



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA
E INDUSTRIAL**

**CARRERA DE INGENIERÍA EN SISTEMAS
COMPUTACIONALES E INFORMÁTICOS**

Tema:

**DISEÑO E IMPLEMENTACIÓN DE MÓDULOS ONLINE MEDIANTE
EL USO DE WEB SERVICE PARA INGRESO DE FORMULARIOS DE:
APERTURA DE CUENTA, PRESTAMOS, CONSULTA DE: SALDOS,
DEPÓSITOS, PAGOS DE CRÉDITOS E INGRESO DE PETICIONES DE
CERTIFICADOS EN LA COOPERATIVA DE AHORRO Y CRÉDITO
CHIBULEO LTDA DE LA CIUDAD DE AMBATO PROVINCIA DE
TUNGURAHUA.**

Trabajo de Graduación. Modalidad: TEMI. Trabajo Estructurado de Manera Independiente, presentado previo la obtención del título de Ingeniero en Sistemas Computacionales e Informáticos.

AUTOR: Sr. Cunalata Wilson

TUTOR: Ing. Jaime Ruiz

Ambato – Ecuador
Diciembre 2010

APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de investigación sobre el tema: “**DISEÑO E IMPLEMENTACIÓN DE MÓDULOS ONLINE MEDIANTE EL USO DE WEB SERVICE PARA INGRESO DE FORMULARIOS DE: APERTURA DE CUENTA, PRESTAMOS, CONSULTA DE: SALDOS, DEPÓSITOS, PAGOS DE CRÉDITOS E INGRESO DE PETICIONES DE CERTIFICADOS EN LA COOPERATIVA DE AHORRO Y CRÉDITO CHIBULEO LTDA DE LA CIUDAD DE AMBATO PROVINCIA DE TUNGURAHUA.**”, del señor **Cunalata Wilson**, egresado de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos , de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el Art. 68 del Capítulo IV, del Reglamento de Graduación de Pregrado de la Universidad Técnica de Ambato.

Ambato Diciembre, 2010

EL TUTOR

Ingeniero Jaime Ruiz

AUTORÍA

El presente trabajo de investigación titulado: **“DISEÑO E IMPLEMENTACIÓN DE MÓDULOS ONLINE MEDIANTE EL USO DE WEB SERVICE PARA INGRESO DE FORMULARIOS DE: APERTURA DE CUENTA, PRESTAMOS, CONSULTA DE: SALDOS, DEPÓSITOS, PAGOS DE CRÉDITOS E INGRESO DE PETICIONES DE CERTIFICADOS EN LA COOPERATIVA DE AHORRO Y CRÉDITO CHIBULEO LTDA DE LA CIUDAD DE AMBATO PROVINCIA DE TUNGURAHUA.”**, es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato Diciembre, 2010

Wilson Roberto Cunalata
CC: 1804029187

APROBACIÓN DE LA COMISIÓN CALIFICADORA

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ingeniero Aldas Clay , Ingeniero Guevara David , revisó y aprobó el Informe Final del trabajo de graduación titulado **“DISEÑO E IMPLEMENTACIÓN DE MÓDULOS ONLINE MEDIANTE EL USO DE WEB SERVICE PARA INGRESO DE FORMULARIOS DE: APERTURA DE CUENTA, PRESTAMOS, CONSULTA DE: SALDOS, DEPÓSITOS, PAGOS DE CRÉDITOS E INGRESO DE PETICIONES DE CERTIFICADOS EN LA COOPERATIVA DE AHORRO Y CRÉDITO CHIBULEO LTDA DE LA CIUDAD DE AMBATO PROVINCIA DE TUNGURAHUA.”**, presentado por el señor Wilson Cunalata de acuerdo al Art. 57 del Reglamento de Graduación para obtener el título Terminal del tercer nivel de la Universidad Técnica de Ambato.

Ambato Diciembre 2010

Ing. M.Sc. Oswaldo Paredes
PRESIDENTE DEL TRIBUNAL

Ing. Clay Aldas
DOCENTE CALIFICADOR

Ing. David Guevara
DOCENTE CALIFICADOR

DEDICATORIA:

Quiero expresar mi agradecimiento

A mi Tutor de TEMI, Ing. Jaime Ruiz por su generosidad al brindarme la oportunidad de recurrir a su capacidad y experiencia científica en un marco de confianza, afecto y amistad, fundamentales para la concreción de este trabajo.

A mis compañeros de la Facultad de Ingeniería en Sistemas por su continuo y afectuoso aliento.

A mi esposa por su cariño, comprensión y constante estímulo.

A mi hija por su paciencia y por enseñarme a enfrentar los obstáculos con alegría.

A mis padres y hermanos por brindarme un hogar cálido y enseñarme que la perseverancia y el esfuerzo son el camino para lograr objetivos.

A la Facultad de Ingeniería en Sistemas, Electrónica e Industrial por ser templo del saber y poder inculcarme los conocimientos de hoy tengo y por prepararme como profesional

Wilson Roberto Cunalata

AGRADECIMIENTO:

Doy infinitas gracias a Dios, por el camino recorrido.

A mi hija y esposa, por ser mi fuerza y templanza que significan en mi vida.

A mis padres, por su amor, apoyo y confianza, siempre estuvieron a mi lado.

A mis profesores que me han encaminado.

A la vida.

Por lo aprendido y aprenderé.

Wilson Roberto Cunalata Pachacama

Índice

DESCRIPCIÓN	Pag
Carátula.....	i
Aprobación del tutor.....	ii
Autoría.....	iii
Aprobación de la comisión calificadora.....	iv
Dedicatoria.....	v
Agradecimiento.....	vi
Índice.....	vii
Índice de Figuras	xi
Índice de Tablas.....	xiii
Resumen ejecutivo.....	xv
Introducción.....	xvi

CAPÍTULO I

1 El Problema.....	1
1.2 Planteamiento del Problema.....	1
1.2.1 Contextualización.....	1
1.2.2 Análisis crítico.....	2
1.2.3 Prognosis.....	3
1.3 Formulación del Problema.....	4
1.3.1 Preguntas Directrices.....	4
1.3.2 Delimitación del Problema.....	4
1.4. Justificación.....	5
1.4.1 Objetivo general.....	6
1.5 Objetivos de la Investigación.....	6
1.5.1 Objetivo General.....	6
1.5.2 Objetivos específicos.....	6

CAPÍTULO II

2.1 Antecedentes investigativos.....	7
2.2. Fundamentación.....	8
2.2.1 Fundamentación legal.....	8
2.2.2. Fundamentación teórica.....	10
2.2.2.1. Internet.....	10
2.2.2.2 World Wide Web	11
2.2.2.3. Web Server.....	13
2.2.2.4. CMS.....	15
2.2.2.5. Lenguajes de informáticos.....	16

2.2.2.6 Web Service.....	31
2.2.2.7. Base de Datos.....	35
2.2.2.8. Sistema operativo.....	38
2.2.3. Lenguaje Organizacional.....	44
2.2.3.1 Fundamentación de la Empresa.....	50
2.3. Variables	52
2.3.1 Variable independiente.....	52
2.3.2 Variable dependiente.....	52
2.4 Hipótesis.....	53

CAPÍTULO III

3.1 Enfoque de la investigación	53
3.2 Modalidad básica de la información.....	53
3.2.1 Investigación de campo.....	54
3.2.3 Proyecto factible.....	54
3.2.2 Investigación documental – bibliográfica.....	54
3.3 Nivel o tipo de investigación.....	55
3.4. Población y muestra	55
3.4.1 Población.....	55
3.4.2 Muestra.....	55
3.5 Operacionalización de las variables.....	56
3.6. Recolección de la información	58
3.6.1 Plan de recolección de la información.....	58
3.6.2 Procesamiento y análisis de la información.	58
3.7 Análisis e interpretación de resultados.	59
3.7.1 Comprobación de la hipótesis	59

CAPÍTULO IV

4.1. Análisis	60
4.2 Interpretación de Datos.....	60

CAPÍTULO V

5.1 Conclusiones.....	69
5.2 Recomendaciones.....	70

CAPÍTULO VI

6.1 Análisis del sistema.....	71
6.2 Análisis de factibilidad.....	74
6.2.1 Factibilidad operativa.....	74
6.2.2 Factibilidad técnica.....	75

6.3. Diseño del sistema.....	76
6.3.1 Modelo Relacional – Físico.....	77
6.3.2 Diccionario de Datos.....	78
6.3.3 Script de la base de datos.....	99
6.4 Implantación.....	115
6.4.1 Características del servidor.....	115
6.4.2 Características del cliente.....	119
6.5 Pruebas.....	119
6.5.1 Pruebas de caja blanca.....	119
6.5.2 Prueba de caja negra.....	120
6.5.3 Pruebas de verificación y validación.....	120
6.6 Conclusiones y Recomendaciones.....	121
Bibliografía.....	124
Glosario de Términos.....	128

ANEXOS

Anexo 1 Diagrama de Inclusión de variables.....	134
Anexo 2 Encuesta.....	135
Anexo 3 Código del sistema.....	137
Anexo 4 Manual de Usuario.....	164
Anexo 5 Manual de Administrador.....	179

ÍNDICE DE FIGURAS

	Pag.
Figura 2.1 Estructura de Funcionamiento de Web Service	31
Figura 4.1 Interpretación de resultado encuestas pregunta 1	61
Figura 4.2 Interpretación de resultado encuestas pregunta 2	63
Figura 4.3 Interpretación de resultado encuestas pregunta 3	65
Figura 4.4 Interpretación de resultado encuestas pregunta 4	67
Figura 4.5 Interpretación de resultado encuestas pregunta 5	69
Figura 6.1 Modelo Relacional – Físico	79
Figura 6.2 Ejecucion Web Service	169
Figura 6.3 Metodo Web Service	169
Figura 6.4 Resultado web service	170
Figura 6.5 Enlace Archivos Jquery	170
Figura 6.6 Información Sobre Campos de un formulario	170
Figura 6.7 Implementacion De Acceso Al Sistema	180
Figura 6.8 Página principal	181
Figura 6.9 Recuperar clave	182
Figura 6.10 Formulario de recuperación de clave	182
Figura 6.11 Acceso al sistema	183
Figura 6.12 Mensaje 1	183
Figura 6.13 Mensaje 2	184
Figura 6.14 Pantalla de Saldos	184
Figura 6.15 Impresión de Saldo Actual	185
Figura 6.16 Saldos por Fechas	186
Figura 6.17 Formulario saldos por Fechas	186
Figura 6.18 Selección de cuenta	186
Figura 6.19 selección de fechas	187
Figura 6.20 selección de fechas inicial y final	187
Figura 6.21 resultado de consultas por fechas	188
Figura 6.22 impresión de resultados	188
Figura 6.23 Mensaje no Créditos	188
Figura 6.24 Visualización de datos de créditos	189
Figura 6.25 principal de certificados	190
Figura 6.26 formulario de certificados.	190
Figura 6.27 Tipos de certificados	191
Figura 6.28 Petición ingresada al sistema	191
Figura 6.29 listado de peticiones de certificados	192
Figura 6.30 Eliminar certificados	192
Figura 6.31 ingreso de cedula	193
Figura 6.32 Lista de ingresos	193
Figura 6.33 Formulario de preapertura de cuenta	194

ÍNDICE DE TABLAS

	Pag
Tabla 2.1 Tipos de datos C#	25
Tabla 2.2 Tipos de datos flotante C#	26
Tabla 2.3 Tipos de datos carácter C#	26
Tabla 2.4 Tipos de datos lógicos C#	26
Tabla 2. 5 Operadores C#	27
Tabla 4.1 Tabulación de datos Pregunta 1 Encuesta	60
Tabla 4.2 Tabulación de datos Pregunta 2 Encuesta	62
Tabla 4.3 Tabulación de datos Pregunta 3 Encuesta	64
Tabla 4.4 Tabulación de datos Pregunta 4 Encuesta	66
Tabla 4.5 Tabulación de datos Pregunta 5 Encuesta	68
Tabla 6.1 Listado de columnas de la tabla Cantones	80
Tabla 6.2 Listado de referencias de la tabla Cantones	80
Tabla 6.3 Listado de columnas de la tabla Certificados	81
Tabla 6.4 Listado de Columnas de la tala Certificado_Clientes	81
Tabla 6.5 Listado de referencias de la tabla Certificado_Clientes	82

Tabla 6.6 Listado de columnas de la tabla Clientes	83
Tabla 6.7 Listado de referencias de la tabla Clientes	83
Tabla 6.8 lista de columnas de la tabla clientes naturales	84
tabla 6.9 listado de referencias de la tabla clientes naturales	84
Tabla 6.10 Lista De Columnas De La Tabla Cuentas	85
Tabla 6.11 Listado De Referencias De La Tabla Cuentas	86
Tabla 6.12 Listado de columnas de la tabla datos crediticios	87
Tabla 6.13 Listado de índices de la tabla datos crediticios	87
Tabla 6.14 Listado de referencias de la tabla datos crediticios	87
Tabla 6.15 Listado de columnas de la tabla garantos prestamos	88
Tabla 6.16 Listado de columnas de la tabla ítems económicos	88
Tabla 6.17 Listado de columnas de la tabla ítems económicos clientes	89
Tabla 6.18 Listado de referencias de la tabla ítem económicos clientes	89
Tabla 6.19 Listado de columnas de ítems patrimoniales	90
Tabla 6.20 Listado de ítems patrimoniales clientes	90
Tabla 6.21 Listado de referencias de la tabla ítems patrimoniales clientes	91
Tabla 6.22 Listado de columnas de la tabla Productos	91
Tabla 6.23 Listado de columnas de la tabla profesiones	92
Tabla 6.24 Listado de columnas de la tabla provincias	92
Tabla 6.25 Listado de columnas de la tabla referencias bancarias	93
Tabla 6.26 Listado de referencias de la tabla referencias bancarias	93
Tabla 6.27 Listado de columnas de la tabla referencias comerciales	94
Tabla 6.28 Listado de referencias de la tabla referencias comerciales	94
Tabla 6.29 Listado de columnas de la tabla referencias comerciales	95
Tabla 6.30 Listado de referencias de la tabla referencias comerciales	95
Tabla 6.31 Listado de columnas de la tabla solicitudes	97
Tabla 6.32 Listado de referencias de la tabla solicitudes	97
Tabla 6.33 Listado de columnas de la tabla web_accesos	97
Tabla 6.34 Listado de referencias la tabla web_accesos	98
Tabla 6.35 Listado de columnas de la tabla web_logins	99
Tabla 6.36 Listado de referencias de la tabla web_logins	99
Tabla 6.37 Listado de columnas de la tabla web_logins_rols	99
Tabla 6.38 Listado de referencias de la tabal web_logins_rols	100
Tabla 6.39 Listado de columnas de la tabla web_rols	100
Tabla 6.40 Características del Servidor	120
Tabla 6.41 Características del Equipo Cliente	121

RESUMEN EJECUTIVO

La Cooperativa de Ahorro y Crédito “Chibuleo” Ltda., nace para satisfacer necesidades de bienes y servicios; como objetivo es la entrega de soluciones efectivas al sistema micro-financiero, con nuevas propuestas fundamentadas en la solidaridad y el servicio.

El 2003 fue determinante para el cooperativismo en la provincia, así lo certificaron los medios de comunicación local y nacional que registraron en sus páginas el trabajo decidido de 27 jóvenes indígenas de entre 18 y 20 años que iniciaron sus actividades en función de ser útiles a la sociedad, el resultado fue: LA COOPERATIVA DE AHORRO Y CRÉDITO CHIBULEO.

Actualmente la información de la Cooperativa se la lleva en el Sistema de Financiamiento Chibuleo en toda su área interna de forma centralizada, más no accesible desde instalaciones fuera de la institución.

En los últimos años en el ámbito financiero se da gran auge a los sitios Web dinámicos por sus múltiples ventajas, maneras de accesibilidad y formas de asegurar la información que se ingresa por parte del cliente mediante HTTPS, encriptación de envío de datos.

Por esta razón el diseño e implementación de módulos online en la Cooperativa de Ahorro y Crédito “Chibuleo” Ltda., en base a los requerimientos de la Institución, ayudara a dar solución al problema, para lo cual a continuación se presenta una investigación que nos facilitará encontrar una propuesta práctica a la Cooperativa.

Introducción

Debido al gran auge tecnológico se hace indispensable que el acceso a la información sea de manera más rápida, eficaz y segura. Por este motivo millones de personas están utilizando el Internet como herramienta para tratar de solucionar este problema, y que mejor manera que con un portal Web, al que pueden acceder desde cualquier parte del mundo.

El hecho de no contar con los módulos online para ingreso de formularios de: apertura de cuenta, prestamos, consulta de: saldos, depósitos, pagos de créditos e ingreso de peticiones de certificados, ha hecho que muchos clientes y futuros clientes desconocen todos servicios y beneficios que trae consigo ser un socio de la Cooperativa. Además se ve perjudicada la Institución que no puede darse a conocer de manera adecuada al país.

Por medio de este trabajo se pretende dar a conocer a la colectividad y al país, los nuevos servicios que ofrece a sus socios. Los usuarios se mantendrán siempre actualizados, contando con un acceso a la información de manera rápida y segura.

La Cooperativa de Ahorro y Crédito “CHIBULEO”, está conformado por 7 sucursales: Quito, Chillo Gallo, Riobamba, Salcedo, Macachí, Latacunga y su principal en Ambato, 12 de Noviembre y Mariano Egeuz.

CAPÍTULO I

EL PROBLEMA

1.1 Tema:

“DISEÑO E IMPLEMENTACIÓN DE MÓDULOS ONLINE MEDIANTE EL USO DE WEB SERVICE PARA INGRESO DE FORMULARIOS DE: APERTURA DE CUENTA, PRESTAMOS, CONSULTA DE: SALDOS, DEPÓSITOS, PAGOS DE CRÉDITOS E INGRESO DE PETICIONES DE CERTIFICADOS EN LA COOPERATIVA DE AHORRO Y CRÉDITO CHIBULEO LTDA DE LA CIUDAD DE AMBATO PROVINCIA DE TUNGURAHUA.”

1.2 Planteamiento del problema

1.2.1 Contextualización

El Ecuador, en las últimas décadas está sufriendo un proceso de transformación económica y social muy importante. El crecimiento económico es notable; pero esto ha dado lugar a que las instituciones financieras tradicionales y nuevas, lleven una competencia entre sí, cada día más fuerte. La mejor forma para poder conllevar la competitividad en el mundo empresarial, es el uso de tecnologías de punta a fin de dar un buen servicio a sus clientes, como es la de presentación de datos financieros en línea, aunque siempre, el temor del fraude o la suplantación de identidad, han hecho que algunas Cooperativas carezcan de sitios Web que ofrezcan estos servicios.

En los últimos años en el ámbito provincial los sitios Web dinámicos para entidades financieras ha tenido un gran auge por el conocimiento de las múltiples ventajas y formas seguras de acceso a la información de los clientes, perdiendo en

gran parte el miedo a las diferentes formas de fraude que son los principales factores por los que no se implementaban este tipo de servicios.

Además en últimas encuestas realizadas se determina que el 70% de las personas que tienen acceso a Internet demandan las 24 horas, 7 días a la semana, de los productos y servicios adaptados a sus necesidades y no basta una presencia física como es el caso de una estructura de oficinas.

En la actualidad la Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda. trabaja con un personal capacitado en las diferentes áreas pero no cuenta con un sistema ONLINE que le permita satisfacer las necesidades de sus clientela en relación con el ingreso de formularios de préstamos, consulta de saldos de depósitos, consulta de pago de créditos e ingreso de peticiones de certificados mediante el uso de Web Service en el sitio Web, lo que evitaría que los usuarios pierdan tiempo al realizar las largas colas para poder acceder a los servicios que ellos requieren.

1.2.2 Análisis crítico

Siendo la Cooperativa de Ahorro y Crédito “CHILULEO” Ltda. una Institución que presta servicio a un alto índice de la población Tungurahuese no se ha integrado a los nuevos avances tecnológicos que muchas instituciones financieras lo están empleando debido en cierta forma al desconocimiento de nuevas tecnologías y de falta de ofertas de servicios ONLINE por parte de quienes manejan estos sistemas, a más de esto el personal que labora en la Cooperativa si bien es cierto está especializado en diferentes áreas principalmente en la contable, no están capacitados en el manejo de tecnología WEB, lo que hace que se produzca insatisfacción en los usuarios por el servicio centralizado que se le viene brindando lo que se traduce en pérdidas económicas y falta de competitividad en el mercado financiero.

La manera actual en que se lleva la información, necesita cambios enfocados a la eliminación del control centralizado. Es decir que solo en la Institución o sus sucursales se puede realizar consultas para los servicios que se brinda, mientras que con la implementación de módulos Online para ingreso de formularios de préstamos, consulta de saldos de depósitos, consulta de pago de créditos e ingreso de peticiones de certificados, los clientes podrán realizar este tipo de operaciones bancarias desde cualquier parte del Ecuador.

1.2.3 Prognosis

Si no se implementara un sistema que tenga como finalidad disminuir las filas que tienen que formar los clientes para realizar una consulta, ya sea para préstamos, consulta de saldos de depósitos, consulta de pago de créditos o peticiones de certificados, continuará con la existencia de clientes descontentos por no brindar un servicio en una vía tecnológica que sea asequible todos los días y las 24 horas del día, y que evite la pérdida de tiempo que tiene el cliente y que puede llevar al decrecimiento o rebaja en el capital financiero.

Actualmente la Cooperativa requiere contar con módulos para el sitio Web dinámicos capaces de proporcionar información de formularios de: apertura de cuenta, préstamos, consulta de saldos de depósitos, consulta de pago de créditos e ingreso de peticiones de certificados de las cuentas de sus clientes, que ayudará a ahorrar tiempo al cliente, así como también se beneficiaran las personas encargadas de brindar dicha información puesto que los clientes no necesitarían acudir a la Cooperativa para obtener una consulta precisa en corto tiempo, ya que el trámite normal se demora aproximadamente una media hora para conocer su saldo u otro servicio, mientras que con el nuevo módulo será solo cuestión de pocos minutos.

Para la realización de estos módulos se ha buscado técnicas de seguridad, con el fin de evitar la pérdida de información valiosa para los clientes y socios.

1.3 Formulación del problema

¿Cómo incide el diseño e implementación de módulos Online mediante el uso de Web service en el ingreso de formularios de: apertura de cuentas, préstamos, consulta de: saldos, depósitos, pagos de créditos e ingreso de peticiones de certificados en la Cooperativa de Ahorro y Crédito CHIBULEO Ltda., de la ciudad de Ambato provincia de Tungurahua.?

1.3.1 Preguntas Directrices

- 1.- ¿El sistema que utiliza la empresa permite el ingreso de documentos de préstamos y depósitos online?
- 2.- ¿Qué datos son necesarios para el diseño de módulos Online mediante Web service?
- 3.- ¿Qué tipo de seguridad se necesitara implantar para que el acceso a los módulos?
- 4.- ¿Qué clase de software se considera apropiada para el desarrollo de los módulos Online?

1.3.2 Delimitación del problema

Para el diseño e implementación de módulos Online se utilizara la tecnología PHP-MYSQL para ingreso de formularios de: apertura de cuenta, préstamos, consulta de saldos de depósitos, consulta de pago de créditos e ingreso de peticiones de certificados mediante el uso de Web Service en el sitio Web de la Cooperativa de Ahorro y Crédito CHIBULEO Ltda., de la ciudad de Ambato provincia de Tungurahua, este proyecto se realizará en el período comprendido entre Septiembre del 2009 y Febrero del 2010 y como población se trabaja con 50

personas seleccionadas del departamento de sistemas, diferentes áreas administrativas y clientes .

1.4 Justificación

La realización de este proyecto es de suma importancia debido que en el campo de los negocios electrónicos la competencia mundial se intensifica, es por eso que las empresas financieras buscan ampliar las redes de atención al cliente, y aumentar los esfuerzos para reducir el tiempo que sus clientes deben utilizar al acudir a las instalaciones de las mismas, por lo que estas necesitan que se brinde acceso las 24 horas, 7 días a la semana, a los productos y servicios que ofrecen las entidades, mediante la utilización de sitios Web dinámicos que le permitan este acceso.

Por esta razón se realizara el desarrollo de los presentes módulos, que es indispensables para darle dirección y promoción a la Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda., ya que sus esfuerzos deben concentrarse en adquirir nuevas oportunidades, determinando el mercado financiero y un sitio de origen en la banca por internet apropiada para el desenvolvimiento y ampliación de la misma, de tal manera que sean beneficiarios de este proyecto todos los usuarios que diariamente están recibiendo servicios de la Institución, la misma que busca incrementar la confianza y dinamismo en cuanto a los ingreso de formularios de préstamos, consulta de saldos y pagos de créditos , ya que actualmente debido a los cambios avances tecnológicos que enfrenta el país , ha provocado que todos optemos por los negocios en línea, y esto ha difundido que las familias ecuatorianas ya no solo realicen consultas en horas de oficina sino las 24 horas del día.

La implantación de los módulos ONLINE mediante WEB SERVICE permitirá aplicar en base a la teoría y práctica adquirida durante los años de estudio, solucionar el problema; logrando captar la atención de las grandes y medianas empresas financieras del mercado, expandiendo a nivel mundial los productos y servicios que ofrece la Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda., al

hacer esto se logra estabilizar la empresa e incrementar la confiabilidad en los clientes, generando nuevas aperturas de cuentas y otros servicios que la misma brinda.

Además de utilizar un medio electrónico, y beneficiar al cliente brindando accesibilidad y cobertura de los servicios de la Cooperativa, sin importar las barreras geográficas, el cliente puede tener acceso a sus cuentas, desde localidades donde el mismo no cuente con oficinas e incluso cuando se encuentre en el extranjero.

1.5 Objetivos de la investigación

1.5.1 Objetivo General

Diseñar e Implementar un sistema de módulos Online mediante Web Service para ingreso de formularios de: apertura de cuenta, préstamos, consulta de: saldos, depósitos, pagos de créditos e ingreso de peticiones de certificados en la Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda., de la Ciudad de Ambato provincia de Tungurahua.

1.5.2 Objetivos Específicos

1. Analizar el sistema actual de ingreso de datos de préstamos, depósitos y documentos.
2. Seleccionar el diseño de módulos Online mediante Web Service.
3. Establecer el tipo de software que se requiere para el diseño del módulo Online mediante WEB SERVICE.
4. Determinar las seguridades para el ingreso a los módulos.
5. Diseñar e Implementar el sistema de módulos Online mediante WEB SERVICE

CAPÍTULO II

MARCO TEÓRICO

2.1 Antecedentes investigativos

Investigando en la biblioteca de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial se ha encontrado una pasantía previa realizada por la Ingeniera Jeanette Patricia Villegas Ocaña con el tema “MÓDULO OSCUS-ONLINE PARA CONSULTAS DE SALDOS DE DEPOSITOS Y PAGO DE CRÉDITOS EN EL SITIO WEB DE LA COOPERATIVA DE AHORRO Y CRÉDITO OSCUS LTDA.”, y se ha llegado a las siguientes conclusiones y recomendaciones:

CONCLUSIONES

- La implementación de los módulos favorecen al cliente o socio para poder realizar sus transacciones de manera más rápida y oportuna.
- Ayudo a la empresa a disminuir las colas que se realizaban para consultas.
- Se debe dotar de un manual completo de la forma de utilizar los módulos implementados.
- A los clientes se les debe dar a conocer la importancia de su clave de acceso a los módulos, ya que el mal uso o compartirlo puede llevar a causar pérdidas tanto para el cliente como para la Institución.

RECOMENDACIONES

- A pesar de que el módulo tiene un fácil manejo, los clientes o socios deben tomar en cuenta los mensajes que se generen en el módulo por el mal ingreso de los datos del cliente para poder acceder a la información requerida.
- Es recomendable aclarar al cliente o socio que no debe divulgar su clave de acceso ya que la información que se maneja en el módulo es confidencial y personal. Es importante tomar en cuenta que el socio o cliente tiene la posibilidad de cambio de clave o solicitar la clave si la olvido, todo esto a través del módulo.

2.2 Fundamentación

2.2.1 Fundamentación legal

Propiedad intelectual – Software y base de datos – Protección – Normas.

Artículo 1°.- A los efectos de la aplicación del presente decreto y de las demás normativas vigentes en la materia:

- a) Se entenderá por obras de software, incluidas entre las obras del artículo 1° de la ley N° 11.723, a las producciones constituidas por una o varias de las siguientes expresiones:
 - Los diseños tanto generales como detallados, del flujo lógico de los datos en un sistema de computación
 - Los programas de computación, tanto en su versión “fuente”, principalmente destinada al lector humano, como es su versión “objeto”, principalmente destinada a ser ejecutada por el computador.
- b) Se entenderá por obra de base de datos, incluidas en la categoría de obras literarias, a las producciones constituidas por un conjunto organizado de datos interrelacionados, compilado con miras a su almacenamiento, procesamiento y recuperación mediante técnicas y sistemas informáticos.
- c) Se considerarán procedimientos idóneos para reproducir obras de software o de base de datos a los escritos o diagramas, directa o indirectamente

perceptibles por los sentidos humanos, así como a los registros realizados mediante cualquier técnica, directa o indirectamente procesables por equipos de procesamiento de información.

- d) Se considerará que una obra de software o de base de datos tiene el carácter de publicada cuando ha sido puesta a disposición del público en general, ya sea mediante su reproducción sobre múltiples ejemplares distribuidos comercialmente o mediante la oferta generalizada de su transmisión a distancia con fines de explotación.
- e) Se considerará que una obra de software o de base de datos tiene el carácter de inédita, cuando su autor, titular o derecho habiente la mantiene en reserva o negocia la cesión de sus derechos de propiedad intelectual contratando particularmente con los interesados.

Artículo 2°.- Para proceder al registro de obras de base de datos publicadas, cuya explotación se realice mediante su transmisión a distancia, se depositarán amplios extractos de su contenido y relación escrita de su estructura y organización, así como de sus principales características, que permitan a criterio y riesgo del solicitante individualizar suficientemente la obra y dar la noción más fiel posible de su contenido.

Artículo 3°.- Para proceder al registro de obras de software o de base de datos que tengan el carácter de inéditas, el solicitante incluirá bajo sobre lacrado y firmado todas las expresiones de la obra que juzgue conveniente y suficientes para identificar su creación y garantizar la reserva de su información secreta.

El proyecto a realizarse se efectuará para la Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda., con el siguiente software:

- MySQL.- Es un sistema de gestión de base de datos relacional, multihilo y multiusuario de código abierto

- Lenguaje de programación PHP.- Lenguaje de programación parecido al C o Perl, que permite la creación de páginas Web dinámicas. Se lo puede digitar en un bloc de notas o con algún software.
- Apache.- Es un servidor de Web estable, fiable y veloz para plataformas Unix. De distribución libre y de código abierto.
- HTML.- Lenguaje de marcado que será de utilidad en el diseño del Portal Web.
- JavaScript.- Es un lenguaje de scripting basado en objetos no tipado y liviano, utilizado para acceder a objetos en aplicaciones.
- Sistema operativo Linux de distribución gratuita.- Es un sistema operativo robusto, seguro y rápido que vio sus orígenes en Unix.
- Windows 2003 Server.- Es un sistema operativo de la familia Windows de la marca Microsoft para servidores que salió al mercado en el año 2003. Está basada en tecnología NT y su versión del núcleo NT es la 5.2.
- CSS.- Las hojas de estilo en cascada que permitirán definir formatos de texto e imágenes para el mejor diseño del sitio.
- WEB SERVICE.- Conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones

2.2.2 Fundamentación Teórica

2.2.2.1. Internet

Es una red formada por cientos y cientos de redes computadoras implementado en un conjunto de protocolos denominado TCP/IP y garantiza que redes físicas heterogéneas funcionen como una solo red lógica de alcance mundial. Sus orígenes se remontan a 1969, cuando se estableció la primera conexión de computadoras, conocida como ARPANET, entre tres universidades en California y una en Utah, Estados Unidos.

Servicios disponibles en Internet

- La Web (World Wide Web)

- Sistemas operativos Web (WebOS, EyeOS)
- Acceso remoto a otras máquinas (SSH y telnet)
- Transferencia de archivos (FTP)
- Correo electrónico (SMTP y POP)
- Boletines electrónicos (news o grupos de noticias)
- Conversaciones en línea (IRC y chats)
- Mensajería instantánea,
- Compartición de archivos (P2P, P2M, Descarga Directa)
- Radio a la carta (Podcast)
- Visionado de video a la carta (P2PTV, Miro, Joost, Videocast)
- Juegos en línea.

Internet y su evolución

Año tras año el Internet ofrece más servicios, dándole al cliente un abanico de posibilidades. Hoy en día el crecimiento o más bien la incorporación de tantas personas a la red hace que se conviertan en todo un planeta extremadamente conectado entre sí. El Internet se ha convertido en una fuente inagotable de oportunidades, en donde se puede disfrutar de música, videos, documentos o libros de nuestro interés. Internet se ha convertido sin lugar a duda en una de las herramientas más utilizadas por las personas.

2.2.2.2. World Wide Web

Red de redes, es un sistema de documentos de hipertexto y/o hipermedios enlazados y accesibles a través de Internet. Con un navegador Web, un usuario visualiza páginas Web que pueden contener texto, imágenes, vídeos u otros contenidos multimedia, y navega a través de ellas usando hiperenlaces. Fue creada alrededor de 1990 por el inglés Tim Berners-Lee y el belga Robert Cailliau mientras trabajaban en el CERN en Ginebra, Suiza.

Portal Web

Un portal de Internet es un sitio Web cuyo objetivo es ofrecer al usuario, de forma fácil e integrada, el acceso a una serie de recursos y de servicios, entre los que suelen encontrarse buscadores, foros, documentos, aplicaciones, compra electrónica, etc. Principalmente están dirigidos a resolver necesidades específicas de un grupo de personas o de acceso a la información y servicios de una Institución pública o privada.

Un Portal, es un sitio (localización) en la World Wide Web que contiene documentos (páginas Web) organizados jerárquicamente. Cada documento (página Web) contiene texto y o gráficos que aparecen como información digital en la pantalla de un ordenador. Un sitio puede contener una combinación de gráficos, texto, audio, vídeo, y otros materiales dinámicos o estáticos.

Características

1. Un portal Web tiene una página de inicio, que es el primer documento que ve el usuario cuando entra. Normalmente tiene otros documentos (páginas Web) adicionales. Cada sitio es administrado por una persona, una compañía o una organización.
2. Un portal Web es un medio para manipular imágenes, texto y vídeo que puede servir como medio de comunicación o para publicitar diversos negocios. Un portal Web está en una red de ordenadores y está codificado de manera que permite que los usuarios interactúen.
3. Están escritos en HTML, o dinámicamente convertidos a éste y se acceden usando un software llamado navegador Web. Los sitios Web pueden ser visualizados o accedidos desde computadoras personales, computadores portátiles, PDAs y teléfonos móviles.

4. Un portal Web está alojado en un servidor Web, y estos términos también pueden referirse al software que se ejecuta en esta computadora y que recupera y entrega las páginas de un sitio Web en respuesta a peticiones.

Modalidades

Portales horizontales.- También llamados portales masivos o de propósito general, se dirigen a una audiencia amplia, tratando de llegar a toda la gente con muchas cosas. Como ejemplo de portales de esta categoría están Terra, AOL, AltaVista, UOL, Lycos, Yahoo, MSN.

Portales verticales.- Se dirigen a usuarios para ofrecer contenido dentro de un tema específico como puede ser un portal de música, empleo, inmobiliario, un portal de finanzas personales, arte o de deportes. Los portales normalmente tienen programación que requiere muchos recursos computacionales y por su alto tráfico generalmente se hospedan en servidores de Internet dedicados.

Página Web

Una página Web es parte de un sitio Web y es un único archivo con un nombre de archivo asignado, mientras que un sitio Web es un conjunto de archivos llamados páginas Web. Una página Web es un documento HTML/XHTML accesible generalmente mediante el protocolo HTTP de Internet. A las páginas de un sitio Web se accede desde una URL raíz común llamada portada, que normalmente reside en el mismo servidor físico. Las URLs organizan las páginas en una jerarquía, aunque los hiperenlaces entre ellas controlan cómo el lector percibe la estructura general y cómo el tráfico Web fluye entre las diferentes partes.

2.2.2.3. Web Server

La Web tiene una arquitectura simple: Los usuarios navegan por Internet enviando peticiones a un servidor Web, los cuales a su vez procesan la información y devuelven una respuesta a los usuarios. Esta es, de manera muy elemental, la

función de los Servidores Web. Hoy en día, las funciones de éstos se han ampliado.

Definiciones

- Cualquier computadora que entrega Páginas Web identificada por una Dirección IP y, la mayoría de las veces, por un Nombre de Dominio. Toda computadora puede hacer la función de Web Server si se le instala el software de Web Server y se conecta a Internet.
- La máquina (hardware) que almacena Páginas Web, en cuyo caso el servidor está haciendo la función de Host.
- El software capaz de interpretar y responder peticiones de usuarios que deséen ver una página determinada.

Función del Web Server

Los primeros prototipos servían simplemente para buscar y entregar documentos e imágenes en formato HTML, entregando solo contenido estático a un navegador Web. Esta simplicidad ha dado paso formas de intercambio de información entre navegadores y Web Servers mucho más complejas gracias basicamente a dos avances: CGI Script y al HyperText, que permite la comunicación segura entre browser y Web Server, es decir, permite la transmisión encriptada de datos confidenciales. Este avance ha supuesto el desarrollo del comercio electrónico.

Rendimiento de los Web Servers

La misión de un Web Server es la de servir contenido a los clientes mediante el protocolo http. La lógica del proceso implica para el Web Server las siguientes tareas:

1. Aceptar la conexión de un browser: “escuchar”
2. Generar el contenido estático o dinámico (buscar datos)

3. Transmitirlo al browser
4. Cerrar la conexión
5. Volver a repetir el proceso y aceptar una nueva conexión

Apache Web Server

Apache, sustancialmente, es un proyecto nacido para crear un servidor de Web estable, fiable y veloz para plataformas Unix. Apache nace, por una parte, de un código ya existente y de una serie de patch para mejorar su fiabilidad y sus características; de ahí su nombre: ¡A PAtCHy sErver!.

Características

- Apache se demuestra más rápido que otros Web server libres y algunos Web Server comerciales.
- Es el Web server más difundido.
- La principal competencia de Apache es el IIS de Microsoft.
- Apache es desarrollado y mantenido por una comunidad abierta de desarrolladores bajo el auspicio de la Apache Software Foundation.
- La aplicación permite ejecutarse en múltiples sistemas operativos como Windows, Novell NetWare, Mac OS X y los sistemas basados en Unix.
- Soporte para los lenguajes Perl, Python, Tcl() y PHP().
- Soporte para SSL() y TLS().
- Permite la configuración de mensajes de errores personalizados y negociación de contenido.
- Permite autenticación de base de datos basada en SGBD.

2.2.2.4. CMS

Es un programa que permite crear una estructura de soporte (framework) para la creación y administración de contenidos por parte de los participantes principalmente en páginas Web.

Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio. El sistema permite manejar de manera independiente el contenido y el diseño. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores. Un ejemplo clásico es el de editores que cargan el contenido al sistema y otro de nivel superior que permite que estos contenidos sean visibles a todo el público.

Operación

Un sistema de administración de contenido a menudo funciona en el servidor del sitio Web. Muchos sistemas proporcionan diferentes niveles de acceso dependiendo del usuario, variando si es el administrador, editor, o creador de contenido. El acceso al CMS es generalmente vía el navegador, y a veces se requiere el uso de FTP para subir contenido, generalmente fotografías o audio.

Los creadores de contenido crean sus documentos en el sistema. Los editores comentan, aceptan o rechazan los documentos. El editor en jefe es responsable por publicar el trabajo en el sitio. El CMS controla y ayuda a manejar cada paso de este proceso, incluyendo las labores técnicas de publicar los documentos a uno o más sitios. En muchos sitios con CMS una sola persona hace el papel de creador y editor, los blogs generalmente funcionan de esta manera.

2.2.2.5. Lenguajes de informáticos

Es un lenguaje usado por, o asociado con, ordenadores. Muchas veces, este término es usado como sinónimo de lenguaje de programación, pero un lenguaje informático no tiene por qué ser un lenguaje de programación. En general, como cualquier otro lenguaje, un lenguaje de ordenador es creado cuando hay que transmitir una información de algo a alguien basado en computadora.

Clasificación de los lenguajes informáticos

Los Lenguajes informáticos pueden ser clasificados en varias clases, entre las que se incluyen las siguientes:

- Lenguaje de programación
- Lenguaje de especificación
- Lenguaje de consultas, como SQL o XQuery
- Lenguaje de marcas, como XML y otros más ligeros
- Lenguaje de transformación, como XSLT
- Protocolo de comunicaciones, como http, ftp
- Lenguaje de sonido, para crear sonidos,
- Lenguaje gráfico, para crear figuras y dibujos. Metapost,
- Pseudocódigo

Lenguajes de Programación

Es un lenguaje que puede ser utilizado para controlar el comportamiento de una máquina, particularmente una computadora. Consiste en un conjunto de reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos, respectivamente.

Un lenguaje de programación permite a un programador especificar de manera precisa: sobre qué datos una computadora debe operar, cómo deben ser estos almacenados y transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias. Todo esto, a través de un lenguaje que intenta estar relativamente próximo al lenguaje humano o natural, tal como sucede con el lenguaje Léxico.

Clasificación de los lenguajes de programación

Los lenguajes de programación se determinan según el nivel de abstracción, según la forma de ejecución y según el paradigma de programación que poseen cada uno de ellos y esos pueden ser:

1) Lenguajes de bajo nivel

Los lenguajes de bajo nivel son lenguajes de programación que se acercan al funcionamiento de una computadora. El lenguaje de más bajo nivel es, por excelencia, el código máquina. A éste le sigue el lenguaje ensamblador, ya que al programar en ensamblador se trabajan con los registros de memoria de la computadora de forma directa.

2) Lenguajes de medio nivel

Hay lenguajes de programación que son considerados por algunos expertos como lenguajes de medio nivel (como es el caso del lenguaje C) al tener ciertas características que los acercan a los lenguajes de bajo nivel pero teniendo, al mismo tiempo, ciertas cualidades que lo hacen un lenguaje más cercano al humano y, por tanto, de alto nivel.

3) Lenguajes de alto nivel

Los lenguajes de alto nivel son normalmente fáciles de aprender porque están formados por elementos de lenguajes naturales, como el inglés. En Basic, el lenguaje de alto nivel más conocido, los comandos como "IF CONTADOR = 10 THEN STOP" pueden utilizarse para pedir a la computadora que pare si CONTADOR es igual a 10. Por desgracia para muchas personas esta forma de trabajar es un poco frustrante, dado que a pesar de que las computadoras parecen comprender un lenguaje natural, lo hacen en realidad de una forma rígida y sistemática.

Lenguaje de programación PHP

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores crear aplicaciones complejas. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones.

Aunque todo en su diseño está orientado a facilitar la creación de página Web, es posible crear aplicaciones con una interfaz gráfica para el usuario. También puede ser usado desde la línea de órdenes, de la misma manera como Perl o Python pueden hacerlo. PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como UNIX (y de ese tipo, como Linux o Mac OS X) y Windows, y puede interactuar con los servidores de Web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI

Historia

Fue originalmente diseñado en Perl, con base en la escritura de un grupo de CGI binarios escritos en el lenguaje C por el programador danés-canadiense Rasmus Lerdorf en el año 1994 para mostrar su currículum vitae y guardar ciertos datos, como la cantidad de tráfico que su página Web recibía. El 8 de junio de 1995 fue publicado "Personal Home Page Tools" después de que Lerdorf lo combinara con su propio *Form Interpreter* para crear PHP/FI.

- **PHP 3**

Dos programadores israelíes del Technion, Zeev Suraski y Andi Gutmans, reescribieron el analizador sintáctico (*parser* en inglés) en el año 1997 y crearon la base del PHP3, cambiando el nombre del lenguaje a la forma actual. Inmediatamente comenzaron experimentaciones públicas de PHP3 y fue publicado oficialmente en junio del 1998.

Para 1999, Suraski y Gutmans reescribieron el código de PHP, produciendo lo que hoy se conoce como motor Zend. También fundaron Zend Technologies en Ramat Gan, Israel.

- **PHP 4**

En mayo de 2000 PHP 4 fue lanzado bajo el poder del motor Zend Engine 1.0. El día 13 de julio de 2007 se anunció la suspensión del soporte y desarrollo de la versión 4 de PHP, a pesar de lo anunciado se ha liberado una nueva versión con mejoras de seguridad, la 4.4.8 publicada el 13 de enero del 2008 y posteriormente la versión 4.4.9 publicada el 7 de agosto de 2008. Según esta noticia se dará soporte a fallos críticos hasta el 2008.

- **PHP 5**

El 13 de julio de 2004, fue lanzado PHP 5, utilizando el motor Zend Engine 2.0 (o Zend Engine 2). La versión más reciente de PHP es la 5.3.3 (22 de julio de 2010), que incluye todas las ventajas que provee el nuevo Zend Engine 2 como:

- Mejor soporte para la Programación Orientada a Objetos, que en versiones anteriores era extremadamente rudimentario.
- Mejoras de rendimiento.
- Mejor soporte para MySQL con extensión completamente reescrita.
- Mejor soporte a XML (XPath, DOM, etc.).
- Soporte nativo para SQLite.
- Iteradores de datos.
- Manejo de excepciones.
- Mejoras con la implementación con Oracle.

Aún se siguen publicando versiones de la rama 5.2.X, siendo publicada la versión 5.2.14 el 22 de Julio de 2010, aunque la mayoría son actualizaciones de seguridad

- **PHP 6**

Está previsto el lanzamiento en breve de la rama 6 de PHP. Cuando se lance esta nueva versión quedarán solo dos ramas activas en desarrollo (PHP 5 y 6), pues se abandonó el desarrollo y soporte de PHP 4 el 13 de julio de 2007.

Las diferencias que encontraremos frente a PHP 5.* son:

- Soportará Unicode.
- Limpieza de funcionalidades obsoletas como `register_globals`, `safe_mode`
- PECL y eliminación de soporte `ereg` (Coincidencia de expresiones regulares).
- Mejoras en orientación a objetos;
- Inclusión en el núcleo de XML Reader y XML Writer así como `Fileinfo`;
- Return por referencia devolverá un error;
- Se retira el soporte de las bibliotecas `FreeType1` y `GD21`.

Programación Orientada a Objetos en PHP

PHP no es en sí un lenguaje de programación orientado a objetos (POO), pero desde hace ya unas versiones se ha implementado el uso de clases y objetos. Aquí tenemos un ejemplo de una clase sencilla en PHP:

```
class Clase {
    var $Una_variable; //Creamos las variables
    var $Otra_variable;

    function Clase($var1, $var2){ //Creamos un constructor que defina las variables
"Una_variable" y "Otra_variable" dentro de la clase.
        $this->Una_variable = $var1;
        $this->Otra_variable = $var2;
    }
}
```

Ahora veremos cómo se llama a la clase previamente creada:

```
$clase = new Clase("Esta es la variable 1", "Esta es la variable 2");
```

El ejemplo anterior es válido para php4, el mismo ejemplo escrito php5, varía en el nombre del constructor y en la forma en como se declara la variable.

```
<?php

class Clase {
    public $Una_variable; //Creamos las variables para php5
    public $Otra_variable;

    function __construct($var1, $var2){ //Creamos un constructor que defina las
variables "Una_variable" y "Otra_variable" dentro de la clase.
        $this->Una_variable = $var1;
        $this->Otra_variable = $var2;
    }
}

$instancia = new Clase('rojo','blanco');

echo "El color de la variable \$Una_variable es : ". $instancia->Una_variable . "
El color de la otra variable \$Otra_variable es: " .
$instancia->Otra_variable;

?>
```

El valor public indica que la propiedad o el método están disponible para todo código que desee acceder a los mismos. PHP 5 ofrece tres niveles de visibilidad para utilizar en clases: public, private, protected. A la vez, cambia la forma en la que se declara el constructor de la clase, ya que en PHP 4 se usaba el mismo nombre de esta, y en PHP 5 se crea el método especial *__construct*

Usos de PHP

- Programación de páginas Web dinámicas, habitualmente en combinación con el motor de base de datos MySQL, aunque cuenta con soporte nativo para otros motores, incluyendo el estándar ODBC, lo que amplía en gran medida sus posibilidades de conexión.
- Programación en consola, al estilo de Perl o Shell scripting.
- Creación de aplicaciones gráficas independientes del navegador, por medio de la combinación de PHP y Qt/GTK+, lo que permite desarrollar aplicaciones de escritorio.

Ventajas

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL
- Es libre, por lo que se presenta como una alternativa de fácil acceso.
- Permite las técnicas de Programación Orientada a Objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables.
- Tiene manejo de excepciones.

Desventajas

- No posee una abstracción de base de datos estándar, sino bibliotecas especializadas para cada motor (a veces más de una para el mismo motor).
- No posee adecuado manejo de internacionalización, unicode, etc.
- Por su diseño dinámico no puede ser compilado y es muy difícil de optimizar.
- Por sus características promueve la creación de código desordenado y complejo de mantener.
- Está diseñado especialmente para un modo de hacer aplicaciones Web que es ampliamente considerado problemático y obsoleto (mezclar el código con la creación de la página Web).

Lenguaje de programación C#

Es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, que después fue aprobado como un estándar por la ECMA e ISO.

Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma.NET, similar al de Java aunque incluye mejoras derivadas de otros lenguajes (entre ellos Delphi).

La creación del nombre del lenguaje, C#, proviene de dibujar dos signos positivos encima de los dos signos positivos de "C++", queriendo dar una imagen de salto evolutivo, del mismo modo que ocurrió con el paso de C a C++.

Como parte de la plataforma.NET, está normalizado por ECMA desde diciembre de 2001. El 7 de noviembre de 2005 salió la versión 2.0 del lenguaje, que incluía mejoras tales como tipos genéricos, métodos anónimos, iteradores, tipos parciales y tipos anulables. El 19 de noviembre de 2007 salió la versión 3.0 de C#, destacando entre las mejoras los tipos implícitos, tipos anónimos y LINQ (Language Integrated Query -consulta integrada en el lenguaje).

Aunque C# forma parte de la plataforma.NET, ésta es una interfaz de programación de aplicaciones (API), mientras que C# es un lenguaje de programación independiente diseñado para generar programas sobre dicha plataforma. Ya existe un compilador implementado que provee el marco de DotGNU - Mono que genera programas para distintas plataformas como Win32, UNIX y Linux.

Tipos de Datos

C# contiene dos categorías generales de tipos de datos integrados: tipos de valor y tipos de referencia. El término tipo de valor indica que esos tipos contienen directamente sus valores.

C# define ocho tipos de enteros, a saber:

Tipo de datos de enteros			
Tipo	Ancho en bits	Rango	Significado
byte	8	De 0 a 255	Entero sin signo
sbyte	8	De -128 a 127	Entero con signo
short	16	De -32.768 a 32.767	Entero corto con signo
ushort	16	De 0 a 65.535	Entero corto sin signo
Int	32	De -2.147.483.648 a 2.147.483.647	Entero medio con signo
uint	32	De 0 a 4.294.967.295	Entero medio sin signo
long	64	De -9.223.372.036.854.775.808 a 9.223.372.036.854.775.807	Entero largo con signo
ulong	64	De 0 a 18.446.744.073.709.551.615	Entero largo sin signo

TABLA 2.1 TIPOS DE DATOS C#

Los tipos de punto flotante pueden representar números con componentes fraccionales. Existen dos clases de tipos de punto flotante; float y double. El tipo double es el más utilizado porque muchas funciones matemáticas de la biblioteca de clases de C# usan valores double. Quizá, el tipo flotante más interesante de C# es decimal, dirigido al uso de cálculos monetarios. La aritmética de punto flotante normal está sujeta a una variedad de errores de redondeo cuando se aplica a valores decimales. El tipo decimal elimina estos errores y puede representar hasta 28 lugares decimales.

Tipo de datos de punto flotante			
Tipo	Ancho en bits	Rango	Significado

float	32	De 1,5E-45 a 3,4E+38	Punto flotante corto
double	64	De 5E-324 a 1,7E+308	Punto flotante largo
decimal	128	De 1E-28 a 7,9E+28	Punto flotante monetario

TABLA 2.2 TIPOS DE DATOS FLOTANTE C#

Los caracteres en C# no son cantidades de 8 bits como en otros muchos lenguajes de programación. Por el contrario, C# usa un tipo de caracteres de 16 bits llamado Unicode al cual se le llama char. No existen conversiones automáticas de tipo entero a char.

Tipo de datos de caracteres			
Tipo	Ancho en bits	Rango	Significado
char	16	De 0 a 65,535 (código Unicode)	Carácter

TABLA 2.3 TIPOS DE DATOS CARÁCTER C#

Tipo de datos lógicos			
Tip	Ancho en bits	Rango	Significado
bool	1	true or false, no se usa 1 ó 0 ya que no hay conversión definida	true or false

TABLA 2.4 TIPOS DE DATOS LÓGICOS C#

No existe una conversión definida entre bool y los valores enteros (1 no se convierte a verdadero ni 0 se convierte a falso).

C# tiene cuatro clases generales de operadores: aritméticos, a nivel de bit, relacionales y lógicos.

Operadores		
Operador	Significado	Tipo
+	Suma	Aritmético
-	Resta	Aritmético
*	Producto	Aritmético
/	División	Aritmético
%	Módulo (residuo entero)	Aritmético
++	Incremento	Aritmético

--	Decremento	Aritmético
==	Igual que	Relacional
!=	Distinto que	Relacional
>	Mayor que	Relacional
<	Menor que	Relacional
>=	Mayor o igual que	Relacional
<=	Menor o igual que	Relacional
&	AND	Lógico y a nivel de bits
	OR	Lógico y a nivel de bits
^	XOR y de nivel de bits	Lógico
	OR de cortocircuito	Lógico
&&	AND de cortocircuito	Lógico
!	NOT	Lógico
~	Complemento a uno	A nivel de bits
<<	Desplazamiento a la izquierda	A nivel de bits
>>	Desplazamiento a la derecha	A nivel de bits

TABLA 2. 5 OPERADORES C#

- Los operadores aritméticos funcionan igual que en C y C++.
- El resultado de los operadores relacionales y lógicos es un valor bool.
- Los operadores de cortocircuito evalúan el segundo operando solo cuando es necesario.
- Los operadores a nivel de bit no se pueden aplicar a tipos bool, float, double o decimal.

Instrucciones de control

- La instrucción if-else es básicamente igual que en C, C++ y Java.
- La diferencia de la instrucción switch con la versión de C, C++ y Java es que todo cuerpo perteneciente a un case debe de toparse con un break o un goto antes de toparse con otro case, a menos que dicho cuerpo esté vacío.
- La instrucción for es básicamente igual que en C, C++ y Java.
- La instrucción while es básicamente igual que en C, C++ y Java.
- La instrucción do-while es básicamente igual que en C, C++ y Java.
- La instrucción foreach realiza un ciclo a través de los elementos de una colección (grupo de objetos). El formato de esta instrucción es: foreach(tipo variable in coleccion) instruccion;. En este ciclo se recorre la

colección y la variable recibe un respectivo elemento de dicha colección en cada iteración.

- Al igual que en C y C++, la instrucción `break` permite forzar la salida de un ciclo omitiendo el código restante en el cuerpo del ciclo.
- Al igual que en C y C++, la instrucción `continue` permite forzar la repetición temprana de un ciclo omitiendo el código restante en el cuerpo del ciclo.
- La instrucción `return` es básicamente igual que en C, C++. Se utiliza para devolver un valor y salir de un método.
- La instrucción `goto` se sigue utilizando en C+ a pesar de toda la polémica que esto conlleva.

Lenguaje de marcado

Un lenguaje de marcado o lenguaje de marcas es una forma de codificar un documento que, junto con el texto, incorpora etiquetas o marcas que contienen información adicional acerca de la estructura del texto o su presentación. Html es el lenguaje de marcado mas conocido. El marcado se usaba y se usa en la industria editorial y de la comunicación, así como entre autores, editores e impresores.

Clases de lenguajes de marcado

1. Marcado de presentación

El marcado de presentación es aquel que indica el formato del texto. El marcado de presentación es muy útil para el diseño de una página Web, pero no sirve para el procesamiento automático de la información. El marcado de presentación resulta más fácil de elaborar, sobre todo para cantidades pequeñas de información. Sin embargo resulta complicado de mantener o modificar.

2. Marcado de procedimientos

El marcado de procedimientos está enfocado hacia la presentación del texto, sin embargo, también es visible para el usuario que edita el texto. El programa que representa el documento debe interpretar el código en el mismo orden en que aparece.

Por ejemplo, para formatear un título, debe haber una serie de directivas inmediatamente antes del texto en cuestión, indicándole al software instrucciones tales como centrar, aumentar el tamaño de la fuente, o cambiar a negrita. Inmediatamente después del título deberá haber etiquetas inversas que reviertan estos efectos. En sistemas más avanzados se utilizan macros o pilas que facilitan el trabajo.

Este tipo de marcado se ha usado extensivamente en aplicaciones de edición profesional, manipulados por tipógrafos cualificados, ya que puede llegar a ser extremadamente complejo.

3. Marcado descriptivo

El marcado descriptivo o semántico utiliza etiquetas para describir los fragmentos de texto, pero sin especificar cómo deben ser representados, o en que orden. Los lenguajes expresamente diseñados para generar marcado descriptivo son el SGML y el XML.

Una de las virtudes del marcado descriptivo es su flexibilidad: los fragmentos de texto se etiquetan tal como son, y no tal como deben aparecer. Estos fragmentos pueden utilizarse para más usos de los previstos inicialmente. Por ejemplo, los hiper enlaces fueron diseñados en un principio para que un usuario que lee el texto los pulse. Sin embargo, los buscadores los emplean para localizar nuevas páginas con información relacionada, o para evaluar la popularidad de determinado sitio

Web. El marcado descriptivo también simplifica la tarea de reformatear un texto, debido a que la información del formato está separada del propio contenido. Por ejemplo, un fragmento indicado como cursiva (<i>texto</i>), puede emplearse para marcar énfasis o bien para señalar palabras en otro idioma.

Lenguaje de marcado HTML

HTML es un lenguaje que hace posible presentar información en Internet. Lo que se visualiza en una página en Internet es la interpretación que hace el navegador del código HTML.

HTML es la abreviatura de "HyperText Mark-up Language", "Lenguaje de marcado hipertextual". Fue inventado por un científico llamado Tim Berners-Lee allá por 1990. El objetivo era facilitar a científicos de diferentes universidades el acceso a los documentos de investigación de cada uno de ellos.

Usos de HTML

Necesario para la creación de sitios Web, incluso si se utiliza una herramienta de diseño como Dreamweaver, se necesita un conocimiento básico de HTML. HTML es fácil de aprender y de usar, no se requiere de gran conocimiento.

2.2.2.6. Web Service

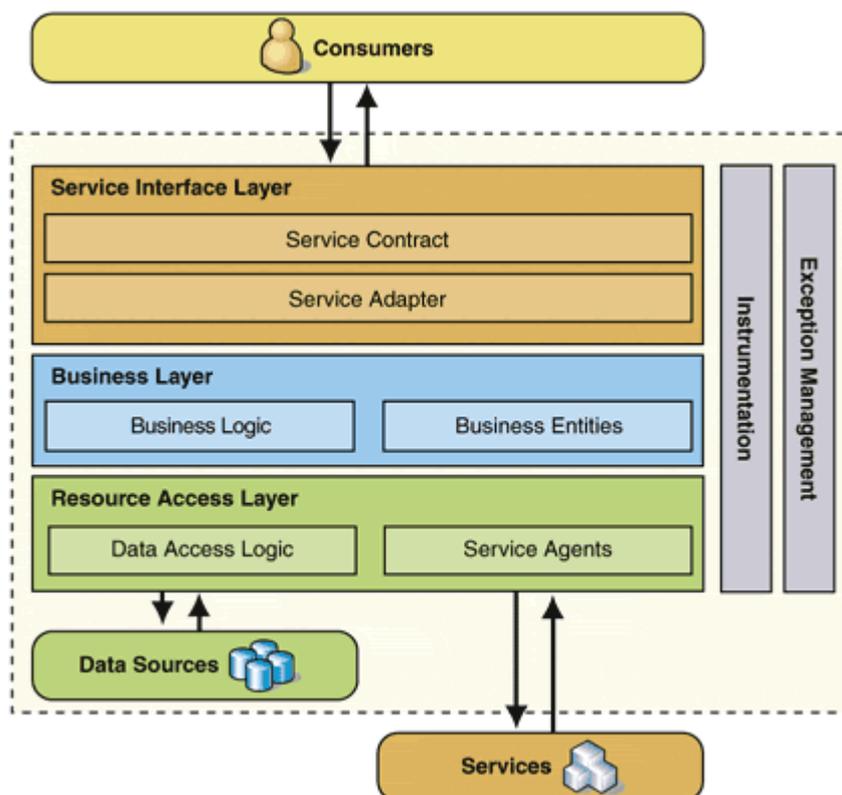


Figura 2.1.- Estructura de Funcionamiento Web Service

Web Service(WS).-Un servicio Web es un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios Web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios Web.

Estándares empleados

- Web Services Protocol Stack: Así se denomina al conjunto de servicios y protocolos de los servicios Web.
- XML (Extensible Markup Language): Es el formato estándar para los datos que se vayan a intercambiar.
- SOAP (Simple Object Access Protocol) o XML-RPC (XML Remote Procedure Call): Protocolos sobre los que se establece el intercambio.
- Otros protocolos: los datos en XML también pueden enviarse de una aplicación a otra mediante protocolos normales como HTTP (Hypertext Transfer Protocol), FTP (File Transfer Protocol), o SMTP (Simple Mail Transfer Protocol).
- WSDL (Web Services Description Language): Es el lenguaje de la interfaz pública para los servicios Web. Es una descripción basada en XML de los requisitos funcionales necesarios para establecer una comunicación con los servicios Web.
- UDDI (Universal Description, Discovery and Integration): Protocolo para publicar la información de los servicios Web. Permite comprobar qué servicios Web están disponibles.
- WS-Security (Web Service Security): Protocolo de seguridad aceptado como estándar por OASIS (Organization for the Advancement of Structured Information Standards). Garantiza la autenticación de los actores y la confidencialidad de los mensajes enviados.

Ventajas de los WS

Aportan interoperabilidad entre aplicaciones de software independientemente de sus propiedades o de las plataformas sobre las que se instalen.

Los servicios Web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento.

Al apoyarse en HTTP, los servicios Web pueden aprovecharse de los sistemas de seguridad firewall sin necesidad de cambiar las reglas de filtrado.

Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.

Permiten la interoperabilidad entre plataformas de distintos fabricantes por medio de protocolos estándar y abiertos. Las especificaciones son gestionadas por una organización abierta, la W3C, por tanto no hay secretismos por intereses particulares de fabricantes concretos y se garantiza la plena interoperabilidad entre aplicaciones.

Inconvenientes de los WS

Para realizar transacciones no pueden compararse en su grado de desarrollo con los estándares abiertos de computación distribuida como CORBA (Common Object Request Broker Architecture).

Su rendimiento es bajo si se compara con otros modelos de computación distribuida, tales como RMI (Remote Method Invocation), CORBA o DCOM (Distributed Component Object Model). Es uno de los inconvenientes derivados de adoptar un formato basado en texto. Y es que entre los objetivos de XML no se encuentra la concisión ni la eficacia de procesamiento.

Al apoyarse en HTTP, pueden esquivar medidas de seguridad basadas en firewall cuyas reglas tratan de bloquear o auditar la comunicación entre programas a ambos lados de la barrera.

Razones para crear WS

La principal razón para usar servicios Web es que se basan en HTTP sobre TCP (Transmission Control Protocol) en el puerto 80. Dado que las organizaciones protegen sus redes mediante firewalls -que filtran y bloquean gran parte del tráfico de Internet, cierran casi todos los puertos TCP salvo el 80, que es, precisamente, el que usan los navegadores. Los servicios Web utilizan este puerto, por la simple razón de que no resultan bloqueados.

Otra razón es que, antes de que existiera SOAP, no había buenas interfaces para acceder a las funcionalidades de otros ordenadores en red. Las que había eran ad hoc y poco conocidas, tales como EDI (Electronic Data Interchange), RPC (Remote Procedure Call), u otras APIs.

Una tercera razón por la que los servicios Web son muy prácticos es que pueden aportar gran independencia entre la aplicación que usa el servicio Web y el propio servicio. De esta forma, los cambios a lo largo del tiempo en uno no deben afectar al otro. Esta flexibilidad será cada vez más importante, dado que la tendencia a construir grandes aplicaciones a partir de componentes distribuidos más pequeños es cada día más utilizada.

Plataformas

Servidores de aplicaciones para servicios Web:

- JBoss servidor de aplicaciones J2EE Open Source de Red Hat inc.
- Oracle Fusion Middleware
- IBM Lotus Domino a partir de la versión 7.0
- Axis y el servidor Jakarta Tomcat (de Apache)
- Mono develop (Apache)
- ColdFusion MX de Macromediahttpd.
- Java Web Services Development Pack (JWSDP) de Sun Microsystems (basado en Jakarta Tomcat)

- JOnAS (parte de *ObjectWeb* una iniciativa de código abierto)
- Microsoft .NET
- Novell exteNd (basado en la plataforma J2EE)
- WebLogic
- WebSphere
- JAX-WS con GlassFish
- Zope es un servidor de aplicaciones Web orientado a objetos desarrollado en el lenguaje de programación Python
- VERASTREAM de AttachmateWRQ para modernizar o integrar aplicaciones host IBM y VT

2.2.2.7. Base de Datos

Base de datos.- Una base de datos o banco de datos es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta.

Ventajas de las bases de datos

- Reducción de espacio físico
- Rapidez de acceso a la información
- Reducción de duplicidad de la información
- Reducción de tiempo para transacciones
- Respaldo de la información
- Se mantiene la integridad de los datos por que estos se mantienen en un dispositivo magnético.
- Puede servir a múltiples usuarios
- Almacena mayor cantidad de información en menos espacio.
- Actualización constante de la información

Desventajas de las bases de datos

- Dependencia total de la base de datos.
- Se puede perder la información sino se respalda.

Elementos de una base de datos

Tablas.-Las bases de datos están conformadas por bloques de información (que pueden ser uno o más) denominados tablas con características en común. Una tabla es un conjunto de información del mismo tipo.

Registros.- Una tabla, a su vez está conformada por registros. Se denomina registro a la unidad elemental de información de una tabla.

Atributos.- Un registro está formado por elementos llamados atributos o campos. Un atributo o campo es cada una de las informaciones que interesa almacenar en cada registro.

Tipos de bases de datos

- **Bases de datos estáticas**

Éstas son bases de datos de sólo lectura, utilizadas primordialmente para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos a través del tiempo, realizar proyecciones y tomar decisiones.

- **Bases de datos dinámicas**

Éstas son bases de datos donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualización y adición de datos, además

de las operaciones fundamentales de consulta. Un ejemplo de esto puede ser la base de datos utilizada en un sistema de información de una tienda de abarrotes, una farmacia, un videoclub, etc.

Sistema de Gestión de Base de Datos

Un Sistema de Gestión de Base de Datos (SGBD) es un conjunto coordinado de programas, procedimientos, lenguajes, herramientas, etc., que suministra, tanto a los usuarios no informáticos como a los analistas, programadores o administradores de una BD., lo medios necesarios para describir y manipular los datos integrados en la BD, manteniendo su integridad, confidencialidad y disponibilidad

SGBD MySQL

MySQL es la base de datos open source más popular y, posiblemente, mejor del mundo. MySQL es un sistema de gestión de bases de datos relacional, de distribución libre. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente. MySQL fue creada por la empresa sueca MySQL AB, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca.

Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración.

Características de MySQL

- Aprovecha la potencia de sistemas multiprocesador.

- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc).
- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y passwords, manteniendo un buen nivel de seguridad.

2.2.2.8. Sistema operativo

Un sistema operativo es un conjunto de programas destinados a permitir la comunicación del usuario con un computador y gestionar sus recursos de una forma eficaz. Comienza a trabajar cuando se enciende el computador, y gestiona el hardware de la máquina desde los niveles más básicos.

Componentes

Los sistemas operativos considerados como un programa, han alcanzado un tamaño muy grande, debido a que tienen que hacer muchas tareas. Por esta razón para construir un SO es conveniente dividirlo en componentes más pequeños.

- Gestión de procesos.
- Gestión de memoria.
- Gestión de archivos y directorios.
- Gestión de la E/S (Entrada/Salida).
- Seguridad y protección.
- Comunicación y sincronización entre procesos.
- Intérprete de órdenes

Sistema Operativo Linux

Es una implementación de libre distribución UNIX para computadoras personales (PC), servidores, y estaciones de trabajo. Linux es muy eficiente y tiene un excelente diseño. Es multitarea, multiusuario, multiplataforma y multiprocesador.

Características de Linux

- **Multitarea:** La habilidad de ejecutar varios programas al mismo tiempo. LINUX utiliza la llamada multitarea preventiva, la cual asegura que todos los programas que se están utilizando en un momento dado serán ejecutados.
- **Multiusuario:** Muchos usuarios usando la misma maquina al mismo tiempo.
- **Multiplataforma:** Las plataformas en las que en un principio se puede utilizar Linux son 386-, 486-, Pentium, Pentium Pro, Pentium II, Amiga y Atari, también existen versiones para su utilización en otras plataformas.
- **Multiprocesador:** Soporte para sistemas con mas de un procesador esta disponible para Intel y SPARC.
- **Protección de la memoria entre procesos,** de manera que uno de ellos no pueda colgar el sistema.
- **Carga de ejecutables por demanda:** Linux sólo lee del disco aquellas partes de un programa que están siendo usadas actualmente.
- **Política de copia en escritura para la comparación de páginas entre ejecutables:** esto significa que varios procesos pueden usar la misma zona de memoria para ejecutarse.
- **Librerías compartidas de carga dinámica (DLL's) y librerías estáticas.**
- **Se realizan volcados de estado (core dumps) para posibilitar los análisis post-mortem,** permitiendo el uso de depuradores sobre los programas no sólo en ejecución sino también tras abortar éstos por cualquier motivo.
- **Todo el código fuente está disponible, incluyendo el núcleo completo y todos los drivers, las herramientas de desarrollo y todos los programas de usuario; además todo ello se puede distribuir libremente.** Hay algunos

programas comerciales que están siendo ofrecidos para Linux actualmente sin código fuente, pero todo lo que ha sido gratuito sigue siendo gratuito.

- TCP/IP, incluyendo ftp, telnet, NFS, etc.
- Software cliente y servidor Netware.
- Diversos protocolos de red incluidos en el kernel: TCP, IPv4, IPv6, AX.25, X.25, IPX, DDP, Netrom, etc.

Sistema Operativo Windows

Windows es el sistema operativo de la compañía Microsoft que fue lanzado al mercado a fines de 1985 con su versión 1.0, como una aplicación para utilizar con el sistema MS-DOS. Desde ese entonces hasta el momento, sus diversas ediciones fueron ganando popularidad hasta convertirse en lo que es hoy: el sistema operativo más utilizado a nivel mundial.

Con casi 20 versiones diferentes desde su aparición, la primera que realmente obtuvo una aceptación comercial fue la 3.0, de 1990, aunque recién para 1995 se la comenzó a considerar realmente un sistema operativo, con su Windows 95.

Clasificación

Microsoft ha tomado dos caminos paralelos en el desarrollo de sus sistemas operativos: una línea para el hogar y una línea para empresas. Estas líneas se fusionaron en Windows XP, pero este incluye ediciones para el hogar, empresa, etc.

La primera versión de Windows fue la 1.0, lanzada en noviembre de 1985, carecía de funcionalidades y consiguió un poco de popularidad. No era un sistema operativo en sí mismo, sino que era una extensión del sistema MS-DOS.

La versión 2.0 de Windows fue lanzada en noviembre de 1987, y fue un poco más popular que su predecesora. La versión 2.03, lanzada en enero de 1988, cambió su

interfaz de ventanas, haciéndose muy parecido al sistema operativo de las Apple, y por esta razón esta empresa alegó que Microsoft infringía sus derechos.

La versión 3.0 de Microsoft Windows fue lanzada en 1990, fue la primera versión en alcanzar éxito comercial, llegando a vender dos millones de copias en sus primeros seis meses. Tuvo mejoras en la interfaz de usuario y las capacidades de multitarea. Hubo un pequeño cambio en su versión 3.1 lanzada el 1ero de marzo de 1992 (el soporte para Windows 3.1 terminó el 31 de diciembre de 2001).

En julio de 1993, Microsoft lanzó Windows NT basado en un nuevo kernel. NT es considerado de la línea profesional de los sistemas operativos Windows. Ambas líneas fueron fusionadas más tarde en Windows XP.

En agosto de 1995, Microsoft lanzó Windows 95, que es considerado realmente un sistema operativo (aunque seguía dependiendo de MS-DOS, por lo tanto algunos no lo consideran como tal). Windows 95 cambió completamente su interfaz y se hizo más fácil de usar, además fue acompañada de una gran campaña publicitaria pocas veces vista hasta el día de hoy en este tipo de lanzamientos.

En junio de 1998, se lanzó Microsoft Windows 98, el cual necesitó una Second Edition en 1999 para resolver severos problemas de seguridad.

En febrero de 2000, se lanzó Windows 2000, que pertenecía a la línea profesional sistemas de Microsoft.

En tanto en la línea de hogar a Windows 98 le siguió Windows ME (Millennium Edition), lanzado en septiembre de 2000. Este sistema implementó varias nuevas tecnologías como Plug&Play universal. De todas maneras este sistema operativo fue sumamente criticado por sus debilidades en la estabilidad y compatibilidad.

En octubre de 2001, Microsoft lanza Windows XP, versión basada en el kernel de Windows NT que también tenía las características de la línea de hogar. Esta

versión fue sumamente elogiada en revistas de computación. Poseía dos ediciones "Home" y "Professional". De todas maneras necesitó dos Service Pack de actualización para lograr una seguridad robusta. El soporte para Windows XP terminará el 14 de abril de 2009, y el soporte extendido el 8 de abril de 2014.

En abril de 2003, fue lanzado Windows Server 2003 en reemplazo de la línea de productos para servidores Windows 2000. Esta versión poseía muchas mejoras y una fuerte seguridad. Fue seguido de Windows 2003 R2 en diciembre de 2005.

La siguiente versión del sistema operativo fue Windows Vista, lanzada el 30 de noviembre de 2006 para clientes de negocios. En cambio la versión para consumidores fue lanzada el 30 de enero de 2007. Windows Vista trajo mejoras en la seguridad, características más avanzadas en sus gráficos con una interfaz opcional llamada Windows Aero, además de múltiples nuevas aplicaciones. Sus ediciones de empresa son Windows Vista Business y Enterprise, y las del hogar son Home Basic y Home Premium. La versión con características de empresas y de hogar es Windows Vista Ultimate.

Windows 2003 Server

Windows Server 2003 es un sistema operativo desarrollado por Microsoft, lanzado el 24 de abril de 2003 como sucesor de Windows 2000 Server. Es más escalable y posee un mejor rendimiento que su predecesor.

Sus ediciones son: Standard, Enterprise, Datacenter, Web y XP Pro x64.

Su sistema de archivos es NTFS, que permite cuotas, encriptación, compresión de archivos, carpetas y unidades completas. Permite montar dispositivos de almacenamiento sobre sistemas de archivos de otros dispositivos al estilo Linux, Active Directory, etc.

Servicios o Servidores

Los servidores que maneja Windows 2003 son:

- Servidor de archivos.
- Servidor de impresiones.
- Servidor de aplicaciones.
- Servidor de correo (SMTP/POP).
- Servidor de terminal.
- Servidor de Redes privadas virtuales (VPN) (o acceso remoto al servidor).
- Controlador de Dominios (mediante Active Directory).
- Servidor DNS.
- Servidor DHCP.
- Servidor de Streaming de Video.
- Servidor WINS.

Ventajas

- Infraestructura segura.
- Fácil de implementar y usar.
- Confiabilidad, disponibilidad, escalabilidad y rendimiento de nivel empresarial.
- Creación fácil de sitios Web de Internet e intranet dinámicos.
- Desarrollo rápido con el servidor de aplicaciones integrado.
- Servicios Web XML fáciles de encontrar, compartir y reutilizar.
- Herramientas de administración sólidas.
- Potenciación de los empleados disminuyendo los costes de soporte técnico.

Desventajas

Como es de esperarse en la familia Microsoft el costo de las licencias. Por otra parte, a pesar de la publicidad que pueda hacerle Microsoft a su sistema de servidor, este es uno de los Servers con más bugs o fallos de seguridad conocidos; probablemente por ser la familia de sistemas operativos más popular en el mundo. En cuestiones de manejo e instalación, es un sistema que requiere altísimos conocimientos para poderle configurar a tope, también contrario a lo que afirma Microsoft con su facilidad de adaptación, es decir que aparte de comprar un sistema servidor de alto costo, el usuario debe pagar por un personal altamente especializado, lo cual desde luego no es tan económico.

2.2.3. Lenguaje Organizacional

- **La Banca Online:** Banca en línea, *e-banking* o genéricamente Banca electrónica, es la banca a la que se puede acceder mediante Internet. Pueden ser entidades con sucursales físicas o que sólo operan por Internet o por teléfono.
- **Cuenta:** Dinero depositado en una Institución financiera con propósitos de inversión y/o seguridad.
- **Cuenta Corriente:** Contrato entre dos personas que convienen que el saldo solo sea exigible al finalizar aquel. Cuenta corriente bancaria, cuenta abierta por un banquero a sus clientes alimentados por los depósitos de cada titular.
- **Cuenta de Ahorros:** Es un depósito ordinario a la vista (producto pasivo), en la que los fondos depositados por el cuentahabiente tienen disponibilidad inmediata y le generan cierta rentabilidad o intereses durante un periodo determinado según el monto ahorrado. Las condiciones de remuneración varían en función del producto concreto. Así, pueden existir cuentas de ahorro que abonen los intereses por ejemplo mensual, trimestral o anualmente. Igualmente, la remuneración puede ser

lineal (mismo interés para cualquier saldo), por tramos de saldo o incluso en especie.

Este tipo de cuentas también pueden estar sujetas al pago de comisiones, aunque desde la popularización de la Banca Online la tendencia es a suprimir el pago de comisiones en los productos de ahorro.

Los servicios asociados a una cuenta de ahorro también varían según la entidad. Pueden ir desde los más básicos, como ingresos y pagos de cheques o transferencias, hasta otros más complejos como tarjetas de débito o crédito asociadas a la cuenta.

Las cuentas de ahorro o caja de ahorro son utilizadas principalmente por los particulares y entidades civiles.

Sirven para tres cosas:

- Para ganar un interés durante el tiempo que estos fondos no se usan.
 - Los fondos pueden extraerse en cualquier momento mediante un comprobante llamado “nota de débito”.
 - Los fondos depositados ganan un interés moderado. También debe tenerse en cuenta que se suman al capital cada 6 meses o anualmente.
- **Saldo:** En términos bancarios, el saldo se refiere a la cantidad de dinero que hay en una cuenta particular. En términos de crédito, el saldo se refiere a la cantidad que se debe.
 - **Créditos:** Contrato por el cual una entidad financiera pone a disposición del cliente cierta cantidad de dinero, que este deberá de devolver con intereses y según los plazos pactados. A diferencia del préstamo, ese capital no ha de tener necesariamente una finalidad prefija (como la compra de un inmueble en el caso de un préstamo hipotecario).

- **Seguridad:** Son medidas generalmente de seguridad que un banco en internet adopta son un aspecto en el que no se debe escatimar esfuerzos, ni recursos, dada la especial naturaleza del negocio. Es un factor fundamental si se quiere cimentar una relación de confianza con los clientes, seriamente preocupados por su intimidad y la seguridad de su dinero.
- **Atención al Cliente:** Servicio que proporciona una empresa prestadora de un servicio o de un producto, para resolver dudas o resolver problemas que pueden surgir en un producto. Se puede contactar por teléfono, correo electrónico o a veces incluso por chat.
- **Mercado:** Mecanismo que permite a los agentes económicos el intercambio de activos financieros. En general, cualquier mercado de materias primas podría ser considerado como un mercado financiero si el propósito del comprador no es el consumo inmediato del producto, sino el retraso del consumo en el tiempo.

Los mercados financieros están afectados por las fuerzas de oferta y demanda. Los mercados colocan a todos los vendedores en el mismo lugar, haciendo así más fácil encontrar posibles compradores. A la economía que confía ante todo en la interacción entre compradores y vendedores para destinar los recursos se le llama economía de mercado, en contraste con la economía planificada.

- **Comercio electrónico:** También conocido como e-commerce (electronic commerce en inglés), consiste en la compra y venta de productos o de servicios a través de medios electrónicos, tales como Internet y otras redes informáticas. Originalmente el término se aplicaba a la realización de transacciones mediante medios electrónicos tales como el Intercambio electrónico de datos, sin embargo con el advenimiento de la Internet y la World Wide Web a mediados de los años 90 comenzó a referirse principalmente a la venta de bienes y servicios a través de Internet, usando

como forma de pago medios electrónicos, tales como las tarjetas de crédito.

La cantidad de comercio llevada a cabo electrónicamente ha crecido extraordinariamente debido a la propagación de Internet. Una gran variedad de comercio se realiza de esta manera, estimulando la creación y utilización de innovaciones como la transferencia de fondos electrónica, la administración de cadenas de suministro, el marketing en Internet, el procesamiento de transacciones en línea, el intercambio electrónico de datos (EDI), los sistemas de administración del inventario, y los sistemas automatizados de recolección de datos.

La mayor parte del comercio electrónico consiste en la compra y venta de productos o servicios entre personas y empresas, sin embargo un porcentaje considerable del comercio electrónico consiste en la adquisición de artículos virtuales (software y derivados en su mayoría), tales como el acceso a contenido "premium" de un sitio Web.

- **Online:** en línea, estado conectado en que se encuentra una computadora cuando se conecta directamente con la red a través de un dispositivo, por ejemplo, un modem.
- **Beneficio del Lugar:** Se refiere al hecho de llevar un producto cerca del consumidor para que este no tenga que recorrer grandes distancias para obtenerlo y satisfacer así una necesidad.
- **El beneficio de tiempo:** Es consecuencia del anterior ya que si no existe el beneficio del lugar, tampoco este puede darse. Consiste en llevar un producto al consumidor al momento más adecuado.
- **Confianza:** Es un elemento básico en la relación del banco y su cliente, es la esencia de esta. Tomando en cuenta que el cliente le está confiando al banco una de sus posesiones más apreciadas: su dinero. Esta afirmación adquiere mayor valor en el caso de la banca por Internet, dado que los

servicios financieros en línea dependen aún más de la confianza de los usuarios. No solo se trata de la credibilidad de las propias entidades y sus productos, sino las estrategias de comunicación de la entidad utilice para justificar al cliente su decisión de salir a Internet, a prestar sus servicios a través de una red LAN pública como popular. Un lugar donde todo el mundo puede acceder: los que tienen las mejores intenciones, pero también los que tienen las peores.

- **Claves:** Conocidas como clave personal, PIN (del inglés, personal identification number, número de identificación personal) o clave secreta. Cuando se accede al banco en Internet, se requiere un código de usuario y una contraseña, que se otorgan al contratar el servicio. En muchos casos, como medida adicional de seguridad al tercer intento consecutivo erróneo el usuario es expulsado y deberá notificarlo a la entidad para la reactivación del servicio.
- **Servidores Seguros:** Establece una conexión con el cliente de manera que la información circula a través del Internet escrutada, es decir codificada, lo que asegura que sea inteligible solo para el servidor y el navegador que accede a la página Web, entendiéndose ambos mediante un protocolo especial de comunicación. De este modo, ninguna persona externa, que eventualmente estuviera espionando la transmisión de la información, podrá descifrar los datos confidenciales mientras viajan hacia y desde la red del banco.
- **Accesibilidad:** El surgimiento de los servicios de la banca a distancia, personalizados en sus primeros pasos en banca telefónica, ha traído como consecuencia un cambio del tipo de relación entre el cliente y el banco, tradicionalmente basado en la inevitable presencia física del cliente en la sucursal.
Se pueden realizar gestiones desde cualquier parte y no se requiere avanzados conocimientos de informática, pues los sistemas están

elaborados en función a la intuición de los usuarios y la familiaridad con los mismos se obtiene en poco tiempo.

Componentes de la plataforma de atención al cliente: Las expectativas de los clientes han cambiado drásticamente. El cliente reclama un servicio e interacción más rápida y personalizada, un mayor y más fácil acceso a información.

- **Herramientas de atención al cliente:** La figura tradicional del ejecutivo de oficina bancaria, que conocía la situación financiera, e incluso personal, de sus clientes tiende a desaparecer con la generalización de la banca electrónica. Ahora el cliente bancario se relaciona con su banco a través de un teclado y una pantalla y eso va acompañado de la sensación de frialdad e impersonalidad que trasmite la red. Una correcta gestión de la atención al cliente puede hacer que el usuario sienta que hay alguien al otro lado. Por eso las guías generales de excelencia en la atención y servicio al cliente siguen vigentes.

Además una correcta atención al cliente en línea, personalizada y de calidad, ofrece la oportunidad de marcar diferencias con la competencia, crear ventajas competitivas, establecer barrera de salida a los clientes y en definitiva, posicionarse como líder.

- **Herramientas de información y promoción:** Son aplicaciones incorporadas al contenido del sitio que permiten al usuario obtener información sobre productos o servicios ofrecidos por el banco.
- **Ventaja Competitiva:** Diferencia positiva de una empresa con respecto a la competencia.
- **Publicidad:** Es uno de los elementos que forma parte del Marketing y se puede definir como “Actividades que intervienen en la presentación de un mensaje pagado, no personal de un patrocinador identificado respecto a

una organización y a sus productos, servicios o ideas “. De hecho la publicidad obedece a la necesidad de llegar a un público bastante amplio.

- **Formulario Bancario:** Se llama formulario a una plantilla o página con espacios vacíos que han de ser rellenos con alguna finalidad o servicio, por ejemplo una solicitud de préstamo en la que has de rellenar los espacios libres con la información personal requerida.

Es un conjunto de objetos y procedimientos que se utilizan como diseño de presentación al usuario.

Todos los solicitados por un determinado programa, los cuales se almacenarán para su posterior uso o manipulación.

En la Web, servicio que permite introducir información a enviar a la ubicación Web remota para procesarla.

2.2.3.1. Fundamentación de la Empresa

Nace el 17 de Enero del 2003, y con él la Cooperativa de Ahorro y Crédito "CHIBULEO". Por la constancia de 27 jóvenes, aquellos que, no teniendo nada, juntaron sus ideas y pensamientos en la búsqueda de un firme rumbo que aliviara sus penumbras y tormentos en difíciles momentos en que se veían rodeados las clases menos privilegiadas de nuestro país.

Con el paso de los años la Cooperativa de Ahorro y Crédito "CHIBULEO" está escribiendo una historia de éxito, ha logrado posicionarse en el sistema financiero como una Cooperativa de demostrada capacidad de crecimiento e innovadora, que trabaja por un futuro mejor para nuestra gente, 65 mil socios, 7 sucursales: Quito, Riobamba, Salcedo, Machachi, Latacunga y su principal en Ambato, 12 de Noviembre y Mariano Eguez

Este sitio de honor se ha obtenido gracias al respaldo que se ha recibido de todos los socios que con gran orgullo, lealtad y confianza continúan apoyando esta Cooperativa.

OBJETIVOS

- Llegar a toda las personas a través de promotores de la empresa para así poder dar a conocer los servicios que brinda la Institución, lo que nos permitirá conocer inquietudes de personas que necesitan de una Institución que pueda ayudarlos.
- Todo esto solo se podrá realizar con la construcción de alianzas, la colaboración y la voluntad firme de los representantes de las distintas empresas que será una de las alternativas para el cumplimiento de los objetivos propuestos
- Ejecutar proceso de participación del sector juvenil para la inserción laboral.
- Fortalecer las diferentes actividades a través de créditos oportunos.
- Capacitar a los socios en temas; valor, creación de microempresas y otros.

MISIÓN

- Con vocación social brindar servicios financieros y complementarios de calidad para satisfacer las expectativas de socios y clientes, contribuyendo al desarrollo de la comunidad.

VISIÓN

- Con vocación social brindar servicios financieros y complementarios de calidad para satisfacer las expectativas de socios y clientes, contribuyendo al desarrollo de la comunidad.

VALORES INSTITUCIONALES

- Ética.
- Honestidad.
- Solidaridad.
- Comunicación efectiva.
- Equidad.
- Liderazgo efectivo.
- Trabajo en equipo.
- Creatividad e innovación.
- Mejoramiento continuo.

2.3 Variables

2.3.1 Variable Independiente

Módulos Online mediante el uso de Web Service.

2.3.2 Variable dependiente

Ingreso de formularios de préstamos, consulta de: saldos, depósitos, pagos de créditos e ingreso de peticiones de certificados en la Cooperativa de ahorro y Crédito “CHIBULEO” Ltda., de la ciudad de Ambato Provincia de Tungurahua.

2.4 Hipótesis

El módulo Online mediante Web Service permite un mejor y rápido acceso a los servicios de ingreso de formularios de préstamos, consulta de: saldos, depósitos, pagos de créditos e ingreso de peticiones de certificados en la Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda., de la ciudad de Ambato provincia de Tungurahua.

CAPÍTULO III

METODOLOGÍA

3.1 Enfoque de la investigación

La investigación está sujeta a un enfoque cualitativo, ya que el investigador se adentró en el problema del diseño e implementación de módulos online mediante el uso de WS para ingreso de formularios de: apertura de cuenta, prestamos, consulta de: saldos, depósitos, pagos de créditos e ingreso de peticiones de certificados en la Cooperativa de Ahorro y Crédito Chibuleo Ltda., analizando las necesidades de la Institución, trabajadores, afiliados y demás actores que se ven relacionados de una u otra manera con la Institución, para poder solucionar de mejor forma el problema.

La investigación también tiene un enfoque cuantitativo porque busca determinar las causas y la explicación del problema diseño e implementación de módulos online mediante el uso de WS para ingreso de formularios de: apertura de cuenta, prestamos, consulta de: saldos, depósitos, pagos de créditos e ingreso de peticiones de certificados en la Cooperativa de Ahorro y Crédito Chibuleo Ltda., de la Ciudad de Ambato provincia de Tungurahua, dentro de un enfoque netamente técnico, ya que al implantar el sistema Web se espera resultados rápidos y óptimos; y que sean medibles a corto plazo.

Para implementar esta de investigación se utilizara la encuesta, porque a través de la misma se puede conocer de manera objetiva el nivel de aceptación del sistema por parte de clientes y usuarios.

3.2 Modalidad básica de la investigación

3.2.1 Investigación de campo

Para la realización de este proyecto, se tiene planificado realizar una investigación de campo, porque es estrictamente necesario, realizar un análisis de los hechos, causas y motivos que originan las necesidades de para la implementación de los módulos online mediante el uso de WS para ingreso de formularios de: apertura de cuenta, prestamos, consulta de: saldos, depósitos, pagos de créditos e ingreso de peticiones de certificados en la cooperativa de ahorro y crédito Chibuleo Ltda. de la ciudad de Ambato provincia de Tungurahua.

3.2.2 Investigación documental – bibliográfica

Para la implementación diseño e implementación de módulos online mediante el uso de WS para ingreso de formularios de: apertura de cuenta, prestamos, consulta de: saldos, depósitos, pagos de créditos e ingreso de peticiones de certificados en la Cooperativa de ahorro y crédito Chibuleo Ltda. de la ciudad de Ambato provincia de Tungurahua, es imprescindible, el uso de la investigación bibliográfica para investigar y analizar, las posibles causas que ocasionaron el problema; y de igual manera encontrar las soluciones más viables para resolverlo.

3.2.3 Proyecto factible

El presente tema de investigación, es un proyecto factible, porque permite solucionar algunas carencias de accesibilidad a los servicios fundamentales que posee el sistema financiero de la Cooperativa de Ahorro y Crédito “Chibuleo” Ltda., y además, se lo ha sustentado en el marco teórico propuesto para el tema de investigación. Por consiguiente se considera que la implementación de los módulos de banca en línea, es una alternativa viable para solucionar el problema de accesibilidad a la información de la Institución por una vía Web.

3.3 Nivel o tipo de investigación

El nivel de la investigación es exploratorio, porque es necesario conocer la Cooperativa y como trabajan los diferentes procesos que se realizan en la misma, en especial los que se desarrollan en el área de préstamos e información, para investigar y conocer las causas que originan al problema.

También es correlacional, porque permite establecer la relación entre la variable dependiente y la variable independiente; y propuse la solución de: diseño e implementación de módulos online mediante el uso de Web Service para ingreso de formularios de: apertura de cuenta, prestamos, consulta de: saldos, depósitos, pagos de créditos e ingreso de peticiones de certificados en la Cooperativa de Ahorro y Crédito Chibuleo Ltda. de la ciudad de Ambato provincia de Tungurahua.

3.4 Población y muestra

3.4.1 Población

En la Cooperativa “Chibuleo”, la población con la que se va a trabajar son: 50 seleccionadas de las áreas de:

- Gerencia.
- Departamento de sistemas.
- Asesoría Externo.
- Clientes

3.4.2 Muestra

La muestra es el total de la población.

3.6 Recolección de información

Para la recolección de información de la investigación, se realizará una encuesta a la población.

La encuesta se hace necesario, ya que permitirá obtener datos referentes a las necesidades que tiene la empresa sobre el sistema financiero; la encuesta se realizará a las personas que trabajan en la Cooperativa, y que sus labores se encuentren involucradas o ligadas al sistema en mención; para esto se elaborara un cuestionario estructurado para obtener la información adecuada.

Además, la técnica de la observación también podrá ser utilizada ya que permitirá la apreciación directa de la realidad de la empresa y de su sistema de finanzas.

3.6.1 Plan para la recolección de información

1. Determinación de las personas que van a ser encuestadas.
2. Para la recolección de información, me he permitido seleccionar la entrevista, como técnica para la obtención de datos.
3. Para llevar a cabo lo anterior, el instrumento a utilizarse es un cuestionario estructural.

3.6.2 Plan de procesamiento y análisis de la información

Una vez, aplicados los instrumentos anteriormente mencionados, y analizada la validéz de la información que entregaron los mismos, se procederá a realizar lo siguiente:

- 1.- Revisión del cuestionario
- 4- Tabulación de los datos
- 5.-Estudio estadístico

Al realizar la tabulación de los datos, la información que se obtenga se presentará en gráficos, en términos de porcentaje, para facilitar su comprensión.

3.7 Análisis e interpretación de la información

3.7.1 Comprobación de la hipótesis

El diseño e implementación de módulos online mediante el uso de Web Service para ingreso de formularios de: apertura de cuenta, prestamos, consulta de: saldos, depósitos, pagos de créditos e ingreso de peticiones de certificados en la Cooperativa de Ahorro y Crédito Chibuleo Ltda., de la ciudad de Ambato provincia de Tungurahua, permitió resolver la centralización de la información para las áreas de prestamos, información y certificación que existe al momento en la Cooperativa por medio del sistema financiero chibuleo; y mejoró el tiempo de espera para la ejecución de los servicios financieros en las áreas mencionadas logrando disminución de colas de espera y agilizando la entrega de certificados o prestamos.

Todos los beneficios se los comprobó mediante la encuesta a los actores de la Cooperativa y su respectiva tabulación e interpretación de resultados.

CAPÍTULO IV

ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

4.1 Análisis

Después de realizar las 50 encuestas se obtuvieron los siguientes resultados, que sirven como realidad de la Cooperativa de manera más clara y apegada a la realidad.

4.2 Interpretación de resultados

Al concluir el análisis de resultados de las encuestas, se procede a su interpretación que se presenta a continuación.

PREGUNTA 1

1.- ¿Le gustaría realizar consulta de saldos de sus cuentas por Internet?

Tabla N° 4.1
LE GUSTARÍA REALIZAR CONSULTA DE SALDOS DE SUS
CUENTAS POR INTERNET

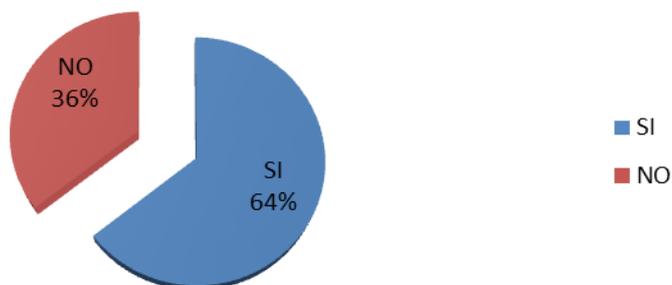
OPCIONES	FRECUENCIA ABSOLUTA	PORCENTAJE %	PORCENTAJE REAL
SI	32	64	64
NO	18	36	36
TOTAL	50	100.00 %	100%

Elaborado por: Wilson Cunalata

Fuente: Encuesta

FIGURA N° 4.1

¿Le gustaría realizar consulta de saldos de sus cuentas por Internet?



Elaborado por: Wilson Cunalata

Fuente: Encuesta

Análisis:

De toda la población (100%) de las personas encuestadas, el 64 % que corresponde a las 32 personas manifiestan que SI les gustaría realizar su consulta de saldos por Internet y el 36 % que corresponden a 18 personas dicen que NO utilizarían dicho servicio.

Interpretación

Entonces podemos interpretar que tanto usuarios como clientes de la Cooperativa si estan dispuestos a utilizar el servicio para poder ahorrar tiempo en estos trasmites.

PREGUNTA N° 2

2.- ¿Cree usted que al habilitar un módulo de pre apertura de cuenta ahorraría tiempo al trámite con la Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda.?

Tabla N° 4.2

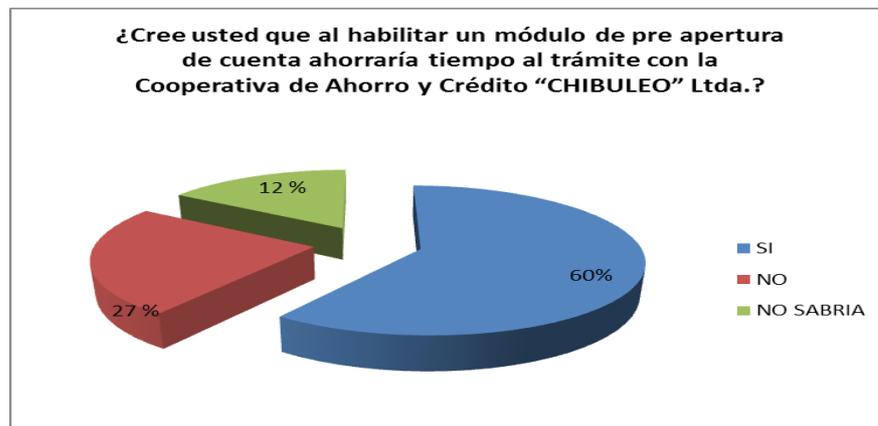
CRÉE USTED QUE AL HABILITAR UN MÓDULO DE PRE APERTURA DE CUENTA AHORRARÍA TIEMPO AL TRÁMITE CON LA COOPERATIVA DE AHORRO Y CRÉDITO “CHIBULEO” LTDA.

OPCIONES	FRECUENCIA ABSOLUTA	PORCENTAJE %	PORCENTAJE REAL
SI	30	60	60
NO	12	24	24
NO SABRIA	8	16	16
TOTAL	50	100.00 %	100%

Elaborado por: Wilson Cunalata

Fuente: Encuesta

FIGURA N° 4.2



Elaborado por: Wilson Cunalata

Fuente: Encuesta

Análisis:

De los encuestados, el 60 % que corresponde a las 30 personas manifiestan que SI les gustaría realizar preaperturas de cuenta por medios computacionales en línea

y el 24% que corresponden a 12 personas dicen que NO utilizarían dicho servicio y un 16% que corresponde a 8 personas no saben su utilidad.

Interpretación

Entonces podemos interpretar que socios y personal de la Institución ven con agrado y de gran uso y utilidad el servicio de pre-apertura de cuentas incrementando la productividad de la Cooperativa.

PREGUNTA N° 3

3.- ¿Mejoraría el tiempo de servicio si el sistema en línea de la Cooperativa pueda realizar ingreso de formulario de préstamos?

Tabla N° 4.3

**MEJORARÍA EL TIEMPO DE SERVICIO SI EL SISTEMA EN LÍNEA
DE LA COOPERATIVA PUEDA REALIZAR INGRESO DE
FORMULARIO DE PRÉSTAMOS**

OPCIONES	FRECUENCIA ABSOLUTA	PORCENTAJE %	PORCENTAJE REAL
SI	40	80	80
NO	10	20	20
TOTAL	48	100.00 %	100%

Elaborado por: Wilson Cunalata

Fuente: Encuesta

FIGURA N° 4.3



Elaborado por: Wilson Cunalata

Fuente: Encuesta

Análisis:

Del 100% de personas encuestadas, el 80% que corresponde a las 40 personas manifiestan que SI les gustaría realizar el ingreso de formularios de préstamos y el 20 % que corresponden a 10 personas dicen que NO utilizarían dicho servicio.

Interpretación

Entonces la interpretación mas lógica para esta pregunta es que la mayoría de personas que fueron encuestadas dentro de la Institución encuentran con gran ayuda el uso del módulo de formulario de prestamos y mejorar el tiempo de servicio para los clientes.

PREGUNTA N° 4

4.- ¿Confiaría en la seguridad de la página Web para el ingreso información personal cuando desee llenar un formulario de préstamos para su respectivo pre aprobación?

Tabla N° 4.4

**CONFIARÍA EN LA SEGURIDAD DE LA PÁGINA WEB PARA EL
INGRESO INFORMACIÓN PERSONAL CUANDO DESEE LLENAR UN
FORMULARIO DE PRÉSTAMOS PARA SU RESPECTIVO PRE
APROBACIÓN**

OPCIONES	FRECUENCIA ABSOLUTA	PORCENTAJE %	PORCENTAJE REAL
SI	30	60	60
NO	13	26	26
A FUTURO	7	14	14
TOTAL	50	100.00 %	100%

Elaborado por Wilson Cunalata

Fuente: Encuesta

FIGURA N° 4.4



Elaborado por: Wilson Cunalata

Fuente: Encuesta

Análisis:

Del total de personas encuestadas, el 60% de ellas que corresponde a las 30 personas manifiestan que SI tendría la confianza para realizar el ingreso de datos

para formularios de préstamo y que 26 % que corresponde a 13 personas no confiarían en este servicio y que un 14% que corresponde a 7 personas no lo realizarían a futuro cuando ya exista un alto margen de efectividad

Interpretación

Entonces podemos interpretar que este servicio mayoría de personas que fueron encuestadas dentro de la Institución encuentran con gran apertura para utilizar este servicio en mejora de tiempos de servicio para los clientes.

PREGUNTA N° 5

5.- ¿Disminuiría el tiempo de realizar una operación bancaria de préstamo o solicitud de certificación el saber los estados dentro del sistema de la Cooperativa por medio de la página Web?

Tabla N° 4.5

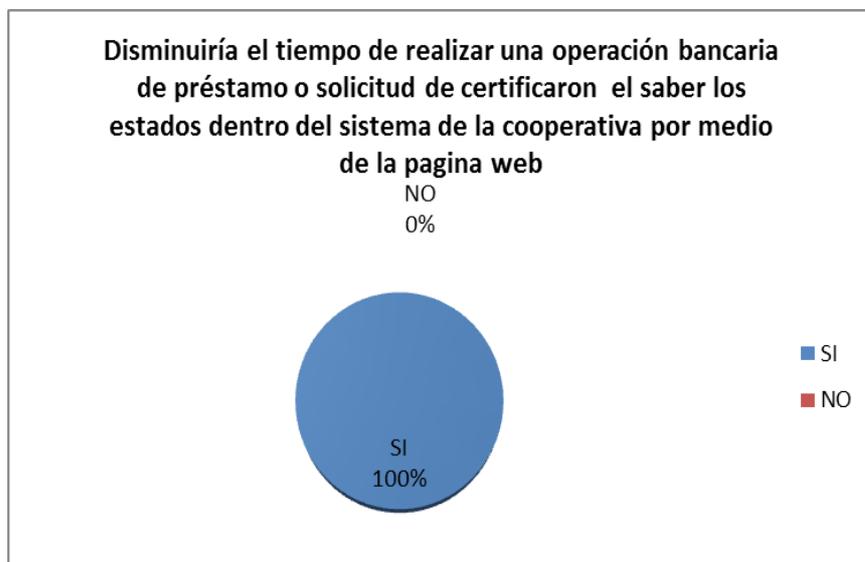
**DISMINUIRÍA EL TIEMPO DE REALIZAR UNA OPERACIÓN
BANCARIA DE PRÉSTAMO O SOLICITUD DE CERTIFICARON EL
SABER LOS ESTADOS DENTRO DEL SISTEMA DE LA
COOPERATIVA POR MEDIO DE LA PAGINA WEB**

OPCIONES	FRECUENCIA ABSOLUTA	PORCENTAJE %	PORCENTAJE REAL
SI	50	100	100
NO	0	0	0
TOTAL	50	100.00 %	100%

Elaborado por Wilson Cunalata

Fuente: Encuesta

FIGURA N° 4.5



Elaborado por: Wilson Cunalata

Fuente: Encuesta

Análisis:

Del total de personas encuestadas, el 100% de personas encuestadas que corresponde a las 50 personas manifiestan que SI tendría la disminuiría el tiempo de realizar una transacción bancaria al saber su estado por medio del internet.

Interpretación

Entonces podemos interpretar que la estos servicios en su totalidad se los ve de manera productiva para disminuir el tiempo que un cliente debe pasar haciendo colas en la Institución para poder utilizar estos servicios.

CAPÍTULOS V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

- Los procesos financieros de la Cooperativa de Ahora y Crédito “CHIBULEO” Ltda., se realizan de una manera centralizada y segura, pero que no permite al cliente conocer información sobre el estado de sus cuentas, pago de créditos, solicitudes de crédito, entre otros de una manera ágil y oportuna causando malestar e inconvenientes en todos sus cuentas ahorristas por tener que realizar colas para obtener esta información solo en la Cooperativa o Sucursales de la misma.
- MySQL y PHP, fueron seleccionados como herramientas de programación para los módulos que se publicaran en internet por los beneficios que ofrece como la rapidez de desarrollo de páginas web que pueden ser multicapas y de acceso simple y universal a todo tipo de información desde cualquier equipo de cómputo; y la aplicación de metodología MVC (Modelo, Vista, Controlado) que permite agrupar secciones de código con funcionalidades comunes.
- C#, fue seleccionado como lenguaje de programación del WS, por la infraestructura que la Institución posee y por ser robusto, rápido y seguro a la momento de programar , permitiendo una interacción tanto con el sistema Web publica en internet como por el sistema interno de la Institución que encuentra funcionando sobre la plataforma .Net.
- Después de la investigación previa se determina que el sistema Web debe ser de fácil manejo, rápido y seguro; explotando los beneficios que prestan las herramientas de programación

5.2 RECOMENDACIONES

- Ordenar los pasos a seguir para crear una propuesta mediante un sistema en línea que le permita obtener la información de manera rápida y fácil,

cambiando de manera radical los procesos financieros para poder descentralizarlos pero sin perder el control y la seguridad de ellos.

- Explotar al máximo las herramientas de programación y las metodologías a aplicarse para desarrollar el sistema Web en sus áreas de clientes como en su área administrativa.
- Determinar las seguridades necesarias a tomar en cuenta para que el acceso a los módulos de banca en línea se realizan de forma segura para que no se exponga la integridad, seguridad de la base de datos de la Cooperativa.
- Desarrollar un sistema que permita el almacenamiento de la información y control de esta para evitar redundancias, por medio de una base de datos relacional.

CAPÍTULO VI

PROPUESTA

6.1 Análisis del sistema

La Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda., no cuenta con un sistema Web dinámico con módulos de banca en línea que permita una expansión más adecuada de toda la información y servicios que ofrece a sus clientes y usuarios.

Destacando el punto anterior la actual directiva dirigida por el Dr. Rodrigo Llambo como Gerente General, se propuso desarrollar un sistema Web dinámico para destacar información de la misma a la colectiva y ofrecer servicios en línea que incrementen la funcionalidad y accesibilidad a los datos por parte de sus clientes.

Para el “DISEÑO E IMPLEMENTACIÓN DE MÓDULOS ONLINE MEDIANTE EL USO DE WEB SERVICE PARA INGRESO DE FORMULARIOS DE: APERTURA DE CUENTA, PRESTAMOS, CONSULTA DE: SALDOS, DEPÓSITOS, PAGOS DE CRÉDITOS E INGRESO DE PETICIONES DE CERTIFICADOS EN LA COOPERATIVA DE AHORRO Y CRÉDITO CHIBULEO LTDA DE LA CIUDAD DE AMBATO PROVINCIA DE TUNGURAHUA.”, se necesita implementar una base de datos robusta, multihilo, multiusuario, segura como lo es MYSQL, que es compatible con PHP, el cual permitirá el diseño de las Páginas Web de forma Dinámica, además que estas herramientas no necesitan de licencias ya que son software libre y permite la

libertad de uso y de modificación de su estructura de una manera sencilla para su manipulación en el sitio WEB.

Entre las características de MYSQL en su versión 5 tenemos que combinar robustez y funcionalidad, seguridad, uso sencillo de XML, flexibilidad del mismo grado de la empresarial como lo provee SQL SERVER 2005 con lo cual la hace una solución rentable para implementarla con los módulos de banca ONLINE ya que no tiene costo su uso.

Finalmente para hacer una combinación robusta se utilizó con PHP (Personal Home Page Tools) en su versión 5 el cual permite realizar scripts avanzados para el diseño de sitios Web dinámicos y fácil acceso a archivos xml, sobre todo en servidores Linux; ya que mediante PHP podemos personalizar la información que deseamos presentar a los usuarios de módulos ONLINE y finalmente es también de software libre permitiendo un alto grado de funcionalidad para su programación.

Como este sistema estará dividido en módulos a continuación destacamos lo principal de cada uno de ellos:

MÓDULO 1.- PRE-APERTURA DE CUENTAS

- Ingreso de datos de los futuros clientes
Este módulo ingresan todos los visitantes que deseen estén interesados en crear obtener una cuenta y puede realizar un pre-ingreso de datos para la creación de cuenta o clientes que tengan y deseen crear una nueva para sus necesidades.
- Verificación y validación de datos ingresados para preapertura
Esta área es revisada por el personal autorizado de la Institución en la cual verifica que los datos ingresados para la pre-apertura de cuenta y los pondrá en estado de espera o en estado de pre-aprobado, para que se

acerque a alguna oficina de la Cooperativa para realizar el proceso final al sistema de la interno de la Institución.

MÓDULO 2.- INGRESO DE FORMULARIO DE PRESTAMOS

- Ingreso de datos para el préstamo

Lo realizan solo clientes que se encuentren activos en el sistema financiero de la Cooperativa y en el Sistema Online de la misma, para lo cual deben ingresan datos referentes a:

- Datos del préstamo
- Datos personales
- Información Financiera
- Referencias Bancarias
- Referencias Personales
- Referencias Comerciales

- Verificación de datos de préstamos y cambio de estados

Lo realiza el personal administrativo de la Cooperativa especializado en el área de prestamos, verificando que los datos ingresados sean correctos y los necesarios para el tipo de préstamo solicitado, y de no serlo, envió de mensajes para que los rectifique para que pueda seguir su proceso bancario; además en caso de ser válido los datos pasan a sus deferentes estados de proceso a aprobarse.

MÓDULO 3.- INGRESO DE PETICIÓN DE CERTIFICADOS

- Ingreso de peticiones de certificados

Esta área es exclusiva para los clientes que se encuentren activos en el sistema ONLINE, en la cual realizan petición de algún tipo de certificado

que la Institución les pueda proveer para sus actividades personales o comerciales.

- **Verificación y Aprobación**

Lo realiza el personal encargado del área de entrega de certificados, dando paso y notificando al usuario que su petición se encuentra aprobado o rechazada según sea el caso y su respectivo retiro en la Cooperativa.

MÓDULO 4.- CONSULTA DE SALDOS

- **Revisión de saldos**

Esta área puede ser utilizada solo por clientes del sistema, en el cual podrán verificar su saldo promedio en todas sus cuentas activas y en dependencia de sus actividades impresión de reporte necesario.

MÓDULO 5.- CONSULTA DE DEPÓSITOS

- **Revisión de saldos**

Esta área similar a la correspondiente a saldos, con la diferencia que se puede obtener un reporte filtrando por fechas y cuentas de los movimientos financieros que ha tenido la respectiva cuenta para un mejor control y verificación de datos por parte de cliente.

6.2 Análisis de factibilidad

6.2.1 Factibilidad operativa

Los módulos ONLINE de la Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda., debe tener una interfaz de fácil manejo y acceso a la información para los visitantes y clientes.

Debe contar con módulos definida para las áreas de consulta de saldos, depósitos, créditos, formularios de preapertura y préstamos con sus respectivas áreas

administrativas en los casos que lo ameriten. La sección información general que será de libre acceso a la comunidad que nos visite.

El usuario de tipo administrador o de tipo usuario se encargará de verificar y procesar toda la información de los módulos que en el sistema los usuarios hayan ingresado por parte de los clientes.

Al final del desarrollo del proyecto se entrega al departamento de Sistemas de la Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda., el instructivo de administración del sistema en línea llamado “Manual del Administrador” y un manual que servirá de guía para el los clientes llamado “Manual de Usuario”.

6.2.2 Factibilidad técnica

Para el desarrollo del sistema en línea y sus respectivos módulos pone a disposición del desarrollador todos los materiales y herramientas necesarias como computadores, dominio, internet, acceso a la base de datos de la Institución, acceso a servidores de pruebas y toda la información necesaria. Además se cuenta con la buena disposición del gerente y departamento de Sistemas para el desarrollo del proyecto en todas sus áreas planteadas.

Software a utilizar

Se lo realiza con software libre (PHP-MySQL) los módulos que se publicaron en el hosting contratado (por su rendimiento que es mucho mas alto que otras conbinaciones de lenguaje de programación y base de datos) y lo correspondiente al WS mediante tecnología .NET C#(debido a la infraestructura que posee la Institución, sistemas en .Net y Sql Sevrer, para pueda ser reutilizado en el momento de agregar nuevos modulos al sistemas Web o consultado por el deparamento de sistemas) que se publica en los servidores de la Cooperativa, lo que no representa ningún inconveniente ni para el desarrollador o para la Institución para conseguirlo, instalarlo e implementarlo en ninguna de las dos

partes, porque poseen las licencias para el uso en el área de Software de pago. A continuación se lista el software necesario.

- PHP como lenguaje de desarrollo Web.
- HTML como lenguaje de marcado para el diseño del portal.
- MySQL como SGBD (sistema de gestión de base de datos). SGBD de libre distribución.
- Wamp (Windows Apache MySQL y PHP v.I) como servidor local.
- Mono Develop para el desarrollo del WS.
- Mono Develop/Apache como servidor de WS en las pruebas realizadas.

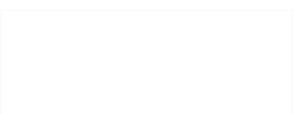
Hardware

La Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda., cuenta con el Departamento de Sistemas el cual en su infraestructura posee equipos de cómputo con las características necesarias para el desarrollo del proyecto planteado.

6.3. Diseño del sistema

Luego de realizar la recolección de la información, ordenar y analizar la misma se ha determinado la necesidad de crear una base de datos para el funcionamiento de los módulos Web y se procede al modelado de la base de datos y su respectiva construcción el MYSQL 5.

6.3.1 Modelo Relacional – Físico



6.3.2 Diccionario de Datos

TABLA CANTONES

Listado de columnas de la tabla Cantones

TABLA CANTONES		Tabla que contiene la información de todos los cantones del Ecuador	
CAMPOS			
Campo	Tipo	Key	Comentarios
Código	char(10)		Contiene el Código del Cantón
Nombre	varchar(50)		Contiene el nombre del Cantón
codigoProvincia	char(10)		
Secuencial	int(11)	PRI	

Tabla 6.1 Listado de columnas de la tabla Cantones

Listado de referencias de la tabla Cantones

FOREIGN KEY RELATIONSHIPS			
FK Id	Tabla de Referencia	Columna Origen	Columna Objetivo
FK_cantones	Provincias	`codigoProvincia`	`codigo`

Tabla 6.2 Listado de referencias de la tabla Cantones

TABLA CERTIFICADOS

Listado de columnas de la tabla Certificados

TABLA CERTIFICADOS		Tabla que contiene certificados existentes en la Cooperativa	
CAMPOS			
Campo	Tipo	Key	Comentarios
Nombre	varchar(100)		Contiene Nombre del Certificado
Descripción	Text		Contiene descripción del Certificado
Id	int(11)	Primary	Contiene el id del certificado

Tabla 6.3 Listado de columnas de la tabla Certificados

TABLA CERTIFICADOS-CLIENTES

Listado de Columnas de la tala Certificado_Clientes

CERTIFICADOS_CLIENTES		Tabla que contiene información de que clientes realizan una petición de certificados	
CAMPOS			
Campo	Tipo	Key	Comentarios
Id	int(11)	PRI	Contiene el identificador de la tabla
Idusuario	varchar(13)	PRI	Contiene el identificador del usuario
idcertificado	int(11)	PRI	Contiene el identificador del tipo de certificado
fechaPeticion	Date		Contiene la fecha de realización de un certificado
Estado	varchar(10)		Contiene le estado en que se encuentra un certificado

Tabla 6.4 Listado de Columnas de la tala Certificado_Clientes

Listado de referencias de la tabla Certificados_Clientes

FOREIGN KEY RELATIONSHIPS			
FK Id	Reference Tabla	Source Column	Target Column
FK_certificados_clientes_2	clientes	`idusuario`	`codigo`
FK_certificados_clientes_4	Certificados	`idcertificado`	`id`

Tabla 6.5 Listado de referencias de la tabla Certificados_Clientes

TABLA CLIENTES

Listado de columnas de la tabla Clientes

TABLA CLIENTES		Tabla que contiene información de los clientes referentes a la Cooperativa	
CAMPOS			
Campo	Tipo	Key	Comentarios
Código	varchar(13)	PRI	Contiene el número de cedula del cliente.
Tipo	char(1)		Contiene que tipo pertenece el usuario o cliente
direccionDomicilio	varchar(80)		Contiene la dirección del usuario o cliente
telefonoDomicilio	varchar(20)		Contiene el número de teléfono del cliente convencional
telefonoCelular	varchar(10)		Contiene el número de teléfono del cliente celular
Email	varchar(35)		Contiene el email del cliente
paisOrigen	varchar(40)		Contiene de que país es el cliente
provinciaReside	char(10)		Contiene a la provincia que pertenece el cliente
cantonReside	char(10)		Contiene a el Cantón que pertenece el cliente
zonaReside	char(10)		Contiene a la zona que pertenece el cliente
Estado	char(1)		Contiene el estado en que se encuentra el cliente.
nombreUnido	varchar(100)		Contiene el nombre de manera unida del cliente
numeroCliente	int(11)	PRI	Contiene el numero de cuenta que posee un cliente
fechaIngreso	mediumblob		Contiene la fecha de ingreso en el sistema del cliente
porAprobar	int(11)		Contiene el estado en el sistema cuando es pre ingreso de cuenta

Tabla 6.6 Listado de columnas de la tabla Clientes

Listado de referencias de la tabla Clientes

FOREIGN KEY RELATIONSHIPS

FK Id	Reference Tabla	Source Column	Target Column	Extra Info
FK_provincias	provincias	`provinciaReside`	`codigo`	

Tabla 6.7 Listado de referencias de la tabla Clientes

TABLA CLIENTESNATURALES

Lista de columnas de la tabla CLIENTESNATURALES

CLIENTESNATURALES		Tabla que contiene informacion de clientes tipo naturales	
CAMPOS			
Campo	Tipo	Ke y	Comentarios
numeroCliente	int(11)	PR I	Contiene el número del cliente
Apellido	varchar(50)		Contiene el apellido del cliente
Nombre	varchar(50)		Contiene el nombre del Cliente
queSexo	char(1)		Contiene el sexo del cliente
estadoCivil	char(1)		Contiene el estado civil de cliente
numeroCargas	tinyint(3) unsigned		Contiene el número de cargas familiares que tiene el cliente
esDependiente	tinyint(3) unsigned		Contiene la información si un clientes es de tipo dependiente o no
nivelEducacion	char(1)		Contiene el nivel de educación
queVivienda	char(1)		Contiene el tipo de vivienda que posee un cliente
fechaNacimiento	mediumblob		Contiene la fecha de nacimiento de un cliente
queProfesion	varchar(10)		Contiene la profesión de un cliente
esNacional	tinyint(3) unsigned		Contiene si es nacional o extranjero un cliente

Tabla 6.8 Lista de columnas de la tabla CLIENTESNATURALES

Listado de referencias de la tabla CLIENTESNATURALES

FOREIGN KEY RELATIONSHIPS			
FK Id	Reference Tabla	Source Column	Target Column
FK_clientesnaturales	clientes	`numeroClien te`	`numeroCliente`
FK_clientesnaturales_pro fesiones	profesiones	`queProfesio n`	`codigo`

TABLA 6.9 Listado de referencias de la tabla CLIENTESNATURALES

TABLA CUENTAS

Lista De Columnas De La Tabla Cuentas

TABLA CUENTAS		Tabla que contiene información de cuentas que posee un cliente habilitado ene el sistema en línea	
CAMPOS			
Campo	Tipo	Key	Comentarios
Código	char(20)	PRI	Contiene el código de la tabla
Secuencial	int(11)		Contiene el numero secuencial de la tabla
numeroCliente	int(11)	PRI	Contiene en número del cliente
codigoProducto	char(10)		Contiene el código del producto de la cuenta
Disponible	decimal(18,2)		Contiene el estado de disponibilidad
Retenciones	decimal(18,2)		Contiene el número de retenciones
Encajes	decimal(18,2)		Contiene el número de encajes
Bloqueos	decimal(18,2)		Contiene el número de bloqueos
Interés	decimal(18,10)		Contiene el interés existente en la cuenta
fechaCreacion	mediumblob		Contiene la fecha de creación
fechaCierre	mediumblob		Contiene la fecha de cierre de la cuenta

Estado	char(1)		Contiene el estado de la cuenta
estaAperturada	tinyint(3) unsigned		Contiene el estado de preapertura
saldoPromedio	decimal(18,2)		Contiene el saldo promedio
fechaActiva	mediumblob		Contiene la fecha de activación de la cuenta

TABLA 6.10 Lista De Columnas De La Tabla Cuentas

Listado De Referencias De La Tabla Cuentas

Foreign Key Relationships				
FK Id	Reference Tabla	Source Column	Target Column	Extra Info
FK_cuentas	Cientes	`numeroCliente`	`numeroCliente`	

Tabla 6.11 Listado De Referencias De La Tabla Cuentas

TABLA DATOS CREDITICIOS

Listado de columnas de la tabla datoscrediticios

TABLA	Tabla que contiene información sobre datos crediticios sobre un cliente en la Web		
DATOS CREDITICIOS			
CAMPOS			
Campo	Tipo	Key	Comentarios
numeroCliente	int(11)	PRI	Contiene le número de cliente
nombreArrendador	varchar(50)		Contiene el nombre del arrendador
costoArriendo	decimal(18,0)		Contiene el costo del arriendo
tiempoResidencia	int(11)		Contiene el tiempo de residencia
numeroConyuge	int(11)		Contiene el número del cónyuge
Ingresos	decimal(18,2)		Contiene el total de los ingresos
Egresos	decimal(18,2)		Contiene el total de los egresos
Activos	decimal(18,2)		Contiene el total de activos
Pasivos	decimal(18,2)		Contiene el total de pasivos

fechaModifico	mediumblob		Contiene la fecha de modificación
descripcionActividad	varchar(200)		Contiene la descripción de actividad
paisResidencia	varchar(20)		Contiene el país que reside
tiempoActividad	int(11)		Contiene le tiempo de actividad

Tala 6.12 Listado de columnas de la tabla datoscrediticios

Listado de referencias de la tabla datos crediticios

FOREIGN KEY RELATIONSHIPS				
FK Id	Reference Tabla	Source Column	Target Column	Extra Info
FK_datoscrediticios	clientes	`numeroCliente`	`numeroCliente`	

Tabla 6.13 Listado de referencias de la tabla datos crediticios

TABLA GARANTES PRÉSTAMOS

Listado de columnas de la tabla garantes prestamos

TABLA	Tabla que contiene información sobre los garantes que se ingresa para realizar un préstamo		
GARANTESPRESTAMOS			
CAMPOS			
Campo	Tipo	Key	Comentarios
numeroSolicitud	int(11)	PRI	Contiene el número de solicitud
numeroCliente	int(11)	PRI	Contiene el número del cliente
esSolidario	tinyint(3) unsigned		Contiene si es de tipo solidario
estaActivo	tinyint(3) unsigned		Contiene es estado
Parentesco	varchar(20)		Contiene el parentesco
fechaProceso	mediumblob		Contiene la fecha de proceso

Tabla 6.14 Listado de columnas de la tabla garantas prestamos

Listado de referencias de la tabla garantas prestamos

FOREIGN KEY RELATIONSHIPS			
FK Id	Reference Tabla	Source Column	Target Column
FK_garantesprestamos	solicitudes	`numeroSolicitud`	`numeroSolicitud`
FK_garantesprestamos2	clientes	`numeroCliente`	`numeroCliente`

Tabla 6.15 Listado de referencias de la tabla garantas prestamos

TABLA ÍTEM ECONÓMICOS

Listado de columnas de la tabla ítems económicos

TABLA		Tabla que contiene información sobre los Items económicos que se tomasn en cuenta para realizar un formulario de préstamo	
ITEMSECONOMICOS			
CAMPOS			
Campo	Tipo	Key	Comentarios
codigo	tinyint(3) unsigned	PRI	Contiene le código
nombre	varchar(21)		Contiene el nombre
esEgreso	tinyint(3) unsigned		Contiene si es de tipo egreso

Tabla 6.16 Listado de columnas de la tabla ítems económicos

TABLA ÍTEMS ECONÓMICOS CLIENTES

Listado de columnas de la tabla ítems económicos clientes

TABLA		Tabla que almacena los ítems económicos que un cliente haya ingresado al momento de realizar una petición de préstamo	
ITEMSECONOMICOSCLIENTES			
CAMPOS			
Campo	Tipo	Key	Comentarios
numeroCliente	int(11)	PRI	Contiene el número del cliente
codigoItem	tinyint(3) unsigned	PRI	Contiene el código de ítem
Valor	decimal(18,2)		Contiene el valor de ítem

Tabla 6.17 Listado de columnas de la tabla ítems económicos clientes

Listado de referencias de la tabla ítem económicos clientes

Foreign Key Relationships			
FK Id	Reference	Source	Target
	Tabla	Column	Column
FK_itemseconomicosclientes	clientes	`numeroCliente`	`numeroCliente`
FK_itemseconomicosclientes2	Itemseconomico	`codigoItem`	`codigo`

Tabla 6.18 Listado de referencias de la tabla ítem económicos clientes

TABLA ÍTEMS PATRIMONIALES

Listado de columnas de ítems patrimoniales

TABLA		Tabla que contiene información sobre los ítems patrimoniales a tomar en cuenta para un forario de préstamo	
ITEMSPATRIMONIALES			
CAMPOS			
Campo	Tipo	Key	Comentarios
codigo	tinyint(3) unsigned	PRI	Contiene el código de ítem
nombre	varchar(20)		Contiene el nombre del ítem
esPasivo	tinyint(3) unsigned		Contiene si es de tipo pasivo

Tabla 6.19 Listado de columnas de ítems patrimoniales

TABLA ÍTEMS PATRIMONIALES CLIENTES

Listado de ítems patrimoniales clientes

TABLA		Tabla que contiene la información de los ítems patrimoniales que un cliente ingresa para un prestamos	
ITEMSPATRIMONIALESCLIENTES			
CAMPOS			
Campo	Tipo	Key	Comentarios
numeroCliente	int(11)	PRI	Contiene el número del cliente
codigoItem	tinyint(3) unsigned	PRI	Contiene el código del ítem
Valor	decimal(18,2)		Contiene el valor
Descripción	varchar(100)		Contiene la descripción

Tabla 6.20 Listado de ítems patrimoniales clientes

Listado de referencias de la tabla ítems patrimoniales clientes

Foreign Key Relationships			
FK Id	Reference	Source Column	Target
	Tabla		Column
FK_itemspatrimon	clientes	`numeroCliente`	`numeroClient

ialesclientes			e`
FK_itemspatrimon	itemspatrimonial	`codigoItem`	`codigo`
ialesclientes2	es		

Tabla 6.21 Listado de referencias de la tabla ítems patrimoniales clientes

TABLA PRODUCTOS

Listado de columnas de la tabla Productos

TABLA		Tabla que contiene tipos de cuentas que posee la	
PRODUCTOS		Cooperatiba	
CAMPOS			
Campo	Tipo	Key	Comentarios
codigo	char(10)	PRI	Contiene el código de producto
nombr	varchar(50)		Contiene el nombre del producto
e			

Tabla 6.22 Listado de columnas de la tabla Productos

TABLA PROFESIONES

Listado de columnas de la tabla profesiones

TABLA	Tabla que contiene una infinidad de tipos de
PROFESIONES	profesiones para selección en un formulario de
	Préstamo

CAMPOS			
Campo	Tipo	Key	Comentarios
codigo	varchar(10)	PRI	Contiene el código de la profesión
nombr e	varchar(50)		Contiene el nombre de la profesion

Tabla 6.23 Listado de columnas de la tabla profesiones

TABLA PROVINCIAS

Listado de columnas de la tabla provincias

TABLA PROVINCIAS		Tabla que contiene información sobre todas la provincias del Ecuador	
CAMPOS			
Campo	Tipo	Key	Comentarios
codigo	char(10)	PRI	Contiene el código de la provincia
nombr e	varchar(50)		Contiene el nombre de la provincia

Tabla 6.24 Listado de columnas de la tabla provincias

TABLA REFERENCIAS BANCARIAS

Listado de columnas de la tabla referencias bancarias

TABLA	Tabla que contiene la información de
--------------	---

REFERENCIASBANCARIAS		referencias bancarias de cada cliente	
CAMPOS			
Campo	Tipo	Key	Comentarios
numeroCliente	int(11)	PRI	Contiene el número del cliente
Prioridad	tinyint(3) unsigned		Contiene la prioridad
nombreBanco	varchar(50)		Contiene el número de banco
numeroCuenta	varchar(20)		Contiene el número de cuenta
tipoCuenta	char(1)		Contiene el tipo de cuenta

Tabla 6.25 Listado de columnas de la tabla referencias bancarias

Listado de referencias de la tabla referencias bancarias

FOREIGN KEY RELATIONSHIPS			
FK Id	Reference	Source	Target Column
	Tabla	Column	
FK_referenciasbancarias	clientes	`numeroCliente`	`numeroCliente`

Tabla 6.26 Listado de referencias de la tabla referencias bancarias

TABLA REFERENCIAS COMERCIALES

Listado de columnas de la tabla referencias comerciales

TABLA		Tabla que contiene las referencias	
REFERENCIASCOMERCIALES		comerciales	
CAMPOS			
Campo	Tipo	Key	Comentarios

numeroCliente	int(11)	PRI	Contiene el número de cliente
Prioridad	tinyint(3) unsigned		Contiene la prioridad
Nombre	varchar(50)		Contiene el nombre de la referencia
Dirección	varchar(50)		Contiene la dirección de la referencia
Teléfono	varchar(20)		Contiene el número de teléfono

Tabla 6.27 Listado de columnas de la tabla referencias comerciales

Listado de referencias de la tabla referenciad comerciales

FOREIGN KEY RELATIONSHIPS			
FK Id	Reference Tabla	Source Column	Target Column
FK_referenciascomerciales	clientes	`numeroCliente`	`numeroCliente`

Tabla 6.28 Listado de índices de la tabla referencias comerciales

TABLA REFERENCIAS COMERCIALES

Listado de columnas de la tabla referencias comerciales

TABLA		Tabla que contiene las referencias	
REFERENCIASPERSONALES		personales de los clientes	
CAMPOS			
Campo	Tipo	Key	Comentarios
numeroCliente	int(11)	PRI	Contiene el número del cliente
Prioridad	tinyint(3) unsigned		Contiene la prioridad

Nombre	varchar(50)		Contiene el nombre de la referencia
Dirección	varchar(50)		Contiene la dirección
Teléfono	varchar(20)		Contiene el número de teléfono
Parentesco	varchar(20)		Contiene le tipo de parentesco

Tabla 6.29 Listado de columnas de la tabla referencias comerciales

Listado de referencias de la tabla referencias comerciales

FOREIGN KEY RELATIONSHIPS			
FK Id	Reference	Source	Target Column
	Tabla	Column	
FK_referenciaspersonales	Cientes	`numeroCliente`	`numeroCliente`

Tabla 6.30 Listado de referencias de la tabla referencias comerciales

TABLA SOLICITUDES

Listado de columnas de la tabla solicitudes

TABLA	Tabla que contiene las solicitudes ingresadas en el sistema para prestamos		
SOLICITUDES			
CAMPOS			
Campo	Tipo	Key	Comentarios
numeroSolicitud	int(11)	PRI	Contiene le número de solicitud
Secuencial	int(11)		Contiene le número de secuencial
numeroCliente	int(11)		Contiene el número de cliente
codigoProducto	char(10)		Contiene el código de producto
Estado	char(1)		Contiene el estado
codigoTipoPrestamo	char(10)		Contiene el tipo de préstamo

codigoTipoCalificacion	tinyint(3) unsigned		Contiene el tipo de calificación
montoSolicitado	decimal(18,2)		Contiene el monto solicitado
montoAprobado	decimal(18,2)		Contiene el monto aprobado
montoEncajadoAhorro	decimal(18,2)		Contiene el monto encajado de ahorro
tasaInteres	decimal(10,2)		Contiene la tasa de interés
numeroCuotas	smallint(6)		Contiene el número de cuotas
valorCuota	decimal(18,2)		Contiene el valor de la cuota
frecuenciaPago	smallint(6)		Contiene la frecuencia de pago
fechaSolicitud	Mediumblob		Contiene la fecha de solicitud
fechaDefinicion	Mediumblob		Contiene la fecha de definición
codigoUsuario	char(10)		Contiene el código del usuario
numeroGarantes	tinyint(3) unsigned		Contiene el número de garantes
numeroGarantias	tinyint(3) unsigned		Contiene el número de garantías

Tabla 6.31 Listado de columnas de la tabla solicitudes

Listado de referencias de la tabla solicitudes

FOREIGN KEY RELATIONSHIPS				
FK Id	Reference Tabla	Source Column	Target Column	Extra Info
FK_solicitudes	productos	`codigoProducto`	`codigo`	,
FK_solicitudes2	tiposprestamos	`codigoTipoPrestamo`	`codigo`	

Tabla 6.32 Listado de referencias de la tabla solicitudes

TABLA WEB_ACCESOS

Listado de columnas de la tabla Web_accesos

TABLA WEB_ACCESOS	Tabla que contiene los accesos realizados al sistema

CAMPOS			
Campo	Tipo	Key	Comentarios
ID	int(11)	PRI	Contiene el código del acceso
IDUSUARIO	varchar(50)	PRI	Contiene el número del usuario que realizo un acceso
IPACCESO	varchar(100)		Contiene la ip de acceso
FECHA	Datetime		Contiene la fecha de acceso
DESCRIPCION	Text		Contiene una descripción del acceso

Tabla 6.33 Listado de columnas de la tabla Web_accesos

Listado de referencias la tabla Web_accesos

FOREIGN KEY RELATIONSHIPS				
FK Id	Reference	Source	Target	Extra
	Tabla	Column	Column	Info
FK_Web_accesos	Web_logins	`IDUSUARIO`	`LOGIN`	

Tabla 6.34 Listado de referencias la tabla Web_accesos

TABLA WEB_LOGINS

Listado de columnas de la tabla Web_logins

TABLA	Contiene información sobre los usuarios del sistema en línea tanto clientes como usuarios		
WEB_LOGINS			
CAMPOS			
Campo	Tipo	Key	Comentarios
LOGIN	varchar(50)	PRI	Contiene el nombre del usuario
CLAVE	varchar(50)		Contiene la clave
CREADOR	varchar(20)		Contiene el nombre del usuario que lo creo
DESCRIPCION	varchar(250)		Contiene la fecha de descripción del usuario
FECHA_CREACION	Date		Contiene la fecha de creación
FECHA_MODIFICADO	Date		Contiene la fecha de

			modificación
ESTADO_SISTEMA	int(11)		Contiene el estado del sistema en línea
OFICINA	varchar(50)		Contiene el tipo
ESTA_ACTIVO	int(2)		Contiene si está activo
EMAIL	varchar(50)	UNI	Contiene le email del usuario

Tabla 6.35 Listado de columnas de la tabla Web_logins

Listado de referencias de la tabla Web_logins

FOREIGN KEY RELATIONSHIPS				
FK Id	Reference	Source	Target	Extra
	Tabla	Column	Column	Info
FK_Web_logins	clientes	`LOGIN`	`codigo`	

Tabla 6.36 Listado de referencias de la tabla Web_logins

TABLA WEB_LOGINS_ROLES

Listado de columnas de la tabla Web_logins_roles

TABLA	Tabla que contiene los privilegios que se le otorga a cada usuario del sistema			
WEB_LOGINS_ROLES				
CAMPOS				
Campo	Tipo	Key	Comentarios	
IDUSUARIO	varchar(50)	PRI	Contiene le id de usuario	
IDROL	int(11)	PRI	Contiene el rol del usuario	
ACTIVO	int(5)		Contiene si está activo	

Tabla 6.37 Listado de columnas de la tabla Web_logins_roles

Listado de referencias de la tabal Web_logins_roles

Foreign Key Relationships				
FK Id	Reference	Source	Target	Extra

	Tabla	Column	Column	Info
FK_Web_logins_roles	Web_logins	`IDUSUARIO`	`LOGIN`	,
FK_Web_logins_roles_2	Web_roles	`IDROL`	`ID`	

Tabla 6.38 Listado de referencias de la tabla Web_logins_roles

Tabla Web_roles

Listado de columnas de la tabla Web_roles

TABLA		Tabla que contiene los roles que tiene el sistema	
WEB_ROLES			
CAMPOS			
Campo	Tipo	Key	Comentarios
ID	int(11)	PRI	Contiene el id del rol
NOMBRE	varchar(50)		Contiene el nombre del rol

Tabla 6.39 Listado de columnas de la tabla Web_roles

6.3.3 SCRIPT DE LA BASE DE DATOS

```
7
47
50 /*
51 MySQL - 5.1.36-community-log : Database - dbchibuleo
52 ****
53 */
54 /*!40101 SET NAMES utf8 */;
55 /*!40101 SET SQL_MODE=""*/;
56 /*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS,
    UNIQUE_CHECKS=0 */;
57 /*!40014 SET
    @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
    FOREIGN_KEY_CHECKS=0 */;
58 /*!40101 SET @OLD_SQL_MODE=@@SQL_MODE,
    SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
59 /*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0
    */;
60 CREATE DATABASE /*!32312 IF NOT EXISTS*/`dbchibuleo` /*!40100
    DEFAULT CHARACTER SET utf8 COLLATE utf8_spanish2_ci */;
61 USE `dbchibuleo`;
62 /*Table structure for table `cantones` */
63 DROP TABLE IF EXISTS `cantones`;
64 CREATE TABLE `cantones` (
65   `codigo` char(10) COLLATE utf8_spanish2_ci DEFAULT NULL,
66   `nombre` varchar(50) COLLATE utf8_spanish2_ci DEFAULT NULL,
67   `codigoProvincia` char(10) COLLATE utf8_spanish2_ci DEFAULT
    NULL,
68   `secuencial` int(11) NOT NULL,
69   PRIMARY KEY (`secuencial`),
70   KEY `FK_Cantones_Provincias` (`codigoProvincia`),
```

```

71  CONSTRAINT `FK_cantones` FOREIGN KEY (`codigoProvincia`)
    REFERENCES `provincias` (`codigo`)
72 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
73 /*Table structure for table `certificados` */
74 DROP TABLE IF EXISTS `certificados`;
75 CREATE TABLE `certificados` (
76  `nombre` varchar(100) COLLATE utf8_spanish2_ci NOT NULL,
77  `descripcion` text COLLATE utf8_spanish2_ci,
78  `id` int(11) NOT NULL AUTO_INCREMENT,
79  PRIMARY KEY (`id`)
80 ) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
81 DROP TABLE IF EXISTS `certificados_clientes`;
82 CREATE TABLE `certificados_clientes` (
83  `id` int(11) NOT NULL AUTO_INCREMENT,
84  `idusuario` varchar(13) COLLATE utf8_spanish2_ci NOT NULL,
85  `idcertificado` int(11) NOT NULL,
86  `fechaPeticion` date NOT NULL,
87  `estado` varchar(10) COLLATE utf8_spanish2_ci DEFAULT 'ESPERA',
88  PRIMARY KEY (`id`,`idusuario`,`idcertificado`),
89  KEY `FK_certificados_clientes_2` (`idusuario`),
90  KEY `FK_certificados_clientes_4` (`idcertificado`),
91  CONSTRAINT `FK_certificados_clientes_2` FOREIGN KEY (`idusuario`)
    REFERENCES `clientes` (`codigo`),
92  CONSTRAINT `FK_certificados_clientes_4` FOREIGN KEY
    (`idcertificado`) REFERENCES `certificados` (`id`)
93 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
94 /*Table structure for table `clientes` */
95 DROP TABLE IF EXISTS `clientes`;
96 CREATE TABLE `clientes` (

```

97 `codigo` varchar(13) COLLATE utf8_spanish2_ci NOT NULL,
 98 `tipo` char(1) COLLATE utf8_spanish2_ci NOT NULL,
 99 `direccionDomicilio` varchar(80) COLLATE utf8_spanish2_ci DEFAULT
 NULL,
 100 `telefonoDomicilio` varchar(20) COLLATE utf8_spanish2_ci DEFAULT
 NULL,
 101 `telefonoCelular` varchar(10) COLLATE utf8_spanish2_ci DEFAULT
 NULL,
 102 `empresaCelular` varchar(1) COLLATE utf8_spanish2_ci DEFAULT
 NULL,
 103 `email` varchar(35) COLLATE utf8_spanish2_ci DEFAULT NULL,
 104 `paisOrigen` varchar(40) COLLATE utf8_spanish2_ci DEFAULT NULL,
 105 `provinciaReside` char(10) COLLATE utf8_spanish2_ci NOT NULL,
 106 `cantonReside` char(10) COLLATE utf8_spanish2_ci NOT NULL,
 107 `zonaReside` char(10) COLLATE utf8_spanish2_ci NOT NULL,
 108 `representantePrincipal` char(10) COLLATE utf8_spanish2_ci DEFAULT
 NULL,
 109 `representanteSecundario` char(10) COLLATE utf8_spanish2_ci
 DEFAULT NULL,
 110 `estado` char(1) COLLATE utf8_spanish2_ci NOT NULL,
 111 `oficina` tinyint(3) unsigned DEFAULT NULL,
 112 `nombreUnido` varchar(100) COLLATE utf8_spanish2_ci DEFAULT
 NULL,
 113 `numeroCliente` int(11) NOT NULL,
 114 `sectorEconomico` char(10) COLLATE utf8_spanish2_ci DEFAULT
 NULL,
 115 `fechaIngreso` mediumblob NOT NULL,
 116 `vinculo` tinyint(3) unsigned NOT NULL,
 117 `relacionCliente` varchar(10) COLLATE utf8_spanish2_ci NOT NULL,
 118 `calificacionInterna` char(1) COLLATE utf8_spanish2_ci NOT NULL,
 119 `sectorInterno` varchar(10) COLLATE utf8_spanish2_ci DEFAULT
 NULL,

```

120 `porAprobar` int(11) DEFAULT '0',
121 `version` int(11) DEFAULT NULL,
122 PRIMARY KEY (`codigo`,`numeroCliente`),
123 UNIQUE KEY `cuenta` (`numeroCliente`),
124 KEY `FK_Clientes_Provincias` (`provinciaReside`),
125 KEY `FK_Clientes_RelacionesClientes` (`relacionCliente`),
126 KEY `FK_Clientes_SectoresInternos` (`sectorInterno`),
127 CONSTRAINT `FK_provincias` FOREIGN KEY (`provinciaReside`)
REFERENCES `provincias` (`codigo`)
128 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
COLLATE=utf8_spanish2_ci;
129 DROP TABLE IF EXISTS `clientesnaturales`;
130 CREATE TABLE `clientesnaturales` (
131 `numeroCliente` int(11) NOT NULL,
132 `apellido` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
133 `nombre` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
134 `queSexo` char(1) COLLATE utf8_spanish2_ci NOT NULL,
135 `estadoCivil` char(1) COLLATE utf8_spanish2_ci NOT NULL,
136 `numeroCargas` tinyint(3) unsigned NOT NULL,
137 `esDependiente` tinyint(3) unsigned NOT NULL,
138 `nivelEducacion` char(1) COLLATE utf8_spanish2_ci NOT NULL,
139 `queVivienda` char(1) COLLATE utf8_spanish2_ci NOT NULL,
140 `fechaNacimiento` mediumblob NOT NULL,
141 `queProfesion` varchar(10) COLLATE utf8_spanish2_ci NOT NULL,
142 `esNacional` tinyint(3) unsigned NOT NULL,
143 `queActividadEconomica` char(10) COLLATE utf8_spanish2_ci NOT
NULL,
144 `estaMultado` tinyint(3) unsigned NOT NULL,
145 PRIMARY KEY (`numeroCliente`),
146 KEY `FK_clientesnaturales_profesiones` (`queProfesion`),
147 CONSTRAINT `FK_clientesnaturales` FOREIGN KEY (`numeroCliente`)
REFERENCES `clientes` (`numeroCliente`),

```

```

148 CONSTRAINT `FK_clientesnaturales_profesiones` FOREIGN KEY
    (`queProfesion`) REFERENCES `profesiones` (`codigo`)
149 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
150 /*Table structure for table `cuentas` */
151 DROP TABLE IF EXISTS `cuentas`;
152 CREATE TABLE `cuentas` (
153   `codigo` char(20) COLLATE utf8_spanish2_ci NOT NULL,
154   `secuencial` int(11) NOT NULL,
155   `numeroCliente` int(11) NOT NULL,
156   `codigoProducto` char(10) COLLATE utf8_spanish2_ci NOT NULL,
157   `disponible` decimal(18,2) NOT NULL,
158   `retenciones` decimal(18,2) NOT NULL,
159   `encajes` decimal(18,2) NOT NULL,
160   `bloqueos` decimal(18,2) NOT NULL,
161   `interes` decimal(18,10) NOT NULL,
162   `numeroLinea` tinyint(3) unsigned NOT NULL,
163   `oficialCuenta` char(10) COLLATE utf8_spanish2_ci NOT NULL,
164   `oficinaAbrio` tinyint(3) unsigned NOT NULL,
165   `fechaCreacion` mediumblob NOT NULL,
166   `fechaCierre` mediumblob NOT NULL,
167   `estado` char(1) COLLATE utf8_spanish2_ci NOT NULL,
168   `estaAperturada` tinyint(3) unsigned NOT NULL,
169   `saldoPromedio` decimal(18,2) NOT NULL,
170   `identificaCuenta` varchar(20) COLLATE utf8_spanish2_ci NOT NULL,
171   `numeroLibreta` tinyint(3) unsigned NOT NULL,
172   `fechaActiva` mediumblob NOT NULL,
173   PRIMARY KEY (`codigo`,`numeroCliente`),
174   KEY `FK_cuentas` (`numeroCliente`),
175   CONSTRAINT `FK_cuentas` FOREIGN KEY (`numeroCliente`)
    REFERENCES `clientes` (`numeroCliente`)

```

```

176 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
177 /*Table structure for table `datoscrediticios` */
178 DROP TABLE IF EXISTS `datoscrediticios`;
179 CREATE TABLE `datoscrediticios` (
180   `numeroCliente` int(11) NOT NULL,
181   `nombreArrendador` varchar(50) COLLATE utf8_spanish2_ci DEFAULT
    NULL,
182   `costoArriendo` decimal(18,0) DEFAULT NULL,
183   `tiempoResidencia` int(11) NOT NULL,
184   `numeroConyuge` int(11) DEFAULT NULL,
185   `ingresos` decimal(18,2) NOT NULL,
186   `egresos` decimal(18,2) NOT NULL,
187   `activos` decimal(18,2) NOT NULL,
188   `pasivos` decimal(18,2) NOT NULL,
189   `version` int(11) NOT NULL,
190   `codigoUsuarioModifico` char(10) COLLATE utf8_spanish2_ci NOT
    NULL,
191   `fechaModifico` mediumblob NOT NULL,
192   `descripcionActividad` varchar(200) COLLATE utf8_spanish2_ci NOT
    NULL,
193   `paisResidencia` varchar(20) COLLATE utf8_spanish2_ci NOT NULL,
194   `tiempoActividad` int(11) NOT NULL,
195   PRIMARY KEY (`numeroCliente`),
196   CONSTRAINT `FK_datoscrediticios` FOREIGN KEY (`numeroCliente`)
    REFERENCES `clientes` (`numeroCliente`)
197 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;

198 /*Table structure for table `garantesprestamos` */
199 DROP TABLE IF EXISTS `garantesprestamos`;
200 CREATE TABLE `garantesprestamos` (

```

```

201 `numeroSolicitud` int(11) NOT NULL,
202 `numeroCliente` int(11) NOT NULL,
203 `esSolidario` tinyint(3) unsigned NOT NULL,
204 `estaActivo` tinyint(3) unsigned NOT NULL,
205 `parentesco` varchar(20) COLLATE utf8_spanish2_ci NOT NULL,
206 `secuencial` int(11) NOT NULL,
207 `fechaProceso` mediumblob,
208 `usuarioProceso` char(10) COLLATE utf8_spanish2_ci DEFAULT NULL,
209 PRIMARY KEY (`numeroSolicitud`,`numeroCliente`),
210 KEY `FK_garantesprestamos2` (`numeroCliente`),
211 CONSTRAINT `FK_garantesprestamos` FOREIGN KEY
    (`numeroSolicitud`) REFERENCES `solicitudes` (`numeroSolicitud`),
212 CONSTRAINT `FK_garantesprestamos2` FOREIGN KEY
    (`numeroCliente`) REFERENCES `clientes` (`numeroCliente`)
213 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
214 /*Table structure for table `itemseconomicos` */
215 DROP TABLE IF EXISTS `itemseconomicos`;
216 CREATE TABLE `itemseconomicos` (
217   `codigo` tinyint(3) unsigned NOT NULL,
218   `nombre` varchar(21) COLLATE utf8_spanish2_ci NOT NULL,
219   `esEgreso` tinyint(3) unsigned NOT NULL,
220   PRIMARY KEY (`codigo`)
221 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
222 /*Table structure for table `itemseconomicosclientes` */
223 DROP TABLE IF EXISTS `itemseconomicosclientes`;
224 CREATE TABLE `itemseconomicosclientes` (
225   `numeroCliente` int(11) NOT NULL,
226   `codigoItem` tinyint(3) unsigned NOT NULL,
227   `valor` decimal(18,2) NOT NULL,
228   PRIMARY KEY (`numeroCliente`,`codigoItem`),

```

```

229 KEY `FK_itemseconomicosclientes2` (`codigoItem`),
230 CONSTRAINT `FK_itemseconomicosclientes` FOREIGN KEY
    (`numeroCliente`) REFERENCES `clientes` (`numeroCliente`),
231 CONSTRAINT `FK_itemseconomicosclientes2` FOREIGN KEY
    (`codigoItem`) REFERENCES `itemseconomicos` (`codigo`)
232 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
233 /*Table structure for table `itemspatrimoniales` */
234 DROP TABLE IF EXISTS `itemspatrimoniales`;
235 CREATE TABLE `itemspatrimoniales` (
236   `codigo` tinyint(3) unsigned NOT NULL,
237   `nombre` varchar(20) COLLATE utf8_spanish2_ci NOT NULL,
238   `esPasivo` tinyint(3) unsigned NOT NULL,
239   PRIMARY KEY (`codigo`)
240 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
241 /*Table structure for table `itemspatrimonialesclientes` */
242 DROP TABLE IF EXISTS `itemspatrimonialesclientes`;
243 CREATE TABLE `itemspatrimonialesclientes` (
244   `numeroCliente` int(11) NOT NULL,
245   `codigoItem` tinyint(3) unsigned NOT NULL,
246   `valor` decimal(18,2) NOT NULL,
247   `descripcion` varchar(100) COLLATE utf8_spanish2_ci NOT NULL,
248   PRIMARY KEY (`numeroCliente`,`codigoItem`),
249   KEY `FK_itemspatrimonialesclientes2` (`codigoItem`),
250   CONSTRAINT `FK_itemspatrimonialesclientes` FOREIGN KEY
    (`numeroCliente`) REFERENCES `clientes` (`numeroCliente`),
251   CONSTRAINT `FK_itemspatrimonialesclientes2` FOREIGN KEY
    (`codigoItem`) REFERENCES `itemspatrimoniales` (`codigo`)
252 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
253 /*Table structure for table `productos` */

```

```

254 DROP TABLE IF EXISTS `productos`;
255 CREATE TABLE `productos` (
256   `codigo` char(10) COLLATE utf8_spanish2_ci NOT NULL,
257   `nombre` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
258   PRIMARY KEY (`codigo`)
259 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
260 /*Table structure for table `profesiones` */
261 DROP TABLE IF EXISTS `profesiones`;
262 CREATE TABLE `profesiones` (
263   `codigo` varchar(10) COLLATE utf8_spanish2_ci NOT NULL,
264   `nombre` varchar(50) COLLATE utf8_spanish2_ci DEFAULT NULL,
265   PRIMARY KEY (`codigo`)
266 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
267 /*Table structure for table `provincias` */
268 DROP TABLE IF EXISTS `provincias`;
269 CREATE TABLE `provincias` (
270   `codigo` char(10) COLLATE utf8_spanish2_ci NOT NULL,
271   `nombre` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
272   PRIMARY KEY (`codigo`)
273 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
274 /*Table structure for table `referenciasbancarias` */
275 DROP TABLE IF EXISTS `referenciasbancarias`;
276 CREATE TABLE `referenciasbancarias` (
277   `numeroCliente` int(11) NOT NULL,
278   `prioridad` tinyint(3) unsigned NOT NULL,
279   `nombreBanco` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
280   `numeroCuenta` varchar(20) COLLATE utf8_spanish2_ci NOT NULL,
281   `tipoCuenta` char(1) COLLATE utf8_spanish2_ci NOT NULL,
282   PRIMARY KEY (`numeroCliente`),

```

```

283 CONSTRAINT `FK_referenciasbancarias` FOREIGN KEY
      (`numeroCliente`) REFERENCES `clientes` (`numeroCliente`)
284 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
      COLLATE=utf8_spanish2_ci;
285 /*Table structure for table `referenciascomerciales` */
286 DROP TABLE IF EXISTS `referenciascomerciales`;
287 CREATE TABLE `referenciascomerciales` (
288   `numeroCliente` int(11) NOT NULL,
289   `prioridad` tinyint(3) unsigned NOT NULL,
290   `nombre` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
291   `direccion` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
292   `telefono` varchar(20) COLLATE utf8_spanish2_ci NOT NULL,
293   PRIMARY KEY (`numeroCliente`),
294   CONSTRAINT `FK_referenciascomerciales` FOREIGN KEY
      (`numeroCliente`) REFERENCES `clientes` (`numeroCliente`)
295 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
      COLLATE=utf8_spanish2_ci;
296 /*Table structure for table `referenciaspersonales` */
297 DROP TABLE IF EXISTS `referenciaspersonales`;
298 CREATE TABLE `referenciaspersonales` (
299   `numeroCliente` int(11) NOT NULL,
300   `prioridad` tinyint(3) unsigned NOT NULL,
301   `nombre` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
302   `direccion` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
303   `telefono` varchar(20) COLLATE utf8_spanish2_ci NOT NULL,
304   `parentesco` varchar(20) COLLATE utf8_spanish2_ci NOT NULL,
305   PRIMARY KEY (`numeroCliente`),
306   CONSTRAINT `FK_referenciaspersonales` FOREIGN KEY
      (`numeroCliente`) REFERENCES `clientes` (`numeroCliente`)
307 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
      COLLATE=utf8_spanish2_ci;
308 /*Table structure for table `solicitudes` */

```

```

309 DROP TABLE IF EXISTS `solicitudes`;
310 CREATE TABLE `solicitudes` (
311   `numeroSolicitud` int(11) NOT NULL,
312   `secuencial` int(11) NOT NULL,
313   `numeroCliente` int(11) NOT NULL,
314   `codigoProducto` char(10) COLLATE utf8_spanish2_ci NOT NULL,
315   `estado` char(1) COLLATE utf8_spanish2_ci NOT NULL,
316   `codigoTipoPrestamo` char(10) COLLATE utf8_spanish2_ci NOT NULL,
317   `codigoTipoCalificacion` tinyint(3) unsigned NOT NULL,
318   `montoSolicitado` decimal(18,2) NOT NULL,
319   `montoAprobado` decimal(18,2) NOT NULL,
320   `montoEncajadoAhorro` decimal(18,2) NOT NULL,
321   `tasaInteres` decimal(10,2) NOT NULL,
322   `numeroCuotas` smallint(6) NOT NULL,
323   `valorCuota` decimal(18,2) NOT NULL,
324   `frecuenciaPago` smallint(6) NOT NULL,
325   `fechaSolicitud` mediumblob NOT NULL,
326   `fechaDefinicion` mediumblob NOT NULL,
327   `codigoUsuario` char(10) COLLATE utf8_spanish2_ci NOT NULL,
328   `codigoOficina` tinyint(3) unsigned NOT NULL,
329   `numeroGarantes` tinyint(3) unsigned NOT NULL,
330   `numeroGarantias` tinyint(3) unsigned NOT NULL,
331   `codigoDestinoFinanciero` char(10) COLLATE utf8_spanish2_ci NOT
NULL,
332   `codigoDestinoInterno` char(10) COLLATE utf8_spanish2_ci NOT NULL,
333   `comentario` mediumtext COLLATE utf8_spanish2_ci NOT NULL,
334   `puedeContinuar` tinyint(3) unsigned NOT NULL,
335   `enviada` tinyint(3) unsigned NOT NULL,
336   `comentarioDevuelve` varchar(200) COLLATE utf8_spanish2_ci NOT
NULL,
337   `codigoUsuarioAfecto` char(10) COLLATE utf8_spanish2_ci NOT NULL,
338   `esVinculada` tinyint(3) unsigned NOT NULL,

```

```

339 `montoEncajadoCertificado` decimal(18,2) DEFAULT NULL,
340 `montoEncajadoInversion` decimal(18,2) DEFAULT NULL,
341 `codigoTipoRecurso` varchar(50) COLLATE utf8_spanish2_ci NOT
NULL,
342 `codigoOficialAsesor` char(10) COLLATE utf8_spanish2_ci DEFAULT
NULL,
343 PRIMARY KEY (`numeroSolicitud`),
344 KEY `FK_solicitudes` (`codigoProducto`),
345 KEY `FK_solicitudes2` (`codigoTipoPrestamo`),
346 CONSTRAINT `FK_solicitudes` FOREIGN KEY (`codigoProducto`)
REFERENCES `productos` (`codigo`),
347 CONSTRAINT `FK_solicitudes2` FOREIGN KEY
(`codigoTipoPrestamo`) REFERENCES `tiposprestamos` (`codigo`)
348 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
COLLATE=utf8_spanish2_ci;
349 /*Table structure for table `tiposprestamos` */
350 DROP TABLE IF EXISTS `tiposprestamos`;
351 CREATE TABLE `tiposprestamos` (
352 `codigo` char(10) COLLATE utf8_spanish2_ci NOT NULL,
353 `nombre` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
354 `siglas` varchar(5) COLLATE utf8_spanish2_ci NOT NULL,
355 `montoMaximo` decimal(18,2) NOT NULL,
356 `plazoMaximo` int(11) NOT NULL,
357 `requiereGarantes` tinyint(3) unsigned NOT NULL,
358 `requiereGarantias` tinyint(3) unsigned NOT NULL,
359 `proporcion` decimal(18,2) NOT NULL,
360 `estaActivo` tinyint(3) unsigned NOT NULL,
361 `codigoOrigenRecurso` char(10) COLLATE utf8_spanish2_ci NOT NULL,
362 `tipoProducto` tinyint(3) unsigned NOT NULL,
363 `encajaCertificados` tinyint(3) unsigned NOT NULL,
364 `requiereVerificacion` tinyint(3) unsigned NOT NULL,
365 `soloCertificados` tinyint(3) unsigned NOT NULL,

```

```

366 `minimaCalificacionSujetoCredito` decimal(18,0) NOT NULL,
367 `aprobacionDirecta` tinyint(3) unsigned NOT NULL,
368 PRIMARY KEY (`codigo`),
369 KEY `FK_TiposPrestamos_OrigenesRecursos` (`codigoOrigenRecurso`)
370 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
371 /*Table structure for table `Web_accesos` */
372 DROP TABLE IF EXISTS `Web_accesos`;
373 CREATE TABLE `Web_accesos` (
374 `ID` int(11) NOT NULL AUTO_INCREMENT COMMENT 'VALOR
    AUTOINCREMENTAL',
375 `IDUSUARIO` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
376 `IPACCESO` varchar(100) COLLATE utf8_spanish2_ci NOT NULL,
377 `FECHA` datetime NOT NULL,
378 `DESCRIPCION` text COLLATE utf8_spanish2_ci NOT NULL,
379 PRIMARY KEY (`ID`,`IDUSUARIO`),
380 UNIQUE KEY `ID` (`ID`),
381 KEY `FK_Web_accesos` (`IDUSUARIO`),
382 CONSTRAINT `FK_Web_accesos` FOREIGN KEY (`IDUSUARIO`)
    REFERENCES `Web_logins` (`LOGIN`)
383 ) ENGINE=InnoDB AUTO_INCREMENT=108 DEFAULT
    CHARSET=utf8 COLLATE=utf8_spanish2_ci;
384 /*Table structure for table `Web_logins` */
385 DROP TABLE IF EXISTS `Web_logins`;
386 CREATE TABLE `Web_logins` (
387 `LOGIN` varchar(50) COLLATE utf8_spanish2_ci NOT NULL
    COMMENT 'USUARIO',
388 `CLAVE` varchar(50) COLLATE utf8_spanish2_ci NOT NULL
    COMMENT 'CLAVE',
389 `CREADOR` varchar(20) COLLATE utf8_spanish2_ci NOT NULL
    COMMENT 'QUIEN LO CREO ',

```

```

390 `DESCRIPCION` varchar(250) COLLATE utf8_spanish2_ci NOT NULL
    COMMENT 'QUIEN SERA ',
391 `FECHA_CREACION` date NOT NULL COMMENT 'FECHA DE
    CREACION ',
392 `FECHA_MODIFICADO` date DEFAULT NULL COMMENT 'FECHA
    DE MODIFICAION ',
393 `ESTADO_SISTEMA` int(11) DEFAULT '0',
394 `OFICINA` varchar(50) COLLATE utf8_spanish2_ci DEFAULT NULL,
395 `ESTA_ACTIVO` int(2) DEFAULT '1',
396 `EMAIL` varchar(50) COLLATE utf8_spanish2_ci DEFAULT NULL,
397 PRIMARY KEY (`LOGIN`),
398 UNIQUE KEY `EMAIL` (`EMAIL`),
399 CONSTRAINT `FK_Web_logins` FOREIGN KEY (`LOGIN`)
    REFERENCES `clientes` (`codigo`)
400 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
401 /*Table structure for table `Web_logins_roles` */
402 DROP TABLE IF EXISTS `Web_logins_roles`;
403 CREATE TABLE `Web_logins_roles` (
404 `IDUSUARIO` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
405 `IDROL` int(11) NOT NULL,
406 `ACTIVO` int(5) DEFAULT '1',
407 PRIMARY KEY (`IDUSUARIO`,`IDROL`),
408 KEY `NewIndex1` (`IDUSUARIO`),
409 KEY `FK_Web_logins_roles_2` (`IDROL`),
410 CONSTRAINT `FK_Web_logins_roles` FOREIGN KEY (`IDUSUARIO`)
    REFERENCES `Web_logins` (`LOGIN`),
411 CONSTRAINT `FK_Web_logins_roles_2` FOREIGN KEY (`IDROL`)
    REFERENCES `Web_roles` (`ID`)
412 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
    COLLATE=utf8_spanish2_ci;
413 /*Table structure for table `Web_noticias` */

```

```

414 DROP TABLE IF EXISTS `Web_noticias`;
415 CREATE TABLE `Web_noticias` (
416   `ID` int(11) NOT NULL AUTO_INCREMENT,
417   `TITULO` varchar(255) COLLATE utf8_spanish2_ci NOT NULL,
418   `IMAGEN` varchar(255) COLLATE utf8_spanish2_ci NOT NULL,
419   `CONTENIDO` text COLLATE utf8_spanish2_ci NOT NULL,
420   `PUBLICADO` int(11) NOT NULL DEFAULT '0',
421   `AUTOR` varchar(200) COLLATE utf8_spanish2_ci NOT NULL,
422   `FECHA_INGRESO` date NOT NULL,
423   `FECHA_PUBLICADO` date DEFAULT NULL,
424   `FECHA_MODIFICACION` date DEFAULT NULL,
425   `MODIFICA` varchar(200) COLLATE utf8_spanish2_ci DEFAULT
      NULL,
426   `IDUSUARIO` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
427   PRIMARY KEY (`ID`,`TITULO`,`IDUSUARIO`),
428   KEY `FK_NOTICIAS` (`IDUSUARIO`),
429   CONSTRAINT `FK_Web_noticias` FOREIGN KEY (`IDUSUARIO`)
      REFERENCES `Web_logins` (`LOGIN`)
430 ) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8
      COLLATE=utf8_spanish2_ci;
431 /*Table structure for table `Web_roles` */
432 DROP TABLE IF EXISTS `Web_roles`;
433 CREATE TABLE `Web_roles` (
434   `ID` int(11) NOT NULL AUTO_INCREMENT,
435   `NOMBRE` varchar(50) COLLATE utf8_spanish2_ci DEFAULT NULL,
436   PRIMARY KEY (`ID`)
437 ) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8
      COLLATE=utf8_spanish2_ci;
438 /*Table structure for table `zonas` */
439 DROP TABLE IF EXISTS `zonas`;
440 CREATE TABLE `zonas` (
441   `codigo` char(10) COLLATE utf8_spanish2_ci NOT NULL,

```

```

442 `nombre` varchar(50) COLLATE utf8_spanish2_ci NOT NULL,
443 `secuencialCanton` int(11) NOT NULL,
444 `secuencial` int(11) NOT NULL,
445 PRIMARY KEY (`secuencial`),
446 KEY `FK_Zonas_Cantones` (`secuencialCanton`),
447 CONSTRAINT `FK_zonas` FOREIGN KEY (`secuencialCanton`)
REFERENCES `cantones` (`secuencial`),
448 CONSTRAINT `zonas_ibfk_1` FOREIGN KEY (`secuencialCanton`)
REFERENCES `cantones` (`secuencial`),
449 CONSTRAINT `zonas_ibfk_2` FOREIGN KEY (`secuencialCanton`)
REFERENCES `cantones` (`secuencial`)
450 ) ENGINE=InnoDB DEFAULT CHARSET=utf8
COLLATE=utf8_spanish2_ci;
451 /*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
452 /*!40014 SET
FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
453 /*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
454 /*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

```

6.4 Implantación

La Cooperativa de ahorro y Crédito “CHIBULEO” Ltda., a través de su representante el Dr. Rodrigo Llambo realizó la adquisición del dominio <http://www.chibuleo.fin.ec> para alojar las páginas Web correspondiente a la parte pública del Sistema, ya que lo que se refiere a los archivos xml que se generan en base a peticiones de los usuarios y el WS (archivo asmx) que se creó para la obtención de datos desde el servidor institucional se almacenaran en el servidor LINUX de la misma permitiendo el acceso solo a <http://www.chibuleo.fin.ec>.

La subida de datos al hosting de Internet se la realiza mediante Filezilla, una herramienta libre y muy ampliamente utilizada por su versatilidad y facilidad de uso, ya que su funcionamiento es mediante el protocolo FTP.

6.4.1 Características del servidor

A continuación se detalla las características del equipo en el que se encuentra alojado el Sistema en línea:

PHP Version 5.2.13

System	Linux 2.6.18-194.3.1.el5.028stab069.6PAE #1 SMP Wed May 26 19:09:08 MSD 2010 i686
Build Date	Jun 11 2010 13:27:27
Configure Command	'./configure' '--enable-bcmath' '--enable-calendar' '--enable-dbase' '--enable-exif' '--enable-force-cgi-redirect' '--enable-ftp' '--enable-gd-native-ttf' '--enable-libxml' '--enable-magic-quotes' '--enable-mbstring' '--enable-pdo=shared' '--enable-soap' '--enable-sockets' '--enable-zend-multibyte' '--enable-zip' '--prefix=/usr' '--with-curl=/opt/curlssl/' '--with-

	curlwrappers' '--with-freetype-dir=/usr' '--with-gd' '--with-gettext' '--with-imap=/opt/php_with_imap_client/' '--with-imap-ssl=/usr' '--with-jpeg-dir=/usr' '--with-kerberos' '--with-libexpat-dir=/usr' '--with-libxml-dir=/opt/xml2' '--with-libxml-dir=/opt/xml2/' '--with-mcrypt=/opt/libmcrypt/' '--with-mhash=/opt/mhash/' '--with-mysql=/usr' '--with-mysql-sock=/var/lib/mysql/mysql.sock' '--with-mysqli=/usr/bin/mysql_config' '--with-openssl=/usr' '--with-openssl-dir=/usr' '--with-pcre-regex=/opt/pcre' '--with-pdo-mysql=shared' '--with-pdo-sqlite=shared' '--with-png-dir=/usr' '--with-pspell' '--with-sqlite=shared' '--with-tidy=/opt/tidy/' '--with-ttf' '--with-xmlrpc' '--with-xpm-dir=/usr' '--with-xsl=/opt/xslt/' '--with-zlib' '--with-zlib-dir=/usr'
Server API	CGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/usr/lib
Loaded Configuration File	/home/chibuleo/phpconf/php.ini
Scan this dir for additional .ini files	(none)
additional .ini files parsed	(none)
PHP API	20041225
PHP Extension	20060613
Zend Extension	220060519
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled

IPv6 Support	enabled
Registered PHP Streams	compress.zlib, dict, ftp, ftps, http, https, imap, imaps, pop3, pop3s, rtsp, smtp, smtps, telnet, tftp, php, file, data, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, sslv2, tls
Registered Stream Filters	zlib.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed
Almacenamiento HD de Espacio:	2.000 MB
Transferencia mensual:	20.000 MB
Velocidad de Servidores (mínimo):	6100 MHZ
Server - Memoria (mínimo):	8 GB
Sistema Operativo :	Centos
Cuentas POP3 / FTP :	SI
Acceso via Web :	SI
Auto - Contestadores :	SI
Bloqueo de Correo no Solicitado-Spam :	Ilimitado
Reenvio de Correo :	Ilimitado
Uso del FTP :	SI
Base de Datos - Mysql 4 - 5 :	SI
Protección con	SI

Contraseña :	
Soporta CGI Scripting :	SI
Soporta Php4 - 5 :	SI
Soporta FrontPage 2002 :	SI
Estadísticas de Acceso:	SI
Servidor Apache:	SI
Sendmail :	SI
Perl :	
Server Side Include	SI
Backups auto-administrados :	SI

Tabla 6.40 Características del Servidor

Para un correcto funcionamiento del sistema en línea se recomienda que los equipos con los que se va a trabajar:

Equipo	Características
PCU	2 GHz o superior
Sistema operativo	Soporte en sistema Operativo Windows XP o superior.
Navegadores	IE 6 en adelante Mozilla 2.0 en adelante Opera 9 en adelante y Google Chrome 1 en adelante
Ancho de banda	256 kbps

Tabla 6.41 Características del Equipo Cliente

6.5 Pruebas

6.5.1 Pruebas de caja blanca

Llamadas también pruebas de cobertura, nos ayudaron a determinar posibles fallas o errores en la estructura interna y funcionamiento del portal Web, para lo cual se avaluaron algunos puntos vulnerables a tomarse en cuenta:

- Se revisó y ejecutó (funciones y/o procedimientos que están incluidos pero que nunca son ejecutadas por el programa)
- Se evaluó todas las decisiones lógicas y bucles presentes en los diferentes módulos.
- Se revisó y evaluó la estructura de cada clase y sus llamadas correspondientes.
- Se procedió a la revisión y evaluación del manejo de errores y manejo de validaciones.

Para poder obtener resultados óptimos en cada uno de los puntos antes mencionados se procedió a la técnica de **“Suponiendo”**, es decir **“Qué pasaría si se hiciera esto”**, con lo cual se obtuvo una reducción considerable en los errores del sistema Online.

6.5.2 Prueba de caja negra

También llamadas pruebas de caja opaca o pruebas funcionales, tienen como finalidad intentar encontrar casos en que el módulo no se atiene a su especificación, resultando una prueba de mucha ayuda sobre la interfaz del portal.

Con estas pruebas se realizó y comprobó lo siguiente:

- Ingreso y salida de datos por parte de los usuarios y clientes.
- Comprobar que cada módulo o sección se ajuste a los requerimientos.

- Que la información a presentarse sea la correcta y se visualice de forma adecuada.
- Que la información sea reservada cuando esta sea necesaria.

6.5.3 Pruebas de verificación y validación

Las pruebas de verificación y validación nos ayudaron en el proceso de revisión del portal Web para ver si este cumple con las especificaciones y requerimientos planteados al inicio. Se utilizaron técnicas tales como evaluaciones, inspecciones, y tutoriales.

Con estas pruebas se evaluó el sistema durante y al final del desarrollo realizándonos la siguiente pregunta: ¿Es esto lo que el cliente quiere?

6.6 Conclusiones y Recomendaciones

6.6.1 Conclusiones

- La manera de llevar toda la información es de forma centralizada y segura pero poco accesible de manera rápida para los clientes por lo que con la implementación de los servicios en línea se dio mayor apertura a la información y servicios por parte de la Cooperativa hacia sus usuarios y clientes ya no de manera única en la Cooperativa o Sucursales.
- El trabajo de manera colectiva fue de gran ayuda para el desarrollo del portal Web, ya que las personas involucradas dieron sus puntos de vista y ayudaron a mejorar y hacer más fácil el desarrollo del proyecto. Se contó primordialmente con el apoyo del departamento de Sistemas.
- La información presentada en el sistema Web cuenta con una interfaz agradable para el usuario o cliente y de fácil acceso. Además la misma

posee seguridades de acceso e integridad en la visualización de datos referente a la información presentada.

- Los formatos para ingreso de información por parte de los clientes se realizaron de la manera sencilla pero respetando los estándares institucionales del sistema que se maneja internamente.
- Los módulos se han desarrollado tomando en consideración aspectos fundamentales de la forma de trabajo de la empresa y necesidades primordiales de los usuarios/clientes de la misma.
- El sistema en línea que se ha desarrollado y con software libre (php, mysql), la parte que se ha visualiza en internet por motivos de fiabilidad, flexibilidad y tiempo de respuesta, además que su no tiene costo su uso tanto para pruebas como para implementación.
- En la parte de seguridades se ha implementado en tres niveles que son por medio de sesiones encriptadas, bloqueo de usuario en la base datos y cookie personalizada para el tiempo de uso del sistema.
- El uso de software adecuado y técnicas de programación de nueva generación fueron de un gran aporte, no solo para la elaboración de los módulos sino que también se los puede tomar como punto de partida en proyectos futuros que se plantea la Institución, debido que está desarrollado mediante metodología MVC (Es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. Modelo, Vista y Controlador) y su código se implementó con clases en las áreas de modelo, controladores y por medio de ello es reutilizable en todas sus áreas implementadas.
- El WS se lo ha desarrollado con C# por razones institucionales, debido a que su sistema financiero interno está desarrollado con tecnología .Net y soportado en una base de datos SQL Server y pueda ser reutilizado tanto por la institución como por el sistema Online de MySQL-PHP y se lo ha implementado con Mono Develop en el servidor público de la Cooperativa permitiendo con esto la interacción de la parte publicada en internet con el

servidor de base de datos de la Cooperativa sin realizar ninguna comando DML o DQL de manera directa al servidor Institucional.

- El acceso a la información privada de cada cliente que proviene del servidor Institucional cuando se realiza una conexión con el WS jamás es publicado en el servidor público de Internet sino que se almacena en el servidor de la Cooperativa.
- Para el acceso a la información de la Institución se lo realiza mediante la conexión archivos XML generados por el WS desde la IP pública del lugar donde está almacenado el sistema en línea a la IP pública de la Cooperativa.

6.6.2 RECOMENDACIONES

- Al personal que deba administrar el sistema online debe estar al tanto del funcionamiento de cada una de las módulos que contiene, los administradores deben ser capaces de entender toda información que se a ingresada, y se debe estar siempre en constante monitoreo, para que la información o formularios que se han ingresado puedan ser procesados rápidamente y exista un alto grado de satisfacción por parte de los clientes.
- El administrador del sitio Web debe tener el criterio suficiente en lo que a asignación de roles de usuario se refiere, ya que si un rol es mal asignado podría causar un uso inadecuado de la información y hasta pérdida de la misma.
- El sistema Web fue probado en su mayoría por navegadores como el Opera, Firefox (desde la versión 2 en adelante) e Internet Explorer (desde la versión 6 en adelante), por lo que se recomienda el uso de estos, para que el portal Web se distorsione lo mínimo posible. También cabe indicar que el portal esta optimizado para una resolución de 1152 x 864.
- Para proyectos de desarrollo se debe tomar en cuenta en primer lugar estándares internacionales, se recomienda el uso de los mismos para que el

proyecto sea más fácil de desarrollar y más entendible en el caso de alguna modificación por el desarrollador o cualquier otra persona.

- Para mayor seguridad de la información que se está presentando a los clientes se recomienda la compra de certificación SSL (Secure Socket Layer) en base a los resultados que los módulos ofrezcan a la Cooperativa “Chibuleo” a corto o larga plazo y en dependencia la ampliación de los servicios que se implementen.

Bibliografía

Libros

- KENDALL y KENDALL (1991) *Análisis y Diseño de Sistemas*. Hispanoamérica S.A., México
- KORTH, Henry F. y SILBERSCHATZ, Abraham. (1993) *Fundamentos de bases de datos*. Madrid: McGraw-Hill
- MIGUEL, Adoración y PIATTINI, Mario G. (1997) *Fundamentos y modelos de bases de datos*. Madrid: RA-MA
- GUTIERREZ G. Juan D. (2004) *Desarrollo Web con PHP 5 y MySQL*. Ediciones Amaya, Madrid.

- JAMSA Kris, KING Konrad y ANDERSON (2002) *Andy Superutilidades para HTML y Diseño Web*. Mc Graw Hill / Interamericana de España, S.A.U.
- SPONA, Helma (2010) *Programación de bases de datos con MYSQL y PHP*. Madrid Marcomb.
- PAVON Puertas (2007) *Creación de un portal Web con PHP y MYSQL* 3ª EDICIÓN. Madrid Ra-Ma

Internet

Internet

- Wikipedia la enciclopedia libre.(2010, septiembre 5) Conexión a internet[Online].Disponible:http://es.wikipedia.org/wiki/Acceso_a_Internet
- Información Online sobre tecnologías. (2010, agosto 10) Tipo de Conexiones a Internet [Online]. Disponible: <http://www.idg.es/iworld/impart.asp?id=28672>
- Monografias.com (2010, agosto 01) Definición de Internet [Online]. Disponible: <http://www.monografias.com/trabajos81/introduccion-internet/introduccion-internet.shtml>

Concepto de Internet, su historia, evolución, servicios y difusión.

- Wikipedia la enciclopedia libre.(2010, agosto 5) World Wide Web [Online]. Disponible: http://es.wikipedia.org/wiki/World_Wide_Web
- Ciudad de la Información. (2010, julio 10) Internet de cerquita [Online]. Disponible: <http://www.ciberhabitat.gob.mx/museo/cerquita/ic03.htm>

Definición de www y su funcionamiento

- Servicios y recursos para tener éxito(2010, junio 20) Que es un sitio Web [Online]. Disponible: <http://www.masadelante.com/faq-sitio-Web.htm>

- Wikipedia la enciclopedia libre. (2010, junio 1) La WWW [Online].
Disponible: [http://es.wikipedia.org/wiki/Portal_\(internet\)](http://es.wikipedia.org/wiki/Portal_(internet))

Definiciones de sitio Web y pagina Web

- Wikipedia la enciclopedia libre. (2010, junio 1) Portal Web [Online].
Disponible: <http://es.wikipedia.org/wiki/.php>
- <http://www.desarrolloweb.com/articulos/392.php>
- <http://php.net/manual/es/intro-what-is.php>

Concepto de PHP, MySQL sus usos y características

- Desarrolloweb.com. (2010, junio 20) Que es PHP [Online].
<http://www.espestudio.com/articulo/desarrollo-Web/bases-de-datos-mysql/Que-es-MySQL.htm>
- MySQL y PostGres. (2010, Julio 20) MySQL [Online]. Disponible:
http://www.netpecos.org/docs/mysql_postgres/x57.html
- MySQL. (2010, agosto 01) Panorama MySQL [Online]. Disponible :
<http://dev.mysql.com/doc/refman/5.0/es/what-is-mysql-ab.html>
- Htmlpoint.com. (2010, febrero 13) MySQL y Apache [Online].
Disponible: <http://www.htmlpoint.com/faq/apache/01.htm>

Concepto de Apache y sus características

- Tecnologías Web. (2010, marzo 15) Apache [Online] Disponible:
<http://www.versys.com/apache.htm>
- Más adelante.com. (2010, agosto 20) Que es un servidor Web [Online].
Disponible: <http://www.masadelante.com/faq-servidor-Web.htm>
- Glosario.net. (2010, marzo 19) Definición de Web Server [Online].
Disponible: <http://tecnologia.glosario.net/terminos-tecnicos-internet/Web-server-1752.html>

Definiciones de HTML server

- Web server.es. (2010, mayo 01) Que es HTML [Online]. Disponible :
<http://es.html.net/tutorials/html/lesson2.asp>

- Wikipedia la enciclopedia libre. (2010, mayo 11) Qu es HTML [Online]. Disponible :<http://es.wikipedia.org/wiki/HTML>
- Aplicaciones Web. (2010, marzo 20) Manual de HTML [Online]. Disponible: <http://www-app.etsit.upm.es/~alvaro/manual/manual.html>

Jquery

- Jquery the write less. (2010,septiembre 01) Is a new kind JavaScript library [Online]. Disponible :<http://jquery.com/>
- Jquery Api. (2010, septiembre 10) Jquery Api [Online]. Disponible: <http://api.jquery.com/>
- Jquery Api references. (2010, septiebre 11) Jquery Api references [Online]. Disponible: <http://www.oscarotero.com/jquery/>
- Docs Jquery. (2010, septiembre 2) Main page jquery [Online]. Disponible: http://docs.jquery.com/Main_Page

Conceptos y Usos de Jquey

- Wikipedia la enciclopedia libre. (2010, agosto 4) Extensible Market Language [Online]. Disponible: http://es.wikipedia.org/wiki/Extensible_Markup_Language

Conceptos, formas de creación de XML

- Web Server. (2010, junio 10) .net Web Services [Online]. Disponible:http://www.codeguru.com/csharp/csharp/cs_webservices/tutorials/article.php/c5477
- <http://www.subgurim.net/Articulos/visual-Web-developer/112/crear-y-consumir-Web-services-con-visual-Web-developer.aspx>

- W3C reglas de la web. (2010, junio 1) Extensible Markup Language [Online]. Disponible: <http://www.w3.org/XML/>
- W3C reglas de la web. (2010, julio 20) Tecnologías Web [Online]. Disponible: <http://www.w3c.es/divulgacion/guiasbreves/tecnologiasxml>

Definiciones, Creación De WS

- Consumo de Web services. (2010, mayo 19) Consumo de web services con PHP [Online]. Disponible: <http://bytes.com/topic/php/answers/531394-consuming-net-webservice-php-nusoap>
- Experts-exchange.com. (2010, marzo 14) Como consumir web services asmx [Online]. Disponible : http://www.experts-exchange.com/Web_Development/Web_Languages-Standards/PHP/Q_24184346.html

Glosario de términos utilizados

Browser.- Un navegador o navegador Web (del inglés, *Web browser*) es un programa que permite ver la información que contiene una página Web (ya se encuentre ésta alojada en un servidor dentro de la World Wide Web o en un servidor local).

Dirección IP.- Una dirección IP es una etiqueta numérica que identifica, de manera lógica y jerárquica, a una interfaz (elemento de comunicación/conexión) de un dispositivo (habitualmente una computadora) dentro de una red que utilice el protocolo IP (Internet Protocol), que corresponde al nivel de red del protocolo TCP/IP. Dicho número no se ha de confundir con la dirección MAC que es un número hexadecimal fijo que es asignado a la tarjeta o dispositivo de red por el fabricante, mientras que la dirección IP se puede cambiar.

Dominio de Internet.- Un dominio de Internet es un nombre base que agrupa a un conjunto de equipos o dispositivos y que permite proporcionar nombres de equipo más fácilmente recordables en lugar de una dirección IP numérica.

Hipertext.- O Hipertexto, es el nombre que recibe el texto que en la pantalla de una computadora conduce a su usuario a otro texto relacionado. Es un texto que nos permite enlazarnos con alguna otra parte del documento.

HTTP.- El propósito del protocolo HTTP es permitir la transferencia de archivos (principalmente, en formato HTML). Entre un navegador (el cliente) y un servidor Web (denominado, entre otros, httpd en equipos UNIX) localizado mediante una cadena de caracteres denominada dirección URL.

Interfaz.- Es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo, normalmente suelen ser fáciles de entender y fáciles de accionar.

Links.- Es el botón o una sección resaltada de un texto que cuando es señalizada establece una conexión con otra página de Internet.

Framework.- Un framework, en el desarrollo de software, es una estructura de soporte definida mediante la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

FTP.- Es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP, basado en la arquitectura cliente-servidor. Desde un equipo cliente se puede conectar a un servidor para descargar archivos desde él o

para enviarle archivos, independientemente del sistema operativo utilizado en cada equipo.

Módulo.- Un módulo es un componente autocontrolado de un sistema, el cual posee una interfaz bien definida hacia otros componentes; algo es modular si es construido de manera tal que se facilite su ensamblaje, acomodamiento flexible y reparación de sus componentes.

Opera.- Opera es un navegador Web y suite de Internet creado por la empresa noruega Opera Software. La aplicación es gratuita desde su versión 8.50, habiendo sido previamente shareware y, antes de su versión 5.0, únicamente de pago.

TCP/IP.- Es la base de Internet, y sirve para enlazar computadoras que utilizan diferentes sistemas operativos, incluyendo PC, minicomputadoras y computadoras centrales sobre redes de área local (LAN) y área extensa (WAN).

Usuario.- Persona que tiene una cuenta en una determinada computadora por medio de la cual puede acceder a los recursos y servicios que ofrece una red. Puede ser tanto usuario de correo electrónico como de acceso al servidor en modo terminal.

Ciente.- Es quien accede a un producto o servicio por medio de una transacción financiera (dinero) u otro medio de pago. Quien compra, es el comprador, y quien consume el consumidor. Normalmente, cliente, comprador y consumidor son la misma persona.

XML.- Es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML).

DOM.- El Document Object Model (una traducción al español no literal, pero apropiada, podría ser modelo en objetos para la representación de documentos o también modelo de objetos del documento), abreviado DOM, es esencialmente una interfaz de programación de aplicaciones que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos. A través del DOM, los programas pueden acceder y modificar el contenido, estructura y estilo de los documentos HTML y XML, que es para lo que se diseñó principalmente.

Evento.- Define como la ejecución de los programas que van determinados por los sucesos que ocurran en el sistema, definidos por el usuario o que ellos mismos provoquen.

JavaScript.- Es un lenguaje de scripting basado en objetos no tipado y liviano, utilizado para acceder a objetos en aplicaciones. Principalmente, se utiliza integrado en un navegador Web permitiendo el desarrollo de interfaces de usuario mejoradas y páginas Web dinámicas. JavaScript es un dialecto de ECMA-Script y se caracteriza por ser un lenguaje basado en prototipos, con entrada dinámica y con funciones de primera clase. JavaScript ha tenido influencia de múltiples lenguajes y se diseñó con una sintaxis similar al lenguaje de programación Java, aunque más fácil de utilizar para personas que no programan.

JQuery.- JQuery es una biblioteca o framework de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la tecnología AJAX a páginas Web. Fue presentada el 14 de enero de 2006 en el BarCamp NYC.

JQuery es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privativos. JQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de

mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

Ajax.- Ajax, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo Web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se requieren al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. JavaScript es el lenguaje interpretado (scripting language) en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante XMLHttpRequest, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML.

Ajax es una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores dados que está basado en estándares abiertos como JavaScript y Document Object Model (DOM).

Función \$().- La forma de interactuar de jquery con la página es mediante la función \$(), un alias de JQuery(), que recibe como parámetro una expresión CSS o el nombre de una etiqueta HTML y devuelve todos los nodos (elementos) que concuerden con la expresión.

```
$("#tablaAlumnos") // Devolverá el elemento con id="tablaAlumnos"  
$(".activo") // Devolverá una matriz de elementos con class="activo"
```

Una vez obtenidos los nodos, se les puede aplicar cualquiera de las funciones que facilita la biblioteca.

```
// Se elimina el estilo (con removeClass()) y se aplica uno nuevo (con addClass())  
a todos los nodos con class="activo"  
$(".activo").removeClass("activo").addClass("inactivo");
```

ANEXOS

ANEXOS 1

CONTRIBUCIÓN DE VALOR

ADMINISTRACIÓN INTERNET FINANCIERO

Lenguaje De Programación Y Software Especializado

WS para Ingreso de formularios de:
Apertura de cuenta,
Prestamos, Consulta de: saldos, depósitos,
Pagos de Créditos e Ingreso de Peticiones de Certificados

45

Acceso a los Servicios para Mercadeo

BENEFICIAR a los Cliente en servicios de Depósitos, Préstamos, Créditos.

Variable Independiente



Variable Dependiente

ANEXO 2

ENCUESTA

Cooperativa de Ahorro y Crédito Chibuleo



Encuesta sobre la incidencia del “diseño e implementación de módulos online en la Cooperativa de ahorro y Crédito Chibuleo Ltda., de la ciudad de Ambato provincia de Tungurahua.”

Indicación : Por favor conteste el cuestionario según su criterio

Marque con una X en la respuesta

1.- ¿Le gustaría realizar consulta de saldos de sus cuentas por Internet?

SI

NO

2.- ¿Cree usted que al habilitar un módulo de pre apertura de cuenta ahorraría tiempo al trámite con la Cooperativa de Ahorro y Crédito “CHIBULEO” Ltda.?

SI

NO

NINGUNO

3.- ¿Mejoraría el tiempo de servicio si el sistema en línea de la Cooperativa pueda realizar ingreso de formulario de préstamos?

SI

NO

4.- ¿Confiaría en la seguridad de la página Web para el ingreso información personal cuando desee llenar un formulario de préstamos para su respectivo pre aprobación?

SI

NO

NO SABRIA

5.- ¿Disminuiría el tiempo de realizar una operación bancaria de préstamo o solicitud de certificaron el saber los estados dentro del sistema de la Cooperativa por medio de la pagina Web?

SI

NO

ANEXO 3

CÓDIGO DEL SISTEMA

A continuación se presenta el código del sistema desde sus diferentes capas que se los ha implementado.

Archivo de configuración.- Archivo de configuración con las correspondientes variables de tipo contante, correspondiente a la config global del sistema.

```
<?php
define("MYSQL_HOSTNAME", "localhost");
define("MYSQL_USER", "root");
define("MYSQL_PASSWORD", "");
define("MYSQL_DBNAME_HTML", "DBCHIBULEO");
define("RAIZ", $_SERVER['DOCUMENT_ROOT'] . "/pasantia/");
define("HOST", "http://" . $_SERVER['HTTP_HOST'] . "/pasantia/");
?>
```

Estructura básica de una clase

Es muy importante conocer la estructura de una clase para entender de mejor manera su funcionamiento y poder manipular su código fácilmente. Caber recordar que el manejo de clases es no tan amplia como en otros lenguajes.

```
<?php
```

```
//Nombre de la clase
```

```
class nombre
```

```
{
```

```
    //variables para la clase
```

```
    var $var_nombre;
```

```
    //Constructor de la clase
```

```
    function nombre($var_nombre)
```

```
    {
```

```
        $this->var_nombre =$var_nombre;
```

```
    }
```

```
    //Métodos de la clase
```

```
    function setVar_nombre($var_nombre)
```

```
    {
```

```
        $this->var_nombre =$var_nombre;
```

```
    }
```

```
    function getVar_nombre()
```

```
    {
```

```
        return $this->var_nombre;
```

```
    }
```

```
    //Funciones de una clase
```

```
    function funcion_nombre()
```

```
    {
```

```
        //Código de la función
```

```
    }
```

```
} //Fin de la clase
```

```
?>
```

Nota: Como se procedió a realizar clases para separar la parte de la vista, modelo y controlador hay que indicar que las clases necesarias para ejecutar las operaciones son llamadas en dependencia que se necesiten hacer para lo cual se va a detallar primero las que corresponden al área del modelo:

Conexión a la Base

```
<?php
include_once
($_SERVER['DOCUMENT_ROOT'].'/pasantia/config/config.inc.php');
class conexion
{
    private $cadenaConexion;
    private $hostname_local;
    private $database_local;
    private $username_local;
    private $password_local;
    private $conexion_error = 'Conectar base de Datos';
    public function conexion()
    {
        $this->hostname_local = MYSQL_HOSTNAME;
        $this->username_local = MYSQL_USER;
        $this->password_local = MYSQL_PASSWORD;
    }
    public function GetCadenaConexion()
    {
        try {

            $this->database_local = MYSQL_DBNAME_HTML;
            if (!($cadenaConexion = mysql_pconnect($this->hostname_local, $this->
```

```

        username_local, $this->password_local))) {
        throw new Exception($this->conexion_error);
    }
}
catch (exception $e) {
    echo "Error = " . $e->getMessage();
}
return $cadenaConexion;
}
public function GetCadenaConexionWSWEB()
{
    try {
        $this->database_local = MYSQL_DBNAME_WSWEB;
        if (!$cadenaConexion = mysql_pconnect($this->hostname_local, $this->
            username_local, $this->password_local))) {
            throw new Exception($this->conexion_error);
        }
    }
    catch (exception $e) {
        echo "Error = " . $e->getMessage();
    }
    return $cadenaConexion;
}
public function UnafilaBaseName()
{
    return $this->database_local;
}
}
?>

```

CLASE PARA EL MANEJO DE LOS SCRIPT

```

<?php
include_once ("conexion.class.php");
class scripts
{
    private $conexion;
    private $cn;
    private $sql;
    private $totalrow;
    private $result;
    private $row_result;
    private $total_rows_result;
    private $resultados = array();
    private $row = array();
    private $error = 0;
    private $error_numero;
    private $mensaje;
    public function scripts()
    {
        $this->conexion = new conexion();
        $this->cn = $this->conexion->GetCadenaConexion();
        $this->database = $this->conexion->UnafilaBaseName();
        mysql_select_db($this->database, $this->cn);
    }
    public function select($sql)
    {
        try {
            $this->sql = strtoupper($sql);
            if ($this->result = mysql_query($this->sql, $this->cn)) {
                $this->total_rows_result = mysql_num_rows($this->result);

                while ($this->row = mysql_fetch_assoc($this->result)) {
                    $this->resultados[] = $this->row;
                }
            }
        }
    }
}

```

```

    }

    mysql_free_result($this->result);

} else {
    $this->resultados=-1;
    $this->codigoerror(mysql_errno($this->cn),"SE-
LECCION",$this->sql);
    throw new Exception($this->mensaje);
}

}

catch (exception $e) {
}

return $this->resultados;
}

public function select_fila($sql)
{
    try {
        $this->sql = strtoupper($sql);
        if ($this->result = mysql_query($this->sql, $this->cn)) {
            $this->total_rows_result = mysql_num_rows($this->result);
            $this->resultados = mysql_fetch_assoc($this->result);
            mysql_free_result($this->result);
        } else {
            $this->resultados=-1;
            $this->codigoerror(mysql_errno($this->cn),"SELECCION",$this->sql);
            throw new Exception($this->mensaje);        }
    }

    catch (exception $e) {
    }

    return $this->resultados;
}

```

```

}
public function TotalFilasAfectadas()
{
    return mysql_affected_rows();
}
public function SQL($sql)
{
    try {
        $this->sql = strtoupper($sql);
        if ($this->result = mysql_query($this->sql, $this->cn)) {
            $this->resultado=mysql_affected_rows();
        } else {
            $this->resultado=-1;
            $this->mensaje=$this-
>codigoerror(mysql_errno($this->cn),"MODIFICACION          DE
DATOS", "PROCESAMIENTO DE DATOS");
            throw new Exception($this->mensaje);        }
        }
    catch (exception $e) {
    }
    return $this->resultado;
}
public function __destruct()
{
    mysql_close($this->cn);
}
public function codigoerror($valor,$operacion,$parametro){
switch ($valor):
case 1: break;
case 1054:
$this->mensaje = "Error En el envio de consulta a la base de datos a la con
parametro ".$parametro." al realizar una ".$operacion;
break;

```

```

    case 1062:
$this->mensaje = "Error al Agregar al sistema con dato clave ".$parametro;
$this->mensaje.= "\n"." ya existe en el sistema ";
return $this->mensaje;
break;
case 1064:
$this->mensaje = "Error En sentencia SQL ";
break;
case 1146:
$this->mensaje = "TABLA ".$parametro ." NO EXISTE ";
break;
default:
$this->mensaje= $valor;
endswitch;
}
public function ObtenerError(){
    return $this->mensaje;
}
}

```

En base a estas clases se maneja toda el capa que se refiere al controlador para conexiones, ejecuciones, maneje de errores y validación de datos.

Acceso a las tablas para realizar operación Dql y Dml

```

<?php
include_once ('config.php');
include_once (INCLUYE_MODELO.'/parametros.class.php');
include_once (INCLUYE_MODELO.'/scripts.class.php');
class accesos
{
    private $sql;

```

```

private $id;
private $idusuario;
private $ipacceso;
private $fecha;
private $fechahora;
private $ejecutar;
private $control;
private $totalfilas;
private $resultado;
public function Todaslasfilas()
{
    $this->sql = "SELECT ID,IDUSUARIO,IPACCESO,FECHA,DESCRIP-
CION FROM Web_accesos";
    $this->ejecutar = new scripts();
    $this->resultado = $this->ejecutar->select($this->sql);
    return $this->resultado;
}
public function Unafila($id)
{
    $this->id = $parametros->escape_campos($id);
    $this->sql = "SELECT ID,IDUSUARIO,IPACCESO,FECHA,DESCRIP-
CION FROM Web_accesos WHERE id='this->usuario'";
    $this->ejecutar = new scripts();
    $this->resultado = $ejecutar->select($this->sql);
    return $this->resultado;
}
public function TotalFilasAfectadas()
{
    return mysql_affected_rows();
}
public function UnaFilafectada()

```

```

    {
        return $this->resultado;
    }
public function InsertarDatos($idusuario,$ipacceso,$fecha,$descripcion)
{
    $this->control = new parametros();
    $this->fechahora = $fecha." ".date("H:i:s");
    $this->idusuario = $this->control->escape_campos($idusuario, "text");
    $this->ipacceso = $this->control->escape_campos($ipacceso,"text");
    $this->descripcion = $this->control->escape_campos($descripcion,"text");
        $this->fechahora = $this->control->escape_campos($this-
>fechahora,"date");
    $this->sql = "INSERT INTO Web_accesos  (IDUSUARIO,
        IPACCESO, FECHA, DESCRIPCION)";
        $this->sql .= "VALUES  ('$this->idusuario','$this->ipacceso','$this-
>fechahora','$this->descripcion')";
    $this->ejecutar = new scripts();
    $this->resultado = $this->ejecutar->SQL($this->sql);
        if ($this->resultado==-1){
            $this->mensaje = $this->ejecutar->ObtenerError();
        }
    return $this->resultado;
}

```

```

public function BorrarFila($id)
{
    $this->control = new parametros();
    $this->id = $this->control->escape_campos($id, "text");
    $this->sql = "DELETE FROM Web_accesos WHERE ID = '$this->id'";
    $this->ejecutar = new scripts();
    $this->resultado = $this->ejecutar->SQL($this->sql);
}

```

```

        return $sejecutar->totalrow();
    }

    public function ObtenerError()
    {
        return $this->mensaje;
    }
}

```

Envío de correos a los clientes

```

<?php
include_once ('config.php');
include_once (INCLUYE_CONTROLADOR.'/accesos.php');
class emails{
    public function emails(){
    public function envioemail($email,$mensaje,$usuario){
        $mensaje = wordwrap($mensaje, 70);
        $replica = "wilsonrobertocunalata@gmail.com";
        $ahora = date("Y-m-d");
        $ip = GetHostByName($_SERVER['REMOTE_ADDR']);
        $email = $email;

        $envio = new accesos();
        $resultado = $envio->InsertarDatos($usuario,$ip,$ahora,$mensaje);
        if ($resultado!=-1){
            // Para enviar correo HTML, la cabecera Content-type debe
definirse

            $cabeceras = 'MIME-Version: 1.0' . "\r\n";
            $cabeceras .= 'Content-type: text/html; charset=iso-8859-1' .
"\r\n";

```

```

        mail($email, 'Informacion Cooperativa CHIBULEO
LTDA.', $mensajehtml,$cabeceras);
        echo "<script>";
        echo "alert('Informacion enviada correctamente a nuestra
base de datos\ny al correo $destinatario');";
        echo "</script>";
        $_matriz['fila_afectada']=1;
        return $_matriz;
    }else{
        $_matriz['fila_afectada']=-1;
        $_matriz['mensaje'] = $envio->ObtenerError();
        return $_matriz;
    }
}
?>

```

Manejo de sesiones para usuarios como para clientes

```

<?php
include_once ('config.php');
include_once (INCLUYE_CONFIG. '/config.inc.php');
class sesiones{
    public function validasesion(){
        if (!isset($_SESSION)) {
            session_start();
        }
        if ((isset($_SESSION['user'])) and (isset($_SESSION['level']))){}
        else{header("Location: index.php" );}
    }
    public function logout(){
        if (!isset($_SESSION)) {

```

```

        session_start();
    }
    session_unset();
    session_unset($_SESSION['user']);
    session_unset($_SESSION['level']);
    session_unset($_SESSION['ipacceso']);
    session_unset($_SESSION['fecha']);
    session_unset($_SESSION['tipo']);
    session_unset($_SESSION['oficina']);
    session_destroy();
    header("Location: index.php" );
}

public function crearsesiones($usuario,$clave,$tipo,$oficina){
    if (!isset($_SESSION)) {
        session_start();
    }
    $this->control = new parametros();
    $this->tools = new tools();
        $_SESSION['user'] = $this->encrypt($this->control->escape_cam-
pos($usuario, "text"));
        $_SESSION['clave']      =$this->encrypt($this->tools->Encrypt-
arClave($clave));
        $_SESSION['ipacceso']=$this-
>encrypt(GetHostByName($_SERVER['REMOTE_ADDR']));
        $_SESSION['fecha']=$this->encrypt(date("Y-m-d"));
        $_SESSION['tipo']=$this->encrypt($this->control->escape_cam-
pos($tipo, "text"));
        $_SESSION['oficina']=$this->encrypt($this->control-
>escape_campos($oficina, "text"));
    }
    function CrearSesiones2($tipo,$usuario){
        if (!isset($_SESSION)) {

```

```

session_start();

/*      session_register(session_id().'login'); // Registramos la variable
        $_SESSION[session_id().'login'] = $usuario;
        session_register(session_id().'tipo'); // Registramos la variable
$_SESSION(session_id().'tipo') = $tipo;
        $_SESSION[session_id().'ipacceso']=GetHostByName($_SERV-
ER['REMOTE_ADDR']);
        $_SESSION[session_id().'fecha']=date("Y-m-d");
*/
    }
}

function encrypt($string) {
    $key = 'CLAVE';
    $result = "";
    for($i=0; $i<strlen($string); $i++) {
        $char = substr($string, $i, 1);
        $keychar = substr($key, ($i % strlen($key))-1, 1);
        $char = chr(ord($char)+ord($keychar));
        $result.=$char;
    }
    return base64_encode($result);
}

function decrypt($string) {
    $key = "CLAVE";
    $result = "";
    $string = base64_decode($string);
    for($i=0; $i<strlen($string); $i++) {
        $char = substr($string, $i, 1);
        $keychar = substr($key, ($i % strlen($key))-1, 1);
        $char = chr(ord($char)-ord($keychar));
        $result.=$char;
    }
}

```

```

        }
        return $result;
    }
}
?>

```

Manejo de claves de manera encriptado y su generación.

```

<?php
class tools
{
    private $pwd = "";
    private $pwd_encrypt = "";
    private $lowercase;
    private $uppercase;
    private $number;
    private $specchar;
    private $len;
    private $prefix;
    function generarclave()
    {
        public function ObtenerClaveEncriptada()
        {

            return $this->pwd_encrypt;
        }
        public function ObtenerClaveNormal()
        {

```

```

        return $this->pwd;
    }
    function EncriptarClave($dato)
    {
        return sha1(md5($dato)); //return $this->pwd_encrypt;
    }
}

```

Validación de usuario o cliente en una página que accede

```

<?php
if (!isset($_SESSION)) {
    session_start();
}
include_once (INCLUYE_CONTROLADOR.'/logins.php');
class validadmin
{
    public function validadmin()
    {
        $_SESSION['acceso'] = "NULL";
        if ((isset($_SESSION['user'])) and (isset($_SESSION['tipo'])) and
(isset($_SESSION['ipacceso']))) {
            $_SESSION['acceso']=$_SESSION['tipo'];
        } else {
            $this->reenvio();
        }
        if ($_SESSION['acceso'] == "NULL") {
            $this->reenvio();
        }
        $dato = $this->decrypt($_SESSION['oficina']);
    }
}

```

```

        if($dato!="W"){      $this->reenvio();}
        $tnUsuarios = new logins();
        $lnUsuario = $this->decrypt($_SESSION['user']);
        $lnResultado= $tnUsuarios->Unafila($lnUsuario);
        if(($lnResultado[0]['OFICINA']=="W") and ($lnResultado[0]
[ESTADO_SISTEMA]=='1')){}else{
            $this->reenviofinal();
        }
    }
    public function reenviofinal()
    {
        echo "<script>";
        echo "parent.location.href = 'logout.php'";
        echo "</script>";
    }
    public function reenvio()
    {
        echo "<script>";
        echo "parent.location.href = '../banca.php'";
        echo "</script>";
    }
    public function validarcookie(){
        if (isset($_COOKIE[strtolower($this-
>decrypt($_SESSION['tipo']))]) //and $_COOKIE[$this-
>decrypt($_SESSION['user'])]=="HORA")
        {}else{
            $this->fincookie();
        }
    }
    public function fincookie()
    {

```

```

        echo "<script>";
            echo "alert('DISCULPE PERO HA ESTADO INACTIVO DURANTE
MAS DE UNA HORA VUELVA A LOGUEARSE')";
        echo "</script>";
        echo "<script>";
        echo "parent.location.href = '../admin/logout.php'";
        echo "</script>";
    }

    function decrypt($string) {
        $key = "clave";
        $result = "";
        $string = base64_decode($string);
        for($i=0; $i<strlen($string); $i++) {
            $char = substr($string, $i, 1);
            $keychar = substr($key, ($i % strlen($key))-1, 1);
            $char = chr(ord($char)-ord($keychar));
            $result.=$char;
        }
        return $result;
    }
}
?>

```

En base a la estructura mostrada se realiza las demás clases que existen en la capa CONTROLADOR, lo cual se ha visto siempre llamando a un config y a sus respectivas clases de control y accesos.

A continuación se detalla la clase para el manejo del WS .

Manejo del WS

Para la implementación de esta clase se ha utilizado una clase de tipo externa llamada NUSOAP para la manipulación del WS y lo detallamos a continuación

Clase para el manejo del WS

```
<?php
include_once ('config.php');
include_once (INCLUYE_CONFIG. '/config.inc.php');
require_once(INCLUYE_CONTROLADOR."/nusoap/lib/nusoap.php"); //
pregunta requiere o include
class service{

function servicio(){
    $this->servicio = new
soapclient('http://localhost/servicios/ws_clientes.asmx?WSDL', true);
    $this->err = $this->servicio->getError();
    if ($this->err) {
        echo '<h2>Constructor error</h2><pre>' . $err . '</pre>';
    }

}

function llamada($parametros,$funcion,$usuario){
    $this->usuario = $usuario;
    $this->result = $this->servicio->call($funcion, array('parameters'
=> $parametros), "", false, true);//pregunta q es esos parametros
    if ($this->servicio->fault) { // Check for a fault
        echo '<h2>Faulteroro </h2><pre>';
        print_r($this->result);
        echo '</pre>';
    } else { // Check for errors
        $this->err = $this->servicio->getError();
        if ($this->err) { // Display the error
            echo '<h2>Error</h2><pre>' . $this->err .
'</pre>';

```



```

System.IO.FileStream myFileStream = new System.IO.FileStream( filename,
System.IO.FileMode.Create );
    System.Xml.XmlTextWriter myXmlWriter =
    new System.Xml.XmlTextWriter ( myFileStream, System.Text.Encoding.Uni-
code );
    thisDataSet.WriteXml ( myXmlWriter );
    myXmlWriter.Close ( );
}
[WebMethod(Description="Devuelve datos de Saldo disponible")]
public DataSet Clientes(string sel)
{
    if( sel != "" ) {
        cedula = sel;
        sel = "SELECT Clientes.codigo,Clientes.numeroCliente, " +
            "Cuentas.secuencial,Cuentas.codigoProducto,Cuentas.disponible,Cuen-
tas.retenciones, " +
            "Cuentas.encajes,Cuentas.bloqueos,Cuentas.OficialCuenta,Cuentas.estado,Cuenta-
s.SaldoPromedio " +
            "FROM Clientes,Cuentas " +
            "WHERE Clientes.numeroCliente = Cuentas.numeroCliente " +
            "AND Clientes.codigo = " + cedula + " ";
    }
    if( sel == "" )
        throw new ArgumentException("La cadena no puede ser nula");
    if( sel.IndexOf("--") > -1 )
        throw new ArgumentException("No se admiten comentarios de SQL en la
cadena de selección");
    if( sel.ToUpper().IndexOf("DROP") > -1 )
        throw new ArgumentException("La cadena debe ser una Cédula de
identidad ");
        da = new SqlDataAdapter(sel,
"server=Qosmio;uid=sa;pwd=sa;database=chibuleo;");

```

```

DataSet ds = new DataSet();
try
{
    da.Fill(ds);
    WriteXmlToFile ( ds,cedula+".xml");
}
catch(Exception ex)
{
    throw ex;
}
return ds;
}

```

El cual en su ejecución lo podemos ver:

The screenshot shows a web service interface with a dark blue header containing the text 'ws_clientes'. Below the header, the text 'Cooperativa Chibuleo Ltda.' is displayed. A paragraph follows: 'Las operaciones siguientes son compatibles. Para una definición formal, revise la [descripción de servicios](#).' Below this, there is a bulleted list of six operations, each with a blue link and a description:

- [Clientes](#)
Devuelve datos de Saldo disponible
- [Creditos](#)
Devuelve datos del Cliente para Creditos
- [Datos](#)
Devuelve datos del Cliente
- [MovimientosCuenta](#)
Devuelve movimientos de una cuenta con filtro
- [Prestamos](#)
Devuelve datos del Cliente para Creditos
- [SaldoPromedio](#)
Devuelve datos del Cliente y Saldo disponible

FIGURA 6.2 WEB SERVICE

• **Uso Del Método Clientes**

ws_clientes

Haga clic [aquí](#) para obtener una lista completa de operaciones.

Cientes

Devuelve datos de Saldo disponible

Prueba

Haga clic en el botón 'Invocar', para probar la operación utilizando el protocolo HTTP POST.

Parámetro	Valor
sel:	<input type="text" value="1801546845"/>

SOAP 1.1

A continuación se muestra un ejemplo de solicitud y respuesta para SOAP 1.1. Es necesario reemplazar los **marcadores de posición** que aparecen con valores reales.

```

POST /servicios/ws_clientes.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://chibuleo.com/web_service/Cientes"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Cientes xmlns="http://chibuleo.com/web_service/">

```

FIGURA 6.3 METODO WEB SERVICE

Y el resultado que obtendriamos seria :

```

1801546845 9 67614 100 115.86 0.00 0.00 0.00 DIANASC A 0.00 1801546845 9 67615 200 2.00 0.00 0.00 0.00 DIANASC A 0.00 1801546845 9 43771 300 0.00 0.00 0.00 0.00 MIGRACION C
21.04

```

FIGURA 6.4 RESULTADO WEB SERVICE

Lo cual se almacena en un archivo xml para con los mtodos en la clase poder visualizarlos.

Con esto se ha terminado de mostrar las clases que corresponden a la capa CONTROLADOR y pasamos a la capa VISTAS.

En la capa vistas se encuentra del manera mas limpia los datos para el uso y se lo ha implementado mediante JQUERY para enriquecer la experiencia con el Usuario.

Ahora vamos a mostrar todas estas clases en uso empezando por el logeo en el sistema que se lo encuentra en le archivo banca.php del cual lo mas importante es:

- La llamada a el jquery y su archivo personalizado

```

<!--<script type="text/javascript" language="javascript" src="js/scr:
<script type="text/javascript" language="javascript" src="js/scripta
<script type="text/javascript" language="javascript" src="js/jsvalida
-->
<script language="javascript" src="js/jquery.js" ></script>
<script language="javascript" src="js/ingreso_banca.js" ></script>

<script src="../../SpryAssets/SpryCollapsiblePanel.js" type="text/javas
<!--
para llamar a funciones de validacion

```

FIGURA 6.5 ENLACE ARCHIVOS JQUERY

- El nombre que se les de a las variables en el formulario de acceso:

```

<tr>
<td>&nbsp;</td>
<td>Cédula :</td>
<td><input name="cedula" type="text" id="cedula" size="15" maxlength="20"/>
<td>&nbsp;</td>
</tr>
<tr>
<td>&nbsp;</td>
<td>Clave :</td>
<td><input name="clave" type="password" id="clave" size="15" maxlength="20"
<td>&nbsp;</td>
</tr>

```

FIGURA 6.6 INFORMACION SOBRE CAMPOS

- El resto del control se lo hace mediante el archivo personalizado en JQUERY :

```

// JavaScript Document
var cedula="";
var clave="";
$(function(){
    $("#process").hide();
    f_valida_cedula_focusout();
    f_valida_clave_focusout();
    f_ingresar();
});
function f_ingresar(){
    $("#ingresar").click(function(){
        event.stopImmediatePropagation();
        validas();
    });
}

```

```

    }
    function f_valida_cedula_focusout(){
        $("#cedula").focusout(function(){
            if(valida($("#cedula").val())==0){
                //alert("POR FAVOR INGRESE DATOS
CORRECTOS");
                $("#cedula").focus();
            }
        });
    }
}

```

Con este código se ha realiza una interacción con el archivo validacionlogins.php en cual interactúa con las clase de la capa CONTROLADOR

```

<?php
if (!isset($_SESSION)) {
    session_start();
}
include_once ('config.php');
include_once (INCLUYE_CONTROLADOR. '/logins.php');
?>
<?php
if (isset($_GET["opcion"])) {
    switch ($_GET['opcion']){
        case 0:
            $lnUsuario = $_GET['id'];
            $lnClave = $_GET['clave'];
            $tnUsuarios = new logins();
            $lnResultado= $tnUsuarios-
>ValidarUsuario($lnUsuario, $lnClave);

```

```

        if ($lnResultado==-1){
            $_matriz['fila_afectada']=-1;
            $_matriz['mensaje']=$tnUsuarios-
>ObtenerError();

            print json_encode($_matriz);
            break;
        }else{

            $lnTotalfilas = $tnUsuarios-
>TotalFilasAfectadas();// obtenermos los roles de cada usuario q se logea

            $_matriz['fila_afectada']=$lnTotalfilas;

            if ($lnTotalfilas>0){
                $lnEstado =
$tnUsuarios->ObtenerResultado();

                $lnTipo = $lnEstado[0]
['NOMBRE'];

                $lnEstadoSistema =
$lnEstado[0]['ESTADO_SISTEMA'];

                $lnOficina=$lnEstado[0]['OFICINA'];

                if
($lnEstadoSistema==0){

                    $sesion
= new sesiones();

                    $sesion-
>CrearSesiones($lnUsuario,$lnClave,$lnTipo,$lnOficina);//creacion de sesiones

                    if ((($se-
sion->decrypt($_SESSION['oficina']))=="W") or (($sesion->decrypt($_SES-
SION['oficina']))=="C")){

```

```

        if($tnUsuarios->ActualizarUsuarioEstado(1,$sesion-
>decrypt($_SESSION['user']))==-1){

            $_matriz['fila_afectada']=-1;

            $_matriz['mensaje']=$tnUsuarios->ObtenerError();

            print json_encode($_matriz);

?>

```

Aquí realizamos la interacción con el método llamado RIA para poder realizar la verificación, logeo y llamada al WS para la obtención del saldo promedio que dispone un cliente o el acceso al panel administrativo de un usuario

Cliente con acceso al sistema

Cuando un usuario o cliente ha ingresado sus datos exitosamente entonces ingresa al área de clientes o usuarios registrados y se realiza sus respectivas validaciones del caso para que no lleguen a ocurrir casos de fraudulencia en los datos de la siguiente manera:

```

<?php
if (!isset($_SESSION)) {
    session_start();
}
include_once ('config.php');
$nombre = strtolower(decrypt($_SESSION['tipc']));
if (isset($_SESSION['user'])) {
    if (isset($_COOKIE[$nombre])) {} else {setcookie($nombre, "HORA", time() + 3600); header('Location: index.php'); }
} else {header('Location: logout.php');}
include_once (INCLUYE_CONTROLADOR. '/validacionadministracion.php');//para validar sesiones
$valida = new validadmin();
$valida->validarcookie();
include_once (INCLUYE_CONTROLADOR. '/menuadmin.php');
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

```

figura 6.7 Implementación de acceso al sistema

En la cual se realiza para todos las páginas que se encuentren en esta área en la cual se crea una cookie de inicio de sesión con validez

ANEXO 4

MANUAL DE USUARIO

Como acceder al portal Web?

Para acceder al portal web de la Cooperativa de Ahorro y Crédito “Chibuleo” Ltda., debemos abrir nuestro navegador y digitar la siguiente URL.

<http://www.chibuleo.fin.ec/>

También existe la posibilidad de utilizar un buscador como Google, Altavista, Yahoo, etc. Una vez iniciado el portal, se cargará la pantalla de acceso al sistema.



FIGURA 6.8 Página principal

La página se encuentra distribuida de la siguiente forma:

- Una cabecera donde se encuentran el logo institucional.

- Un menú principal de fácil acceso para los visitantes
- Zona de logueo para los usuarios del portal
- Un enlace a la parte de recuperar contraseña.
- Un enlace a la preapertura de cuentas.
- Y en la parte céntrica de la misma toda la información necesario y los servicios que ofrece la Cooperativa a sus cliente o usuarios.

¿Cómo recuperar la clave?

Para poder realizar esta operación ingresamos a <http://www.chibuleo.fin.ec/> y nos vamos a



Figura 6.9 Recuperar clave

Para luego ingresar la información que se presenta:

Figura 6.10 Formulario de recuperación de clave

En el cual ingresamos el número de cedula y la nueva clave se enviara a el correo proporcionado a la Cooperativa, en caso de no poseer deberá acercarse a una oficina de la misma para realizar este proceso.

Acceso al Sistema

Figura 6.11 Acceso al sistema

Para acceder al sistema financiero en línea se debe ingresar el número de cedula y la clave proporcionada en la Cooperativa y leer el respectivo mensajes que el sistema genere como puede ser:

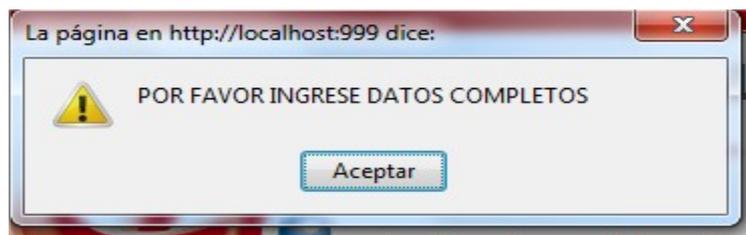


Figura 6.12 Mensaje 1

Este mensaje se genera cuando no se ingresado o el número de cedula/nombre de usuario o la clave y se ha dado clic en ingresar.

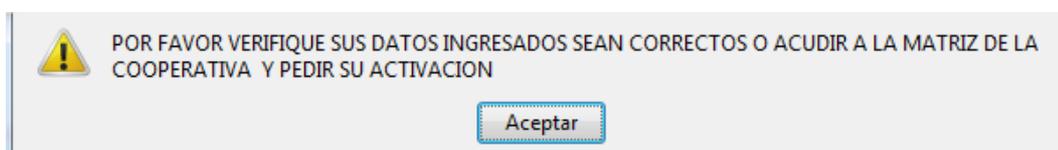


Figura 6.13 Mensaje 2

Este mensaje se genera cuando ha ingresado los datos de manera errores.

Cuando realice un ingreso correcto de los datos sea de un cliente o de un usuario se darán sus respectivas validaciones y se accederá a el área con permisos para el usuario o cliente.

PAGINAS DEL CLIENTE

Una vez ingresado al sistema un cliente vera la siguiente pantalla:

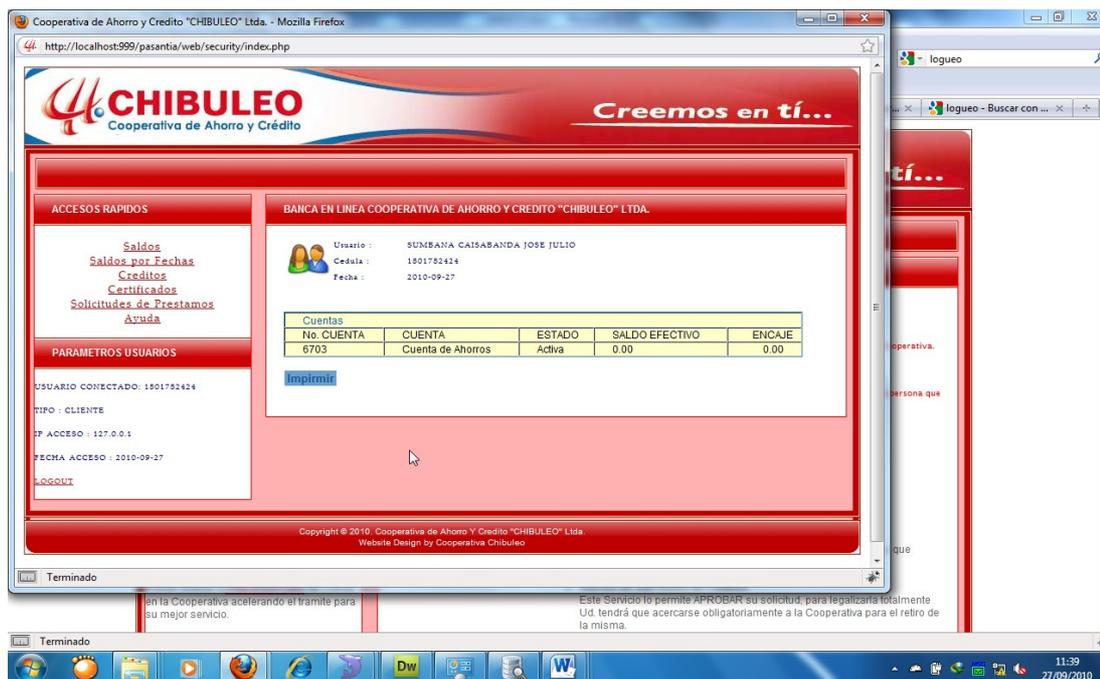


Figura 6.14 Pantalla de Saldos

Esta es la página de saldos actuales que posee un cliente en referencia a su o sus cuentas que mantenga en la Cooperativa, la cual está distribuida de la siguiente forma:

- En la cabecera el logo y nombre de la Cooperativa.
- A la izquierda superior el menú con las opciones implementadas y de acceso más cómodo.
- En la izquierda inferior los parámetros con los que cada usuario accede al sistema incluyendo el logout o salir del sistema.
- Y en la parte céntrica que corresponde al cuerpo de la página la información financiero.

Para imprimir esta información se debe dar click en imprimir y se nos cambiara un poco el formato de la información como lo vemos:

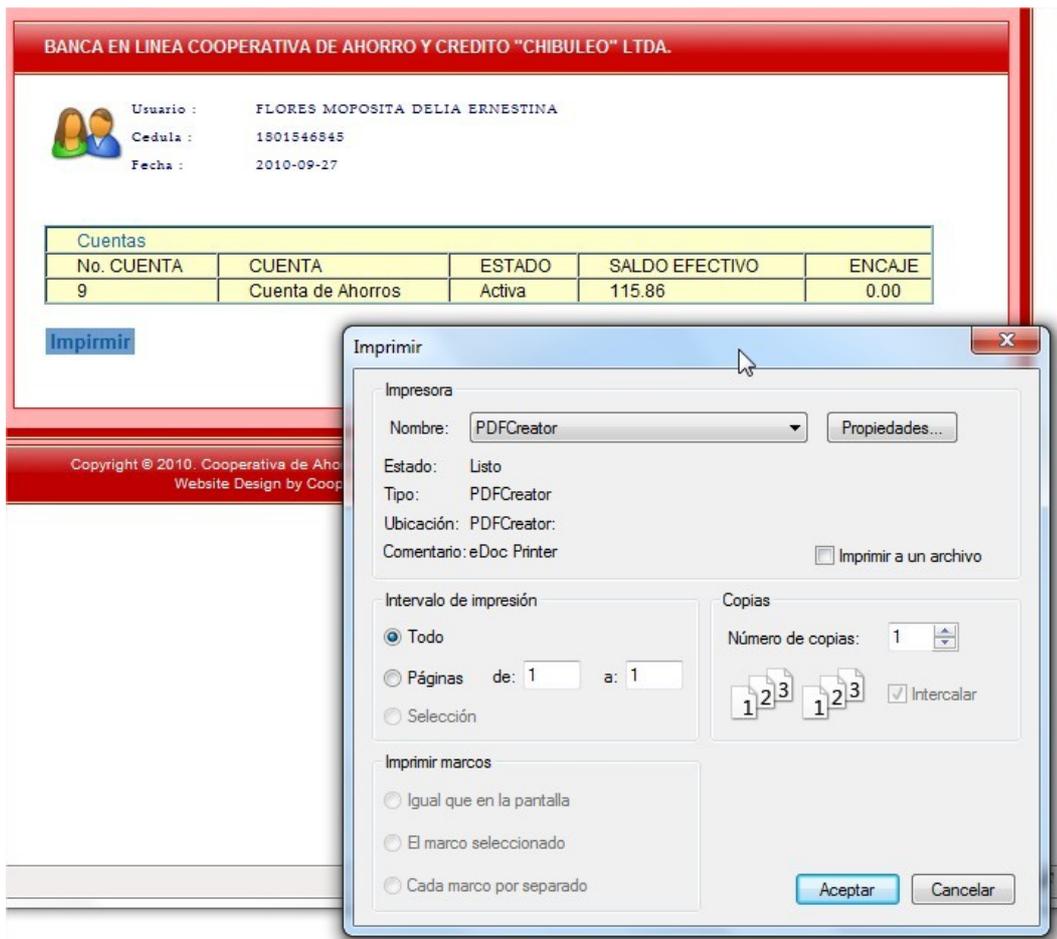


Figura 6.15 Impresión de Saldo Actual

Con lo que se permite obtener la información financiera en un medio impreso.

Movimientos de Cuentas

Si se desea saber que movimientos ha sufrido la cuenta se debe dar click en:



Figura 6.16 Saldos por Fechas

Y se presenta el siguiente formulario:

Figura 6.17 Formulario saldos por Fechas

En le Cual se debe seleccionar el número de cuenta:

Figura 6.18 Selección de cuenta

Y luego definir si desea realizar una consulta total de la cuenta o por fechas como se muestra en las figuras

BANCA EN LINEA COOPERATIVA DE AHORRO Y CREDITO "CHIBULEO" LTDA.


 Usuario : FLORES MOPOSITA DELIA ERNESTINA
 Cedula : 1801546845
 Fecha : 2010-09-27

Selección de Criterios

Cuenta :

Fecha :

Figura 6.19 selección de fechas

BANCA EN LINEA COOPERATIVA DE AHORRO Y CREDITO "CHIBULEO" LTDA.


 Usuario : FLORES MOPOSITA DELIA ERNESTINA
 Cedula : 1801546845
 Fecha : 2010-09-27

Selección de Criterios

Cuenta :

Fecha :

Fecha de Inicio :

Fecha de Finalización :

Septiembre, 2010

sem	Lun	Mar	Mi	Jue	Vie	Sob	Dom
35			1	2	3	4	5
36	6	7	8	9	10	11	12
37	13	14	15	16	17	18	19
38	20	21	22	23	24	25	26
39	27	28	29	30			

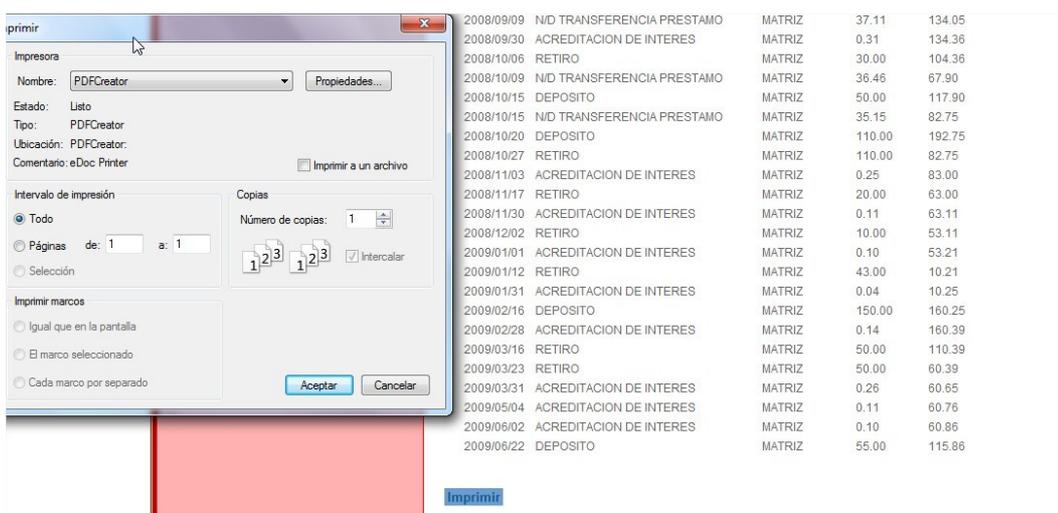
Seleccionar fecha

Figura 6.20 selección de fechas inicial y final

Con estos parámetros escogidos podemos obtener el resultado total de la cuenta o parcial según sea las fechas seleccionadas.

BANCA EN LINEA COOPERATIVA DE AHORRO Y CREDITO "CHIBULEO" LTDA.				
	Usuario :	FLORES MOPOSITA DELIA ERNESTINA		
	Cedula :	1501546845		
	Fecha :	2010-09-27		
	Rango :	DESDE EL 1 de enero del 2008 a la actualidad		
	Moneda :	Dolar		
	Saldo Actual :	115.56		
	Cuenta # :	9		
Fecha	Transaccion	Oficina	Valor	Saldo Disponible
2008/05/02	APERTURA CUENTA	MATRIZ	23.14	28.14
2008/05/02	NOTA DE DEBITO TRANSFERENCIA CUENTAS	MATRIZ	5.00	23.14
2008/05/05	DEPOSITO	MATRIZ	25.00	48.14
2008/05/12	N/C TRANSFERENCIA PRESTAMO	MATRIZ	200.00	248.14
2008/05/12	RETIRO	MATRIZ	200.00	48.14
2008/06/02	ACREDITACION DE INTERES	MATRIZ	0.06	48.20
2008/06/11	N/D TRANSFERENCIA PRESTAMO	MATRIZ	39.37	8.83
2008/06/25	DEPOSITO	MATRIZ	20.00	28.83
2008/06/30	ACREDITACION DE INTERES	MATRIZ	0.03	28.86
2008/07/31	ACREDITACION DE INTERES	MATRIZ	0.04	28.90
2008/08/05	DEPOSITO	MATRIZ	120.00	148.90
2008/08/11	N/D TRANSFERENCIA PRESTAMO	MATRIZ	38.01	110.89
2008/08/31	ACREDITACION DE INTERES	MATRIZ	0.27	111.16
2008/09/09	DEPOSITO	MATRIZ	60.00	171.16
2008/09/09	N/D TRANSFERENCIA PRESTAMO	MATRIZ	37.11	134.05
2008/09/30	ACREDITACION DE INTERES	MATRIZ	0.31	134.36
2008/10/06	RETIRO	MATRIZ	30.00	104.36
2008/10/09	N/D TRANSFERENCIA PRESTAMO	MATRIZ	36.46	67.90
2008/10/15	DEPOSITO	MATRIZ	50.00	117.90
2008/10/15	N/D TRANSFERENCIA PRESTAMO	MATRIZ	35.15	82.75
2008/10/20	DEPOSITO	MATRIZ	110.00	192.75
2008/10/27	RETIRO	MATRIZ	110.00	82.75
2008/11/03	ACREDITACION DE INTERES	MATRIZ	0.25	83.00
2008/11/17	RETIRO	MATRIZ	20.00	63.00
2008/11/30	ACREDITACION DE INTERES	MATRIZ	0.11	63.11
2008/12/02	RETIRO	MATRIZ	10.00	53.11
2009/01/01	ACREDITACION DE INTERES	MATRIZ	0.10	53.21
2009/01/12	RETIRO	MATRIZ	43.00	10.21
2009/01/31	ACREDITACION DE INTERES	MATRIZ	0.04	10.25
2009/02/16	DEPOSITO	MATRIZ	150.00	160.25
2009/02/28	ACREDITACION DE INTERES	MATRIZ	0.14	160.39
2009/03/16	RETIRO	MATRIZ	50.00	110.39
2009/03/23	RETIRO	MATRIZ	50.00	60.39
2009/03/31	ACREDITACION DE INTERES	MATRIZ	0.26	60.65
2009/05/04	ACREDITACION DE INTERES	MATRIZ	0.11	60.76
2009/06/02	ACREDITACION DE INTERES	MATRIZ	0.10	60.86
2009/06/22	DEPOSITO	MATRIZ	55.00	115.86

Figura 6.21 resultado de consultas por fechas



2008/09/09	N/D TRANSFERENCIA PRESTAMO	MATRIZ	37.11	134.05
2008/09/30	ACREDITACION DE INTERES	MATRIZ	0.31	134.36
2008/10/06	RETIRO	MATRIZ	30.00	104.36
2008/10/09	N/D TRANSFERENCIA PRESTAMO	MATRIZ	36.46	67.90
2008/10/15	DEPOSITO	MATRIZ	50.00	117.90
2008/10/15	N/D TRANSFERENCIA PRESTAMO	MATRIZ	35.15	82.75
2008/10/20	DEPOSITO	MATRIZ	110.00	192.75
2008/10/27	RETIRO	MATRIZ	110.00	82.75
2008/11/03	ACREDITACION DE INTERES	MATRIZ	0.25	83.00
2008/11/17	RETIRO	MATRIZ	20.00	63.00
2008/11/30	ACREDITACION DE INTERES	MATRIZ	0.11	63.11
2008/12/02	RETIRO	MATRIZ	10.00	53.11
2009/01/01	ACREDITACION DE INTERES	MATRIZ	0.10	53.21
2009/01/12	RETIRO	MATRIZ	43.00	10.21
2009/01/31	ACREDITACION DE INTERES	MATRIZ	0.04	10.25
2009/02/16	DEPOSITO	MATRIZ	150.00	160.25
2009/02/28	ACREDITACION DE INTERES	MATRIZ	0.14	160.39
2009/03/16	RETIRO	MATRIZ	50.00	110.39
2009/03/23	RETIRO	MATRIZ	50.00	60.39
2009/03/31	ACREDITACION DE INTERES	MATRIZ	0.26	60.65
2009/05/04	ACREDITACION DE INTERES	MATRIZ	0.11	60.76
2009/06/02	ACREDITACION DE INTERES	MATRIZ	0.10	60.86
2009/06/22	DEPOSITO	MATRIZ	55.00	115.86

Figura 6.22 impresión de resultados

Créditos

Para poder ver los datos de pago de un crédito debemos dar click en el enlace [creditos](#) del menú rápido; se nos cargara uno de los dos casos a continuación:



Figura 6.23 Mensaje no Créditos



Figura 6.24 Visualización de datos de créditos

Certificados

Para poder realizar la petición de algún tipo de certificado a la Institución debemos dar click en el enlace certificados del menú rápido y se presenta la siguiente página:

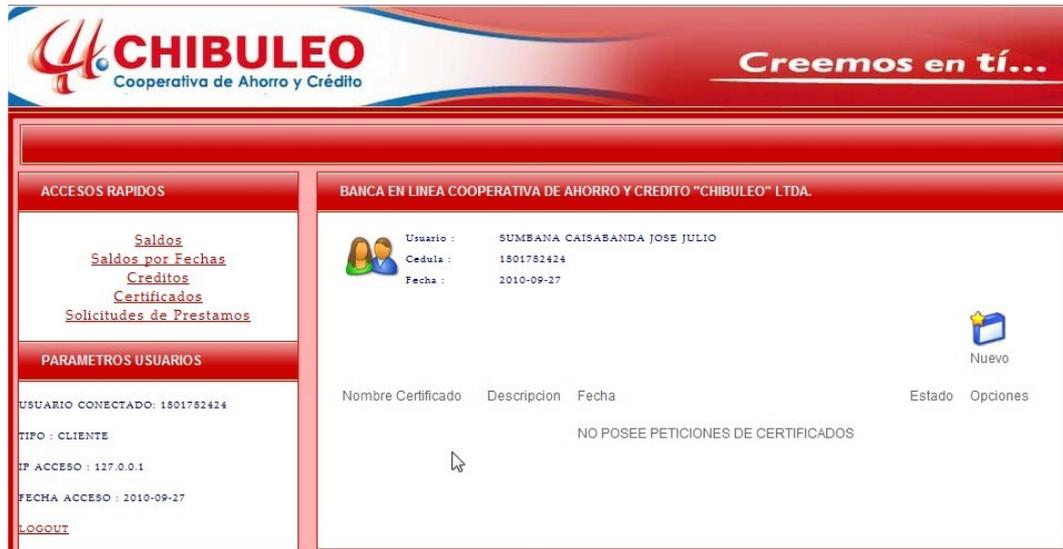


Figura 6.25 principal de certificados

Aquí podemos visualizar en el cuerpo de la página si ha realizado alguna petición de certificado y en que estado se encuentra o la posibilidad de hacer una petición dando click en nueva y se presenta el formulario:



Figura 6.26 formulario de certificados.

Aquí se selecciona que tipo de certificado desea pedir a la Cooperativa que pueden ser:



Figura 6.27 Tipos de certificados

Los cuales tienen un costo de \$2 y se le hará la entrega en las oficinas de la Institución, lo cual se acepta cuando se da click en aceptar y la solicitud para a un estado de espera, para que sea aprobado.



Figura 6.28 Petición ingresada al sistema

Cuando se haya echo estos procesos en la lista de certificados como la vemos abajo se mostraran todos.

BANCA EN LINEA COOPERATIVA DE AHORRO Y CREDITO "CHIBULEO" LTDA.					
 Usuario : SUMBANA CAISABANDA JOSE JULIO Cedula : 1801782424 Fecha : 2010-09-27					
					 Nuevo
Nombre Certificado	Descripcion	Fecha	Estado	Opciones	
Certificado de Plazo Fijo	Obtiene informacion detallada de sus inversiones a plazo fijo	2010-09-27	ESPERA	Eliminar	
Certificado de Creditos al Dia	Detalle de Certificados de Creditos Vigentes	2010-09-27	ESPERA	Eliminar	
Certificado de Creditos al Dia	Detalle de Certificados de Creditos Vigentes	2010-09-27	ESPERA	Eliminar	
Certificado de Creditos al Dia	Detalle de Certificados de Creditos Vigentes	2010-09-27	ESPERA	Eliminar	
Certificado de Creditos al Dia	Detalle de Certificados de Creditos Vigentes	2010-09-27	ESPERA	Eliminar	

Figura 6.29 listado de peticiones de certificados

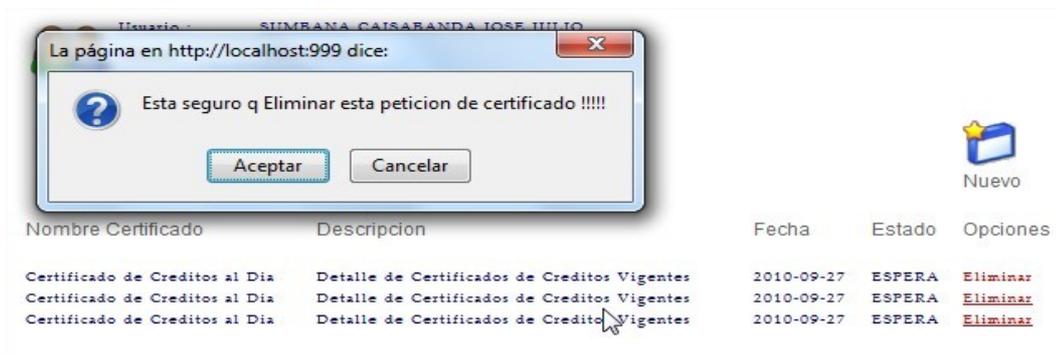


Figura 6.30 Eliminar certificados

Si lo que desea es no adquirir ese certificado o se equivocó al seleccionarlo puede eliminarlo de lista de peticiones dando click en eliminar, el cual no como aun no se ha procesado no tiene ningún costo.

PRE-APERTURA DE CUENTAS

Este módulo esta creado para futuros socios que deseen abrir una cuenta y agilizar este trámite mediante el internet.



Figura 6.31 ingreso de cedula

Aquí se debe ingresar una cedula valida que sera la clave de cada usuario que desee obtener una cuenta en la Cooperativa.

PASO 2 : VERIFICACION DE PREAPERTURA					
PREAPERTURA DE CUENTA					
Sr. Usuario, si usted puede ver esta pagina es porque ya ha ingresado sus datos para una preapertura de cuenta y en este momento esta siendo procesada.					
Cedula	Nombres	Apellidos	Fecha de Ingreso	Estado	Opciones
1804029187	Wilson Roberto	Cunalata Pachacama	10-09-2010	ESPERA	Editar Eliminar

Figura 6.32 Lista de ingresos

Se puede ver si un futuro cliente ya ha ingresado una solicitud de apertura de cuenta, ya que el sistema y por políticas de la Institución solo se permite tener una solicitud de cuenta en espera y no puede ingresar otra.

Por lo cual en la página se visualiza los datos más referencias que existen en petición, el estado y dando las opciones de eliminar o modificar según sea el caso requerido por el cliente.

Y en el caso de que aún no haya hecho una solicitud de cuenta, entonces se puede visualizar el formulario de pre-apertura:

PASO 3 : DATOS CUENTA	
Cliente Naturales	
Identificación : CEDULA	Codigo : [] # Cliente : ?????? Estado PRE APERTURADO
DATOS GENERALES CLIENTES	
Campos Requeridos (*)	
Apellidos :(*) []	Número de Cargas :(*) []
Nombres :(*) []	Fecha Ingreso :(*) []
Dirección :(*) []	Email :(*) []
Telefono :(*) []	Provincia :(*) Seleccione
Celular :(*) []	Canton :(*) Seleccione
Empresa Telefonica :(*) MOVISTAR	Parroquia :(*) Seleccione
Sexo :(*) MASCULINO	Tipo Vivienda :(*) Seleccione
Estado Civil :(*) SOLTERO	Representante 1 :(*) []
Nivel Educacion :(*) NINGUNA	Representante 2 :(*) []
Titulo Profesional :(*) Seleccione	Es nacional :(*) <input type="checkbox"/>
Actividad Económica :(*) Seleccione	Pais :(*) []
Fecha Nacimiento :(*) []	

Figura 6.33 Formulario de preapertura de cuenta

Como informativo podemos decir que todo lo que se encuentra con * son datos obligatorios de ingreso.

En este formulario se debe ingresar todos los datos personales que se solicitan y se deben ingresar de manera obligatoria.

REFERENCIAS CLIENTES

Bancarias : Banco : Cuenta :

Personales : Nombre : Dirección : Telefono :

Comerciales : Nombre : Dirección : Telefono :

Figura 6.34 Formulario de pre-apertura datos referenciales.

Lo último que se ingresa son los datos de referencia de un cliente para poder validar la información antes ingresada sea verdadera y verificable.

Para todo lo referente a la parte de banca en línea el cliente posee una ayuda como enlace en la que puede visualizar ayuda importantes que le servirán para el mejor uso del sistema de banca en línea de la Cooperativa de Ahorro y Crédito “Chibuleo” Ltda.

ANEXO 5

MANUAL DE ADMINISTRADOR

1. Loguearse como Administrador

Se recomienda leer primero las instrucciones de clientes para tener un mejor conocimiento de las áreas que se van a administrar.

En la mayoría de sistemas web existe una sección de logeo de usuarios, donde se procederá a logearse como administrador. Este tipo de usuario solo puede ser creado directamente en la base de datos por el administrador del portal web. También existen usuarios pero con menos privilegios que un administrador.

Para poder loguearnos como administrador o usuarios del sistema se debe acceder a la misma página que los clientes solo que no ingresa ningún tipo de cedula, sino el nombre del usuario que se le ha dado, por ejemplo:



The image shows a web login form with a red header bar containing the word "INGRESO". Below the header, the text "FORMULARIO DE ACCESO" is displayed. There are two input fields: "Cédula:" with the text "admin" entered, and "Clave:" with masked characters represented by dots. Below the fields are two buttons: "Ingresar" and "Cancelar". A red note is present, stating "Nota (*) Formato Incorrecto ." and "(**) Campo de Ingreso requerido.". At the bottom right, there is a red link that says "Olvidaste tu clave??".

Figura 6.35 Logueo Administrador

Entonces se puede ver que no solo se puede ingresar número de cedula en el campo cedula, sino el nombre de un usuario sea administrador o común.

Se realizan pueden llegar a dar los mismos errores que cuando se ingresa un cliente.

Luego de realiza un ingreso correcto de usuario y clave se puede ver el panel de administración:



Figura 6.36 Menú administrador

En el cual podemos distinguir primero:

- Su cabecera con el logo de la Cooperativa
- A su izquierda superior sus accesos rápidos y a continuación sus parámetros de acceso.
- En el centro de la página los enlaces a las áreas implementadas como son:
 - o Usuarios
 - o Clientes
 - o Accesos
 - o Noticias

- o Formulario de prestamos
- o Formularios de certificados
- o Formularios de pre-apertura de cuenta

Las cuales están restringidas según los niveles de acceso que posea el usuario.

Crear Usuarios

Para utilizar esta área debemos dar click en el enlace:



Y se presenta el listado de todos los usuarios existentes en el sistema:

ADMINISTRACION DE USUARIOS						
 LISTADO DE USUARIOS						
						 Nuevo
						
Login	Clave	Descripción	Creacion	Estado Online	Opciones	
ADMIN	****	PROGRAMADOR	2010-09-04	EN LINEA	Editar	Eliminar
ADMINISTRADOR	****	ADMINISTRADOR DE SITIO	2010-09-04	FUERA DE LINEA	Editar	Eliminar
DMAR CERLO	****	USUARIO DEL SISTEMA	2010-09-18	FUERA DE LINEA	Editar	Eliminar
PRUEBA	****	USUARIO DEL SISTEMA	2010-09-08	FUERA DE LINEA	Editar	Eliminar
WCUNALATA	****	USUARIO DEL SISTEMA	2010-09-05	FUERA DE LINEA	Editar	Eliminar
WILSON	****	USUARIO DEL SISTEMA	2010-09-05	FUERA DE LINEA	Editar	Eliminar
WWW	****	USUARIO DEL SISTEMA	2010-09-05	FUERA DE LINEA	Editar	Eliminar

Figura 7.37 Listado de Usuarios

Como se lo puede ver en la figura teniendo la posibilidad de realizar las operaciones de modificar su información o su tipo o hasta de eliminarlo como se puede ver en las opciones del sistema.

- **Nuevo Usuario.-**

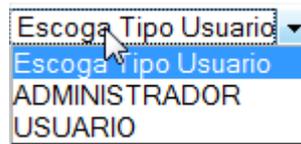
Para la creación de un nuevo usuario se debe dar click en  que se encuentra en la parte superior del listado de los mismos y se carga el formulario:



The screenshot shows a web interface for user management. At the top, there is a red header with the text 'ADMINISTRACION DE USUARIOS'. Below the header, there is a yellow gear icon followed by the text 'NUEVO USUARIO'. The form contains four input fields: 'Nombre Completo', 'Nombre Usuario', 'Clave', and 'Tipo'. The 'Tipo' field is a dropdown menu with the text 'Escoga Tipo Usuario'. At the bottom of the form, there are two buttons: 'Crear' and 'Cancelar'.

Figura 6.38 Nuevo Usuario

En el cual se debe ingresar el nombre completo de la persona, nombre de usuario, su clave que se genera automáticamente y el tipo:

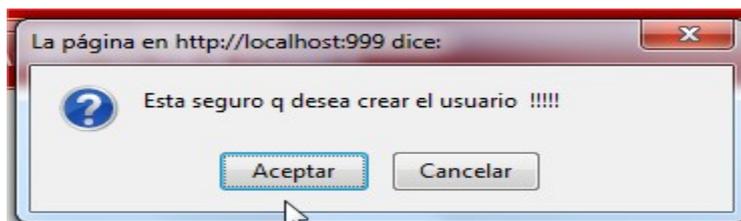


Que es donde se restringe el acceso a las áreas de donde se crean usuarios, se ven accesos.

Luego de estos ingresos dar click en crear y en le caso que no se haya ingresado algún dato se presenta:



Para dar a conocer que falta algún dato de ingresar en el sistema, y si ingresamos correctamente los datos:



El cual pide confirmación de que se desee crear al usuario, para luego permitir, Imprimir y entregar su clave al usuario.

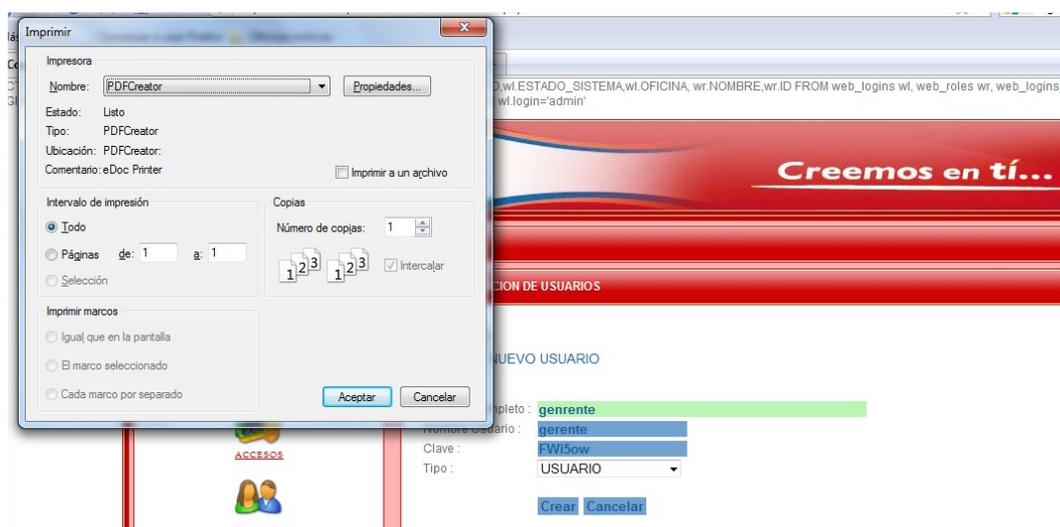


Figura 6.39 Impresión de usuario

Accesos

Esta área fue creada como modulo auditor para tener un conocimiento de quien accede al sistema tomando en cuenta puntos como el nombre o número de cedula, la dirección de su conexión y la operación que ha realizado en el sistema.



Su icono de acceso es  y se lo encuentra en le área de menús rápidos o en la página principal.

A continuación un posible visualización de los datos de este módulo:



USUARIO	IP	FECHA
0010001393	127.0.0.1	2010-09-07 12:44:15
0010001393	127.0.0.1	2010-09-07 12:44:41
0010001393	127.0.0.1	2010-09-07 12:47:10
0010001393	127.0.0.1	2010-09-07 12:47:28
0010001393	127.0.0.1	2010-09-07 12:47:50
ADMIN	127.0.0.1	2010-09-08 11:33:11
ADMIN	127.0.0.1	2010-09-08 11:34:25
PRUEBA	127.0.0.1	2010-09-08 11:34:37
PRUEBA	127.0.0.1	2010-09-08 11:34:55
0010001393	127.0.0.1	2010-09-08 12:22:59
0010001393	127.0.0.1	2010-09-08 12:23:22
0010001393	127.0.0.1	2010-09-08 12:23:34
0010001393	127.0.0.1	2010-09-08 12:23:52
0010001393	127.0.0.1	2010-09-08 12:23:53
ADMIN	127.0.0.1	2010-09-08 12:24:02
ADMIN	127.0.0.1	2010-09-08 12:24:43
0010001393	127.0.0.1	2010-09-08 12:24:58
0010001393	127.0.0.1	2010-09-08 12:24:59

Figura 6.40 Listado de Accesos

Cientes

Esta es el área donde se realiza la creación de clientes al sistema, para lo cual su



icono es , el cual primero pide la cedula del cliente y lo valida con los que se encuentran activos en el Sistema Financiero Chibuleo y me devuelve datos posibles del mismo, para después simplemente pedirme el ingreso de email y la generación de la clave, con lo cual queda activo el usuario en el sistema.

n	Clave	Creacion	Estado Online	Opciones
MIN	****	2010-09-04	EN LINEA	Editar
MINISTRADOR	****	2010-09-04	FUERA DE LINEA	Editar
ARCERLO	****	2010-09-18	FUERA DE LINEA	Editar
EBA	****	2010-09-08	FUERA DE LINEA	Editar
NALATA	****	2010-09-05	FUERA DE LINEA	Editar
ENTE	****	2010-09-27	FUERA DE LINEA	Editar
SON	****	2010-09-05	FUERA DE LINEA	Editar
V	****	2010-09-05	FUERA DE LINEA	Editar

En el cual se visualiza los usuarios activos y si ya están en el sistema, pudiendo editar para regenerar su clave.

NUEVO Cliente

Nombre Completo : CRIOLLO LLANGANATE JOHN STEVEN

Numero de Cedula: 0011003111

Clave : RB0HcM

[Crear](#) [Cancelar](#)

Figura 6.41 Nuevo Cliente

Se puede ver el formulario para creación de un cliente en el que el sistema genera los datos de el cliente, y generando la clave para imprimir y dársela al cliente.

Formularios de certificados

En esta área se realiza la revisión de peticiones de certificados y aprobación de los



mismos su icono es  en el cual después de realizar las operaciones, se envía un mensaje al cliente que lo puede retirar en cualquiera de las oficinas de la Cooperativa.

Formularios de pre-apertura

En esta área se realiza la revisión todas las peticiones de pre-apertura y revisión



de datos de las mismas cambio de estados su icono es  en el cual después de realizar cada operación, se sigue almacenado mensajes de todo el proceso, hasta que llegue al estado de pre aprobado y se acerque a las oficinas para realizar ya el paso final para el préstamo.

Formularios de préstamos

En esta área se realiza la revisión de peticiones de préstamos y aprobación de los



mismos su icono es  en el cual después de realizar las operaciones, se envía un mensaje al cliente que lo puede retirar en cualquiera de las oficinas de la Cooperativa.