	<input type="checkbox"/>	<input type="checkbox"/>	FRANCISCO CHICAIZA	10					
10	0000240	o	<input type="checkbox"/>	<input type="checkbox"/>	MARIA ELENA FLORES	5	2				
10	0000230	o	<input type="checkbox"/>	<input type="checkbox"/>	RENATA FLORES		1				
10	0000228	o	<input type="checkbox"/>	<input type="checkbox"/>	RAQUEL FLORES	10	3				
20	0000227	o	<input type="checkbox"/>	<input type="checkbox"/>	MARTHA FLORES						
20	0000225	o	<input type="checkbox"/>	<input type="checkbox"/>	HILDA FLORES	5	0.5				
40	0000229	o	<input type="checkbox"/>	<input type="checkbox"/>	LUPE GUZMAN	10					
SALDO TOTAL TRANSACCION:						110	9.5				119.5

1. Ingreso de ahorros
2. Ingreso de ahorros de fondo de reserva
3. Ingreso de pago de acciones
4. Ingreso de pago de créditos
5. Ingreso de pago de interés



Permite guardar todos los datos ingresados en el detalle transacción caja de ahorro correspondiente a la cuenta del socio.



Permite cancelar el ingreso de los datos del detalle transacción caja de ahorro.

5.4. Solicitud de crédito

Permite registrar la solicitud de un crédito en la caja de ahorro, cuando se selecciona esta opción el sistema muestra la siguientes pantalla.

Acciones	Nro Cuenta	Tiene Crédito	Solicitud Crédito	Asistencia	Nombre Socio	Ahorro(\$)	Fondo Reserva(\$)	Pago Acciones(\$)	Pago Credito(\$)	Pago Interes(\$)	Saldo Total
	40 0000217	o	<input checked="" type="checkbox"/>	<input type="checkbox"/>	ALEXANDRA ARMAS	10					
	20 0000216	o	<input checked="" type="checkbox"/>	<input type="checkbox"/>	ZOLA AYALA						
	30 0000220	o	<input checked="" type="checkbox"/>	<input type="checkbox"/>	HOMERO BENAVIDES						
	30 0000219	v	<input checked="" type="checkbox"/>	<input type="checkbox"/>	GABRIELA BENAVIDES						
	40 0000218	o	<input type="checkbox"/>	<input type="checkbox"/>	FERNANADA BENAVIDES						
	12 0000223	o	<input type="checkbox"/>	<input type="checkbox"/>	JUANA CARDENAS						
	20 0000222	o	<input type="checkbox"/>	<input type="checkbox"/>	JESÚS CARDENAS	10					
	30 0000221	o	<input type="checkbox"/>	<input type="checkbox"/>	MARCIA CARDENAS						
	40 0000224	v	<input type="checkbox"/>	<input type="checkbox"/>	FRANCISCO CHICAIZA	10					
	10 0000249	o	<input type="checkbox"/>	<input type="checkbox"/>	MARIA ELENA FLORES	5	2				
	10 0000230	o	<input type="checkbox"/>	<input type="checkbox"/>	RENATA FLORES		1				
	10 0000228	o	<input type="checkbox"/>	<input type="checkbox"/>	RAQUEL FLORES	10	3				
	20 0000227	o	<input type="checkbox"/>	<input type="checkbox"/>	MARTHA FLORES						
	20 0000225	o	<input checked="" type="checkbox"/>	<input type="checkbox"/>	HILDA FLORES	5					
	40 0000229	o	<input checked="" type="checkbox"/>	<input type="checkbox"/>	LUPE GUZMAN	10	0.5				
SALDO TOTAL TRANSACCION :						110	9.5				119.5


Una vez registrado las solicitudes de crédito se realiza la verificación del crédito



solicitado presionando el botón el cual permite mostrar la siguiente pantalla dividida en tres secciones.

SALDO TOTAL CAJA AHORRO			
Saldo Total Caja Ahorro:	19.5	1	
CRÉDITOS APROBADOS			
Nro Cuenta	Nombre	Monto (\$)	
0000219	GABRIELA BENAVIDES	50	
0000220	HOMERO BENAVIDES	50	
2			
SOLICITANTES DE CRÉDITOS			
Nro Cuenta	Solicitantes	Crédito Pagado Total	Verificar
0000216	ZOLA AYALA		<input checked="" type="checkbox"/>
0000217	ALEXANDRA ARMAS		<input checked="" type="checkbox"/>
0000229	LUPE GUZMAN		<input checked="" type="checkbox"/>
0000225	HILDA FLORES		<input checked="" type="checkbox"/>
3			

1. El saldo total que tiene la caja de ahorro.
2. Los créditos aprobados
3. Las solicitudes de créditos

Para la verificación de las solicitudes de cada uno de los créditos se presiona el botón  verificar que se encuentra en la columna verificar de las solicitudes de créditos con lo cual se mostrará la siguiente pantalla.



SISTEMA DE CAJA DE AHORRO COMUNITARIO Usuario: ADM215 Fecha Inicio Ciclo: 23-07-2012 Fecha Fin Ciclo: 23-07-2013

Solicitud de Crédito

Registro de Solicitud de Crédito

DATOS SOCIO

Número Cuenta: 0000216 Cédula Identidad: 1001618477
 Nombre Completo: ZOILA AYALA Acciones: 20
 Dirección: AMBUQUI Teléfono Convencional:

DETALLE SOLICITUD CRÉDITO

Saldo Total Caja Ahorro 19.5(\$)
 Interés 5(%)
 Monto 240 (\$)
 Plazo 3(Meses)

En la ficha de registro de crédito se deben ingresarse los siguientes datos:

- **Monto:** Es la cantidad de dinero que se solicita en el crédito.
- **Plazo:** Es el tiempo en el cual se debe cancelar el crédito.

Para registrar la información del nuevo crédito haga clic en el botón aceptar.

5.5. Ver detalle cuenta de ahorros.

Permite ver el detalle del estado de cuenta del socio el mismo que permite revisar los ahorros, los ahorros de fondos de reserva, los créditos como se muestra en la siguiente pantalla.



Registro de Ahorros

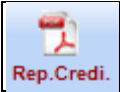
Datos Socio

Número Cuenta: 0000200 Cédula Identidad: 1001705852
 Nombre Completo: ALEXANDRA RUTH ARMAS FUENTES Acciones: 45
 Dirección: BARRIO SAN VICENTE Teléfono Convencional:

1 Ahorros 2 Fondos Reserva 3 Créditos

Código Transacción	Fecha Depósito	Depósito	Saldo	Depósito Faltante	Depósito Excedido
142	2012-06-13	34.0000	34.0000		0.2500
149	2012-06-14	150.0000	184.0000		116.2500

5.6. Reportes Caja de Ahorro

Para realizar reportes de la caja de ahorros se seleccionará los botones  con lo cual permite ver todos los créditos de los socios.





SISTEMA DE CAJA DE AHORRO COMUNITARIO
CAJA DE AHORROS UNION ESPERANZA

Semana N° 2 08/09/12 12:00 AM

Listado de Créditos

Nómina	Fecha	monto(\$)	plazo (meses)	Interés (%)	Estado
GABRIELA BENAVIDES	07-09-2012	100,00	3	5	Activo
	Fecha Pago	Interés(\$)	Capital(\$)	Capital Total	Estado
	05-10-2012	5,00	100,00	105,00	Activo
	02-11-2012	5,25	105,00	110,25	Activo
	30-11-2012	5,51	110,25	115,76	Activo
FRANCISCO CHICAIZA	07-09-2012	100,00	3	5	Activo
	Fecha Pago	Interés(\$)	Capital(\$)	Capital Total	Estado
	05-10-2012	5,00	100,00	105,00	Activo
	02-11-2012	5,25	105,00	110,25	Activo
	30-11-2012	5,51	110,25	115,76	Activo

Para realizar el reporte de las deudas por cobrar seleccionara el botón  que permite ver todos los socios que poseen deudas



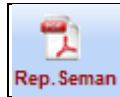
SISTEMA DE CAJA DE AHORRO COMUNITARIO
CAJA DE AHORROS UNION ESPERANZA

Semana N° 2 08/09/12 12:00 AM


Valores Impagos

Nómina	Ahorro Adeudado	Fondo Reserva Adeudado	Pago Accion Adeudado
ALEXANDRA ARMAS	10,00	1,00	40,00
ZOILA AYALA	0,00	1,00	0,00
HOMERO BENAVIDES	15,00	1,00	30,00
FERNANADA BENAVIDES	20,00	1,00	40,00
GABRIELA BENAVIDES	10,00	0,00	10,00
JUANA CARDENAS	3,00	0,50	12,00

Para realizar el reporte de las transacciones que se realizarón en la caja de ahorro se



debe seleccionar el botón

 SISTEMA DE CAJA DE AHORRO COMUNITARIO CAJA DE AHORROS UNION ESPERANZA					
Semana N° 2		08/09/12 12:00 AM			
Nómina Semanal					
Nómina	Ahorro	Fondo Reserva	Pago Acciones	Pago Créditos	Pago Interés
ALEXANDRA ARMAS	10,00	--	--	--	--
ZOILA AYALA	--	--	--	--	--
HOMERO BENAVIDES	--	--	--	--	--
GABRIELA BENAVIDES	--	--	--	--	--
FERNANADA BENAVIDES	--	--	--	--	--
JUANA CARDENAS	--	--	--	--	--
JESÚS CARDENAS	10,00	--	--	--	--
MARCIA CARDENAS	--	--	--	--	--