

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN TECNOLÓGICA

ANÁLISIS DE SISTEMAS INFORMÁTICOS

**“DISEÑO E IMPLEMENTACIÓN DE UN ADMINISTRADOR DE CONTENIDOS
CMS (CONTENT MANAGAMENT SYSTEM) Y PORTAL WEB PARA LA COAC
(COOPERATIVA DE AHORRO Y CRÉDITO) ALIANZA DEL VALLE”**

LUIS RENE GUALOTUÑA GUALOTUÑA

DIRECTORA: ING. ROSA NAVARRETE MSc.

Quito, Septiembre del 2006

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Luis René Gualotuña Gualotuña, bajo mi supervisión.

**Ing. Rosa Navarrete MSc.
DIRECTORA DE PROYECTO**

DECLARACIÓN

Yo, Luis René Gualotuña Gualotuña, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional y que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Luis René Gualotuña Gualotuña

AGRADECIMIENTO

A Dios por su infinito Amor,
Por darme su luz cuando más lo necesitaba
Por ser mi amigo y mi guía en los pasos de mi vida.

A mis padres Luis Enrique y María Luisa por
su apoyo permanente e incondicional

A Elías Martín el PEQUE, por ser la fuente de mi inspiración,
mi alegría, y la fuerza para luchar cada día, Te Amo Hijo.

A Silvia, que con su cariño y comprensión pudo llevarme
a la culminación de esta meta,

A mis hermanos Edison, Marithza, Carmen, Marcel, Blanca
por su preocupación y confianza.

A todos mis amigos en especial a Vero y Mayrita,
Quienes me devolvieron la confianza cuando todo lo sentí perdido.

Al Ing. Rosa Navarrete por el tiempo dedicado para el correcto
desarrollo de esta tesis.

René

DEDICATORIA

Creí siempre en los números, que la lógica llevaba a la razón,
Pero di vida a tal búsqueda y no se aun lo que es lógico,
ni quien define la razón, indague hondamente,
en la programación, en las redes, en lo irreal,
Y he vuelto, con el descubrimiento más importante de mi carrera,
el mas importante de mi vida entera,
y es que solo en la misteriosas ecuaciones del amor,
donde existe una razón verdadera,
DEDICADO PARA USTEDES,
SON MI RAZÓN, SON TODAS MIS RAZONES

M. S.

René

CONTENIDO

CAPITULO 1	1
1. INTRODUCCIÓN	1
1.1 PLANTEAMIENTO DEL PROBLEMA	1
1.2 OBJETIVOS	1
1.2.1. OBJETIVO GENERAL.....	1
1.2.2. OBJETIVOS ESPECÍFICOS	1
1.3 JUSTIFICACIÓN	2
1.4 ALCANCE	2
1.5 LIMITACIONES.....	2
CAPITULO 2	3
2. MARCO TEÓRICO.....	3
2.1 INGENIERÍA WEB	3
2.1.1 INTRODUCCION.....	3
2.1.2 EL PROCESO DE INGENIERÍA WEB	3
2.1.2.1 La Formulación.-	4
2.1.2.2 La Planificación.-	4
2.1.2.3 La Modelización.-.....	4
2.1.2.4 La Generación de páginas.-	4
2.1.2.5 El Test.-	4
2.1.2.6 Finalmente.....	4
2.1.3 CONTROL Y GARANTÍA DE LA CALIDAD	5
2.1.3.1 La Usabilidad	5
2.1.3.1.1 Empírica	5
2.1.3.1.2 Relativa	5
2.1.3.1.3 Definiciones formales	5
2.1.3.1.4 Otros aspectos de la usabilidad	6
2.1.3.1.5 Beneficios de la usabilidad.....	7
2.1.3.2 Fiabilidad.-	8
2.1.3.2.1 Disponibilidad:.....	8

2.1.3.2.2 Calidad en las búsquedas:.....	8
2.1.3.3 Seguridad.-	8
2.1.3.4 Eficiencia.-	8
2.1.3.5 Mantenibilidad.-	8
2.1.4 CONTROL DE LA CONFIGURACIÓN	8
2.1.4.1 Contenido:	9
2.1.4.2 Personal:	9
2.1.4.3 Escalabilidad.....	9
2.1.4.4 Política	9
2.1.5 LA GESTIÓN DEL PROCESO.....	9
2.2 ARQUITECTURA DE LA INFORMACIÓN.....	9
2.2.1 DEFINICIÓN	9
2.2.2 ARQUITECTURA WEB.....	11
2.2.3 ARQUITECTURA DE LA INFORMACIÓN COMO PROCESO	12
2.2.4 ARQUITECTURA DE LA INFORMACIÓN COMO ABSTRACCIÓN ...	13
2.2.5 EL ARQUITECTO DE INFORMACIÓN	13
2.2.6 LA EXPERIENCIA DE USUARIO.....	14
2.2.6.1 Diseño de información	14
2.2.6.1.1 Diseño gráfico.-	15
2.2.6.1.2 Informática.-	15
2.2.6.1.3 Documentación Técnica.-.....	15
2.2.6.2 Interacción	16
2.3 DISEÑO DE PAGINAS WEB.....	16
2.3.1 DISEÑO WEB APLICADO	16
2.3.2 ETAPAS	17
2.4 APLICACIONES WEB Y EL DESARROLLO DE CAPAS	18
2.4.1 ARQUITECTURAS DE TRES NIVELES O PROGRAMACIÓN POR CAPAS.....	18
2.4.1.1 Capas o Niveles.....	19
2.4.1.1.1 Capa de presentación:	19
2.4.1.1.2 Capa de negocio:	19
2.4.1.1.3 Capa de datos:.....	19
2.4.2 SOFTWARE INTERMEDIO (Middleware).....	23

2.5	HERRAMIENTAS DE CONSTRUCCION WEB.....	25
2.5.1	CAPA DE PRESENTACIÓN	25
2.5.1.1	JAVASCRIPT	25
2.5.1.2	HTML	26
2.5.1.2.1	Historia del estándar	26
2.5.2	CAPA DE NEGOCIO.....	27
2.5.2.1	PHP.....	27
2.5.2.1.1	Visión general	27
2.5.2.1.2	Historia.....	28
2.5.2.1.3	Ventajas de PHP.....	29
2.5.2.2	SERVIDOR WEB APACHE.....	29
2.5.3	CAPA DE DATOS	30
2.5.3.1	MySQL	30
2.5.4	MACROMEDIA FLASH MX 2004.....	31
2.5.5	MACROMEDIA DREAMWEAVER MX 2004	32
2.5.6	MACROMEDIA FIREWORKS MX 2004.....	33
CAPITULO 3	34
3.	ASPECTOS METODOLOGICOS.....	34
3.1	PARADIGMA ESPIRAL ORIENTADO A LA WEB	34
3.1.1	COMUNICACIÓN CON EL CLIENTE O FORMULACION	34
3.1.2	PLANIFICACIÓN.....	34
3.1.3	ANÁLISIS DE RIESGOS	35
3.1.4	INGENIERÍA.....	35
3.1.5	GENERACIÓN DE PÁGINAS.....	35
3.1.6	EVALUACIÓN EL CLIENTE	35
3.2	METODOLOGÍA	36
3.3	METODOLOGÍA OOHDM.....	37
3.3.1	INTRODUCCIÓN A OOHDM	37
3.3.2	FASES DE LA METODOLOGÍA OOHDM	37
3.3.3	CAPTURA DE REQUERIMIENTOS.....	38
3.3.4	DISEÑO CONCEPTUAL	39
3.3.5	DISEÑO NAVEGACIONAL	39

3.3.6	DISEÑO ABSTRACTO DE INTERFAZ	40
3.3.7	IMPLEMENTACION	41
3.3.8	PRUEBAS	41
3.4	UML (UNIFIED MODELING LANGUAGE)	42
3.4.1	DIAGRAMAS PARA MODELAMIENTO	43
3.4.1.1	Diagramas de Caso de Uso	43
3.4.1.1.1	Notación del Diagrama de Caso de uso.-	43
3.4.1.1.2	Relaciones de los Casos de uso.-	44
3.4.1.2	Diagrama de Clases	44
3.4.1.2.1	Clase.-	45
3.4.1.2.2	Atributos:.....	45
3.4.1.2.3	Métodos:.....	46
3.4.1.2.4	Relaciones entre Clases:.....	46
3.4.1.2.5	Casos Particulares:.....	49
3.4.1.3	Diagramas de Interacción.....	50
3.4.1.3.1	Diagrama de Secuencia.-	51
3.4.1.3.1	Diagrama de Colaboración.-	52
3.4.1.4	Diagrama de Estados	53
3.4.1.5	Diagrama de Actividades	54
3.4.1.5.1	Estado de Acción.-	56
3.4.1.5.2	Estados de Actividad.-	56
3.4.1.5.3	Inicio y Terminación.-	56
3.4.1.5.4	Transiciones.-.....	57
3.4.1.5.5	Bifurcaciones (Decisiones).-.....	57
3.4.1.5.6	Barras de Sincronización.-	57
3.4.1.5.7	Calles.-.....	58
3.4.1.5.8	Señales.-	58
3.4.1.5.9	Señal de envío.-	58
3.4.1.5.10	Señal de recibo.-	58
CAPITULO 4	60
4.	CONCLUSIONES Y RECOMENDACIONES	60
4.1	CONCLUSIONES	60

4.2 RECOMENDACIONES	60
BIBLIOGRAFÍA	61
ANEXOS	63
MANUAL TÉCNICO	64
MANUAL DE USUARIO	1

CAPITULO 1

1. INTRODUCCIÓN

1.1 PLANTEAMIENTO DEL PROBLEMA

La Cooperativa de Ahorro y Crédito Alianza del Valle Ltda., ubicada su casa matriz en Quito, Valle de los Chillos sector de Chaupitena, vía antigua Conocoto Amaguaña, es una institución financiera, con 36 años de experiencia, dedicada al ahorro y préstamo de capital, bajo el control de la Superintendencia de Bancos y Seguros, dando mayor confiabilidad y seguridad a sus socios.

La necesidad de establecer una imagen institucional sólida y de apertura a nuevos nichos de mercado, hace que la institución decida lanzar su sitio Web.

La propuesta de un sitio Web permitirá contar con una herramienta de entrega de información a sus socios y además, lanzar publicidad y marketing de la Cooperativa de Ahorro y Crédito Alianza del Valle, para atraer potenciales clientes.

1.2 OBJETIVOS

1.2.1. OBJETIVO GENERAL

Crear para la COAC Alianza del Valle una herramienta de marketing y negocios en la captación de nuevos clientes con el desarrollo de un portal Web.

1.2.2. OBJETIVOS ESPECÍFICOS

- a) Desarrollar un CMS para mantener información actualizada en el portal web.
- b) Ofrecer información actualizada sobre los servicios, beneficios y promociones financieras de la institución
- c) Proporcionar información a los clientes de su estado de cuenta mediante un usuario y password asignado por el administrador de la aplicación Web.

1.3 JUSTIFICACIÓN

El desarrollo del portal Web para la Cooperativa de Ahorro y Crédito ALIANZA DEL VALLE, permitirá que la institución y sus clientes se encuentren todo el tiempo comunicados e informados, y servirá además como una herramienta de marketing y negocios en la captación de nuevos clientes, promoviendo el mercado financiero.

1.4 ALCANCE

Diseñar y Programar los componentes necesarios para el CMS y portal Web. de la Cooperativa de Ahorro y Crédito ALIANZA DEL VALLE. Además de Incluir una pantalla de consultas del saldo disponibles por cliente.

1.5 LIMITACIONES

El portal Web desarrollado para la Cooperativa de Ahorro y Crédito ALIANZA DEL VALLE presenta las siguientes limitaciones:

- No se podrá realizar transacciones en línea.
- Temporalmente la conexión para consultar, el estado de la cuenta de clientes, se lo realizara a través de MySql.

CAPITULO 2

2. MARCO TEÓRICO

2.1 INGENIERÍA WEB

2.1.1 INTRODUCCION

La Ingeniería de la Web es la aplicación de metodologías sistemáticas, disciplinadas y cuantificables al desarrollo eficiente, operación y evolución de aplicaciones de alta calidad en la World Wide Web

http://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_la_Web

En 1998, Roger Pressman moderó una mesa redonda virtual con representantes la ingeniería software tradicional y del desarrollo software basado exclusivamente en Internet. El debate principalmente se centró en discutir si valía la pena aplicar un proceso de ingeniería a las aplicaciones con base en Internet, o qué características tenían éstas que justificaran el no utilizarlo. La conclusión general fue que aplicar un proceso de ingeniería nunca es una mala idea pero que éste debería adaptarse a los requerimientos de cambio continuo y rapidez siempre presentes en el proceso de desarrollo Web. De iniciativas como ésta y de otras como la organización de congresos y talleres especializados en el desarrollo para la Web, surge el nacimiento de una nueva disciplina denominada Ingeniería Web.

Escuetamente podemos “traducir” como el proceso utilizado para crear, implantar y mantener aplicaciones y sistemas Web de alta calidad.

Esta breve definición nos lleva a abordar un aspecto clave de cualquier proyecto como es determinar que tipo de proceso es más adecuado en función de las características del mismo.

2.1.2 EL PROCESO DE INGENIERÍA WEB

Características como inmediatez y evolución y crecimiento continuos, nos llevan a un proceso incremental y evolutivo, que permite que el usuario se involucre activamente, facilitando el desarrollo de productos que se ajustan mucho lo que éste busca y necesita. Según Presuman, las actividades que formarían parte del

marco de trabajo incluirían las tareas abajo enumeradas. Dichas tareas serían aplicables a cualquier aplicación Web, independientemente del tamaño y complejidad de la misma.

Las actividades que forman parte del proceso son: *formulación, planificación análisis, modelización, generación de páginas, test y evaluación del cliente.*

2.1.2.1 La Formulación.- identifica objetivos y establece el alcance de la primera entrega.

2.1.2.2 La Planificación.- genera la estimación del coste general del proyecto, la evaluación de riesgos y el calendario del desarrollo y fechas de entrega. El Análisis especifica los requerimientos e identifica el contenido.

2.1.2.3 La Modelización.- se compone de dos secuencias paralelas de tareas. Una consiste en el diseño y producción del contenido que forma parte de la aplicación. La otra, en el diseño de la arquitectura, navegación e interfaz de usuario. Es importante destacar la importancia del diseño de la interfaz. Independientemente del valor del contenido y servicios prestados, una buena interfaz mejora la percepción que el usuario tiene de éstos.

2.1.2.4 La Generación de páginas.- se integra contenido, arquitectura, navegación e interfaz para crear estática o dinámicamente el aspecto más visible de las aplicación, las páginas.

2.1.2.5 El Test.- busca errores a todos lo niveles: contenido, funcional, navegacional, rendimiento, etc. El hecho de que las aplicaciones residan en la red, y que ínter operen en plataformas muy distintas, hace que el proceso de test sea especialmente difícil.

2.1.2.6 Finalmente, el resultado es sometido a la evaluación del cliente.

2.1.3 CONTROL Y GARANTÍA DE LA CALIDAD

Una de las tareas colaterales que forman parte del proceso es el Control y Garantía de la Calidad (CGC). Todas las actividades CGC de la ingeniería software tradicional como son: establecimiento y supervisión de estándares, revisiones técnicas formales, análisis, seguimiento y registro de informes, etc, son igualmente aplicables a la Ingeniería Web. Sin embargo, en la Web toman especial relevancia para valorar la calidad aspectos como: Usabilidad, Funcionalidad, Fiabilidad, Seguridad, Eficiencia y Mantenibilidad.

<http://www.informandote.com/jornadasIngWEB/articulos/jiw01.pdf>

2.1.3.1 La Usabilidad (del inglés usability) es una medida empírica y relativa acerca de lo fácil, rápido y agradable que es utilizar un determinado producto o servicio.

2.1.3.1.1 Empírica porque no se basa en opiniones o sensaciones sino en pruebas (del inglés tests) de usabilidad, realizadas en laboratorio u observadas mediante trabajo de campo.

2.1.3.1.2 Relativa porque el resultado no es ni bueno ni malo, sino que depende de las metas planteadas (por lo menos el 80% de los usuarios de un determinado grupo o tipo definido deben poder instalar con éxito el producto X en N minutos sin más ayuda que la guía rápida) y/o de una comparación con otros sistemas similares.

La usabilidad viene determinada por tres aspectos clave:

- El tipo o tipos específicos de usuario/s,
- La tarea o tareas que efectúan objeto de observación, y
- El contexto en el que se da la interacción.

2.1.3.1.3 Definiciones formales La Organización Internacional para la Estandarización (ISO) ofrece dos definiciones de usabilidad además se añade una de Redish.

- ISO/IEC 9126:

"La usabilidad se refiere a la capacidad de un software de ser comprendido, aprendido, usado y ser atractivo para el usuario, en condiciones específicas de uso"

Esta definición hace énfasis en los atributos internos y externos del producto, los cuales contribuyen a su funcionalidad y eficiencia. La usabilidad depende no sólo del producto sino también del usuario. Por ello un producto no es en ningún caso intrínsecamente usable, sólo tendrá la capacidad de ser usado en un contexto particular y por usuarios particulares. La usabilidad no puede ser valorada estudiando un producto de manera aislada (Bevan, 1994).

- ISO/IEC 9241:

"Usabilidad es la eficiencia y satisfacción con la que un producto permite alcanzar objetivos específicos a usuarios específicos en un contexto de uso específico"

Es una definición centrada en el concepto de calidad en el uso, es decir, se refiere a cómo el usuario realiza tareas específicas en escenarios específicos con efectividad.

- Redish:

" La usabilidad se refiere a encontrar lo que necesitan, entender lo que encuentran y actuar apropiadamente... dentro del tiempo y esfuerzo que ellos consideran adecuado para esa tarea".

2.1.3.1.4 Otros aspectos de la usabilidad

A partir de la conceptualización llevada a cabo por la ISO, se infieren los principios básicos en los que se basa la usabilidad:

Facilidad de Aprendizaje: facilidad con la que nuevos usuarios desarrollan una interacción efectiva con el sistema o producto. Está relacionada con la predicibilidad, sintetización, familiaridad, la generalización de los conocimientos previos y la consistencia.

Flexibilidad: relativa a la variedad de posibilidades con las que el usuario y el sistema pueden intercambiar información. También abarca la posibilidad de

diálogo, la multiplicidad de vías para realizar la tarea, similitud con tareas anteriores y la optimización entre el usuario y el sistema.

Robustez: es el nivel de apoyo al usuario que facilita el cumplimiento de sus objetivos. Está relacionada con la capacidad de observación del usuario, de recuperación de información y de ajuste de la tarea al usuario.

En informática, la usabilidad está muy relacionada con la accesibilidad, hasta el punto de que algunos expertos consideran que una forma parte de la otra o viceversa. Uno de estos expertos y gurú de la usabilidad en los entornos web es Jakob Nielsen, el cual definió la usabilidad en el 2003 como "un atributo de calidad que mide lo fáciles de usar que son las interfaces web".

2.1.3.1.5 Beneficios de la usabilidad

Actualmente la usabilidad está reconocida como un importante atributo de calidad del software, habiéndose ganado un puesto entre atributos más tradicionales como el rendimiento y la fiabilidad. Incluso diversos programas de estudios se centran en ella. También han surgido diversas empresas de consultoría de usabilidad, y las firmas tradicionales de consultoría y diseño están ofreciendo servicios similares. Entre los principales beneficios encontramos:

- Reducción de los costes de aprendizaje.
- Disminución de los costes de asistencia y ayuda al usuario.
- Optimización de los costes de diseño, rediseño y mantenimiento.
- Aumento de la tasa de conversión de visitantes a clientes de un sitio web.
- Mejora la imagen y el prestigio.
- Mejora la calidad de vida de los usuarios, ya que reduce su estrés, incrementa la satisfacción y la productividad.

Todos estos beneficios implican una reducción y optimización general de los costes de producción, así como un aumento en la productividad. La usabilidad permite mayor rapidez en la realización de tareas y reduce las pérdidas de tiempo. Un caso real, después de ser rediseñado prestándose especial atención a la usabilidad, el sitio web de IBM incrementó sus ventas en un 400% (InfoWorld, 1999).

<http://es.wikipedia.org/wiki/Usabilidad>

2.1.3.2 Fiabilidad.- capacidad del portal para llevar a cabo su funcionalidad de forma precisa”. Esta dimensión además se verá afectada por:

2.1.3.2.1 Disponibilidad: El portal debe estar siempre operativo, para que los usuarios puedan acceder a él.

2.1.3.2.2 Calidad en las búsquedas: Los resultados que nos proporciona el portal al realizar una búsqueda deben ser adecuados a la petición realizada por el usuario.

2.1.3.3 Seguridad.- que el acceso a los servicios, datos y meta-datos del repositorio provean distintos niveles de seguridad conforme a distintos roles bien definidos.

2.1.3.4 Eficiencia.- la eficiencia de una aplicación es su capacidad de ayudar a los usuarios finales a alcanzar sus objetivos en menos tiempo que el requerido para alcanzar los mismos objetivos utilizando otros instrumentos Algunos indicadores para medir la eficiencia son:

- El tiempo necesario para completar exitosamente una transacción
- Número de transacciones completadas con éxito en un periodo de tiempo dado

2.1.3.5 Mantenibilidad.- cuando mayor es el número de páginas que componen la aplicación Web más difícil de mantener resultará

[http:// alarcos.inf-cr.uclm.es/doc/calidadSI/Metodos%20De%20Calidad.ppt](http://alarcos.inf-cr.uclm.es/doc/calidadSI/Metodos%20De%20Calidad.ppt)

2.1.4 CONTROL DE LA CONFIGURACIÓN

Establecer mecanismos adecuados de control de la configuración para la Ingeniería Web es uno de los mayores desafíos a los que esta nueva disciplina se enfrenta. La Web tiene características únicas que demandan estrategias y herramientas nuevas. Hay cuatro aspectos importantes a tener en cuenta en el desarrollo de tácticas de control de la configuración para la Web.

2.1.4.1 Contenido: Considerando la dinamicidad con la que el contenido se genera, es tarea compleja organizar racionalmente los objetos que forman la configuración y establecer mecanismos de control.

2.1.4.2 Personal: Cualquiera realiza cambios. Hay mucho personal no especializado que no reconoce la importancia que tiene el control del cambio.

2.1.4.3 Escalabilidad: Es común encontrar aplicaciones que de un día para otro crecen considerablemente. Sin embargo, las técnicas de control no escalan de forma adecuada.

2.1.4.4 Política: ¿Quién posee la información? ¿Quién asume la responsabilidad y coste de mantenerla?

2.1.5 LA GESTIÓN DEL PROCESO

En un proceso tan rápido como es el proceso de Ingeniería Web, donde los tiempos de desarrollo y los ciclos de vida de los productos son tan cortos, ¿merece la pena el esfuerzo requerido por la gestión? La respuesta es que dada su complejidad es imprescindible. Entre los aspectos que añaden dificultad a la gestión destacamos: - alto porcentaje de contratación a terceros, - el desarrollo incluye una gran variedad de personal técnico y no técnico trabajando en paralelo, el equipo de desarrollo debe dominar aspectos tan variopintos como, software basado en componentes, redes, diseño de arquitectura y navegación, diseño gráfico y de interfaces, lenguajes y estándares en Internet, test de aplicaciones Web, etc, lo que hace que el proceso de búsqueda y contratación de personal sea arduo.

<http://www.informandote.com/jornadasIngWEB/articulos/jiw01.pdf>

2.2 ARQUITECTURA DE LA INFORMACIÓN

2.2.1 DEFINICIÓN

El término "Arquitectura de la Información" (AI) fue utilizado por primera vez por Richard Saul Wurman en 1975, quién la define como:

“El estudio de la organización de la información con el objetivo de permitir al usuario encontrar su vía de navegación hacia el conocimiento y la comprensión de la información”

Si nos ceñimos exclusivamente a la AI en el campo de la Web, una de las definiciones que Louis Rosenfeld y Peter Morville ofrecen en su libro "Information Architecture for the World Wide Web 2nd Edition", puede sernos de más fácil comprensión:

“El arte y la ciencia de estructurar y clasificar sitios web e intranets con el fin de ayudar a los usuarios a encontrar y manejar la información.”

<http://www.nosolousabilidad.com/articulos/ai.htm>

En relación a la World Wide Web el Information Architecture Institute define la Arquitectura de la Información como:

- *El diseño estructural en entornos de información compartida.*
- *El arte y la ciencia de organizar y rotular sitios web, intranets, comunidades en línea y software para promover la usabilidad y encontrabilidad.*
- *Una comunidad emergente orientada a aplicar los principios del diseño y la arquitectura en el entorno digital.*

http://es.wikipedia.org/wiki/Arquitectura_de_la_informaci%C3%B3n

El concepto "Arquitectura de la Información" no solo engloba la actividad de organizar información, sino también el resultado de dicha actividad. La arquitectura de la información de un sitio web, como resultado de la actividad, comprende los sistemas de organización y estructuración de los contenidos, los sistemas de rotulado o etiquetado de dichos contenidos, y los sistemas de recuperación de información y navegación que provea el sitio web.

Independientemente de la velocidad, están los problemas de Navegación, de Organización, de Estructura, de Usabilidad, es decir de la Arquitectura de la información. En definitiva de Arquitectura Web.

El "diseño" cumple una importante función, pero no es la única que compone una web atractiva y funcional y, al parecer, algunos lo han olvidado. Un sitio web se construye, no sólo se diseña. Hay que juntar diferentes piezas, que deben encajar convenientemente para que el usuario/navegante pueda, al menos, conocernos. Si no es así, difícilmente podrá convertirse en un posible cliente.

El principal "atractivo" de una web, debería residir en su contenido y funcionalidad. La Arquitectura Web pretende reunir todas las piezas y juntarlas en la forma más adecuada para construir el sitio que desea nuestro público objetivo.

La web debe construirse para el usuario. No para el diseñador, ni para el directivo que lo ha contratado, ni para su Consejo de Administración. Presentación, diseño, navegación, marketing, interactividad, estructura, flujo transaccional, información... son algunas de las piezas que debe contemplar el Arquitecto de la web, para su correcta construcción.

<http://www.webaprendiz.com/viewp/notas/arquiweb/2001/06/id0001.htm>

2.2.2 ARQUITECTURA WEB

De la misma forma en que los arquitectos tradicionales diseñan y coordinan la construcción de edificios, los arquitectos Web diseñan y coordinan el desarrollo de sitios Web. Los sitios Web son una conjunción muy compleja de distintos sistemas integrados entre sí (Bases de datos, servidores, redes, componentes de backup y seguridad, etc...).

El resultado final será un sitio que pueda resolver las necesidades de negocios: Vender productos y servicios online y servir mejor a las necesidades de los clientes. Las aplicaciones deben ser diseñadas sobre de las necesidades de las personas que van a utilizarlas.

<http://www.pabloimpallari.com.ar/articulos/arquitecturaweb.php>

Para abrir una página Web en un navegador, normalmente se tecldea el correspondiente URL o se emplea el hiperenlace oportuno. Una vez que se solicita esta petición mediante el protocolo HTTP y la recibe el servidor Web, éste

localiza la página Web en su sistema de ficheros y la envía de vuelta al navegador que la solicitó, según se muestra en la en la siguiente figura.

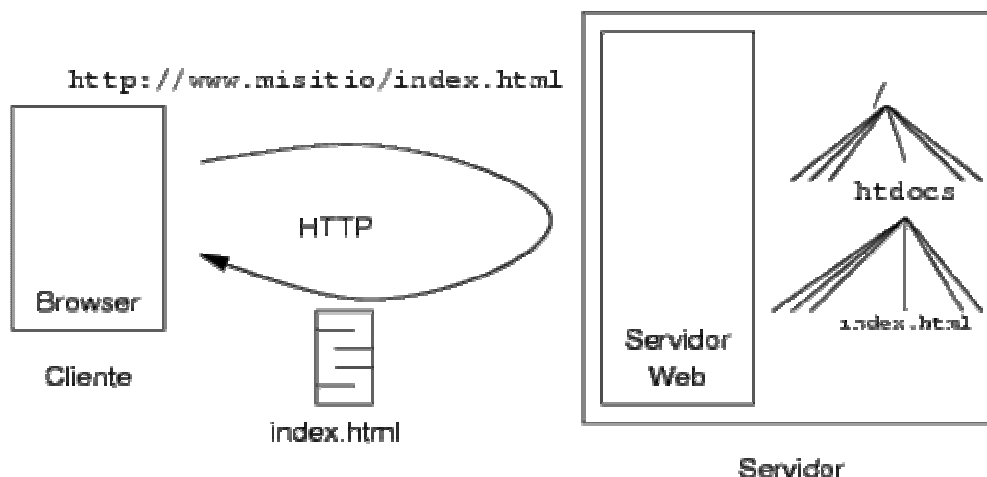


Figura: Arquitectura Web Básica

<http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node18.html>

2.2.3 ARQUITECTURA DE LA INFORMACIÓN COMO PROCESO

La Arquitectura de la Información es un proceso cíclico, transversal, que se da a lo largo del diseño del sitio y en cada una de sus fases, para asegurarse de que los objetivos de su producción y del desarrollo de la interfaz se cumplen de manera efectiva.

El fin de la Arquitectura de la Información no es la búsqueda de una metodología de diseño universal sino la articulación de un conjunto de técnicas para ayudar al desarrollo y producción de los sitios.

Con el fin de que la asimilación de contenidos por parte del usuario sea eficiente y efectiva (accesible y usable) la Arquitectura de la Información como proceso en general, se encarga, durante el proceso de desarrollo de los sitios, de definir:

- El objeto, propósito y fines del sistema de información o sitio
- La definición del público objetivo y los estudios de la audiencia.
- El diseño de la interacción
- El diseño de la navegación
- La planificación, gestión y desarrollo de contenidos
- La facilidad de búsqueda y el diseño de la interfaz de búsqueda.
- La usabilidad.
- El feedback del resultado y el rediseño de la interfaz.

2.2.4 ARQUITECTURA DE LA INFORMACIÓN COMO ABSTRACCIÓN

La Arquitectura de la Información no trata de un conjunto de pasos o guías predefinidas, sino la omnipresente inteligencia que está detrás y a cargo de que interactividad, navegación y contenido sinérgicamente permitan una integración sistémica con el cerebro del usuario y se produzca un fenómeno de persuasión, conocimiento o información simbiótica, que se traspa de un sistema a otro, según sean los objetivos del proyecto en general y los objetivos iniciales del usuario al interactuar con la interfaz.

De esta forma las acciones de buscabilidad, encontrabilidad y recuperabilidad de información se realizan en un contexto óptimo en ambos nodos de información (interfaz y usuario), hacen simbiosis e inician un proceso de comunicación que los enriquecerá de cualquier forma, la interfaz cumple con su objetivo y puede ser mejorada, y el usuario encuentra lo que busca y lo asimila.

2.2.5 EL ARQUITECTO DE INFORMACIÓN

Es la persona encargada de llevar a cabo y verificar el proceso de desarrollo del sitio trabajando estrechamente con los expertos en usabilidad para definir la interfaz de usuario. Está integrado en un equipo y se ocupa desde la fundamentación del proyecto hasta el rediseño del producto, verificando y testeando todas las fases hasta la obtención del resultado final.

Se trata de una nueva profesión surgida en un contexto social, cultural, económico y político fuertemente condicionado por las Nuevas Tecnologías de la Información, las cuales han modificado bruscamente en todos los niveles las formas de comunicación entre seres humanos, así como el medio en que perciben y asimilan información. Estos avances en telecomunicaciones, ciencia, y tecnología en general han producido una cantidad ingente de conocimiento, nuevos conceptos, nuevas ideas, nuevos métodos, nuevos procesos, nuevas visiones, nuevos problemas, nuevas soluciones de tal forma que es necesario hacer llegar al usuario común estos nuevos conceptos y soluciones de manera simple, objetiva y precisa. Es aquí, entre otras instancias, en que interviene la A.I.:

Procesar y dosificar la enorme cantidad de información que se ha producido a causa de los descubrimientos, nuevas investigaciones en todos y nuevos campos, a causa de la revolución de Internet y ponerla de una manera clara, relevante y significativa a disposición del usuario común, se trata, entre otras cosas, de hacer comprensible lo abstracto de alguna forma.

Desarrollar y verificar procesos de producción o diseño de información con el fin de que el usuario pueda hacer recuperación de esa información de manera clara, precisa y sin ambigüedades desde cualquier plataforma o soporte; en especial hablamos de soportes multimediales e interactivos, aunque retomando a Shedroff en la práctica no debemos omitir ningún soporte por plano que este sea y hablar de experiencias de usuario.

Organizar, estructurar, sistematizar (Tufte), rotular, distribuir, diseñar estructuralmente sistemas de información (Baeza, Rivera, Velasco, 2003) con el fin de que el usuario pueda hacer de su experiencia de recuperación algo simple, agradable, eficaz y productivo.

2.2.6 LA EXPERIENCIA DE USUARIO

Se entiende por Experiencia del Usuario el conjunto de factores y elementos que determinan una interacción satisfactoria del usuario con un entorno o dispositivo concretos. En ella influyen desde el diseño gráfico, la estética, hasta la articulación de la interfaz, la funcionalidad definida por el diseño de interacción y la extrapolación de principios del mundo del marketing, la psicología etc. Nathan Shedroff extiende el concepto de Experiencia del usuario más allá de la Web planteando su Teoría Unificada del Diseño que articula en torno a los conceptos clave de diseño de información e interacción.

http://es.wikipedia.org/wiki/Arquitectura_de_la_informaci%C3%B3n

2.2.6.1 Diseño de información

Hay tres áreas principales de donde provienen los practicantes del diseño de información: del diseño gráfico, de la documentación técnica (del inglés technical documentation) y de la informática. No es de extrañarse, pues el conjunto de

estas tres disciplinas representan en gran medida lo que hace un diseñador de información. Explican (documentación técnica), recolectan datos y los transforman en información digerible (informática) y presentan información visualmente (diseño gráfico).

http://es.wikipedia.org/wiki/Dise%C3%B1o_de_informaci%C3%B3n

2.2.6.1.1 Diseño gráfico.- El diseño gráfico es una forma de comunicación visual. Se ocupa de organizar imagen y texto para comunicar un mensaje. Puede aplicarse a muchos medios de comunicación, ya sean impresos, digitales, audiovisuales, u otros. El diseño gráfico publicitario es una rama perteneciente al diseño gráfico, pero con objetivos comerciales definidos.

http://es.wikipedia.org/wiki/Dise%C3%B1o_gr%C3%A1fico

2.2.6.1.2 Informática.- La Informática es la ciencia del tratamiento automático de la información a través de un computador (llamado también ordenador o computadora).

<http://es.wikipedia.org/wiki/Inform%C3%A1tica>

2.2.6.1.3 Documentación Técnica.- La documentación técnica es un aspecto sumamente importante, tanto en el desarrollo de la aplicación como en el mantenimiento de la misma. Mucha gente no hace este parte del desarrollo y no se da cuenta de que pierde la posibilidad de la reutilización de parte del programa en otras aplicaciones, sin necesidad de conocerse el código al dedillo. La documentación de un programa empieza a la vez que la construcción del mismo y finaliza justo antes de la entrega del programa o aplicación al cliente. Así mismo, la documentación que se entrega al cliente tendrá que coincidir con la versión final de los programas que componen la aplicación. Una vez concluido el programa, los documentos que se deben entregar son una guía técnica, una guía de uso y de instalación.

<http://www.desarrolloweb.com/articulos/importancia-documentacion.html>

2.2.6.2 Interacción

Es el proceso que establece un usuario con un dispositivo, sistema u objeto determinado. De Entre otros factores en el diseño de interacción intervienen disciplinas como la usabilidad o la ergonomía.

<http://es.wikipedia.org/wiki/Interacci%C3%B3n>

2.3 DISEÑO DE PAGINAS WEB

El diseño web es una actividad que consiste en la planificación, diseño e implementación de sitios web y páginas web. No es simplemente una aplicación del diseño convencional sobre internet ya que requiere tener en cuenta cuestiones tales como navegabilidad, interactividad, usabilidad, arquitectura de la información y la interacción de medios como el audio, texto, imagen y vídeo.

El diseño web no sólo aporta a la comunicación textual (contenidos) existente en internet una faceta visual, sino que obliga a pensar una mejor estructuración de los mismos en este soporte. La unión de un buen diseño con una jerarquía bien elaborada de contenidos aumenta la eficiencia de la web como canal de comunicación e intercambio de datos, que brinda posibilidades como el contacto directo entre el productor y el consumidor de contenidos, característica destacable del medio internet.

El diseño Web ha visto amplia aplicación en los sectores comerciales de internet especialmente en la World Wide Web. Asimismo, a menudo la Web se utiliza como medio de expresión plástica en sí. Artistas y creadores hacen de las páginas en Internet un medio más para ofrecer sus producciones y utilizarlas como un canal más de difusión de su obra.

2.3.1 DISEÑO WEB APLICADO

El diseño de páginas Web es una amplia área de aplicación del diseño gráfico en la cual se integran conocimientos propios del diseño como son la composición, el uso de color y la tipografía con conocimientos técnicos del medio como son los lenguajes HTML (HiperText Markup Language) y CSS (Cascading Style Sheets),

así como conocimientos sobre usabilidad, accesibilidad y organización de un sitio Web.

El diseño de páginas Web trata básicamente de realizar un documento con información hiperenlazado con otros documentos y asignarle una presentación para diferentes dispositivos de salida (en una pantalla de ordenador, en papel, en un teléfono móvil, etc).

Estos documentos o páginas Web pueden ser creadas:

- Directamente el código HTML con un procesador de texto plano, como el mismo Bloc de Notas de Windows.
- Utilizando un programa WYSIWYG de creación de páginas como Adobe GoLive, Macromedia Dreamweaver, NVU o Microsoft Frontpage.
- Utilizando lenguajes de programación del lado del servidor como ASP.NET, JSP, PHP, Perl, etc para generar la página web
- Usando un sistema CMS (Web Content Management System) que permite separar el contenido del diseño y que permite al diseñador generar solamente las plantillas, de modo que el contenido sea administrado en un área separada. Algunos sistemas CMS son el PhpNuke, Plone y Mambo.

2.3.2 ETAPAS

Para el diseño de páginas web debemos tener en cuenta dos etapas:

La primera es el diseño visual de la información que se desea editar. En esta etapa se trabaja en el papel distribuyendo el texto, los gráficos, los vínculos a otros documentos y otros objetos multimedia que se consideren pertinentes.

La segunda, una vez que se tiene este boceto se pasa a 'programar' la página web. Para esto, y fundamentalmente para manejar los vínculos entre documentos, se creó el lenguaje de marcación de hipertexto o HTML. Los enlaces que aparecen subrayados en este documento y otros de Wikipedia son ejemplos de hipertexto, puesto que al pulsar sobre ellos conducen a otras páginas con información relacionada.

El HTML consta de una serie de etiquetas que estructuran el texto y son presentados en forma de hipertexto por navegadores. Esto se puede hacer con un simple editor de textos (debe guardarse como texto plano, sin ningún tipo de formato). Aprender HTML es relativamente fácil, así que es sencillo crear páginas web de este modo. Esta era la única manera de generarlas hasta que aparecieron, a mediados de 1996, algunos editores visuales de HTML, como MS FrontPage y Macromedia Dreamweaver. Con estas herramientas no es absolutamente necesario aprender HTML (aunque sí aconsejable), con lo cual el desarrollador se concentra en lo más importante, el diseño del documento.

Todo esto teniendo en cuenta el nivel de programación en el diseño de las aplicaciones y del impacto visual que se quiere generar en el usuario.

2.4 APLICACIONES WEB Y EL DESARROLLO DE CAPAS

Al hablar del desarrollo de aplicaciones Web resulta adecuado presentarlas dentro de las aplicaciones multinivel. Los sistemas típicos cliente/servidor pertenecen a la categoría de las aplicaciones de dos niveles. La aplicación reside en el cliente mientras que la base de datos se encuentra en el servidor. En este tipo de aplicaciones el peso del cálculo recae en el cliente, mientras que el servidor hace la parte menos pesada, y eso que los clientes suelen ser máquinas menos potentes que los servidores.

<http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node21.html>

2.4.1 ARQUITECTURAS DE TRES NIVELES O PROGRAMACIÓN POR CAPAS

La programación por capas es un estilo de programación en la que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño, un ejemplo básico de esto es separar la capa de datos de la capa de presentación al usuario.

La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio sólo se ataca al nivel requerido sin tener que revisar entre código mezclado. Un buen ejemplo de este método de programación sería: Modelo de interconexión de sistemas abiertos

Además permite distribuir el trabajo de creación de una aplicación por niveles, de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, simplemente es necesario conocer la API que existe entre niveles.

En el diseño de sistemas informáticos actual se suele usar las arquitecturas multinivel o Programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

2.4.1.1 Capas o Niveles

El diseño más en boga actualmente es el diseño en tres niveles (o en tres capas).

2.4.1.1.1 Capa de presentación: es la que ve el usuario, presenta el sistema al usuario, le comunica la información y captura la información del usuario dando un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio.

2.4.1.1.2 Capa de negocio: es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) pues es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos de él.

2.4.1.1.3 Capa de datos: es donde residen los datos. Está formada por uno o mas gestor de bases de datos que realiza todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

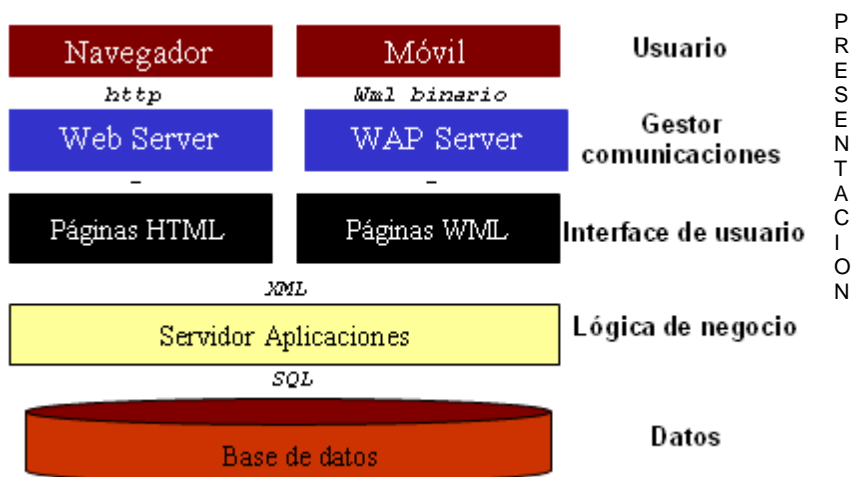
Todas estas capas pueden residir en un único ordenador (no sería lo normal), si bien lo más usual es que haya una multitud de ordenadores donde reside la capa

de presentación (son los clientes de la arquitectura cliente/servidor). Las capas de negocio y de datos pueden residir en el mismo ordenador, y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o mas ordenadores. Así, si el tamaño o complejidad de la base de datos aumenta, se puede separar en varios ordenadores los cuales recibirán las peticiones del ordenador en que resida la capa de negocio.

Si por el contrario fuese la complejidad en la capa de negocio lo que obligase a la separación, esta capa de negocio podría residir en uno o mas ordenadores que realizarían solicitudes a una única base de datos. En sistemas muy complejos se llega a tener una serie de ordenadores sobre los cuales corre la capa de datos, y otra serie de ordenadores sobre los cuales corre la base de datos. En una arquitectura de tres niveles, los términos "capas" y "niveles" no significan lo mismo ni son similares.

El término "capa" hace referencia a la forma como una solución es segmentada desde el punto de vista lógico:

Presentación/ Lógica de Negocio/ Datos.



En cambio, el término "nivel", corresponde a la forma en que las capas lógicas se encuentran distribuidas de forma física. Por ejemplo:

- Una solución de tres capas (presentación, lógica, datos) que residen en un solo ordenador (Presentación+lógica+datos). Se dice, que la arquitectura de la solución es de tres capas y un nivel.

- Una solución de tres capas (presentación, lógica, datos) que residen en dos ordenadores (presentación+lógica, lógica+datos). Se dice que la arquitectura de la solución es de tres capas y dos niveles.
- Una solución de tres capas (presentación, lógica, datos) que residen en tres ordenadores (presentación, lógica, datos). La arquitectura que la define es: solución de tres capas y tres niveles.

http://es.wikipedia.org/wiki/Arquitectura_de_tres_niveles

La capa intermedia es el código que el usuario invoca para recuperar los datos deseados. La capa de presentación recibe los datos y los formatea para mostrarlos adecuadamente. Esta división entre la capa de presentación y la de la lógica permite una gran flexibilidad a la hora de construir aplicaciones, ya que se pueden tener múltiples interfaces sin cambiar la lógica de la aplicación.

La tercera capa consiste en los datos que gestiona la aplicación. Estos datos pueden ser cualquier fuente de información como una base de datos o documentos XML.

Convertir un sistema de tres niveles a otro multinivel es fácil ya que consiste en extender la capa intermedia permitiendo que convivan múltiples aplicaciones en lugar de una sola.

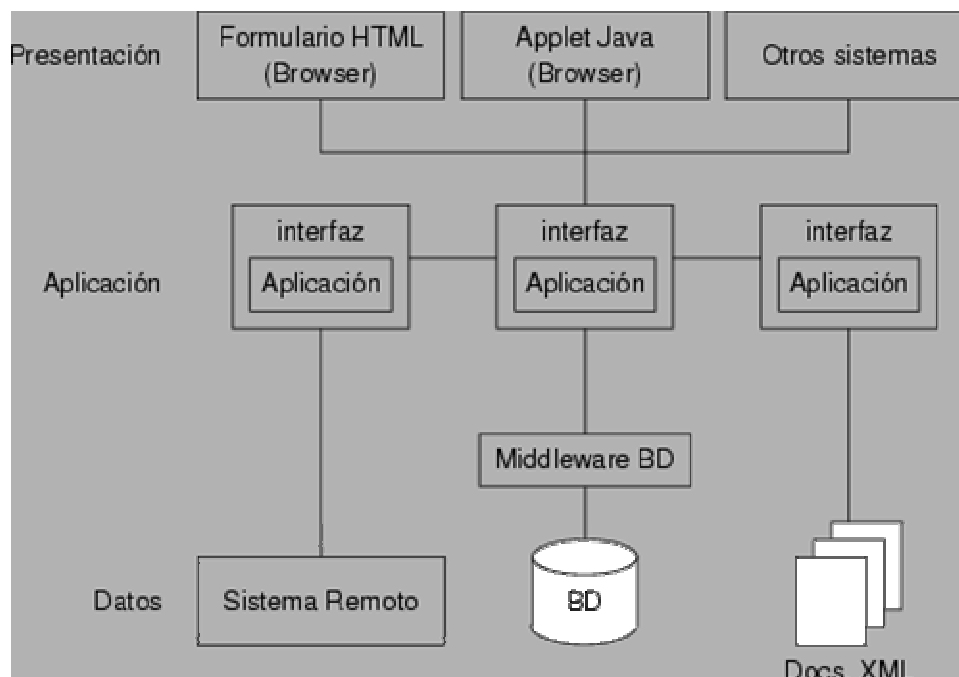


Figura : Arquitectura Multinivel.

La arquitectura de las aplicaciones Web suelen presentar un esquema de tres niveles. El primer nivel consiste en la capa de presentación que incluye no sólo el navegador, sino también el servidor web que es el responsable de dar a los datos un formato adecuado. El segundo nivel está referido habitualmente a algún tipo de programa o *script*. Finalmente, el tercer nivel proporciona al segundo los datos necesarios para su ejecución.

Una aplicación Web típica recogerá datos del usuario (primer nivel), los enviará al servidor, que ejecutará un programa (segundo y tercer nivel) y cuyo resultado será formateado y presentado al usuario en el navegador (primer nivel otra vez).

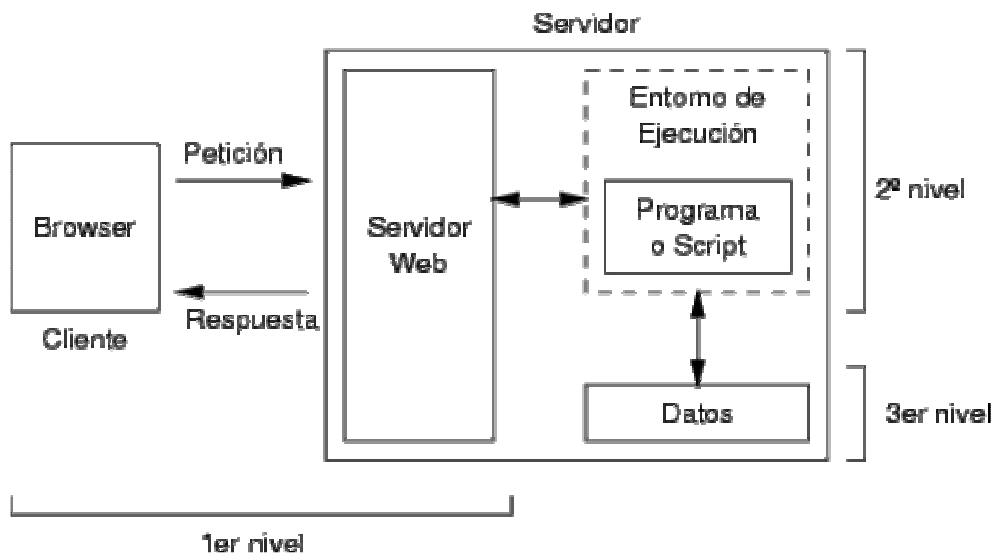
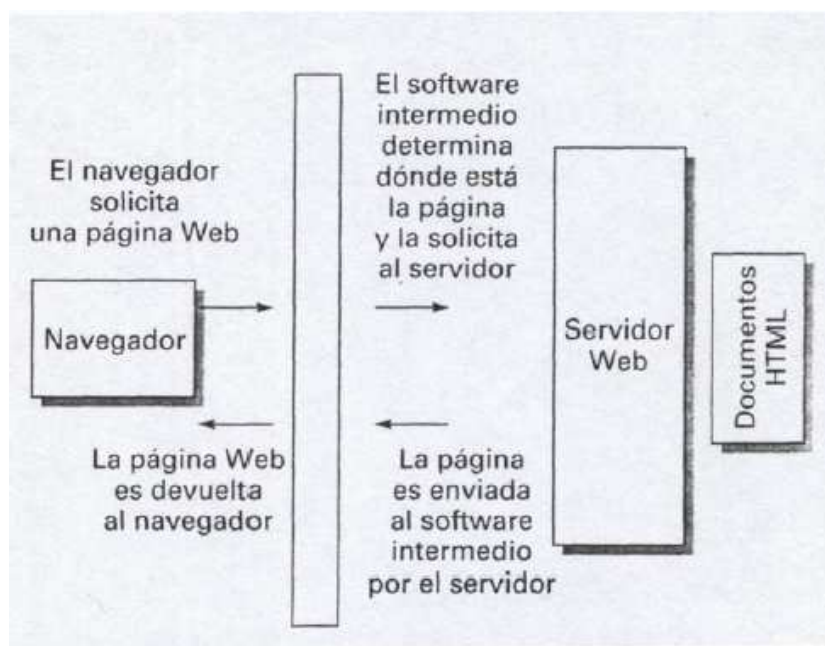


Figura : Arquitectura Web de tres niveles.

<http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node21.html>

2.4.2 SOFTWARE INTERMEDIO (Middleware)

Hasta el momento probablemente tengamos la impresión de que la comunicación entre cliente y servidor es directa. Desgraciadamente, esto no es verdad: normalmente existe por lo menos una capa de software entre ellos. Esta capa se llama software intermedio (*middleware*). Esta muestra la comunicación entre un cliente ejecutando un navegador como Internet Explorer y un servidor Web.



Fuente: <http://www.desarrolloweb.com/articulos/middleware>

Aquí el software intermedio que se encuentra entre el servidor Web y el cliente que ejecuta el navegador Web intercepta las peticiones que proceden del navegador. Si se hace una petición para una página Web entonces determina la localización del documento Web y envía una petición para esa página. El servidor responde a la petición y devuelve la página al software intermedio, quien la dirige al navegador que la visualizará en la pantalla del monitor que utiliza el cliente.

Existen dos categorías de software intermedio: el software intermedio general y el software intermedio de servicios.

El software intermedio general es el que está asociado a los servicios generales que requieren todos los clientes y servidores. El software típico que se utiliza como tal es:

- El software para llevar a cabo comunicaciones utilizando el protocolo TCP/IP y otros protocolos de red.
- El software del sistema operativo que por ejemplo, mantiene un almacenamiento distribuido de archivos.
- El software de autenticación, el cual comprueba que un usuario que desee utilizar un sistema distribuido pueda en efecto hacerlo.
- El software intermedio orientado a mensajes que gestiona el envío de mensajes desde clientes a servidores y viceversa.

El software intermedio de servicios es el software asociado a un servicio en particular. Entre los ejemplos típicos de este tipo de software se incluyen:

- Un software que permite a bases de datos diferentes conectarse a una red cliente/servidor.
- Un software específico de objetos distribuidos.
- Un software intermedio de red asociado al protocolo http
- Un software intermedio asociado a productos de seguridad específicos.

(Enciclopedia Wikipedia, 2005)

<http://www.monografias.com/trabajos16/sistemas-distribuidos/sistemas-distribuidos.shtml#MIDDLE>

2.5 HERRAMIENTAS DE CONSTRUCCION WEB

En el presente capítulo se describirá las características mas importantes de las software utilizado en el desarrollo del Sitio Web, dicho software proporcionan una combinación potente como herramientas visuales para el desarrollo de aplicaciones Web.

2.5.1 CAPA DE PRESENTACIÓN

2.5.1.1 JAVASCRIPT

JavaScript es un lenguaje interpretado orientado a las páginas web basado en el paradigma prototipo, con una sintaxis semejante a la del lenguaje Java.

El lenguaje fue inventado por Brendan Eich en la empresa Netscape Communications, que es la que fabricó los primeros navegadores de Internet comerciales. Apareció por primera vez en el producto de Netscape llamado Netscape Navigator 2.0.

Tradicionalmente, se venía utilizando en páginas web HTML, para realizar tareas y operaciones en el marco de la aplicación cliente servidor.

Los autores inicialmente lo llamaron Mocha y más tarde LiveScript pero fue rebautizado como JavaScript en un anuncio conjunto entre Sun Microsystems y Netscape, el 4 de diciembre de 1995.

En 1997 los autores propusieron JavaScript para que fuera adoptado como estándar de la the European Computer Manufacturers' Association ECMA, que a pesar de su nombre no es europeo sino internacional, con sede en Ginebra. En junio de 1997 fue adoptado como un estándar ECMA, con el nombre de ECMAScript. Poco después también lo fue como un estándar ISO.

JScript es la implementación de ECMAScript de Microsoft, muy similar al JavaScript de Netscape, pero con ciertas diferencias en el modelo de objetos del navegador que hacen a ambas versiones con frecuencia incompatibles.

Para evitar estas incompatibilidades, el World Wide Web Consortium diseñó el estándar Document Object Model (DOM, ó Modelo de Objetos del Documento en castellano), que incorporan Konqueror, las versiones 6 de Internet Explorer y Netscape Navigator, Opera versión 7, y Mozilla desde su primera versión.

<http://es.wikipedia.org/wiki/JavaScript>

2.5.1.2 HTML

El HTML, acrónimo inglés de HyperText Markup Language (lenguaje de marcas hipertextuales), es un lenguaje de marcación diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web. Gracias a Internet y a los navegadores del tipo Internet Explorer, Opera, Firefox o Netscape, el HTML se ha convertido en uno de los formatos más populares que existen para la construcción de documentos y también de los más fáciles de aprender.

HTML es una aplicación de SGML conforme al estándar internacional ISO 8879. XHTML es una reformulación de HTML 4 como aplicación XML 1.0, y que supone la base para la evolución estable de este lenguaje. Además XHTML permite la compatibilidad con los agentes de usuario que ya admitían HTML 4 siguiendo un conjunto de reglas.

2.5.1.2.1 Historia del estándar

En 1989 existían dos técnicas que permitían vincular documentos electrónicos, por un lado los hipervínculos (links) y por otro lado un poderoso lenguaje de etiquetas denominado SGML. Por entonces un usuario conocedor de ambas opciones, Tim Berners-Lee físico nuclear del Centro Europeo de Investigaciones Nucleares da a conocer a la prensa que estaba trabajando en un sistema que permitirá acceder a ficheros en línea, funcionando sobre redes de computadoras o máquinas electrónicas basadas en el protocolo TCP/IP.

Principios de 1990, Tim Berners-Lee define por fin el HTML como un subconjunto del conocido SGML y crea algo más valioso aun, el World Wide Web. En 1991, Tim Berners-Lee crea el primer navegador de HTML que funcionaría en modo texto y para UNIX.

Los trabajos para crear un sucesor del HTML, posteriormente llamado 'HTML+', comenzaron a finales de 1993. El HTML+ se diseñó originalmente para ser un superconjunto del HTML que permitiera evolucionar gradualmente desde el formato HTML anterior. A la primera especificación formal de HTML+ se le dio, por lo tanto, el número de versión 2.0 para distinguirla de esos "estándares no oficiales" previos. Los trabajos sobre HTML+ continuaron, pero nunca se convirtió en un estándar.

El borrador del estándar HTML 3.0 fue propuesto por el recién formado W3C en marzo de 1995. Con él se introdujeron muchas nuevas capacidades, tales como facilidades para crear tablas, hacer que el texto fluyese alrededor de las figuras y mostrar elementos matemáticos complejos. Aunque se diseñó para ser compatible con HTML 2.0, era demasiado complejo para ser implementado con la tecnología de la época y, cuando el borrador del estándar expiró en septiembre de 1995, se abandonó debido a la carencia de apoyos de los fabricantes de navegadores web. El HTML 3.1 nunca llegó a ser propuesto oficialmente, y el estándar siguiente fue el HTML 3.2, que abandonaba la mayoría de las nuevas características del HTML 3.0 y, a cambio, adoptaba muchos elementos desarrollados inicialmente por los navegadores web Netscape y Mosaic. La posibilidad de trabajar con fórmulas matemáticas que se había propuesto en el HTML 3.0 pasó a quedar integrada en un estándar distinto llamado MathML.

El HTML 4.0 también adoptó muchos elementos específicos desarrollados inicialmente para un navegador web concreto, pero al mismo tiempo comenzó a limpiar el HTML señalando algunos de ellos como 'desaprobados'.

Ya no va a haber nuevas versiones del HTML. Sin embargo, la herencia del HTML se mantiene en XHTML, que se basa en XML.

http://es.wikipedia.org/wiki/HTML#Historia_del_est.C3.A1ndar

2.5.2 CAPA DE NEGOCIO

2.5.2.1 PHP

PHP es un lenguaje de programación usado generalmente para la creación de contenido para sitios web. El nombre es el acrónimo recursivo de "PHP: Hypertext Preprocessor" (inicialmente PHP Tools, o, Personal Home Page Tools), y se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios web. Últimamente también para la creación de otro tipo de programas incluyendo aplicaciones con interfaz gráfica usando la librería GTK+.

2.5.2.1.1 Visión general

El fácil uso y la similitud con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores experimentados crear aplicaciones complejas con una curva de aprendizaje muy

suave. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones y prácticas.

Debido al diseño de PHP, también es posible crear aplicaciones con una interfaz gráfica para el usuario (también llamada GUI), utilizando la extensión PHP-GTK. También puede ser usado desde la línea de órdenes, de la misma manera como Perl o Python pueden hacerlo, esta versión de PHP se llama PHP CLI (Command Line Interface).

Su interpretación y ejecución se da en el servidor, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente. Además es posible utilizar PHP para generar archivos PDF, Flash, así como imágenes en diferentes formatos, entre otras cosas.

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de Aplicaciones web muy robustas.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos tales como UNIX (y de ese tipo, como Linux), Windows y Mac OS X, y puede interactuar con los servidores de web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

El modelo PHP puede ser visto como una alternativa al sistema de Microsoft que utiliza ASP.NET/C#/VB.NET, a ColdFusion de la compañía Macromedia, a JSP/Java de Sun Microsystems, y al famoso CGI/Perl. Aunque su creación y desarrollo se da en el ámbito de los sistemas libres, bajo la licencia GNU, existe además un compilador comercial llamado Zend Optimizer.

2.5.2.1.2 Historia

PHP fue originalmente diseñado en Perl, seguidos por 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 del

1995 fue publicado "Personal Home Page Tools" después de que Lerdorf lo combinara con su propio Form Interpreter para crear PHP/FI.

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 PHP 3, cambiando el nombre del lenguaje a la forma actual. Inmediatamente comenzaron experimentaciones públicas de PHP 3 y fue lanzado oficialmente en junio del 1998.

Para 1999, Suraski y Gutmans reescribieron el código de PHP, produciendo lo que hoy se conoce como Zend Engine o motor Zend. También conformaron Zend Technologies en Ratmat Gan, Israel. En mayo de 2000 PHP 4 fue lanzado bajo el poder del motor Zend Engine 1.0. El 13 de julio de 2004, fue lanzado PHP 5, utilizando el motor Zend Engine II (o Zend Engine 2). La versión más reciente de PHP es la 5.1.6, que incluye el novedoso PDO (Objetos de Datos de PHP o PHP Data Objects) y mejoras utilizando todas las ventajas que provee el nuevo Zend Engine 2.

2.5.2.1.3 Ventajas de PHP

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad.
- Leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial ([1]).
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos.
- Nos permite crear los formularios para la web

<http://es.wikipedia.org/wiki/PHP>

2.5.2.2 SERVIDOR WEB APACHE

Apache es uno de los servidores Web mas potentes del mercado, ofreciendo una perfecta combinación entre estabilidad y sencillez.

Las principales características de Apache son:

- Funcionalidad en múltiples plataformas (Unix / Linux), Windows y otras).
- Elaborado índice de directorios.
- Soporte del último protocolo http 1.1.
- Sencilla administración basada en la configuración de un único archivo.
- Soporte para CGI (Common Gateway Interface) y FastCGI

Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

En la actualidad (2006), Apache es el servidor HTTP más usado, siendo el servidor HTTP del 68% de los sitios web en el mundo y creciendo aún su cuota de mercado (estadísticas históricas y de uso diario proporcionadas por Netcraft.

http://es.wikipedia.org/wiki/Servidor_web_Apache

2.5.3 CAPA DE DATOS

2.5.3.1 MySQL

MySQL es uno de los Sistemas Gestores de bases de Datos (SQL) más populares desarrolladas bajo la filosofía de código abierto.

La desarrolla y mantiene la empresa MySQL AB pero puede utilizarse gratuitamente y su código fuente está disponible libremente.

Inicialmente, MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de ello, atrajo a los desarrolladores de páginas web con contenido dinámico, justamente por su simplicidad; aquellos elementos faltantes fueron llenados por la vía de las aplicaciones que la utilizan.

Poco a poco los elementos faltantes en MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre. Entre las características disponibles en las últimas versiones se puede destacar:

- Amplio subconjunto del lenguaje SQL.
- Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.

- Diferentes opciones de almacenamiento según si se desea velocidad en las operaciones o el mayor número de operaciones disponibles.
- Transacciones y claves foráneas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.

<http://es.wikipedia.org/wiki/MySQL>

2.5.4 MACROMEDIA FLASH MX 2004

Flash es la plataforma de software de predominancia indiscutible, usada por más de un millón de profesionales y con una presencia en más del 97% de los equipos de escritorio con conexión a Internet en todo el mundo, así como en una amplia gama de dispositivos.

Macromedia Flash MX 2004 le permite a los diseñadores y desarrolladores integrar video, texto, audio y gráficos en experiencias dinámicas que le permiten al cliente imbuirse en su vivencia y que producen resultados superiores para marketing y presentaciones interactivas, aprendizaje electrónico e interfaces de usuario de aplicaciones. Los Timeline Effects permiten a los diseñadores añadir transiciones comunes como difusiones y sombras sin necesidad de scripting. Comportamientos (“Behaviours”) pre-definidos, eliminan la necesidad de un scripting complejo de navegación. El compilador de alto desempeño mejora significativamente el nivel de reproducción del sonido, incluso con contenido creado para versiones anteriores de Flash Player. Mejoras en Flash Player 7 entregan un desempeño adicional que ofrecen la mejor experiencia para el usuario final. Flash MX 2004 ofrece integración única a Macromedia Dreamweaver, Fireworks y las otras herramientas de diseño

Flash MX 2004 también incluye Elementos MX para Flash, componentes de aplicación que facilitan a los desarrolladores el crear rápidamente grandes experiencias con una imagen visual consistente. (*Horizonteweb.com, 2005*)

El nuevo lenguaje ActionScript 2.0 permite a los desarrolladores añadir interactividad sofisticada a su contenido. Nuevo soporte para Hojas de Estilo en Cascada –CSS- permite a los usuarios entregar sitios híbridos que mezclan Flash y HTML con una imagen visual consistente.

Los Elementos MX para Flash incluyen plantillas para proyectos comunes como presentaciones y anuncios para la Web, así como componentes útiles para la interfase de usuario como los “data grid”, un “player” para multimedia y un “accordion pane”.

<http://www.horizonteweb.com/revision/flashMX.htm>

2.5.5 MACROMEDIA DREAMWEAVER MX 2004

Dreamweaver MX 2004 es la opción profesional para la creación de sitios y aplicaciones web. Proporciona una combinación potente de herramientas visuales de disposición, características de desarrollo de aplicaciones y soporte para la edición de código. Gracias a las robustas características para la integración y diseño basado en CSS, Dreamweaver permite que los diseñadores y desarrolladores web creen y manejen cualquier sitio web con toda facilidad.

Dreamweaver MX 2004 incluye potentes controles basados en normas para asegurar un diseño de alta calidad. Un entorno de diseño construido en torno a las hojas de estilo en cascada (CSS) hace posible un desarrollo más rápido y más eficiente de sitios profesionales creados con código limpio.

Dreamweaver MX 2004 es abierto y ampliable y cuenta con los productos y las tecnologías que usted usa, para darle la libertad y la flexibilidad de elegir la tecnología que más le conviene, ahora y en el futuro. Ya que permite desarrollar sitios web de HTML, XHTML, XML, ASP, ASP.NET, JSP, PHP y Macromedia ColdFusion. Además Dreamweaver MX 2004 ofrece un FTP seguro para codificar completamente todas las transferencias de archivos y evitar el acceso no autorizado a los datos, contenido de archivos, nombres de usuario y contraseñas.

<http://www.horizonteweb.com/revision/dreamweaverMX.htm>

2.5.6 MACROMEDIA FIREWORKS MX 2004

Fireworks MX 2004 tiene las herramientas que los profesionales del web necesitan para crear de todo, desde botones gráficos sencillos hasta sofisticados efectos de rollover. Permite Importar, editar e integrar fácilmente todos los principales formatos gráficos, incluidos imágenes vectoriales y de mapas de bits. Con Fireworks MX 2004 podemos exportar fácilmente imágenes de Fireworks a Flash, Dreamweaver y aplicaciones de terceros.

Fireworks MX 2004 permite crear rápidamente gráficos web de alta calidad e interactividad compleja. Fireworks ofrece todas las herramientas familiares que usted necesita: edición de fotos robusta, control de texto preciso y creación de imágenes profesional.

Fireworks proporciona un conjunto de herramientas completo y profesional para una producción rápida y actualizaciones fáciles con lo cual se reduce el tiempo necesario para el diseño y el desarrollo de imágenes, aun al trabajar con imágenes grandes.

<http://www.horizonteweb.com/revision/fireworks.htm>

CAPITULO 3

3. ASPECTOS METODOLOGICOS

3.1 PARADIGMA ESPIRAL ORIENTADO A LA WEB

A medida que la evolución de las WebApps (Sistemas se aplicación basadas en Web) pasa de utilizar recursos estáticos de información controlada por el contenido a utilizar entornos de aplicación dinámicos controlados por el usuario, cada ves es mas importante la necesidad de implementar una gestión sólida y unos principios de ingeniería. Para conseguir esto, es necesario desarrollar un marco de trabajo IWeb (Ingeniera Web) que acompañe a un modelo de proceso eficaz, popularizado por la actividades del marco de trabajo y por las tarea de Ingeniería.

El modelo en espiral trata de desarrollar incrementalmente el proyecto, dividiéndolo en muchos subproyectos. Uno de los puntos mas importantes del proceso es concentrarse primero en los aspectos mas críticos del proyecto. La idea es definir e implementar las características mas importantes primero, y con el conocimiento adquirido para hacerlo, volver hacia atrás y reimplementar las características siguientes en pequeños subproyectos.

El modelo en espiral orientado a la web se divide en un número de actividades estructurales, también llamadas regiones de tareas. Generalmente, existen entre tres y seis regiones de tareas.

3.1.1 COMUNICACIÓN CON EL CLIENTE O FORMULACION

Las tareas requeridas para establecer comunicación entre el desarrollador y el cliente. Actividad que identifica las metas y lo objetivos de la WebApp, y establece el ámbito del primer incremento.

3.1.2 PLANIFICACIÓN

La planificación estima el coste global del proyecto, evalúa los riesgos asociados con el esfuerzo del desarrollo, y define una planificación del desarrollo bien

granulada para el incremento final de la WebApp, con una aplicación mas toscamente granulada para los incrementos subsiguientes.. Son todos los requerimientos.

3.1.3 ANÁLISIS DE RIESGOS

Establece los requerimientos técnicos para la WebApp e identifica los elementos del contenido que se van a incorporar. También se define los requisitos de diseño gráfico (estética). Es decir se identifica los datos y requisitos funcionales y de comportamiento para la aplicación web.

3.1.4 INGENIERÍA

Aquí se incorpora dos tareas paralelas, el Diseño del Contenido y la Producción, son tareas llevadas a cabo por personas no técnicas del equipo IWeb. El objetivo de estas tareas es diseñar, producir y/o adquirir todo el contenido de texto, gráfico y video que se vayan a integrar en la WebApp. Al mismo tiempo se lleva a cabo un conjunto de tareas de diseño.

3.1.5 GENERACIÓN DE PÁGINAS

La generación de páginas es una actividad de construcción que hace mucho uso de las herramientas automatizadas para la creación de la WebApp. El contenido definido en la actividad de ingeniería se fusiona con los diseños arquitectónicos, de navegación y de la interfaz para la elaboración de páginas Web ejecutables en HTML, XML y otros lenguajes orientados a procesos (java). Durante esta actividad también se lleva a cabo la integración con el software intermedio (Middleware) de componentes (es decir: COBRA, DCOM o JavaBEan). Las pruebas ejercitan la navegación, intentan descubrir los errores de los applets, guiones y formularios, y ayuda a asegurar que la WeApp funcionará correctamente en diferentes entornos (por ejemplo, con diferentes navegadores)

3.1.6 EVALUACIÓN EL CLIENTE

En este punto es donde se solicitan cambios, tiene lugar la ampliación del ámbito. (Pressman, 2002)

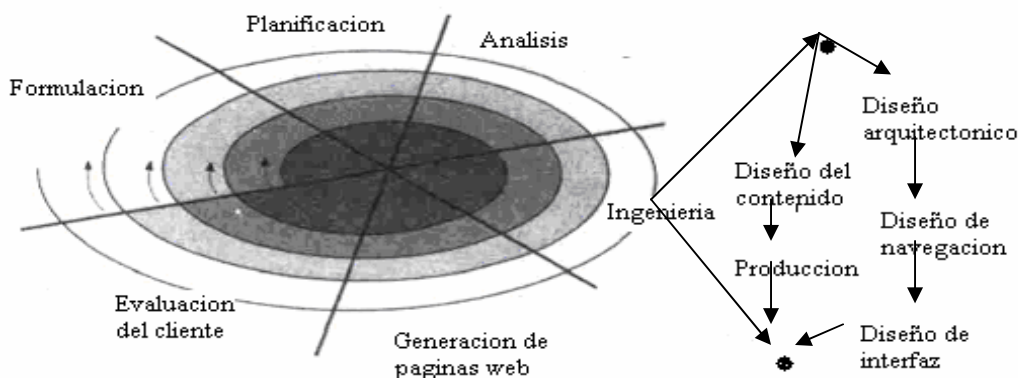


Figura : El modelo de proceso IWeb.

3.2 METODOLOGÍA

La Metodología es el establecimiento de teorías sobre el Método. Entonces la metodología es la descripción y el análisis de los métodos. Podríamos afirmar que la metodología es el estudio analítico y crítico de los métodos de investigación y de prueba, esto incluye:

La descripción, el análisis y la valoración crítica de los métodos que conciernen a la investigación.

Lo trascendente de la metodología es que le interesa más el proceso de investigación que los mismos resultados.

En la era espacial en la cual nos desarrollamos, el bombardeo permanente de información demanda de cada uno de los sujetos la búsqueda de diferentes procedimientos y mecanismos de acción, es decir, las herramientas más idóneas para aplicar esa información y transformarla en conocimiento. La información dista mucho del significado de conocimiento, a la primera se la puede conceptualizar como los datos que obtenemos por diversos medios y la segunda como la puesta en práctica de aquellos datos que han sido obtenidos.

Al hablar de metodología hacemos referencia a los diversos mecanismos de rigor lógico-científico, que ayudan a desarrollar el conocimiento, dentro de cada una de las disciplinas científicas. La metodología une y procesa todos los componentes de las áreas del saber, de igual forma, construye sistemas que permiten llegar a los objetivos, metas, desafíos y por ende los consabidos resultados. (C. Villalba, 2004)

3.3 METODOLOGÍA OOHDM

3.3.1 INTRODUCCIÓN A OOHDM

Las metodologías tradicionales de ingeniería de software, o las metodologías para sistemas de desarrollo de información, no contienen una buena abstracción capaz de facilitar la tarea de especificar aplicaciones hipermedia. El tamaño, la complejidad y el número de aplicaciones crecen en forma acelerada en la actualidad, por lo cual una metodología de diseño sistemática es necesaria para disminuir la complejidad y admitir evolución y usabilidad.

Producir aplicaciones en las cuales el usuario pueda aprovechar el potencial del paradigma de la navegación de sitios web, mientras ejecuta transacciones sobre bases de información, es una tarea muy difícil de lograr. En primer lugar, la navegación posee algunos problemas. Una estructura de navegación robusta es una de las claves del éxito en las aplicaciones hipermedia. Si el usuario entiende dónde puede ir y cómo llegar al lugar deseado, es una buena señal de que la aplicación ha sido bien diseñada.

Construir la interfaz de una aplicación web es también una tarea compleja; no sólo se necesita especificar cuáles son los objetos de la interfaz que deberían ser implementados, sino también la manera en la cual estos objetos interactuarán con el resto de la aplicación. En hipermedia existen requerimientos que deben ser satisfechos en un entorno de desarrollo unificado.

Por un lado, la navegación y el comportamiento funcional de la aplicación deberían ser integrados. Por otro lado, durante el proceso de diseño se debería poder desacoplar las decisiones de diseño relacionadas con la estructura navegacional de la aplicación, de aquellas relacionadas con el modelo del dominio.

http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art5_c.pdf

3.3.2 FASES DE LA METODOLOGÍA OOHDM

La metodología aplicada para el desarrollo del proyecto y propuesta, es la OOHDM (Object – Oriented Hypermedia Design Method, Método de Diseño

Hipermedia Orientado a Objeto). La misma fue seleccionada ya que satisface los requerimientos de la aplicación, especialmente en lo referente al desarrollo Web.

- Captura de Requerimientos.- Uno de los procesos esenciales para el desarrollo de un software de calidad, es la correcta captura de requerimientos por parte de los diseñadores del sistema a desarrollar.
- Diseño Conceptual.- El resultado de ésta fase, son los requerimientos expresados en los diagramas de clases Graig, L.(1998).
- Diseño Navegacional.- En esta fase, se hace un bosquejo de la conexión entre las páginas de la aplicación Web a desarrollar.
- Diseño Abstracto de Interfaz.- El Diseño de interfaz es una actividad crítica en las aplicaciones interactivas, incluyendo la hipermedia. Aunque los objetos han sido usados por años en el campo del diseño de interfaz de usuario, el enfoque ha sido principalmente aplicado al software elaborado y no para la especificación de contenidos.
- Implementación.- En esta fase se implementa el software hipermedia. Los modelos construidos independientemente de la plataforma de implementación son traducidos a un ambiente de ejecución
- Pruebas.- En esta fase se llevan a cabo las pruebas a la aplicación antes de colocarla en producción.

<http://edutec2004.lmi.ub.es/pdf/157.pdf>

3.3.3 CAPTURA DE REQUERIMIENTOS

En el desarrollo de software de calidad, es esencial la captura correcta de requerimientos por parte de los diseñadores del sistema a desarrollar. es necesario incorporar esta fase al OOHDM. Aquí, es primordial identificar a los actores (stakeholders) y las tareas que ellos van a ejecutar. Luego, los escenarios son recolectados para cada tarea y tipo de actor, produciendo los Casos de Uso, los cuales son una representación gráfica concisa de la interacción entre el usuario y el sistema durante la ejecución de una tarea.

3.3.4 DISEÑO CONCEPTUAL

Durante esta actividad se construye un esquema conceptual representado por los objetos del dominio, las relaciones y colaboraciones existentes establecidas entre ellos. En las aplicaciones hipermedia convencionales, cuyos componentes de hipermedia no son modificados durante la ejecución, se podría usar un modelo de datos semántico estructural (como el modelo de entidades y relaciones). De este modo, en los casos en que la información base pueda cambiar dinámicamente o se intenten ejecutar cálculos complejos, se necesitará enriquecer el comportamiento del modelo de objetos.

En OOHD, el esquema conceptual está construido por clases, relaciones y subsistemas. Las clases son descritas como en los modelos orientados a objetos tradicionales. Sin embargo, los atributos pueden ser de múltiples tipos para representar perspectivas diferentes de las mismas entidades del mundo real.

Se usa notación similar a UML (Lenguaje de Modelado Unificado).

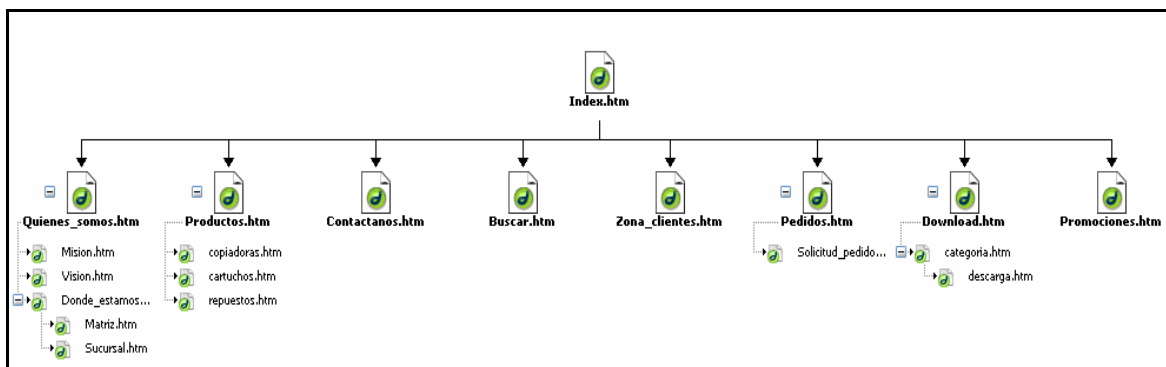
<http://edutec2004.lmi.ub.es/pdf/157.pdf>

3.3.5 DISEÑO NAVEGACIONAL

La primera generación de aplicaciones web fue pensada para realizar navegación a través del espacio de información, utilizando un simple modelo de datos de hipermedia. En OOHD, la navegación es considerada un paso crítico en el diseño aplicaciones. Un modelo navegacional es construido como una vista sobre un diseño conceptual, admitiendo la construcción de modelos diferentes de acuerdo con los diferentes perfiles de usuarios. Cada modelo navegacional provee una vista subjetiva del diseño conceptual.

El diseño de navegación es expresado en dos esquemas: el esquema de clases navegacionales y el esquema de contextos navegacionales. En OOHD existe un conjunto de tipos predefinidos de clases navegacionales:

nodos, enlaces y estructuras de acceso. La semántica de los nodos y los enlaces son las tradicionales de las aplicaciones hipermedia, y las estructuras de acceso, tales como índices o recorridos guiados, representan los posibles caminos de acceso a los nodos.

GRÁFICO 3.4.

La principal estructura primitiva del espacio navegacional es la noción de contexto navegacional. Un contexto navegacional es un conjunto de nodos, enlaces, clases de contextos, y otros contextos navegacionales (contextos anidados). Pueden ser definidos por comprensión o extensión, o por enumeración de sus miembros.

Los contextos navegacionales juegan un rol similar a las colecciones y fueron inspirados sobre el concepto de contextos anidados. Organizan el espacio navegacional en conjuntos convenientes que pueden ser recorridos en un orden particular y que deberían ser definidos como caminos para ayudar al usuario a lograr la tarea deseada.

Los nodos son enriquecidos con un conjunto de clases especiales que permiten de un nodo observar y presentar atributos (incluidos las anclas), así como métodos (comportamiento) cuando se navega en un particular contexto.

http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art5_c.pdf

3.3.6 DISEÑO ABSTRACTO DE INTERFAZ

Una vez que las estructuras navegacionales son definidas, se deben especificar los aspectos de interfaz. Esto significa definir la forma en la cual los objetos navegacionales pueden aparecer, cómo los objetos de interfaz activarán la navegación y el resto de la funcionalidad de la aplicación, qué transformaciones de la interfaz son pertinentes y cuándo es necesario realizarlas.

Una clara separación entre diseño navegacional y diseño de interfaz abstracta permite construir diferentes interfaces para el mismo modelo navegacional, dejando un alto grado de independencia de la tecnología de interfaz de usuario.

El aspecto de la interfaz de usuario de aplicaciones interactivas (en particular las aplicaciones web) es un punto crítico en el desarrollo que las modernas metodologías tienden a descuidar. En OOHDM se utiliza el diseño de interfaz abstracta para describir la interfaz del usuario de la aplicación de hipermedia.

El modelo de interfaz ADVs (Vista de Datos Abstracta) especifica la organización y comportamiento de la interfaz, pero la apariencia física real o de los atributos, y la disposición de las propiedades de las ADVs en la pantalla real son hechas en la fase de implementación.

http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art5_c.pdf

3.3.7 IMPLEMENTACION

En esta fase, el diseñador debe implementar el diseño. Hasta ahora, todos los modelos fueron construidos en forma independiente de la plataforma de implementación; en esta fase es tenido en cuenta el entorno particular en el cual se va a correr la aplicación.

Al llegar a esta fase, el primer paso que debe realizar el diseñador es definir los ítems de información que son parte del dominio del problema. Debe identificar también, cómo son organizados los ítems de acuerdo con el perfil del usuario y su tarea; decidir qué interfaz debería ver y cómo debería comportarse. A fin de implementar todo en un entorno web, el diseñador debe decidir además qué información debe ser almacenada.

http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r22_art5_c.pdf

3.3.8 PRUEBAS

En este caso a medida que se codifiquen las tareas, se van a ir aplicando las pruebas unitarias a cada una de ellas, con el fin de ir constatando el buen funcionamiento de los módulos, para luego poder ensamblar el sistema y así realizar las pruebas definitivas con el cliente.

3.4 UML (Unified Modeling Language)

El Lenguaje de Modelado Unificado (UML) es la sucesión de una serie de métodos de análisis y diseño orientadas a objetos que aparecen a fines de los 80's y principios de los 90s. Directamente unifica los métodos de Booch, Rumbaugh (OMT), y Jacobson, y algo más.

UML es llamado un lenguaje de modelado, no un método. Los métodos consisten de ambos de un lenguaje de modelado y de un proceso.

<http://agamenon.uniandes.edu.co/~pfigueroa/soo/uml>

UML Es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Se usa para entender, diseñar, configurar, mantener y controlar la información sobre los sistemas a construir.

UML capta la información sobre la estructura estática y el comportamiento dinámico de un sistema. Un sistema se modela como una colección de objetos discretos que interactúan para realizar un trabajo que finalmente beneficia a un usuario externo.

El lenguaje de modelado pretende unificar la experiencia pasada sobre técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar.

UML no es un lenguaje de programación. Las herramientas pueden ofrecer generadores de código de UML para una gran variedad de lenguaje de programación, así como construir modelos por ingeniería inversa a partir de programas existentes.

Es un lenguaje de propósito general para el modelado orientado a objetos. UML es también un lenguaje de modelamiento visual que permite una abstracción del sistema y sus componentes.

Existían diversos métodos y técnicas Orientadas a Objetos, con muchos aspectos en común pero utilizando distintas notaciones, se presentaban inconvenientes para el aprendizaje, aplicación, construcción y uso de herramientas, etc., además de pugnas entre enfoques, lo que genero la creacion del UML como estándar para

el modelamiento de sistemas de software principalmente, pero con posibilidades de ser aplicado a todo tipo de proyectos.

<http://www.creangel.com/uml/intro.php>

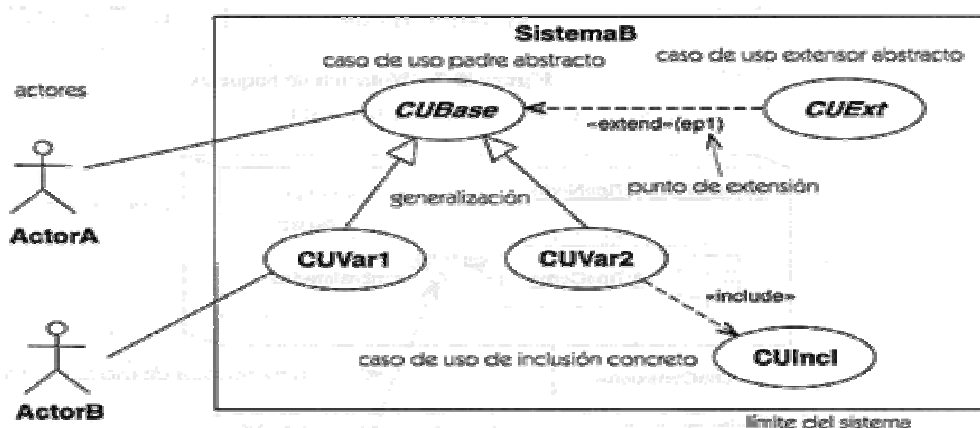
3.4.1 DIAGRAMAS PARA MODELAMIENTO

3.4.1.1 Diagramas de Caso de Uso

Jacobsson en define a los casos de uso y a los actores como:

“Los Actores representan lo que interactúa con el sistema. Ellos representan a todo lo que necesita intercambiar información con el sistema. Las instancias de los actores son los usuarios del sistema, ellos llevan a cabo un número de operaciones con el sistema y desarrollan una secuencia de transacciones en comunicación con el sistema. A esta secuencia de acciones se llama Caso de uso.”

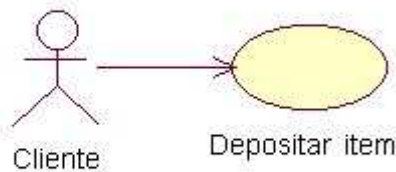
Los casos de uso se usan para especificar el comportamiento del sistema sin definir su estructura, la forma de que un modelo de Caso de uso es realizado en términos de objetos que son definidos por clases dentro del sistema se puede describir con diagramas de colaboración.



<http://www.creangel.com/uml/casouso.php#>

3.4.1.1.1 Notación del Diagrama de Caso de uso.- Un diagrama de caso de uso es una gráfica de actores, un conjunto de casos de uso, posiblemente algunas interfaces y las relaciones entre estos elementos. Las relaciones son asociaciones

entre los actores y los casos de uso generalizaciones entre los actores y generalizaciones, extensiones e inclusiones entre los casos de uso.



3.4.1.1.2 *Relaciones de los Casos de uso.*- Hay varias relaciones estándar entre los casos de uso o entre los actores y los casos de uso.

- Asociación (____). La participación de un actor en el Caso de uso, i. e. instancias de el actor e instancias de el caso de uso se comunican entre sí. **Esta es la única relación entre los actores y los Casos de uso.**
- Extensión (__e__). Una relación de extensión entre el caso de uso A y el caso de uso B indica que una instancia de el caso de uso B puede ser aumentada (Dependiendo de ciertas condiciones especificadas en la extensión) por el comportamiento de especificado en el Caso de uso B. El comportamiento es insertado el punto definido como punto de extensión del Caso de uso B, el cual es referenciado por la relación externa.
- Generalización (←—) (Una generalización de un caso de uso A hacia el caso de uso B indica que A es una especialización de B
- Inclusión (__i__). Una relación de inclusión del caso de uso A hacia el Caso de uso B indica que una instancia del caso de uso A también contendrá el comportamiento especificado por B. El comportamiento es incluido en el punto definido en A.

<http://www.geocities.com/txmetsb/#Que es un Use Case>

3.4.1.2 Diagrama de Clases

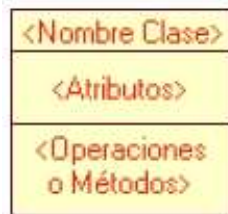
Un diagrama de clases sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de contenido.

Un diagrama de clases esta compuesto por los siguientes elementos:

- Clase: atributos, métodos y visibilidad.
- Relaciones: Herencia, Composición, Agregación, Asociación y Uso.

3.4.1.2.1 *Clase.*- Es la unidad básica que encapsula toda la información de un Objeto (un objeto es una instancia de una clase). A través de ella podemos modelar el entorno en estudio (una Casa, un Auto, una Cuenta Corriente, etc.).

En UML, una clase es representada por un rectángulo que posee tres divisiones:



En donde:

Superior: Contiene el nombre de la Clase

Intermedio: Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser private, protected o public).

Inferior: Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: private, protected o public).

Ejemplo:

Una Cuenta Corriente que posee como característica:

Balance

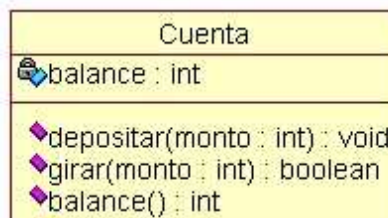
Puede realizar las operaciones de:

Depositar


Girar


y Balance


El diseño asociado es:




3.4.1.2.2 *Atributos:* Los atributos o características de una Clase pueden ser de tres tipos, los que definen el grado de comunicación y visibilidad de ellos con el entorno, estos son:


Public (+, 


Private (-, 

Protected (#, 

3.4.1.2.3 *Métodos:* Los métodos u operaciones de una clase son la forma en como ésta interactúa con su entorno, éstos pueden tener las características:

Public (+, 

Private (-, 

Protected (#, 

<http://www.clikear.com/manuales/uml/diagramasestructuraestatica.asp>

3.4.1.2.4 *Relaciones entre Clases:* Ahora ya definido el concepto de Clase, es necesario explicar como se pueden interrelacionar dos o más clases (cada uno con características y objetivos diferentes).

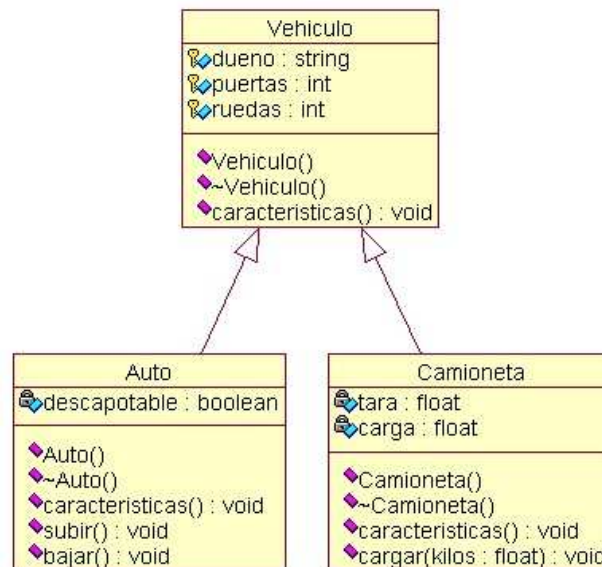
Antes es necesario explicar el concepto de cardinalidad de relaciones:

En UML, la cardinalidad de las relaciones indica el grado y nivel de dependencia, se anotan en cada extremo de la relación y éstas pueden ser:

- **uno o muchos:** 1..* (1..n)
- **0 o muchos:** 0..* (0..n)
- **número fijo:** m (m denota el número).

Herencia (Especialización/Generalización):

Indica que una subclase hereda los métodos y atributos especificados por una Super Clase, por ende la Subclase además de poseer sus propios métodos y atributos, poseerá las características y atributos visibles de la Super Clase (public y protected), ejemplo:



En la figura se especifica que Auto y Camión heredan de Vehículo, es decir, Auto posee las Características de Vehículo (Precio, VelMax, etc) además posee algo particular que es Descapotable, en cambio Camión también hereda las características de Vehículo (Precio, VelMax, etc) pero posee como particularidad propia Acoplado, Tara y Carga.

Cabe destacar que fuera de este entorno, lo único "visible" es el método Características aplicable a instancias de Vehículo, Auto y Camión, pues tiene definición pública, en cambio atributos como Descapotable no son visibles por ser privados.

Agregación:

Para modelar objetos complejos, no bastan los tipos de datos básicos que proveen los lenguajes: enteros, reales y secuencias de caracteres. Cuando se requiere componer objetos que son instancias de clases definidas por el desarrollador de la aplicación, tenemos dos posibilidades:

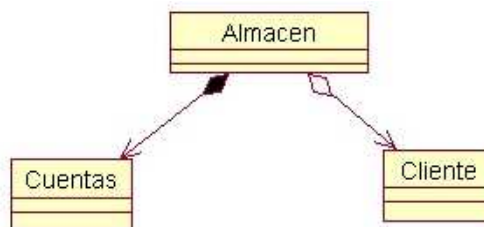
Por Valor: Es un tipo de relación estática, en donde el tiempo de vida del objeto incluido esta condicionado por el tiempo de vida del que lo incluye. Este tipo de relación es comúnmente llamada:

Composición (el Objeto base se construye a partir del objeto incluido, es decir, es "parte/todo").

Por Referencia: Es un tipo de relación dinámica, en donde el tiempo de vida del objeto incluido es independiente del que lo incluye. Este tipo de relación es comúnmente llamada

Agregación (el objeto base utiliza al incluido para su funcionamiento).

Un Ejemplo es el siguiente:



En donde se destaca que:

Un almacén posee Clientes y Cuentas (los rombos van en el objeto que posee las referencias).

Cuando se destruye el Objeto almacén también son destruidos los objetos Cuenta asociados, en cambio no son afectados los objetos Cliente asociados.

La composición (por Valor) se destaca por un rombo relleno.

La agregación (por Referencia) se destaca por un rombo transparente.

La flecha en este tipo de relación indica la navegabilidad del objeto referenciado.

Cuando no existe este tipo de particularidad la flecha se elimina.

Asociación: 

La relación entre clases conocida como Asociación, permite asociar objetos que colaboran entre si. Cabe destacar que no es una relación fuerte, es decir, el tiempo de vida de un objeto no depende del otro.

Ejemplo:



Un cliente puede tener asociadas muchas Ordenes de Compra, en cambio una orden de compra solo puede tener asociado un cliente.

Dependencia o Instanciación (uso):

Representa un tipo de relación muy particular, en la que una clase es instanciada (su instanciación es dependiente de otro objeto/clase). Se denota por una flecha punteada.

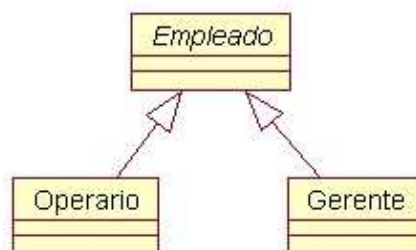
El uso más particular de este tipo de relación es para denotar la dependencia que tiene una clase de otra, como por ejemplo una aplicación grafica que instancia una ventana (la creación del Objeto Ventana esta condicionado a la instanciación proveniente desde el objeto Aplicación):



Cabe destacar que el objeto creado (en este caso la Ventana gráfica) no se almacena dentro del objeto que lo crea (en este caso la Aplicación).

3.4.1.2.5 Casos Particulares:

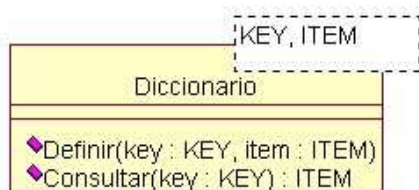
Clase Abstracta:



Una clase abstracta se denota con el nombre de la clase y de los métodos con letra "itálica". Esto indica que la clase definida no puede ser instanciada pues posee métodos abstractos (aún no han sido definidos, es decir, sin

implementación). La única forma de utilizarla es definiendo subclases, que implementan los métodos abstractos definidos.

Clase parametrizada:



Una clase parametrizada se denota con un subcuadro en el extremo superior de la clase, en donde se especifican los parámetros que deben ser pasados a la clase para que esta pueda ser instanciada. El ejemplo más típico es el caso de un Diccionario en donde una llave o palabra tiene asociado un significado, pero en este caso las llaves y elementos pueden ser genéricos.

<http://www.dcc.uchile.cl/~psalinas/uml/modelo.html>

3.4.1.3 Diagramas de Interacción

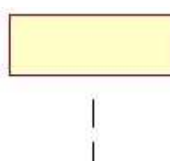
Los diagramas de interacción, representan la forma en como un Cliente (Actor) u Objetos (Clases) se comunican entre si en petición a un evento. Esto implica recorrer toda la secuencia de llamadas, de donde se obtienen las responsabilidades claramente.

Dichos diagramas pueden ser obtenidos de dos partes, desde el Diagrama Estático de Clases o el de Casos de Uso (son diferentes).

Los componentes de un diagrama de interacción son:

- Un Objeto o Actor.
- Mensaje de un objeto a otro objeto.
- Mensaje de un objeto a si mismo.

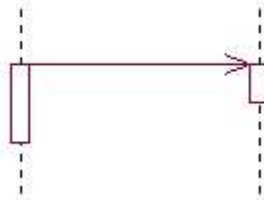
Objeto/Actor:



El rectángulo representa una instancia de un Objeto en particular, y la línea punteada representa las llamadas a métodos del objeto.

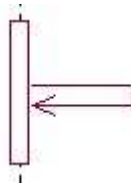
<http://www.dcc.uchile.cl/~psalinas/uml/interaccion.html>

Mensaje a Otro Objeto:



Se representa por una flecha entre un objeto y otro, representa la llamada de un método (operación) de un objeto en particular.

Mensaje al Mismo Objeto:



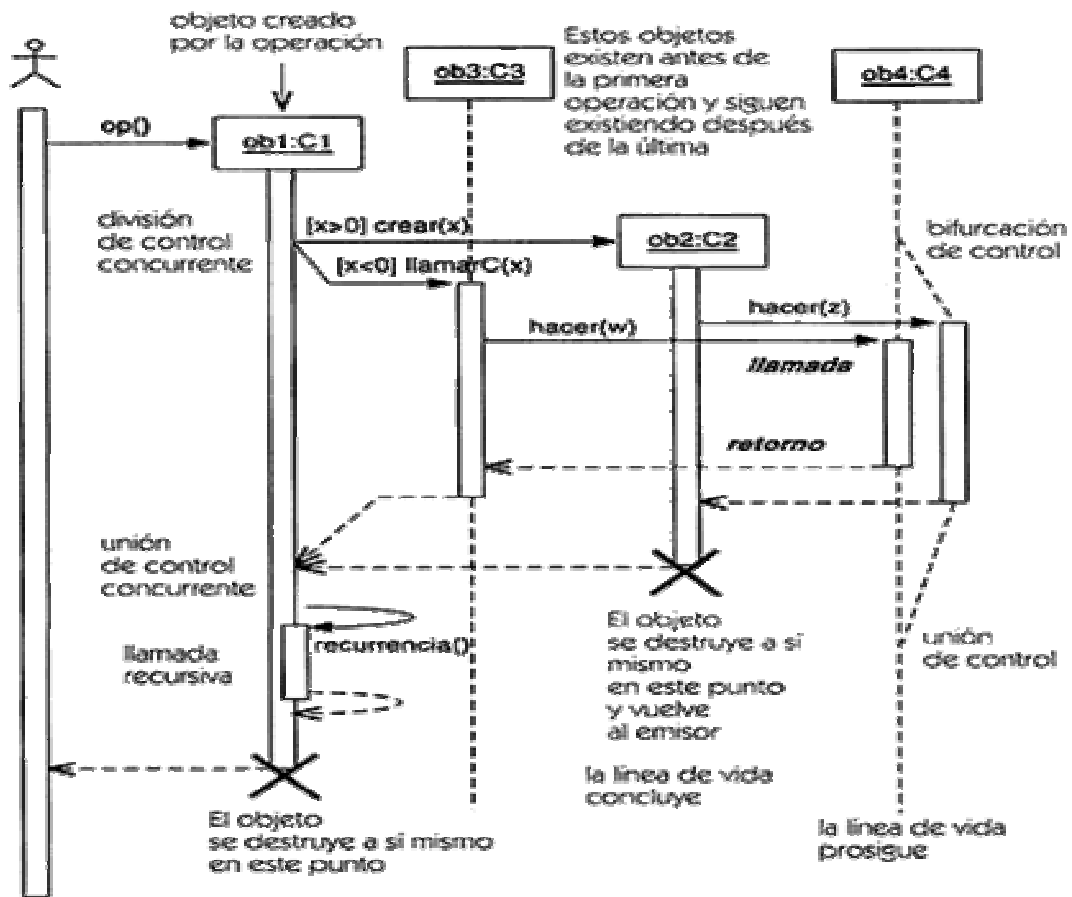
No solo llamadas a métodos de objetos externos pueden realizarse, también es posible visualizar llamadas a métodos desde el mismo objeto en estudio.

Hay dos tipos de diagrama de interacción, ambos basados en la misma información, pero cada uno enfatizando un aspecto particular: Diagramas de Secuencia y Diagramas de Colaboración.

3.4.1.3.1 Diagrama de Secuencia.- Un diagrama de Secuencia muestra una interacción ordenada según la secuencia temporal de eventos. En particular, muestra los objetos participantes en la interacción y los mensajes que intercambian ordenados según su secuencia en el tiempo. El eje vertical representa el tiempo, y en el eje horizontal se colocan los objetos y actores participantes en la interacción, sin un orden prefijado. Cada objeto o actor tiene una línea vertical, y los mensajes se representan mediante flechas entre los distintos objetos. El tiempo fluye de arriba abajo. Se pueden colocar etiquetas (como restricciones de tiempo, descripciones de acciones, etc.) bien en el margen izquierdo o bien junto a las transiciones o activaciones a las que se refieren.

<http://www.clikear.com/manuales/uml/diagramainteraccion.asp>

El diagrama de Secuencia Muestra las interacciones entre los objetos organizadas en una secuencia temporal. En particular muestra los objetos participantes en la interacción y la secuencia de mensajes intercambiados.

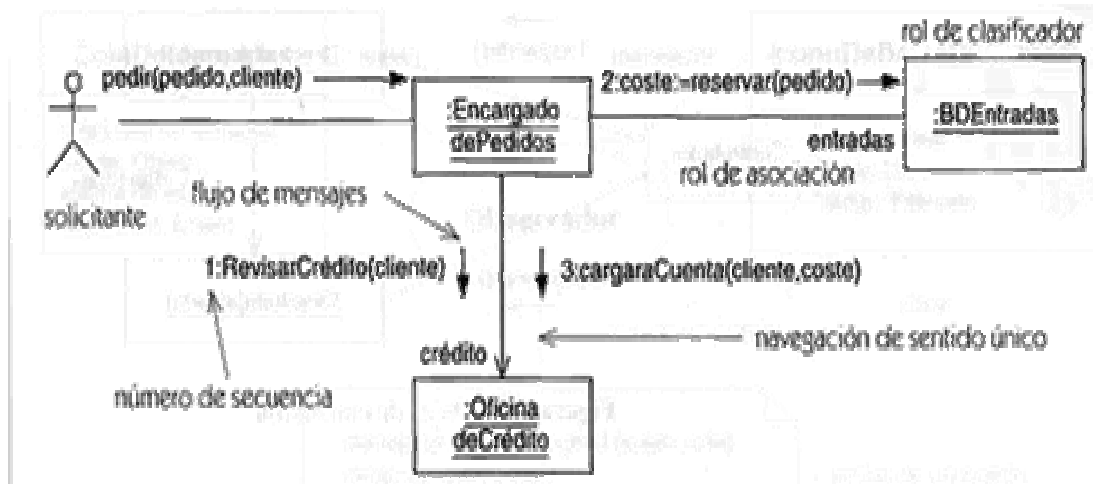


<http://www.creangel.com/uml/secuencia.php#>

3.4.1.3.1 *Diagrama de Colaboración.*- Un Diagrama de Colaboración muestra una interacción organizada basándose en los objetos que toman parte en la interacción y los enlaces entre los mismos (en cuanto a la interacción se refiere). A diferencia de los Diagramas de Secuencia, los Diagramas de Colaboración muestran las relaciones entre los roles de los objetos. La secuencia de los mensajes y los flujos de ejecución concurrentes deben determinarse explícitamente mediante números de secuencia.

<http://www.dcc.uchile.cl/~psalinas/uml/interaccion.html>

Un Diagrama de Colaboración muestra interacciones organizadas alrededor de los roles. A diferencia de los diagramas de secuencia, los diagramas de colaboración muestran explícitamente las relaciones de los roles.



<http://www.creangel.com/uml/colaboracion.php#>

3.4.1.4 Diagrama de Estados

Un Diagrama de Estados muestra la secuencia de estados por los que pasa bien un caso de uso, bien un objeto a lo largo de su vida, o bien todo el sistema. En él se indican qué eventos hacen que se pase de un estado a otro y cuáles son las respuestas y acciones que genera.

En cuanto a la representación, un diagrama de estados es un grafo cuyos nodos son estados y cuyos arcos dirigidos son transiciones etiquetadas con los nombres de los eventos. Un estado se representa como una caja redondeada con el nombre del estado en su interior. Una transición se representa como una flecha desde el estado origen al estado destino.

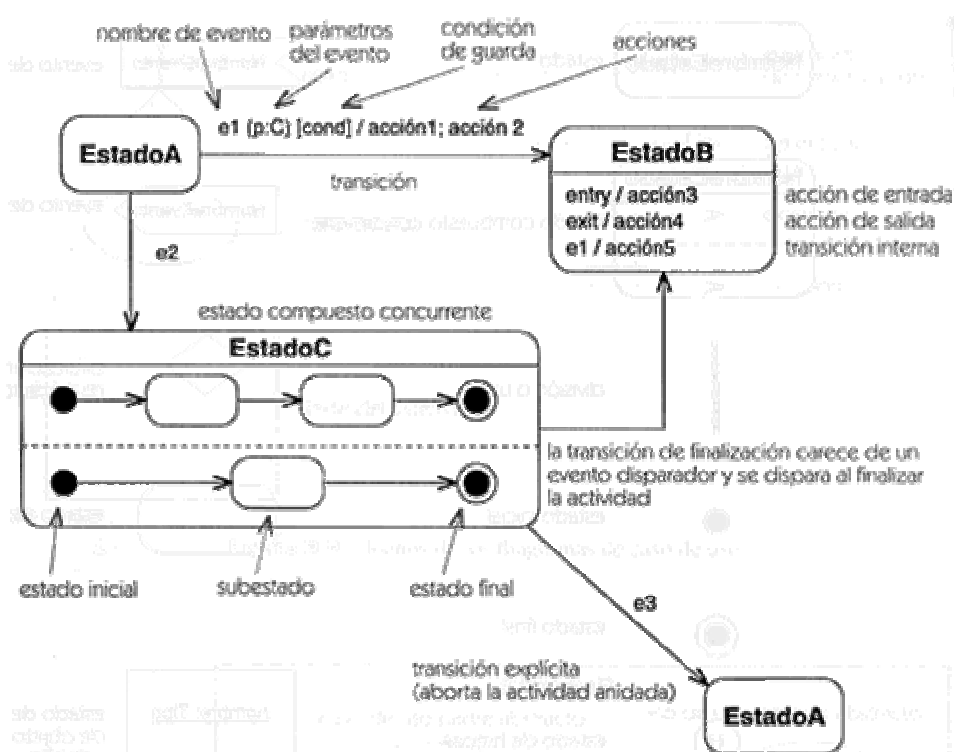
La caja de un estado puede tener 1 o 2 compartimentos.

- En el primer compartimento aparece el nombre del estado.
- El segundo compartimento es opcional, y en él pueden aparecer acciones de entrada, de salida y acciones internas.

Una acción de entrada aparece en la forma entrada/acción_asociada donde acción_asociada es el nombre de la acción que se realiza al entrar en ese estado. Cada vez que se entra al estado por medio de una transición la acción de entrada se ejecuta.

<http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/multiple-html/x98.html>

Una acción de salida aparece en la forma salida/acción_asociada. Cada vez que se sale del estado por una transición de salida la acción de salida se ejecuta. Una acción interna es una acción que se ejecuta cuando se recibe un determinado evento en ese estado, pero que no causa una transición a otro estado. Se indica en la forma nombre_de_evento/acción_asociada. Un diagrama de estados puede representar ciclos continuos o bien una vida finita, en la que hay un estado inicial de creación y un estado final de destrucción (finalización del caso de uso o destrucción del objeto).



El estado inicial se muestra como un círculo sólido y el estado final como un círculo sólido rodeado de otro círculo. En realidad, los estados inicial y final son pseudoestados, pues un objeto no puede “estar” en esos estados, pero nos sirven para saber cuáles son las transiciones inicial y final(es).

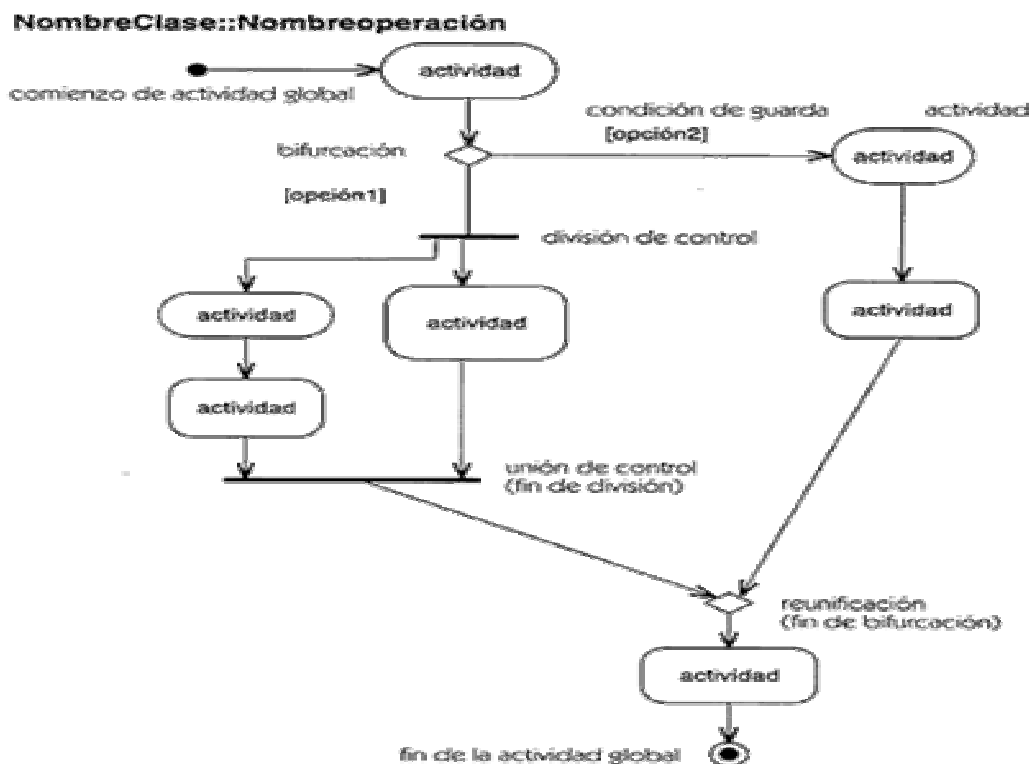
<http://www.clikear.com/manuales/uml/diagramasestados.asp>

3.4.1.5 Diagrama de Actividades

Un diagrama de actividad muestra la realización de operaciones para conseguir un objetivo. Presentan una visión simplificada de lo que ocurre en un proceso,

mostrando los pasos que se realizan, constituyéndose en uno de los diagramas que modelan los aspectos dinámicos del sistema.

Los diagramas de actividad son una extensión de los diagramas de estado. Los diagramas de estado resaltan los estados y muestran las actividades que dan lugar a cambios de estado, mientras que los diagramas de actividad resaltan justamente las actividades.



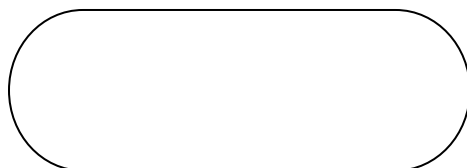
<http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/multiple-html/x277.html>

Básicamente un diagrama de actividades contiene:

- Estados de acción
- Estados de actividad
- Inicio y Terminación
- Transiciones
- Bifurcaciones
- Calles
- Señales.

3.4.1.5.1 Estado de Acción.- La representación es un rectángulo con las puntas redondeadas, en cuyo interior se representa una acción. La idea central es la siguiente:

“Un estado que represente una acción es atómico, lo que significa que su ejecución se puede considerar instantánea y no puede ser interrumpida”



3.4.1.5.2 Estados de Actividad.- Un estado de actividad representa la realización de una o varias tareas que causa un cambio en el sistema. La representación es un rectángulo con las puntas redondeadas

Un estado de actividad puede descomponerse en más sub-actividades representadas a través de otros diagramas de actividades. Además estos estados sí pueden ser interrumpidos y tardan un cierto tiempo en completarse.

En los estados de actividad podemos encontrar otros elementos adicionales como son: acciones de entrada (entry) y de salida (exit) del estado en cuestión.

http://www.embarcadero.com/support/what_is_uml.asp

3.1.4.5.3 Inicio y Terminación.- Un flujo de control debe empezar en algún sitio y terminar El inicio se representa mediante un círculo completamente relleno.

● Inicio del flujo

○ Término de flujo

Mientras que el término se representa con un círculo relleno dentro de una circunferencia.

3.4.1.5.4 Transiciones.- Las transiciones reflejan el paso de un estado a otro, bien sea de actividad o de acción. Esta transición se produce como resultado de la finalización del estado del que parte la transición.

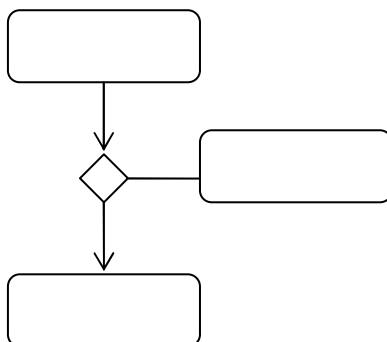
Como todo flujo de control debe empezar y terminar en algún momento, podemos indicar esto utilizando dos disparadores de inicio y fin.

Se representa mediante una línea dirigida que va de la actividad que finaliza hacia la actividad a ejecutar. Las flechas que indican la transición no deben ser etiquetadas pues son disparadas automáticamente por la finalización de la primera actividad.

3.4.1.5.5 Bifurcaciones (Decisiones).- En los diagramas de actividad se pueden incluir caminos alternativos según se cumpla alguna condición. Estas condiciones se representan mediante un rombo al cual llega la transición de la actividad origen y del cual salen las múltiples transiciones a cada actividad destino, cada una con una condición diferente.

Se pueden incluir la palabra else que indica que ninguna de las expresiones de guarda es verdad.

Una expresión de guarda es aquella que puede tomar un valor de verdadero o falso. Las alternativas de la bifurcación han de ser excluyentes y contemplar todos los casos ya que de otro modo la ejecución quedaría interrumpida.



<http://www.monografias.com/trabajos5/insof/insof.shtml>

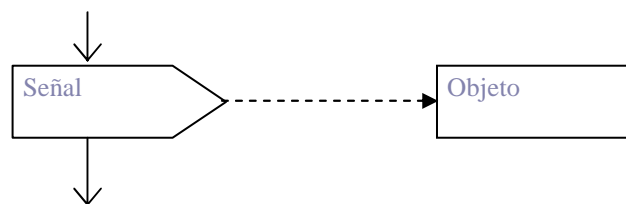
3.4.1.5.6 Barras de Sincronización.- Al modelar flujos de procesos podemos encontrar actividades que se pueden realizar simultáneamente. A una barra de sincronización se la representa como una línea vertical u horizontal gruesa.

Las barras de sincronización muestran actividades concurrentes es decir que la ejecución de procesos es al mismo tiempo; de forma paralela.

3.4.1.5.7 *Calles*.- Cuando se modelan flujos de trabajo de organizaciones, es especialmente útil dividir los estados de actividades en grupos, cada grupo tiene un nombre concreto y se denominan calles. Cada calle representa a la parte de la organización responsable de las actividades que aparecen en esa calle .

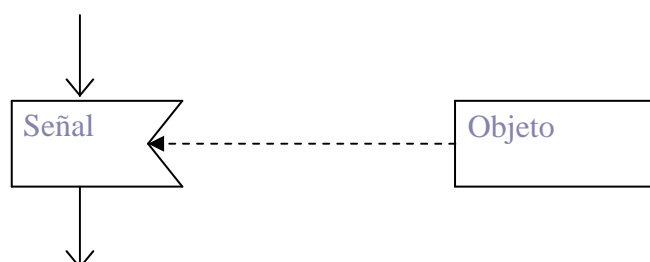
3.4.1.5.8 *Señales*.- Algunos objetos pueden enviar señales para la realización de alguna actividad. Estas señales pueden ser de dos tipos: señal de envío y señal de recibo, aunque no son indispensables para la construcción de diagramas de actividades y pueden evitarse.

3.4.1.5.9 *Señal de envío*.- El envío es una señal, desde nuestro diagrama de actividades hacia un objeto, puede ser mostrado como un pentágono convexo que luce como un rectángulo con un triángulo apuntando hacia fuera. El significado de esta señal es mostrada dentro del símbolo.



Esta señal se coloca dentro de dos transiciones una entrando desde una actividad y otra saliendo hacia otra actividad. Opcionalmente se puede dibujar una flecha de línea discontinua desde el pentágono convexo, hacia el objeto receptor de la señal

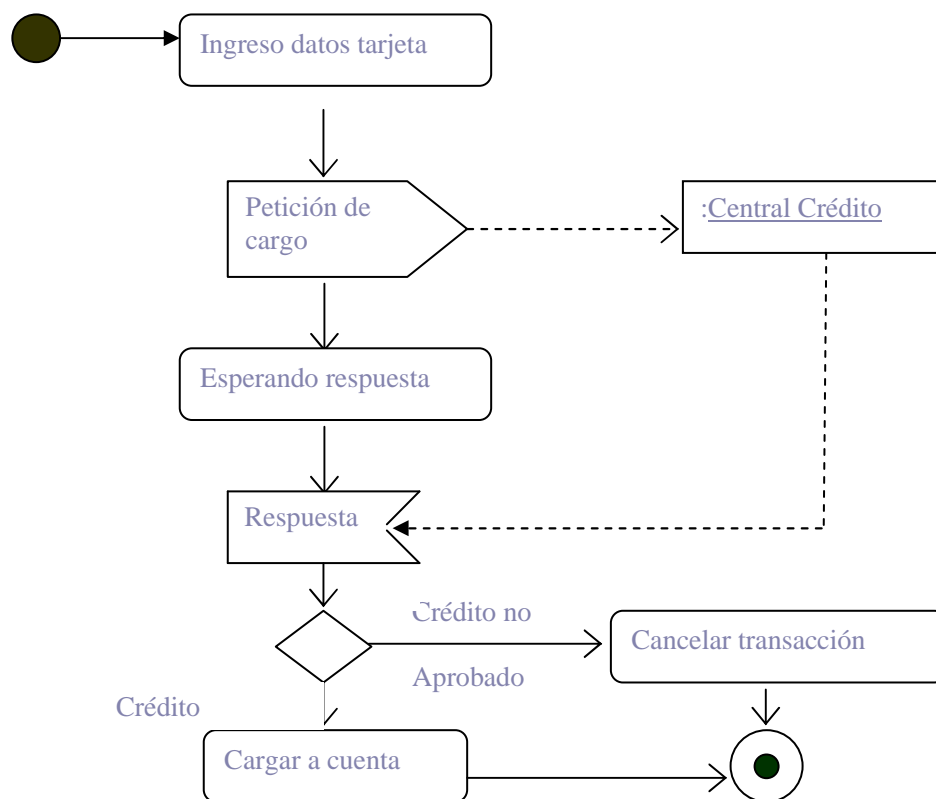
3.4.1.5.10 *Señal de recibo*.- El recibo de una señal, por nuestro diagrama de actividades desde un objeto puede ser mostrado como un pentágono cóncavo que luce como un rectángulo con un triángulo entrante en uno de sus lados. El significado de esta señal es mostrado dentro del símbolo.



Esta señal se coloca dentro de dos transiciones una entrando desde una actividad y la otra saliendo hacia otra actividad. Opcionalmente se puede dibujar una flecha de línea discontinua desde el objeto que envía la señal hacia el pentágono cóncavo

Ejemplo:

La siguiente descripción corresponde a un proceso típico de pago con tarjeta de crédito. Cuando se realizan pagos con tarjeta de crédito debe introducirse los datos de la tarjeta, luego se dispara una señal de petición de cargo, hacia la central de crédito mientras esto ocurre estaremos esperando a que la central de crédito nos envíe la señal de autorizado, para finalmente hacer el cargo respectivo a la tarjeta.



<http://lucas.hispalinux.es/Tutoriales/doc-modelado-sistemas-UML/multiple-html/index.html>

CAPITULO 4

4. CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

- El utilizar la metodología OOHDM como técnica de diseño. A permitido dedicar un tiempo importante en las fases previas a la implementación. Esta inversión de tiempo está ampliamente justificada no sólo porque simplifica el proceso de desarrollo, sino que también permitirá el mantenimiento de la aplicación.
- OOHDM propone un conjunto de tareas que en principio pueden involucrar mayores costos de diseño, pero que a mediano y largo plazo reducen notablemente los tiempos de desarrollo al tener como objetivo principal la reusabilidad de diseño, y así simplificar la evolución y el mantenimiento de toda aplicaron web.
- EL software de Open Source fue muy eficaz y útil en el desarrollo de este Portal WEB, no solo por el lenguaje de programación al momento de utilizar PHP, sino que el fácil manejo de creación tablas e información con Mysql, a esto se puede agregar y recalcar la confiabilidad, seguridad el este presta,
- Con el desarrollo de este Portal Web, se fortaleció conocimientos en ámbitos de Diseño, Programación, Seguridades y algo muy importante es que permitió ampliar la visión y los objetivos Profesionales.

4.2 RECOMENDACIONES

- Es importante que en la brevedad posible se permita la interfaz con la base de datos, específicamente a la consulta de saldos de los clientes que actualmente maneja la COAC. Alianza del Valle.

- Es importante capacitar al personal que administrará el sitio web para tener un correcto funcionamiento de la aplicación, con el fin de brindar un buen servicio al usuario y cliente de la institución
- El sitio debe ser actualizado constantemente, por el personal asignado para el mismo, ya que el Portal Web va ser la herramienta de marketing y negocios en la captación de los nuevos clientes.
- Se recomienda crear varios templates para el Portal Web, los mismos deberían tener relación con fechas importantes como son: El Aniversario de la Institución, Navidad,
- Es sumamente importante sacar respaldos de la base de datos semanalmente o cada vez que se ingresen o actualice la información y así evitar perdidas o inconsistencia de lo almacenado en la base de datos.
- Se recomienda instalar seguridades como fireware, antivirus, antispay, etc que protejan al sitio de intrusos y ataques. Así evitar que la información, tanto de Base de Datos y el Portal, sea alterada.
- Contratar un buen servicio de Internet para que no colapse el portal al navegar en el mismo, solo así tendremos clientes y usuarios satisfechos.

BIBLIOGRAFÍA

- PRESSMAN Roger, Ingeniería de Software, 2002, Quinta Edicion.
- Graig, L.(1998). UML y Patrones, Introducción al Análisis y Diseño Orientado a Objeto. México: Editorial Pretince Hall.
- SILVA Darío, MERCERAT Bárbara; Construyendo Aplicaciones Web con una Metodología Orientada a Objetos; Buenos Aires; 2002.
- WESLEY Addison, JACOBSON Ivar, BOOCH Grady, RUMBAUGH James; El Lenguaje Unificado de Modelado; ED. Pearson Education S.A.; Madrid; 2000
- http://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_la_Web
- <http://www.informandote.com/jornadasIngWEB/articulos/jiw01.pdf>
- <http://es.wikipedia.org/wiki/Usabilidad>
- [http:// alarcos.inf-cr.uclm.es/doc/calidadSI/Metodos%20De%20Calidad.ppt](http://alarcos.inf-cr.uclm.es/doc/calidadSI/Metodos%20De%20Calidad.ppt)

- <http://www.nosolousabilidad.com/articulos/ai.htm>
- http://es.wikipedia.org/wiki/Arquitectura_de_la_informaci%C3%B3n
- <http://www.webaprendiz.com/viewp/notas/arquiweb/2001/06/id0001.htm>
- <http://www.pabloimpallari.com.ar/articulos/arquitecturaweb.php>
- <http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node18.html>
- http://es.wikipedia.org/wiki/Dise%C3%B1o_gr%C3%A1fico
- <http://es.wikipedia.org/wiki/Inform%C3%A1tica>
- <http://www.desarrolloweb.com/articulos/importancia-documentacion.html>
- http://es.wikipedia.org/wiki/Dise%C3%B1o_de_p%C3%A1ginas_web
- <http://es.wikipedia.org/wiki/W3C>
- http://es.wikipedia.org/wiki/HTML#Historia_del_est.C3.A1ndar
- <http://es.wikipedia.org/wiki/JavaScript>
- <http://es.wikipedia.org/wiki/PHP>
- <http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node21.html>
- <http://agamenon.uniandes.edu.co/~pfigueroa/soo/uml>
- <http://www.creangel.com/uml/secuencia.php#>
- http://www.embarcadero.com/support/what_is_uml.asp
- <http://www.monografias.com/trabajos5/insof/insof.shtml>
- <http://lucas.hispalinux.es/Tutoriales/doc-modelado-sistemas-UML/multiple-html/index.html>

ANEXOS

MANUAL TÉCNICO

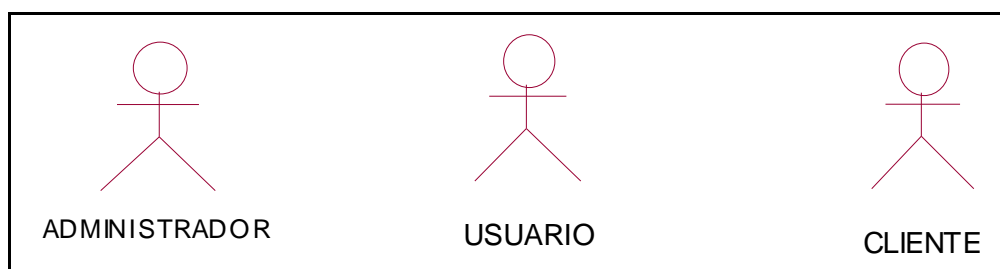
CONTENIDO

1.	MODELO ESTÁTICO	1
1.1	IDENTIFICACIÓN DE ACTORES	2
1.2	DIAGRAMA DE CASOS DE USO.....	3
1.3	DICCIONARIO DE CASOS DE USO	7
1.4	DIAGRAMA DE CLASES.....	8
1.5	DICCIONARIO DE CLASES	9
2.	MODELO DINÁMICO	13
2.1	DIAGRAMAS DE INTERACCIÓN	14
2.2	DIAGRAMAS DE SECUENCIA	16
3.	MODELO FUNCIONAL.....	18
3.1	DIAGRAMA DE ACTIVIDADES.....	19
4.	MODELO ARQUITECTÓNICO.....	21
4.1	DIAGRAMA ARQUITECTÓNICO.....	22
5.	INGENIERÍA	1

1. MODELO ESTÁTICO

1.1 IDENTIFICACIÓN DE ACTORES

Para una mejor comprensión del rol que tienen los usuarios del Sistema se realizó una breve descripción de cada uno de ellos colocando su descripción, responsabilidades y uso del sistema.

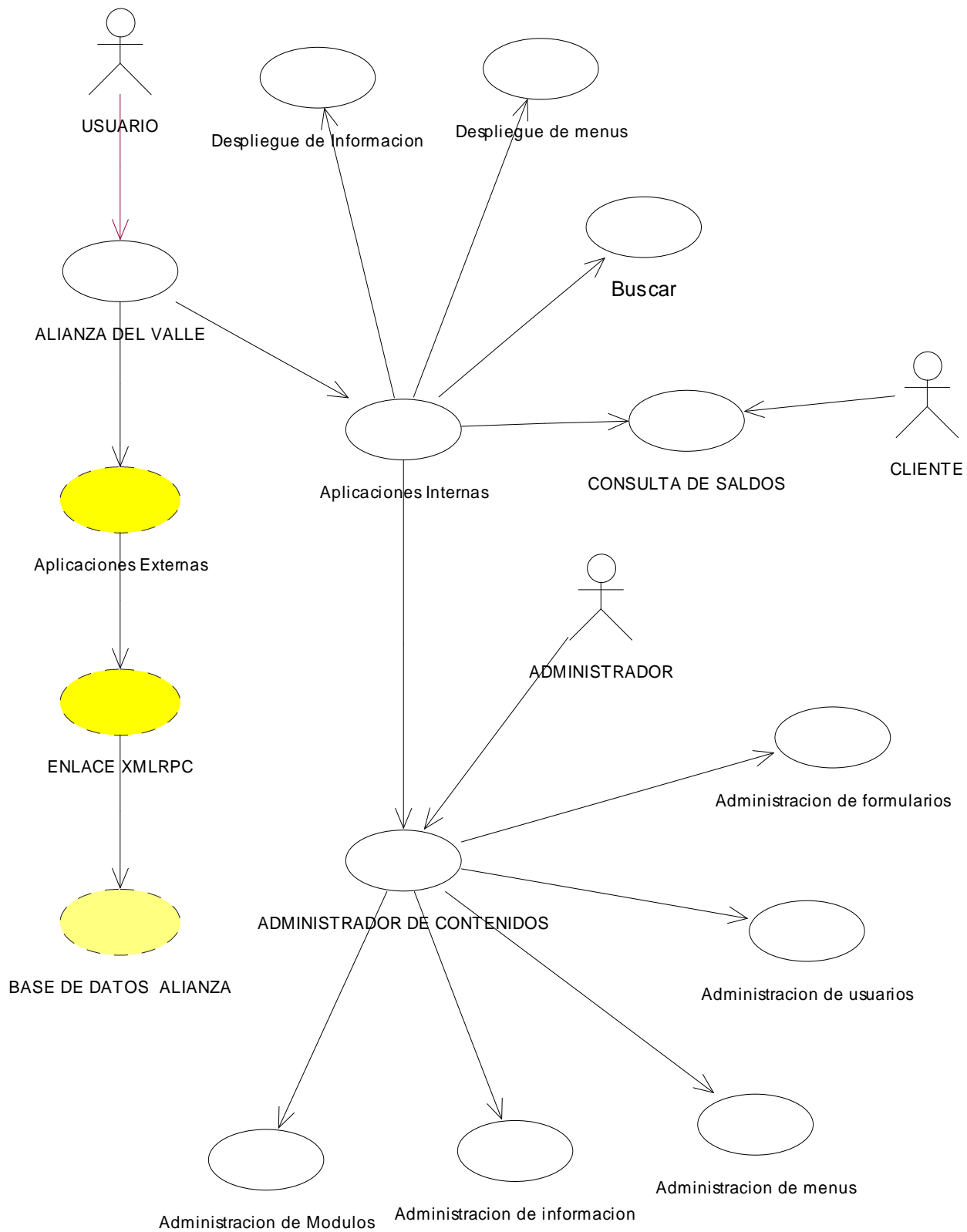


ADMINISTRADOR.- Es la persona que se encarga de administrar todo lo referente al Sitio Web, actualiza la Información propuesta en el mismo. También es el encargado de dar los accesos respectivos a los CLIENTES.

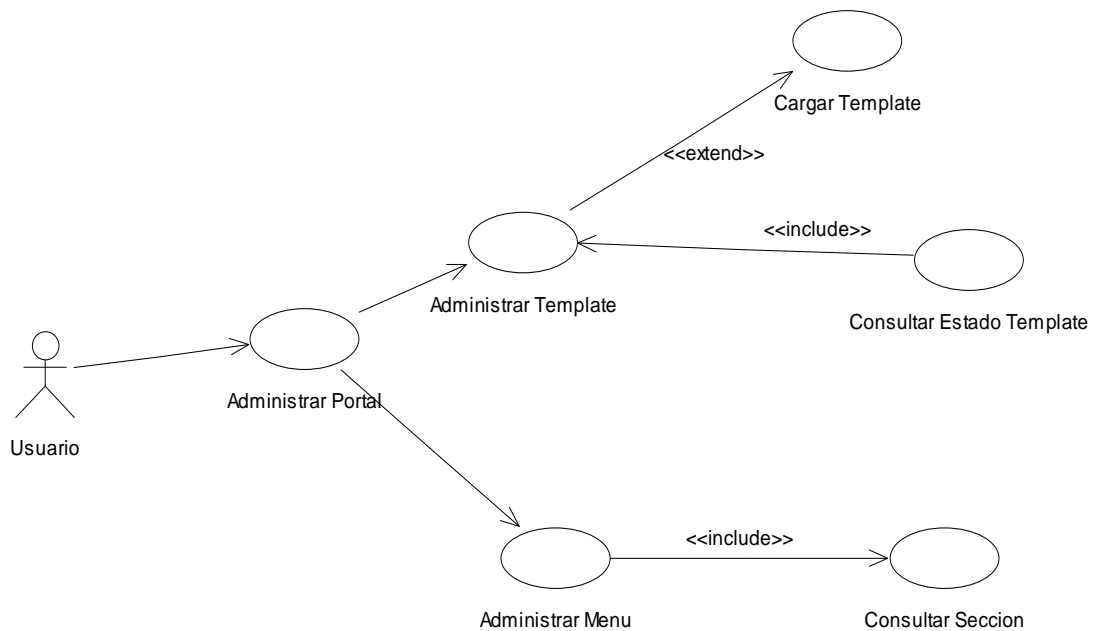
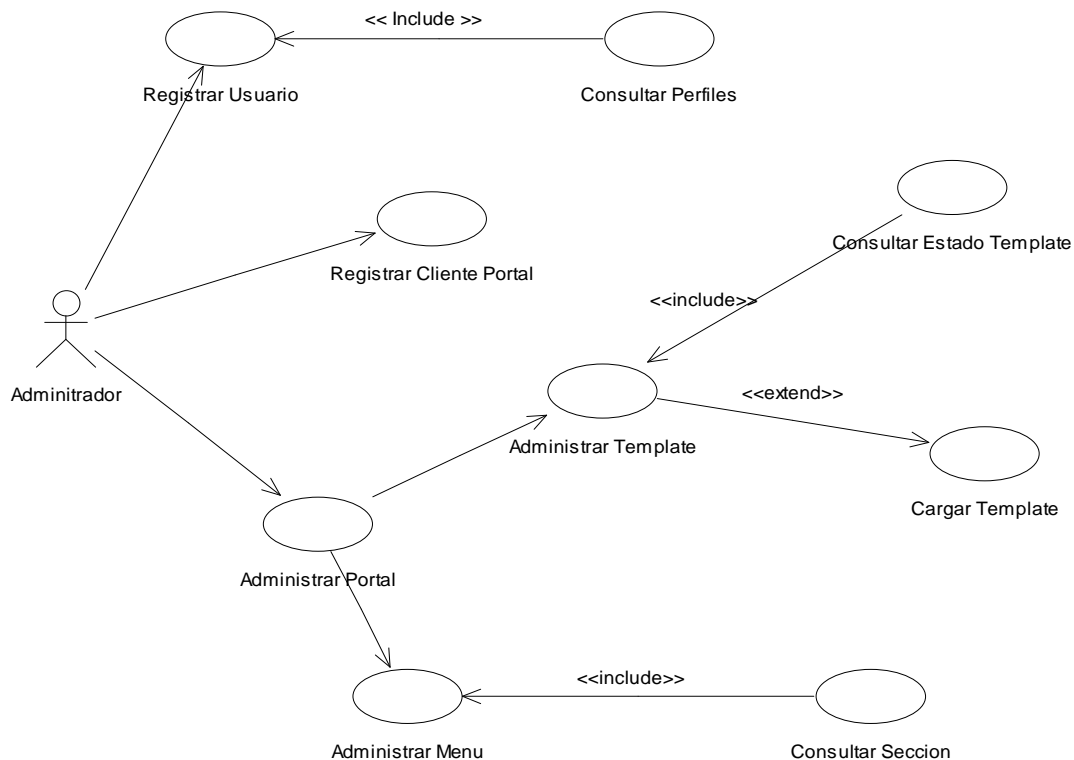
USUARIO.- Es la Persona que visita el Sitio Web, puede revisar la información de los productos y servicios que ofrece la COAC ALIANZA DEL VALLE

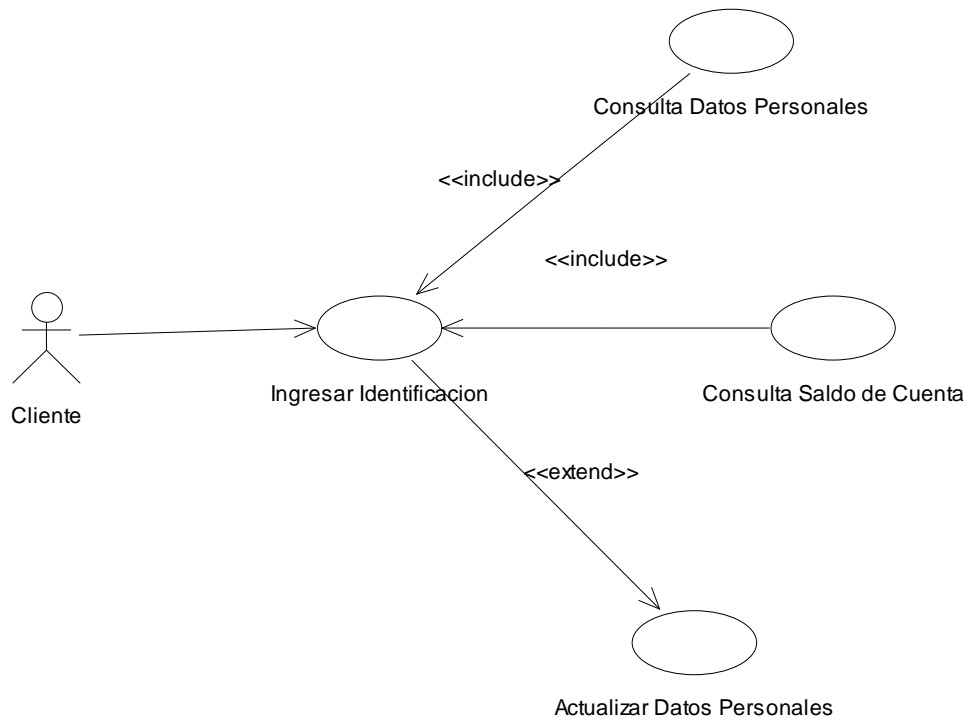
CLIENTE.- Es un Usuario que esta registrado en el Sistema, este tendrá acceso a la información de sus saldos de su cuenta de Ahorros.

1.2 DIAGRAMA DE CASOS DE USO



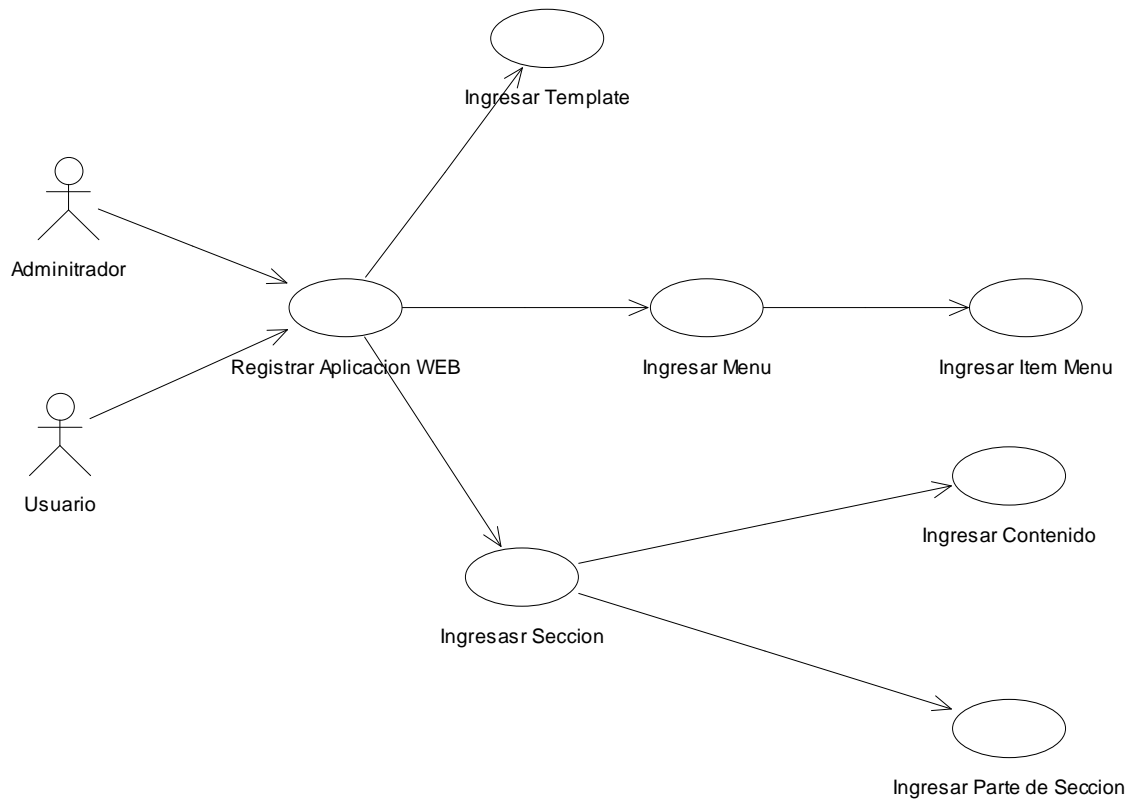
DIAGRAMAS DE CASOS DE USO POR ACTORES



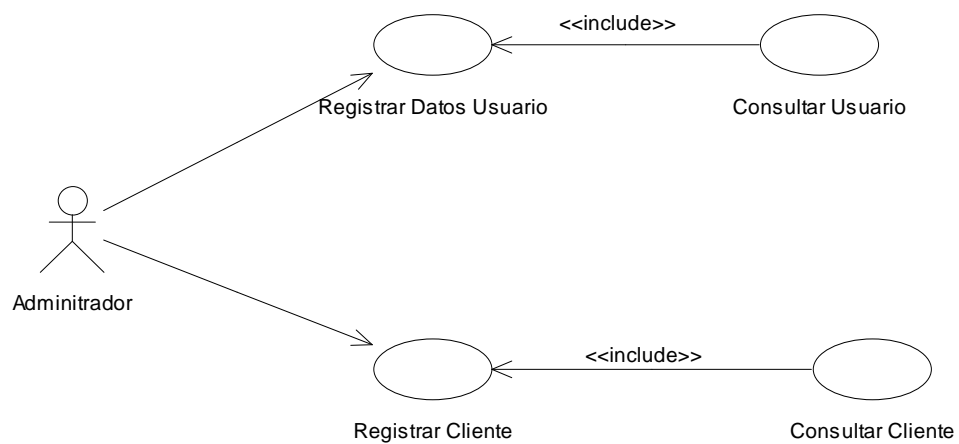


DIAGRAMAS DE CASOS DE USO POR ACTIVIDADES

ADMINISTRAR CONTENIDOS DEL PORTAL WEB



ADMINISTRAR CLIENTES Y USUARIOS



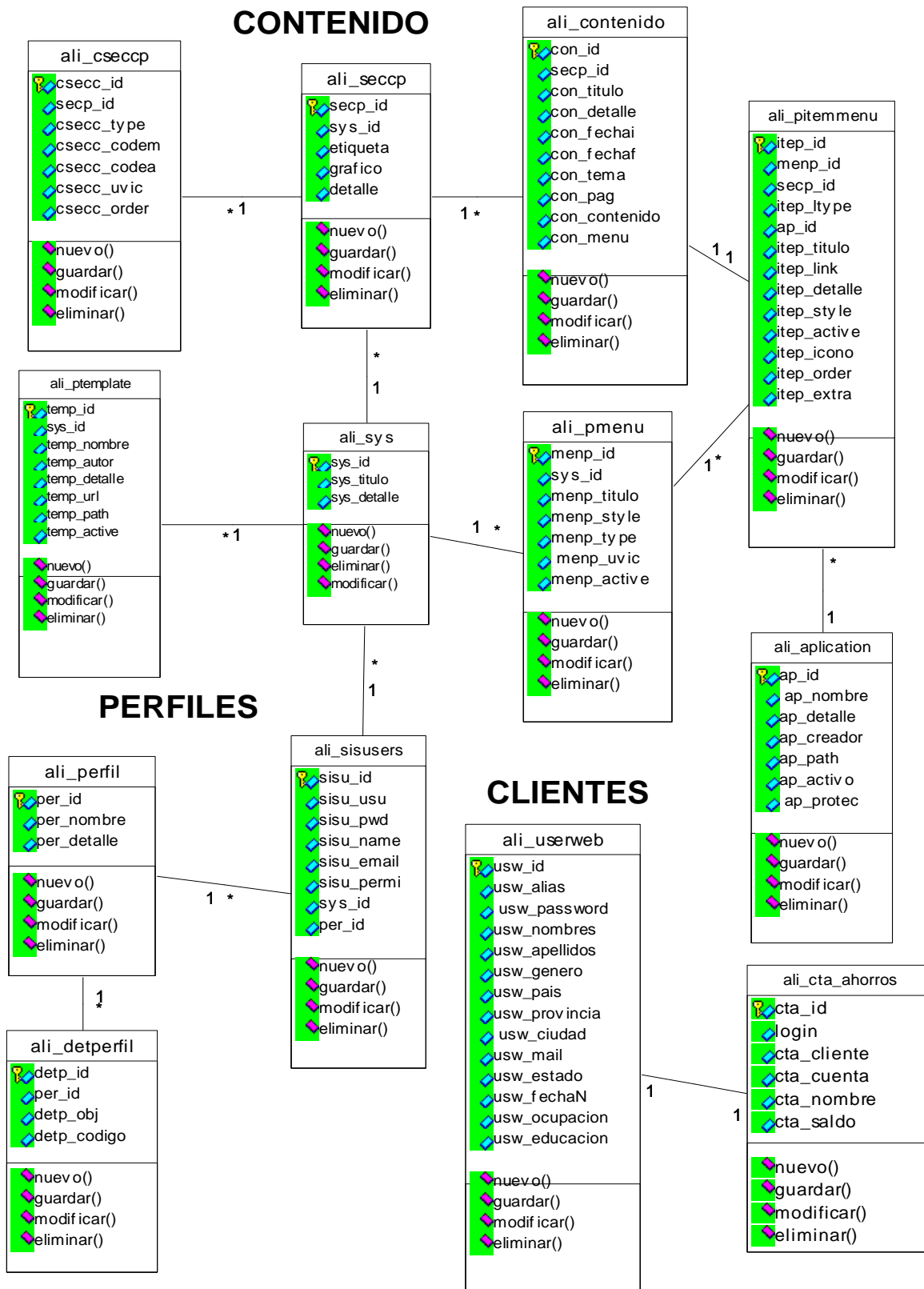
1.3 DICCIONARIO DE CASOS DE USO

Administrar Contenidos Del Portal Web.- En este proceso el Usuario asignado para la administración de contenidos del portal WEB, ingresa a través de su perfil de usuario (clave personal), al Administrador De Contenidos, este tendrá el link “Portales”, donde puede acceder a la actualización o eliminación de “Template, Menús y Secciones”, tomando en cuenta el orden la siguiente de creación:

- **Template**, en este punto debemos asignar o activar el t mplate a utilizar para el portal Web.
- **Secciones**, Este puede tener varias secciones, y cada secci n debe tener la ubicaci n o la parte de secci n con sus respectivos contenidos (Informaci n a mostrar en el Portal Web)
- **Menus**, en este punto se debe crear los men s que puede tener el portal, con sus respectivos submen s, los mismos que estar n enlazados a cada secci n.

Administrar Clientes Y Usuarios.- En este proceso el Super Usuario ingresa a trav s de su perfil de usuario (clave personal), al Administrador De Contenidos, por este podr  registrar, actualizar y eliminar nuevos Usuarios (Administrador de contenidos) y nuevos Clientes (Para consulta de saldos).

1.4 DIAGRAMA DE CLASES



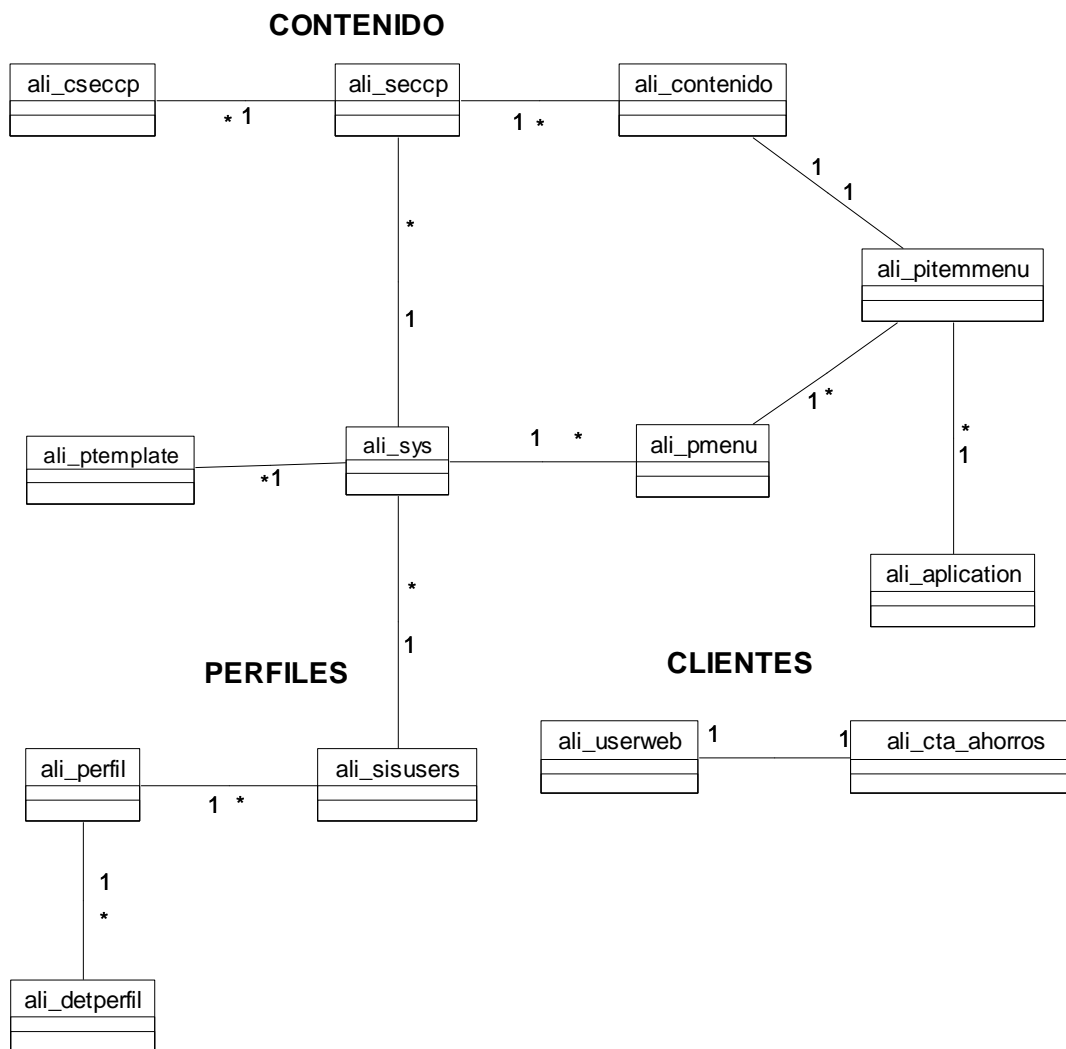
1.5 DICCIONARIO DE CLASES

Clase	Atributos	Descripción
CONTENIDO		
ali_sys	sys_id sys_titulo sys_detalle	Almacena el sistema de contenidos
ali_ptemplate	temp_id sys_id temp_nombre temp_autor temp_detalle temp_url temp_path temp_active	Almacena el template asignado para el portal
ali_pmenu	menp_id sys_id menp_titulo menp_style menp_type menp_uvic menp_active	Almacena la ubicación de los menús asignados para el sistema de contenidos
ali_seccp	secp_id sys_id etiqueta grafico detalle	Almacena las secciones del portal
ali_cseccp	csecc_id secp_id csecc_type csecc_codem csecc_codea csecc_uvic csecc_order	Almacena los objetos que componen cada sección
ali_contenido	con_id secp_id con_titulo con_detalle con_fecha con_fecha con_tema con_pag con_contenido con_menu	Almacena el contenido del portal por sección
ali_pitemmenu	itep_id menp_id	Almacena los menús asignados para los

	secp_id itep_ltype ap_id itep_titulo itep_link itep_detalle itep_style itep_active itep_icono itep_order itep_order itep_extra	contenidos del portal
USUARIOS		
ali_perfil	per_id per_nombre per_detalle	Almacena los perfiles del usuario
ali_detperfil	detp_id per_id detp_obj detp_codigo	Almacena los atributos de acceso para cada perfil
ali_sisusers	sisu_id sisu_usu sisu_pwd sisu_name sisu_email sisu_permi sys_id per_id	Almacena los usuarios que tienen el acceso para el sistema de contenidos
CLIENTES		
ali_aplication	ap_id ap_nombre ap_detalle ap_creador ap_path ap_activo ap_protec	Almacena la aplicación a la cual pueden acceder los clientes en este caso la de consulta de saldos
ali_userweb	usw_id usw_alias usw_password usw_nombres usw_apellidos usw_genero usw_pais usw_provincia usw_ciudad	Almacena los datos de los clientes que tendrán acceso al portal en este caso consultas de saldos

	usw_mail usw_estado usw_fechaN usw_ocupacion usw_educacion	
ali_cta_ahorros	cta_id login cta_cliente cta_cuenta cta_nombre cta_saldo	Almacena el estado de cuenta de los clientes

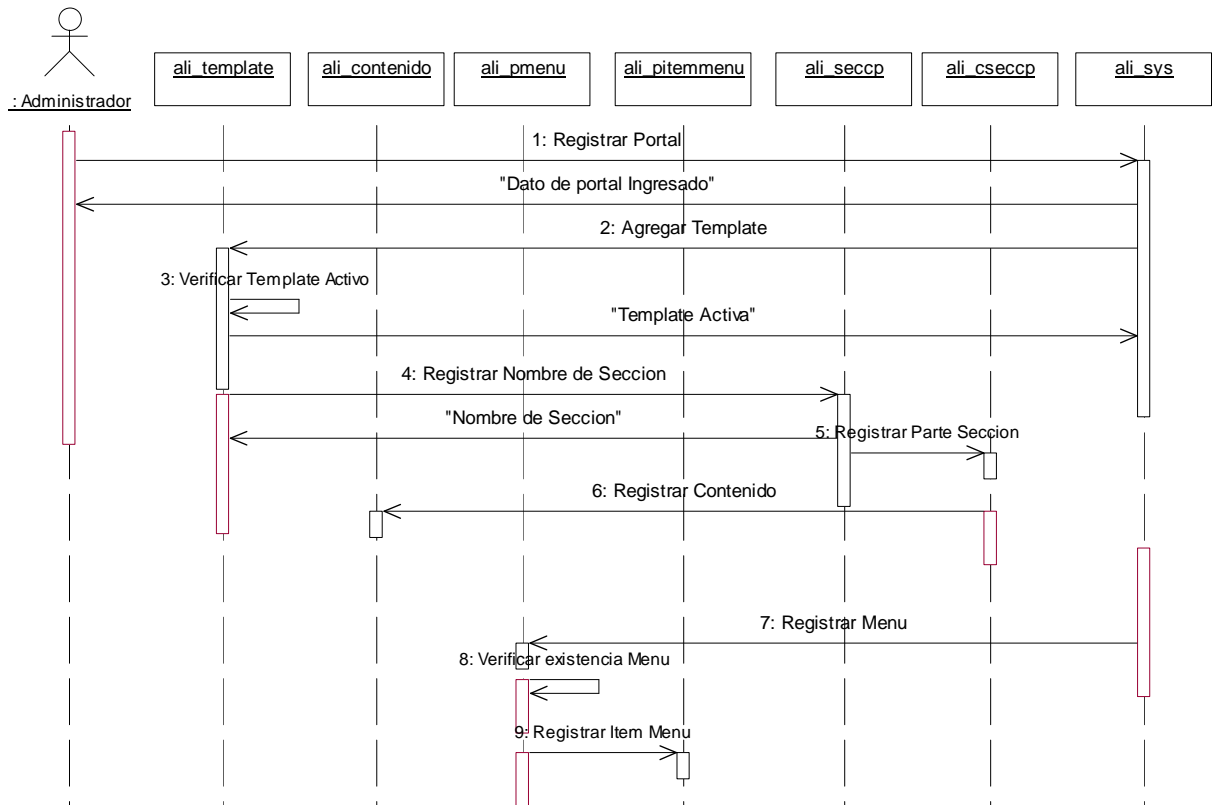
DIAGRAMA DE OBJETOS



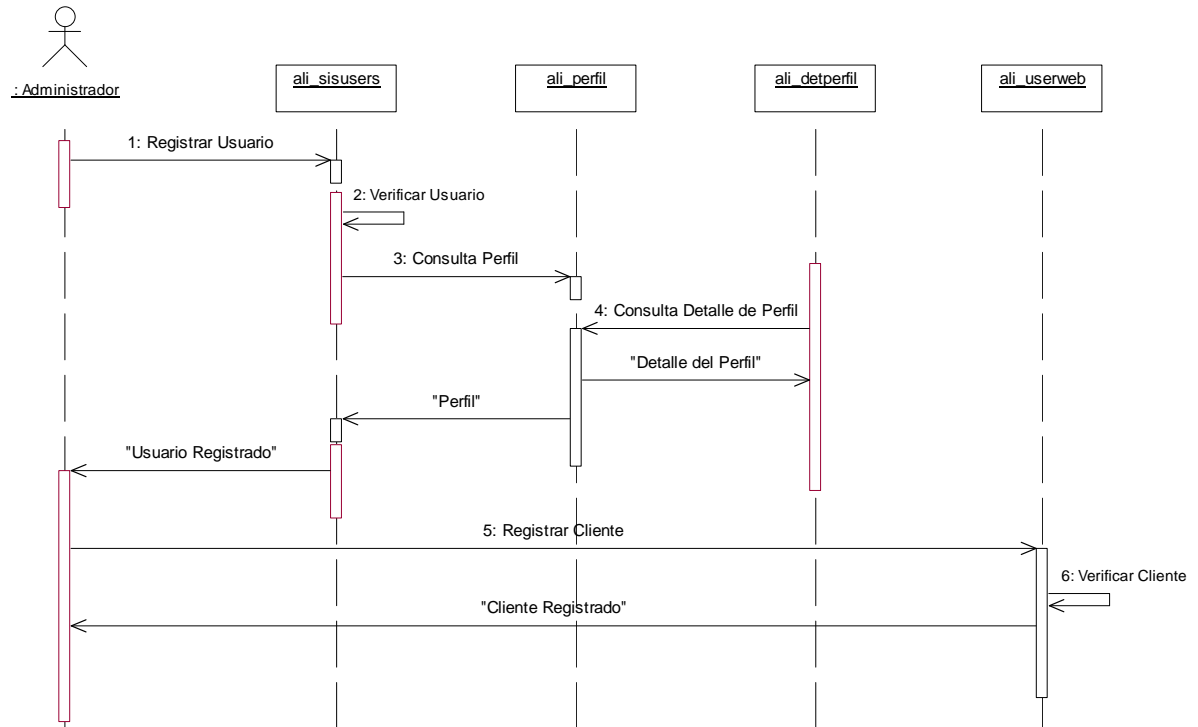
2. MODELO DINÁMICO

2.1 DIAGRAMAS DE INTERACCIÓN

REGISTRAR CONTENIDOS DEL PORTAL WEB

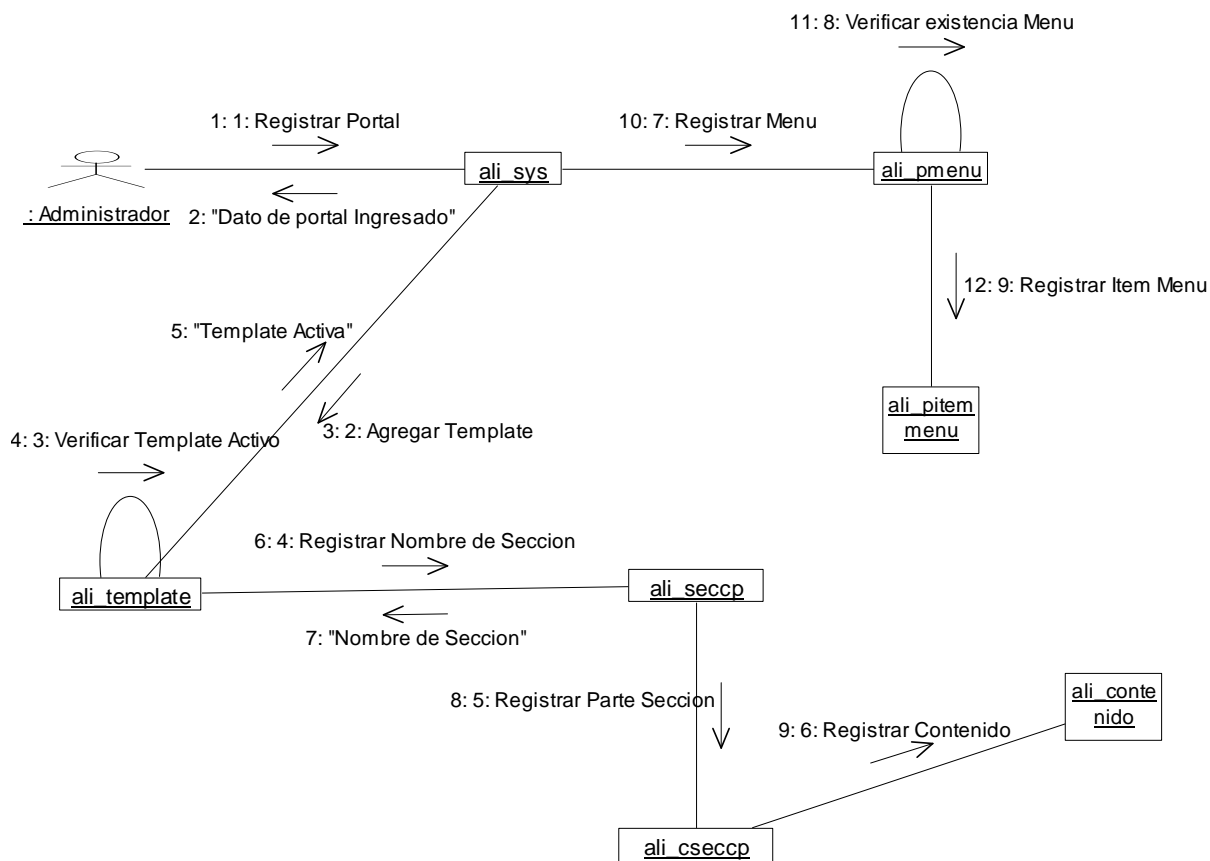


REGISTRAR CLIENTES Y USUARIOS

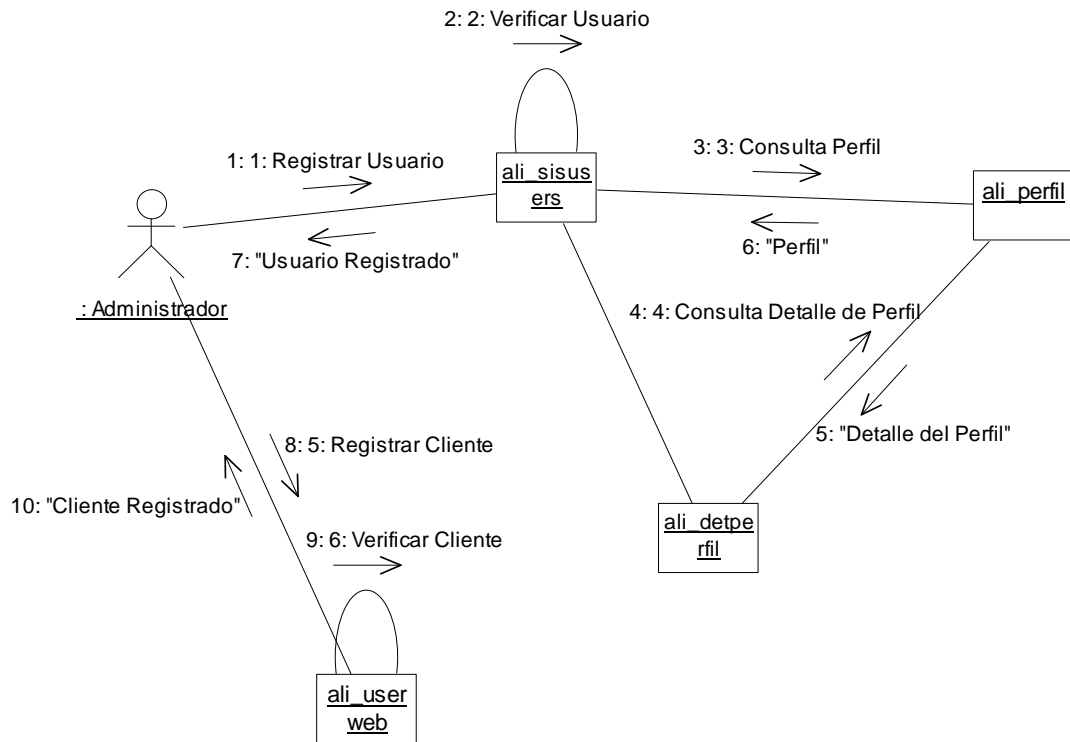


2.2 DIAGRAMAS DE SECUENCIA

REGISTRAR CONTENIDOS DEL PORTAL WEB



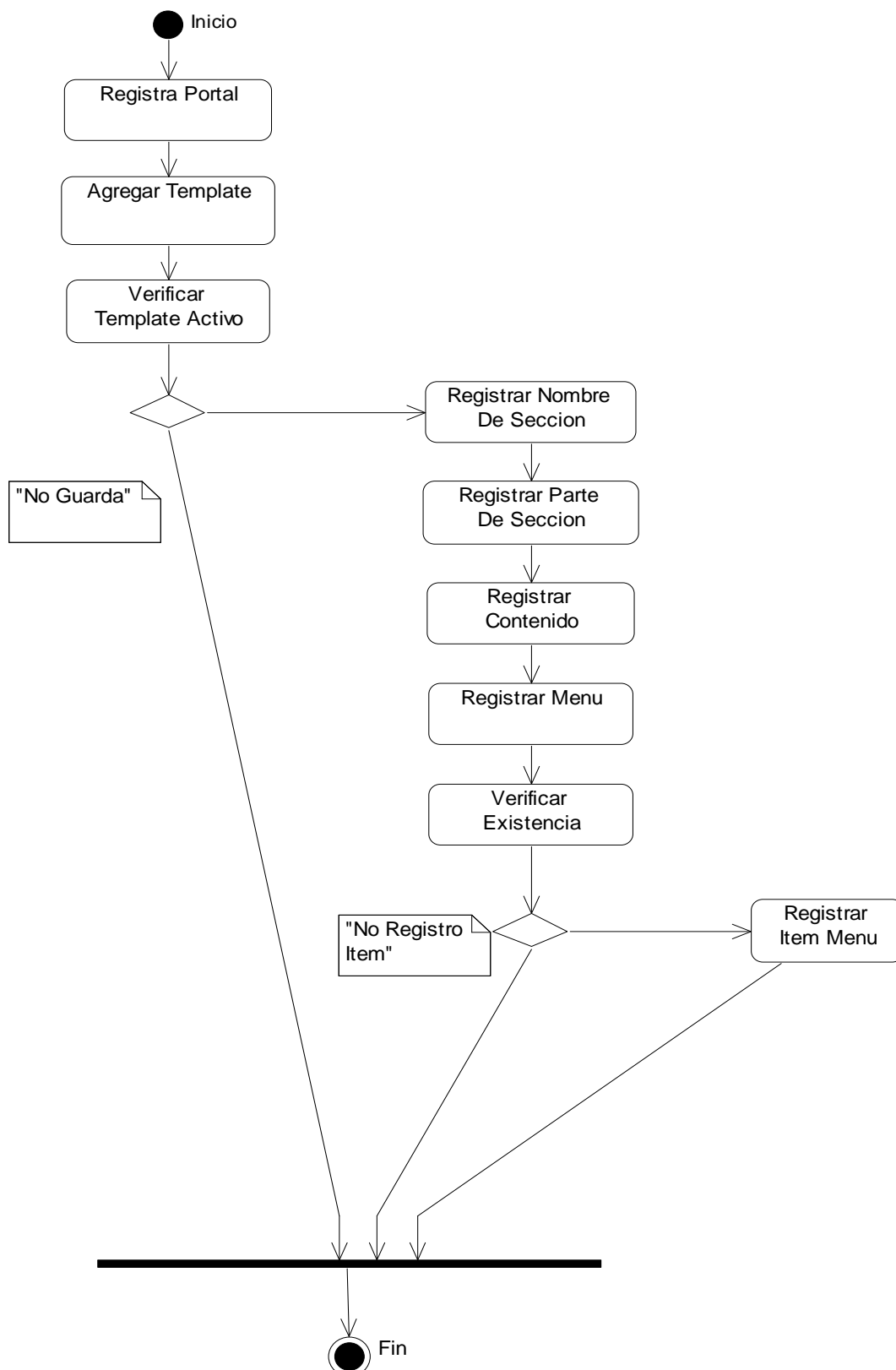
REGISTRAR CLIENTES Y USUARIOS

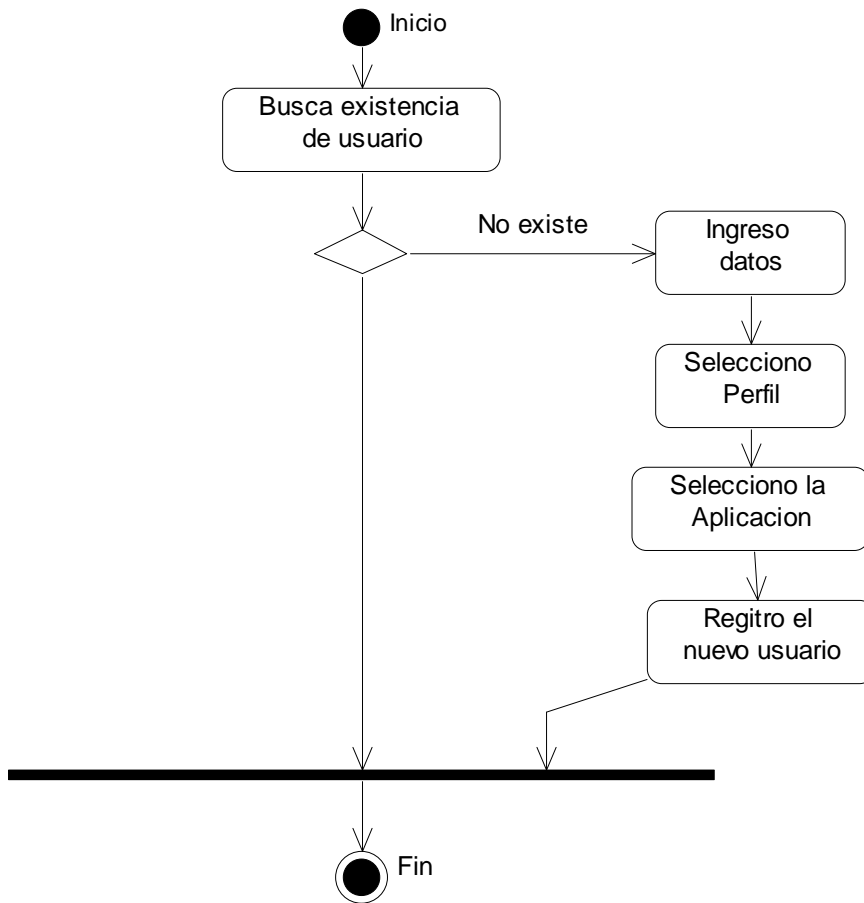


3. MODELO FUNCIONAL

3.1 DIAGRAMA DE ACTIVIDADES

REGISTRAR CONTENIDOS DEL PORTAL WEB



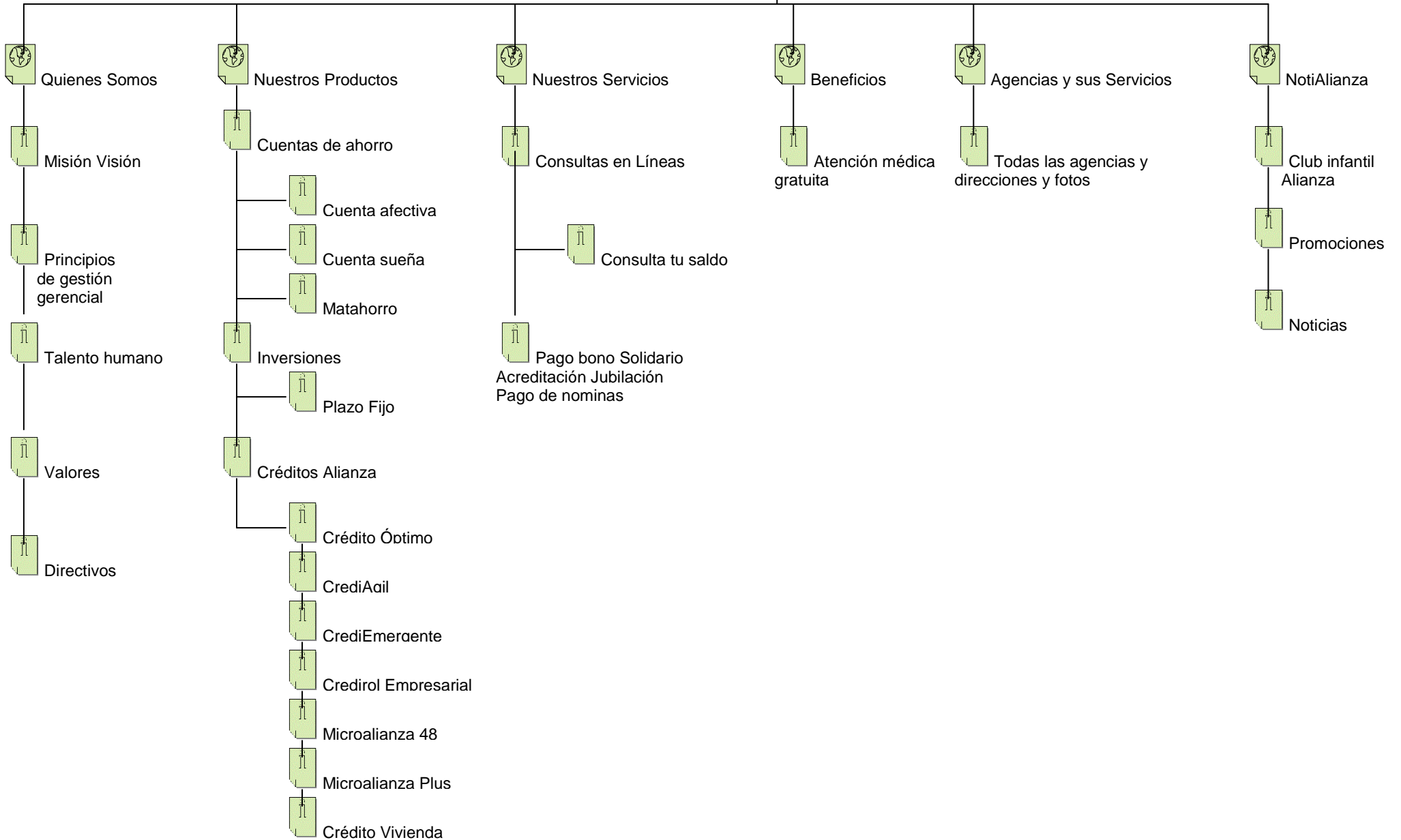
REGISTRAR USUARIO

4. MODELO ARQUITECTÓNICO



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

4.1 DIAGRAMA ARQUITECTÓNICO



5. INGENIERÍA

CÓDIGO FUENTE

indexr.php

```
<?php
//session_cache_limiter('nocache,private');
session_name('MMusuarios');
session_start();
session_register('MM_Username');
session_register('MM_passwd');
session_register('userid');
session_register('MM_UserAuthorization');

?>
<?php
include ("cfgclases/config.php");
include ("cfgclases/clases.php");
//Sesiones
$objacceso_system = new acceso_system();
$objacceso_session = new session_system();
//Conexion a la base de datos
$objBd = new conecc();
//Objeto menu
$objmenup = new menup();
//Objeto template
$objtemplatep = new templatep();
//Objeto contenido
$objcontenido = new contenidop();
//Objeto tablas templates
$objtableform= new templateform();
//Conexion ejecutada
$objBd->conectardb($basededatos,$host,$userdb,$passwd);
//formulario
$objform= new formulario();
```

```
//Objeto estaditicas
$objest= new estadisticas();
$objest->visitas_new($ar,$apl,$seccionp,$system);
?>
<?php
//Sesiones de usuarios
if ($close==1)
{

    $_SESSION['nov_ususer']="";
    $_SESSION['nov_pwd']="";
    $_SESSION['nov_name']="";
}

?>
<?php
$objtemplatep->select_templatep($system);
$objcontenido->select_contenidop(1);
$objacceso_session->select_session($sessid);
if ($system)
{
    include($objtemplatep->path_template."index.php");
}
?>
```

index.php

```

<?php
$segundo =trim((date("s"))); $graf= $segundo % 3 ;
?>
<html>
<head>
<title>alianzadelvalle.fin.ec</title>
<meta name="keywords" content="Alianza del Valle, Cooperativa de Ahorro y
Credito, Financiera, Cooperativa, Ahorro, CreditoEcuador, Quito, Sudamerica">
<meta name="description" content=" Cooperativa de ahorro y credito
alianzadelvalle.fin.ec e-mail : cooperativa@alianzadelvalle.fin.ec *Tel. (5932)
2335372* Quito Ecuador">

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<script language="JavaScript">
<!-- Uculto para browsers
function openWindow(url) {
    popupWin = window.open(url,'new_page','width=550,height=500')
}
function email(url) {
    popupWin = window.open(url,'new_page','width=250,height=350')
}
function aviso(url) {
    popupWin = window.open(url,'new_page','width=450,height=400')
}
function politicas(url) {
    popupWin = window.open(url,'new_page','width=450,height=400')
}
function prensa(url) {
    popupWin = window.open(url,'new_page','width=450,height=400')
}
// done hiding -->
</script>
<script language="JavaScript">
<!--
function MM_goToURL() { //v3.0
    var i, args=MM_goToURL.arguments; document.MM_returnValue = false;
    for (i=0; i<(args.length-1); i+=2) eval(args[i]+"."+location+"="+args[i+1]+"");
}
//-->
</script>
<script language="JavaScript">
<!--
<!--
function MM_reloadPage(init) { //reloads the window if Nav4 resized
    if (init==true) with (navigator) {if
((appName=="Netscape")&&(parseInt(appVersion)==4)) {

```

```

    document.MM_pgW=innerWidth;          document.MM_pgH=innerHeight;
onresize=MM_reloadPage; }}
    else if (innerWidth!=document.MM_pgW || innerHeight!=document.MM_pgH)
location.reload();
}
MM_reloadPage(true);
// -->

```

```

function MM_openBrWindow(theURL,winName,features) { //v2.0
    window.open(theURL,winName,features);
}
//-->
</script>

```

```

<link href="<?php echo $objtemplatep->path_template ?>styles/formato.css"
rel="stylesheet" type="text/css">
<link href="<?php echo $objtemplatep->path_template ?>styles/cebra_con.css"
rel="stylesheet" type="text/css">
<link href="<?php echo $objtemplatep->path_template ?>styles/estilo.css"
rel="stylesheet" type="text/css">
<link href="styles/formato.css" rel="stylesheet" type="text/css">
<style type="text/css">
<!--
body {
    margin-left: 0px;
    margin-top: 0px;
    margin-right: 0px;
    margin-bottom: 0px;
}
.Estilo3 {
    font-family: Verdana, Arial, Helvetica, sans-serif;
    font-size: 11px;
    font-weight: bold;
    color: #006699;
}
.Estilo4 {font-size: 11px; color: #003300; font-family: Verdana, Arial, Helvetica,
sans-serif;}
-->
</style>
</head>
<body>
<table width="663" border="0" align="center" cellpadding="0" cellspacing="0"
bgcolor="#FFFFFF">
<tr>
<td width="791"><table width="659" border="0" align="center" cellpadding="0"
cellspacing="0">
<!-- fwtable fwsrc="head.png" fwbase="head.jpg" fwstyle="Dreamweaver"
fwdocid = "650028513" fwnested="0" -->
<tr>

```

```

        <td></td>
        <td></td>
        <td></td>
        <td></td>
        <td></td>
        <td></td>
        <td></td>
        <td></td>
        <td></td>
        <td></td>
        <td></td>
    </tr>
    <tr>
        <td rowspan="4" colspan="2"></td>
        <td colspan="2"><a href="indexr.php?system=10" target="_top"></a></td>
        <td colspan="2"><a href="#"></a></td>
        <td colspan="2"><a href="#"></a></td>
        <td rowspan="3"></td>
        <td><a href="#"></a></td>
        <td></td>
    </tr>
    <tr>
        <td colspan="6"></td>

```

```

        <td rowspan="2"></td>
        <td></td>
    </tr>
    <tr>
        <td rowspan="2"></td>
        <td rowspan="2" colspan="2"><a href="indexr.php?system=10"
target="_top"></a></td>
        <td rowspan="2" colspan="3"></td>
        <td></td>
    </tr>
    <tr>
        <td colspan="2"></td>
        <td></td>
    </tr>
    <tr>
        <td rowspan="3"></td>
        <td colspan="6" rowspan="2" background="<?php echo $objtemplatep-
>path_template ?>images/head_r5_c2.jpg"><?php
$objcontenido->objetos_seccion(34,'i',$varsend,$system);
?></td>
        <td colspan="3"></td>
        <td></td>
    </tr>
    <tr>
        <td colspan="3" rowspan="2" background="<?php echo $objtemplatep-
>path_template ?>images/head_r6_c8.jpg">
        <table width="90%" border="0" align="center" cellpadding="0"
cellspacing="0">
            <tr>
                <td bgcolor="#FFFF99"><?php

```



```

                                $objcontenido-
>objetos_seccion($seccionp,'t',$varsend,$system);
                                ?></td>
                                </tr>
                                </table></td>
                                <td></td>
                                </tr>
                                <tr>
                                <td colspan="6"></td>
                                <td></td>
                                </tr>
                                </table></td>
                                </tr>
                                <tr>
                                <td      background="<?php      echo      $objtemplatep->path_template
?>images/fondoc.jpg"><?php

switch ($secc)
{
case 1:
{
        $objcontenido->select_articulo($ar);
        include($objtemplatep->path_template."contenido.php");
    }
    break;
case 2:
{
        include($objtemplatep->path_template."buscar.php");
    }
    break;
case 5:
{
        include($objtemplatep->path_template."registro.php");
    }
    break;
case 6:
{
    if ($sessid)
    {
        $sqlf = "select * from ali_userweb where usw_alias like
'$objacceso_session->sess_name'";
        //echo $sql;
        $resultf = mysql_query($sqlf);
        //VERIFICACION DE USUARIOS REPETIDOS MEDIANTE $RESULT
        if ($resultf)

```

```

        {
            while($rowf = mysql_fetch_array($resultf))
            {
                $actvf=$rowf["usw_estado"];
            }
        }

    if ($actvf==1)
    {
        include($objtemplatep->path_template."foro.php");
    }
    else
    {
        echo "<br><br><br><br><center><div class=titulo>Para
ingresar a esta seccion debe estar registrado como funcionario de
Swisscontac</div>";
        echo "<a href='indexr.php?secc=5&menu=2' target='_top'></a></center><br><br><br><br>";
    }
}
else
{
    echo "<br><br><br><br><center><div class=titulo>Para ingresar a esta
seccion debe estar registrado</div>";
    echo "<a href='indexr.php?secc=5&menu=2' target='_top'></a></center><br><br><br><br>";
}
}
break;
case 7:
{
    //Aplicaciones
    include("module/aplication.php");
    //include($objtemplatep->path_template."registroconsul.php");
}
break;

    case 8:
    {
        //propiedades
        include("propiedades/detalles.php");
    }
    break;

    case 9:

```

```

    {
        //Buscar Propiedad
        include("propiedades/busqueda.php");
    }
    break;

    case 10:
    {
        //Buscar Propiedad
        include("propiedades/index.php");
    }
    break;

    default:
        include($objtemplatep->path_template."centro.php");
    }
?></td>
</tr>
<tr>
    <td bgcolor="#FFFFFF"><p align="center" class="titulo"></p></td>
</tr>
<tr>
    <td>&nbsp;</td>
</tr>
</table>
</body>
</html>

```

MANUAL DE USUARIO

CONTENIDO

1.	MANUAL DE INSTALACIÓN	98
1.1	XAMPP 1.4.15.....	99
2.	CMS ALIANZA	107
2.1	ADMINISTRACIÓN DE USUARIOS.....	108
2.2	ADMINISTRACIÓN DE PERFILES	111
2.3	ADMINISTRACIÓN DE CLIENTES	112
2.4	ADMINISTRACIÓN DE APLICACIONES	113
2.5	CONEXIÓN.....	114
2.6	ADMINISTRACIÓN DE PORTALES	114
2.6	CONSULTA DE SALDOS	119

1. MANUAL DE INSTALACIÓN

1.1 XAMPP 1.4.15

En el presente manual se detalla la instalación del Software utilizado en el desarrollo del CMS para la Cooperativa de Ahorro y Crédito Alianza del Valle, XAMPP 1.4.15 es un Software libre que contiene varias aplicaciones como son:

- Apache HTTPD 2.0.54,
- MySQL 4.1.13,
- PHP 5.0.4 + 4.4.0 + PEAR + Switch, MiniPerl 5.8.7, mod_ssl 2.0.54,
- PHPMyAdmin 2.6.3
- FileZilla FTP Server 0.9.8c
- Openssl 0.9.8,

Pasos para la instalación:

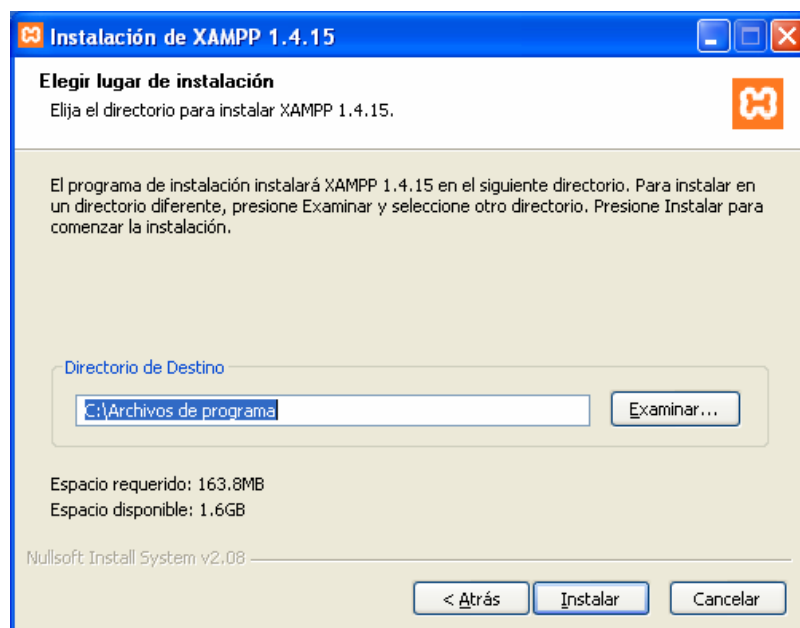
1.- Ejecutar el archivo xampp-win32-1.4.15-installer.exe, el mismo que presentará la siguiente pantalla:



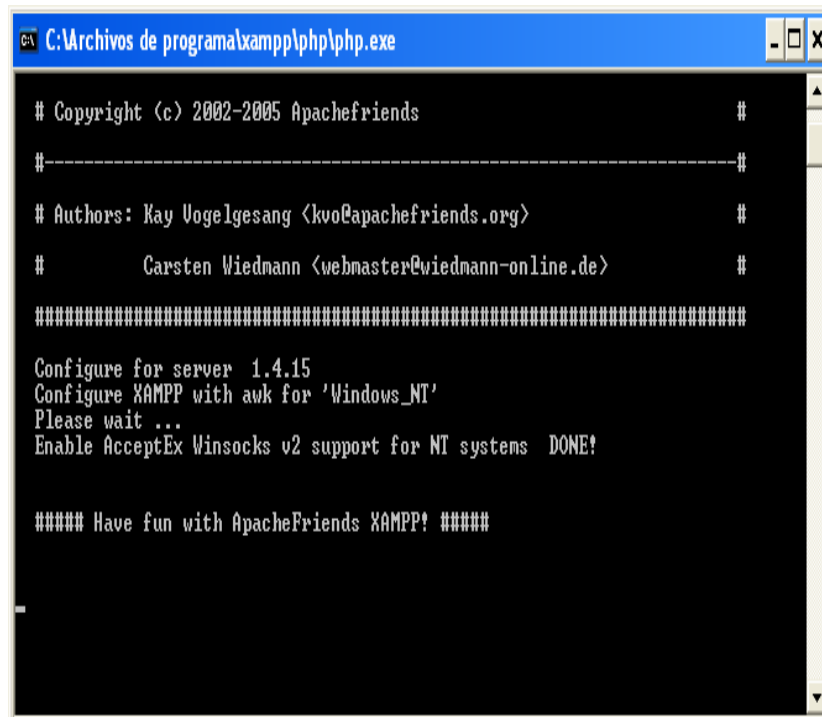
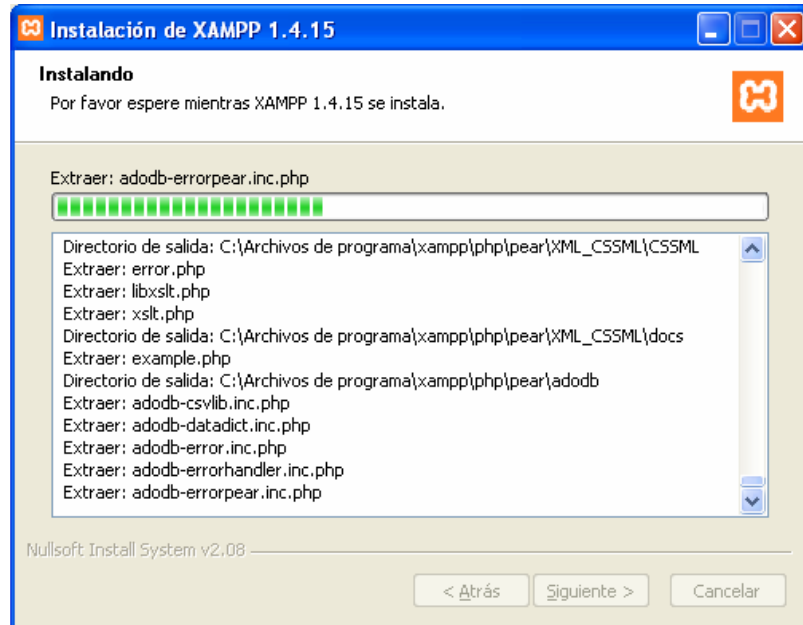
2.- Aquí escogemos “OK” el mismo que nos llevara a la siguiente pantalla:

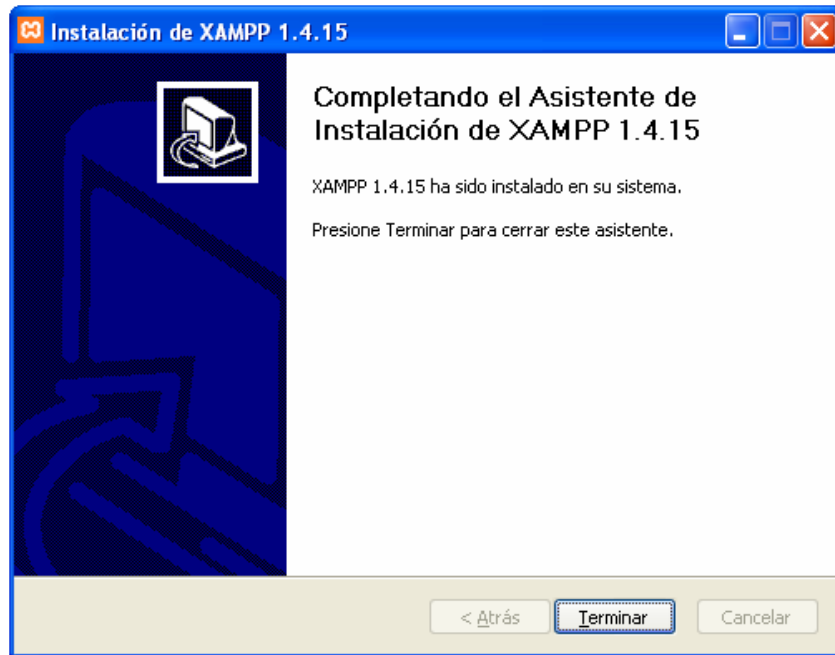


3.- Seleccionamos “siguiente” y presenta la pantalla donde se elije el directorio donde queremos que se instale, es recomendable dejar la carpeta predeterminada, caso contrario le indicamos el directorio de instalación.

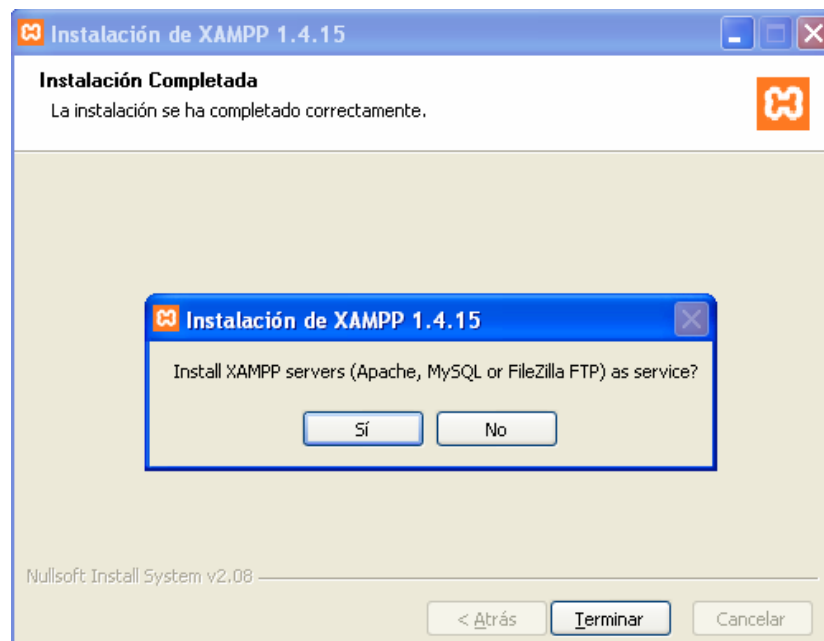


4.- Escogemos “instalar”, y procederá a la instalación de los paquetes mencionados:

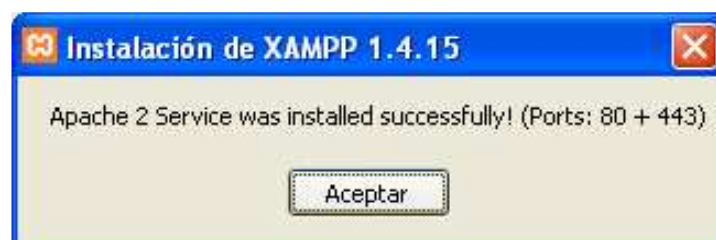
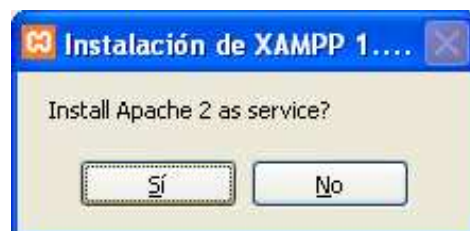
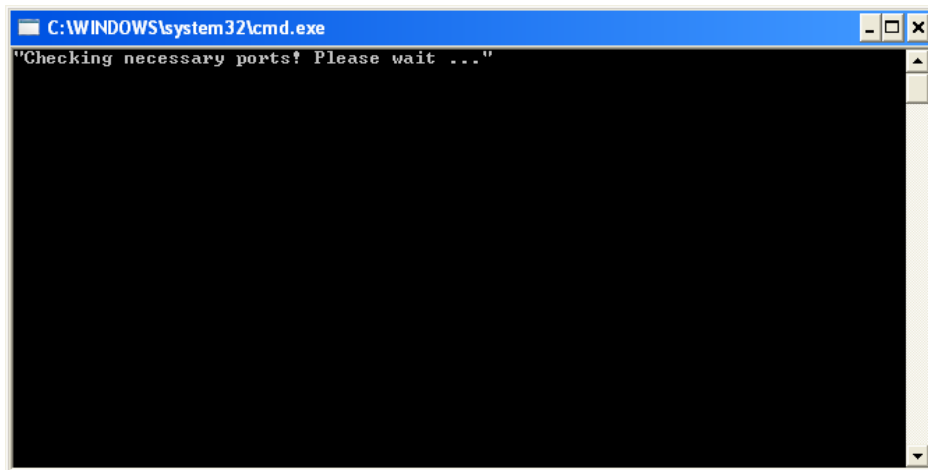




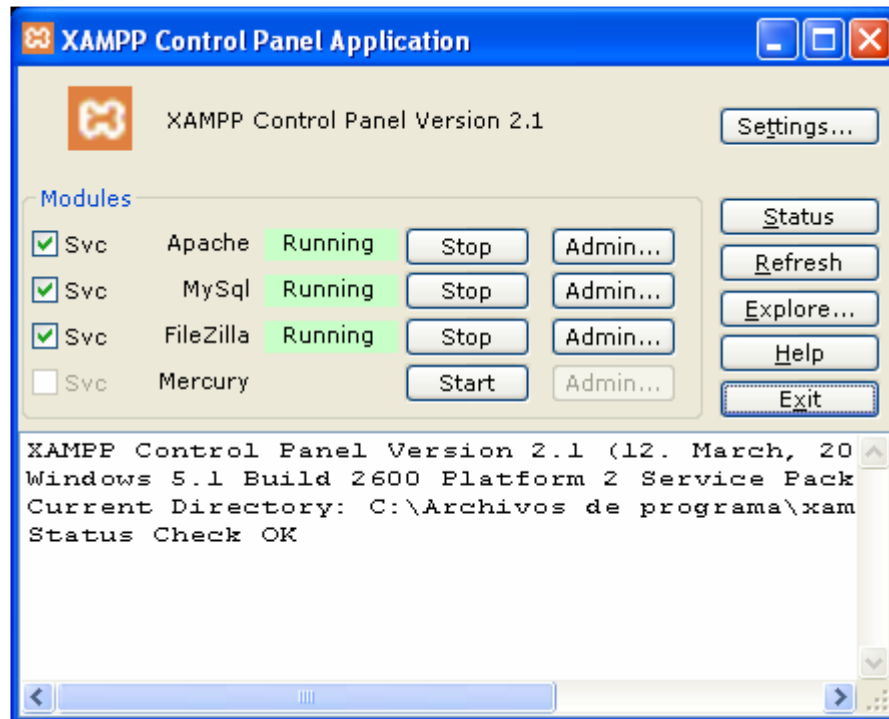
5.- Una vez que escogemos "Terminar", presentara las siguientes pantallas de mensajes, los mismos que permiten levantar las aplicaciones instaladas.



6.- Estas ventanas se muestran para confirmar que servicios, se requiere levantar, por lo tanto para cada aplicación presentara su mensaje respectivo.



6.- Una vez instalada la aplicación nos presenta la siguiente pantalla para indicarnos que aplicaciones están siendo corridas en ese momento.



Con esto se concluye la instalación de los aplicativos necesarios para el buen funcionamiento de nuestro sitio, por lo tanto podemos configura la base de datos "alianzadb" mediante el PHPMYADMIN, la misma que utilizaremos en el desarrollo del portal.

127.0.0.1 >> localhost | phpMyAdmin 2.6.3-pl1 - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

My Web Search Buscar Dirección <http://127.0.0.1/phpmyadmin/>

127.0.0.1 >> localho...

Bienvenido a phpMyAdmin 2.6.3-pl1

MySQL 4.1.13-nt ejecutándose en localhost como root@localhost

phpMyAdmin

Base de datos: (Bases de datos) ...

Seleccione una base de datos

MySQL

Crear nueva base de datos Collation

Mostrar información de marcha de MySQL

Mostrar las variables del sistema MySQL

Mostrar procesos

Juego de caracteres y sus cotejamientos (collations)

Motores de almacenamiento

Reinicio de MySQL

Privilegios

Bases de datos

Exportar

phpMyAdmin

Language: Spanish (es-utf-8)

Juegos de caracteres de MySQL: UTF-8 Unicode (utf8)

Cotejamiento (collation) de las conexiones MySQL: utf8_general_ci

Tema / Estilo: XAMPP (ApacheFriends.org)

Documentación de phpMyAdmin

Mostrar información de PHP

Página oficial de phpMyAdmin

[ChangeLog] [CVS] [Lists]

! Su archivo de configuración contiene parámetros (root sin contraseña) que corresponden a la cuenta privilegiada predeterminada de MySQL. Su servidor de MySQL está usando estos valores, que constituyen una vulnerabilidad. Se le recomienda corregir esta brecha de seguridad.

Inicio COBIS HELIOT (G) Sin título - ... 2 Windo... manual usr... WinFutura... 127.0.0.1 ... http://lsta... 12:35

127.0.0.1 >> localhost >> alianzadb | phpMyAdmin 2.6.3-pl1 - Microsoft Internet Exp

Archivo Edición Ver Favoritos Herramientas Ayuda

My Web Search Buscar Dirección <http://127.0.0.1/phpmy...>

127.0.0.1 >> localho...

Servidor: localhost Base de datos: alianzadb

Estructura SQL Exportar Buscar Genera

Base de datos alianzadb se creó.

consulta SQL:

CREATE DATABASE `alianzadb`;

[Editar] [Crear código PHP]

Base de datos: alianzadb (-)

alianzadb

No se han encontrado tablas en la base de datos.

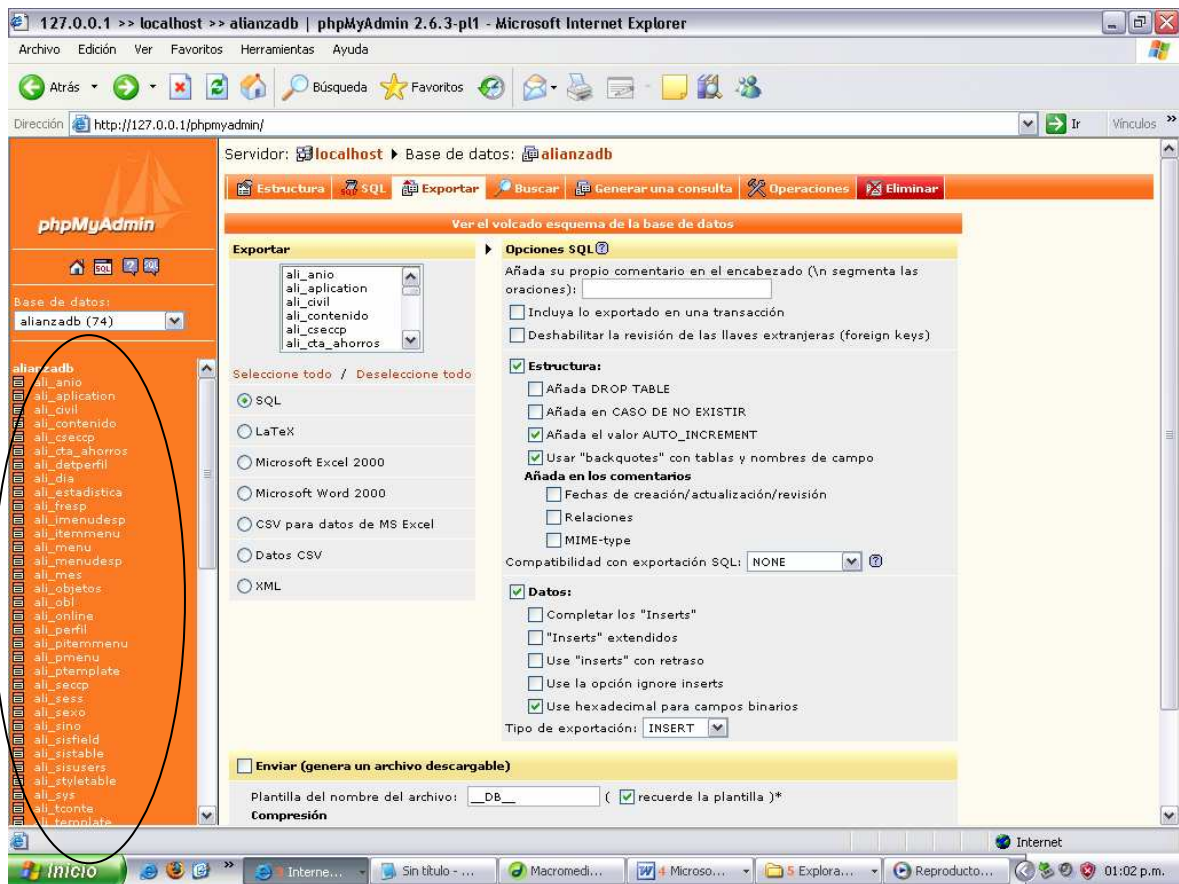
Crear nueva tabla en la base de datos alianzadb:

Nombre:

Campos:

Continúe

Una vez creada la base de datos, levantamos las tablas mediante el SQL de esta misma venta, los cuales utilizaremos para el CMS de la Cooperativa de ahorro y crédito Alianza del Valle. Las tablas levantadas se muestran a lado derecho de la ventana.

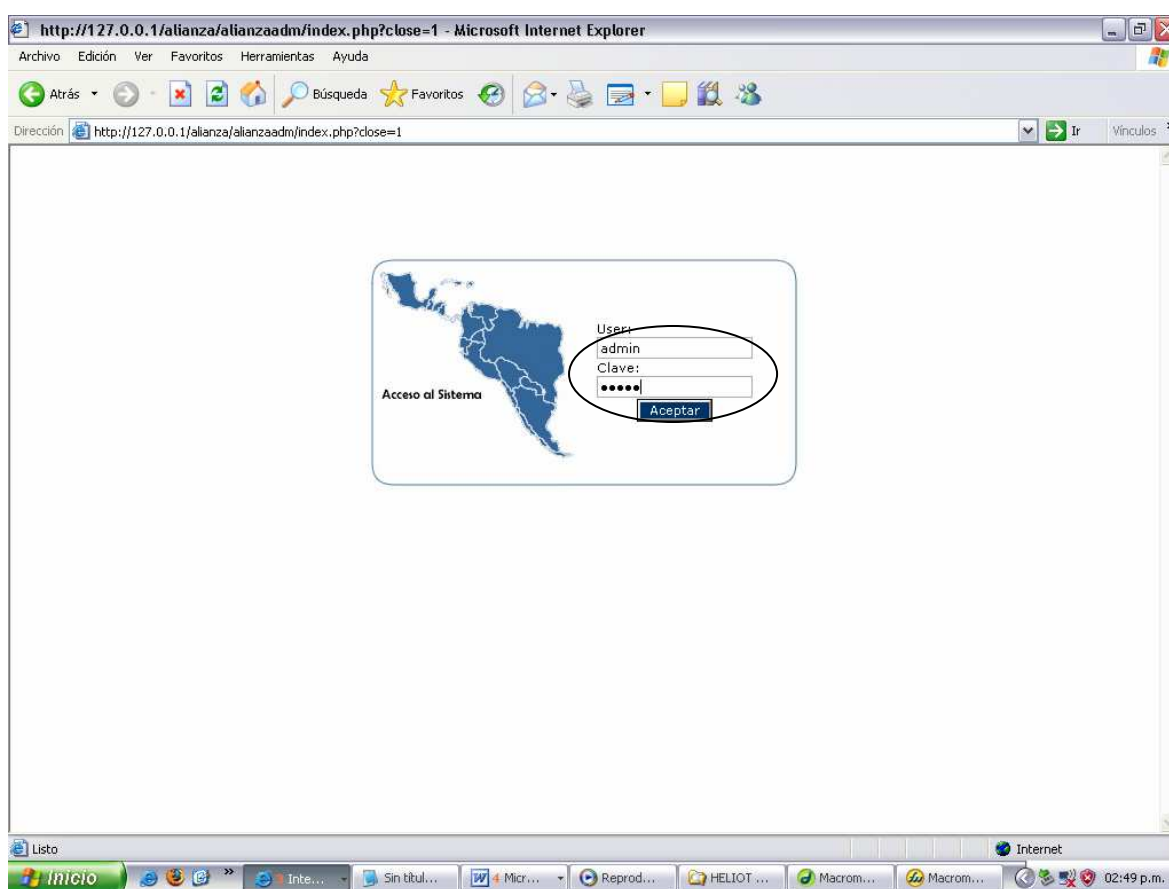


2. CMS ALIANZA

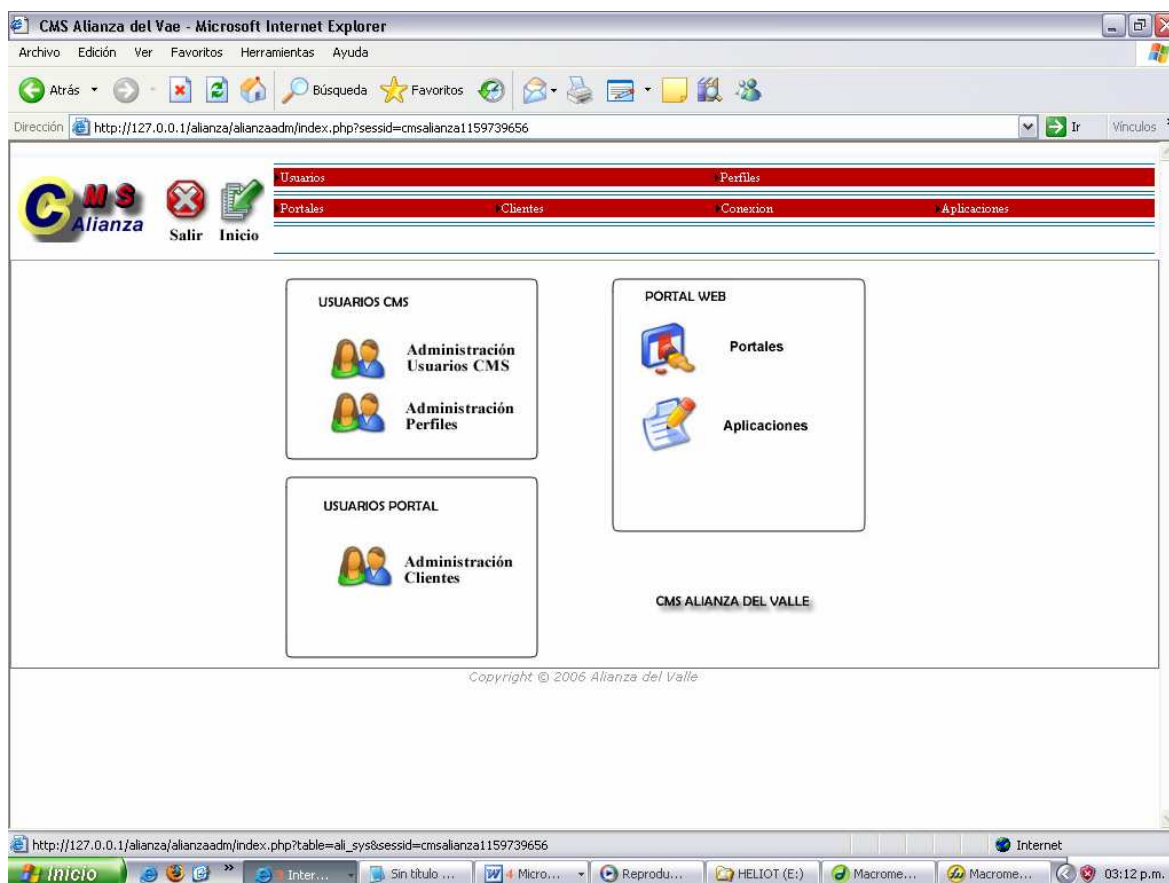
2.1 ADMINISTRACIÓN DE USUARIOS

En el presente manual se detallan las características principales del portal Web para la Cooperativa alianza del Valle, en este punto guiaremos las utilidades de un administrador de contenidos.

1.- Pantalla de ingreso al CMS Alianza del Valle, en el cual solo el usuarios admin, tendrá los accesos totales a la aplicación, si el usuario es diferente a este, solo tendrá acceso a realizar cambios del portal web.



1.- Ya dentro de CMS Alianza del Valle, tenemos para el usuario “admin” el menu siguientes:



1. **Usuarios:** Por donde se realizara la administración de los usuarios del CMS Alianza del Valle, los mismos que tendrán accesos restringidos.
2. **Perfiles:** En esta parte se puede crear los perfiles para cada usuario, por recomendación y agilidad solo existirán dos usuarios que son: “admin” y “usuario”.
3. **Portales:** Esta es la aplicación que realiza la administración de contenidos.
4. **Clientes:** por esta ventana se realizara la administración de CLIENTES que tendrán acceso a la consulta de su cuenta de ahorros.
5. **Conexión:** esta ventana permite la conexión con la base de datos de la Cooperativa de ahorro y crédito Alianza del Valle, la misma que no se encuentra en funcionamiento ya que el acceso lo mantiene restringido.

6. **Aplicaciones:** esta ventana nos permite administrar la aplicación Consulta de Saldos, por lo tanto será muy útil, si la aplicación no quiere ser mostrada ya sea esta por actualización de la base de datos

Para realizar la administración de los usuarios del CMS Alianza del Valle, nos dirigimos por el link Usuarios, el mismo que desplegara la siguiente ventana, en la cual tenemos las siguientes opciones.

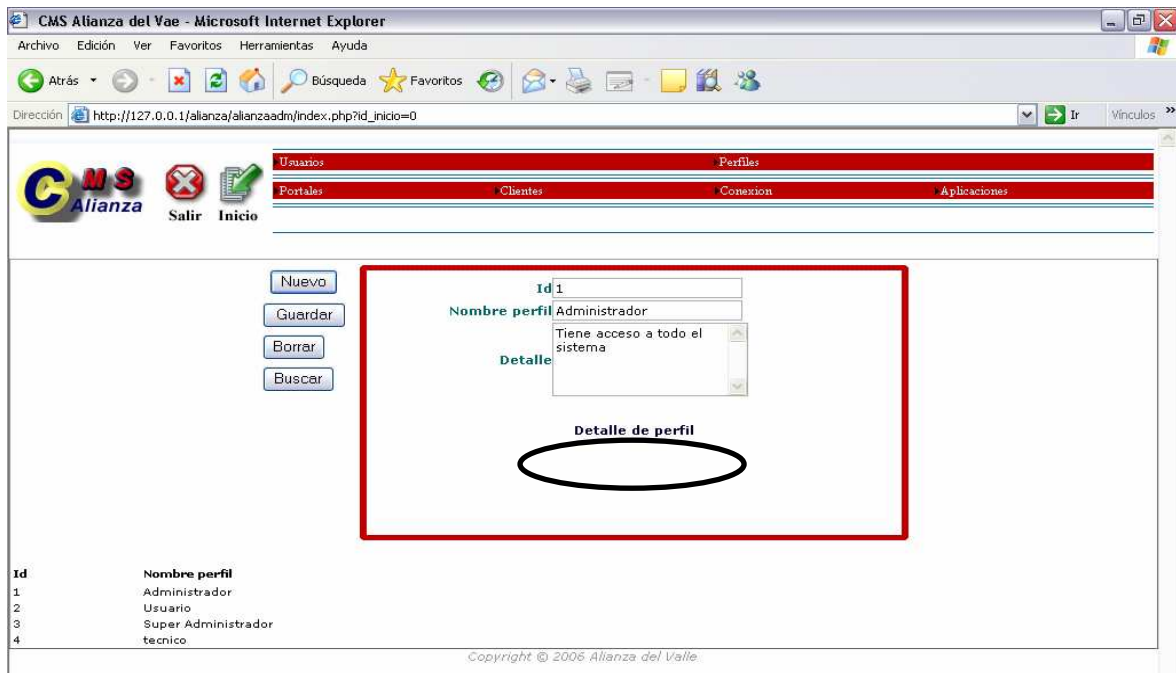
1. **Nuevo:** Limpia la pantalla que se está mostrando en ese momento
2. **Guardar:** Guarda en la base de datos la información ingresada.
3. **Borrar:** Elimina la información de la base de datos mostrada en la ventana actual.
4. **Buscar:** Realiza búsquedas en la base de datos dependiendo de los parámetros enviados.

Usuario ID	Usuario	Nombres y apellidos	Email	Sistema	Perfil
1	admin	Rene Gualotuña	simarene@hotmail.com	10	Administrador
3	guess	Franklin	frank@labibliadice.org	1	Usuario
4	sadmin	Super usuario	super@hotmail.com	1	Super Administrador
5	alfredoo	Alfredo Oña	aona@alianzadelvalle.fin.ec	10	tecnico

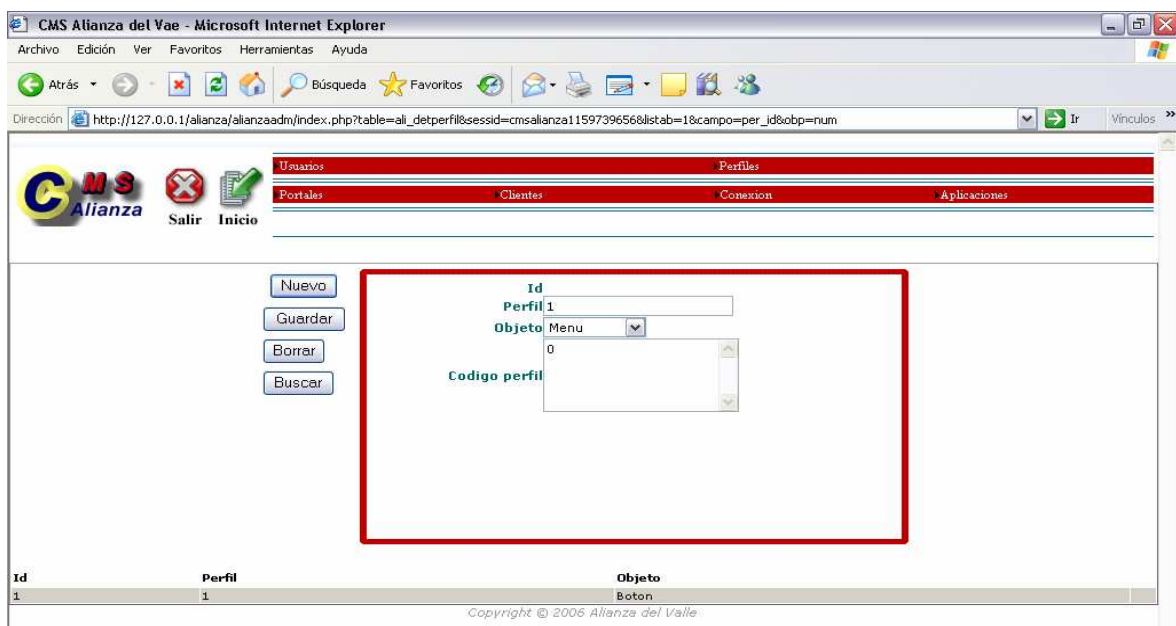
Copyright © 2006 Alianza del Valle

2.2 ADMINISTRACIÓN DE PERFILES

Mediante el Link indicada nos permite ingresa nuevos perfiles para el CMS Alianza del Valle, los mismo que serán analizados antes de crearlos, la administración solo lo realizara el usuario “admin”, además se recomienda mantener los perfiles ya expuestos, la pantalla de administración de perfiles es la siguiente:



Por lo tanto cada perfil tendrá su detalle de perfil:



2.3 ADMINISTRACIÓN DE CLIENTES

Mediante el Link ingresamos a la pantalla de clientes, esta actividad solo lo realiza el usuario Administrador, por esta ventana registraremos a los CLIENTES quienes tendrán acceso a ver su estado de cuenta desde el exterior.

The screenshot shows a web browser window titled 'CMS Alianza del Vae - Microsoft Internet Explorer'. The address bar displays the URL: `http://127.0.0.1/alianza/alianzaadm/index.php?table=mas_usuarios&sessid=cmsalianza1159739656`. The page features a navigation menu with links for 'Usuarios', 'Perfiles', 'Portales', 'Clientes', 'Conexion', and 'Aplicaciones'. Below the menu, there are buttons for 'Nuevo', 'Guardar', 'Borrar', and 'Buscar'. The main content area is titled 'DATOS PERSONALES' and contains the following form fields:

- Login (Cédula de Identidad o RUC): [Text Input]
- Password: [Text Input]
- Confirmación Password: [Text Input]
- Nombre o Razón Social: [Text Input]
- Ciudad: [Text Input]
- Dirección: [Text Input]
- Teléfono: [Text Input]
- email: [Text Input]

At the bottom of the page, there is a copyright notice: 'Copyright © 2006 Alianza del Valle'.

Este punto hay que recalcar que para el ingreso de un nuevo cliente se debe realizar una búsqueda para determinar si el cliente fue registrado, por la siguiente pantalla:

The screenshot shows a dialog box titled 'Search... - Diálogo Web' with the question '¿Ingrese el campo que desea buscar?'. The dialog contains a list of search criteria with checkboxes and corresponding text input fields:

- Id: Texto de búsqueda
- Login (Cédula de Identidad o RUC): 1716194046
- Nombre o Razón Social: Texto de búsqueda
- Ciudad: Texto de búsqueda
- Dirección: Texto de búsqueda
- Teléfono: Texto de búsqueda
- email: Texto de búsqueda

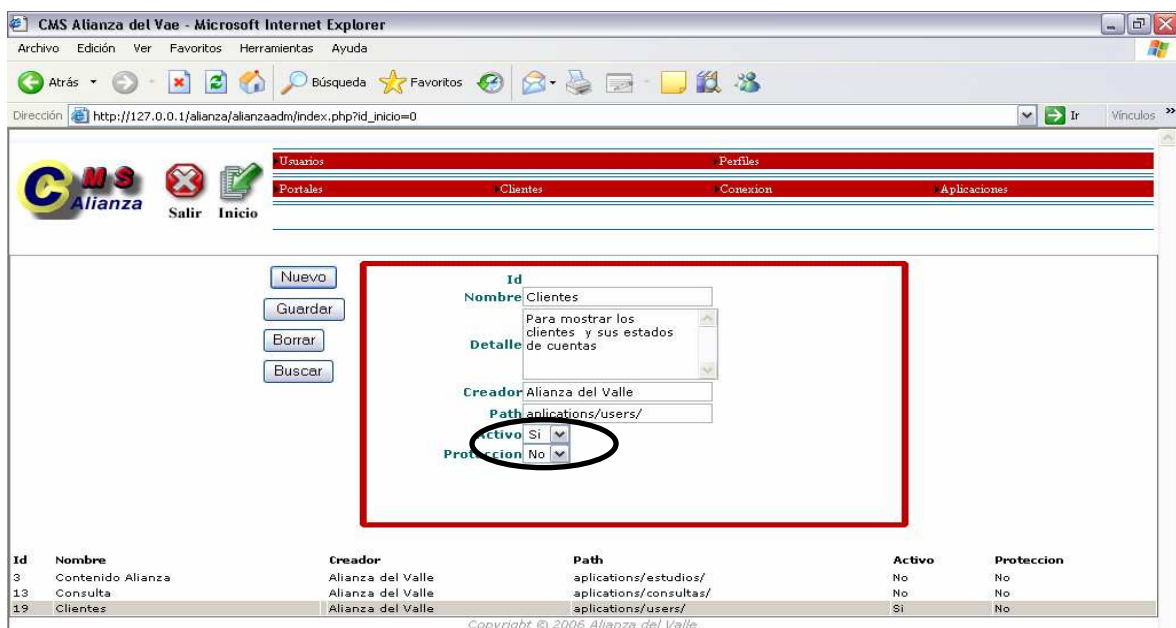
At the bottom of the dialog, there are 'Aceptar' and 'Cancelar' buttons. The status bar at the bottom shows the URL: `http://127.0.0.1/alianza/alianzaadm/bdialog/search.php?busq=1&table=mas`.

Si el cliente fue encontrado presenta la información con sus datos personales, como se muestra en la siguiente pantalla:



2.4 ADMINISTRACIÓN DE APLICACIONES

Mediante el Link ingresamos a la pantalla de aplicaciones, este actividad solo lo realizara el usuario Administrador, habilitamos o deshabilitamos la aplicación consulta de saldos.



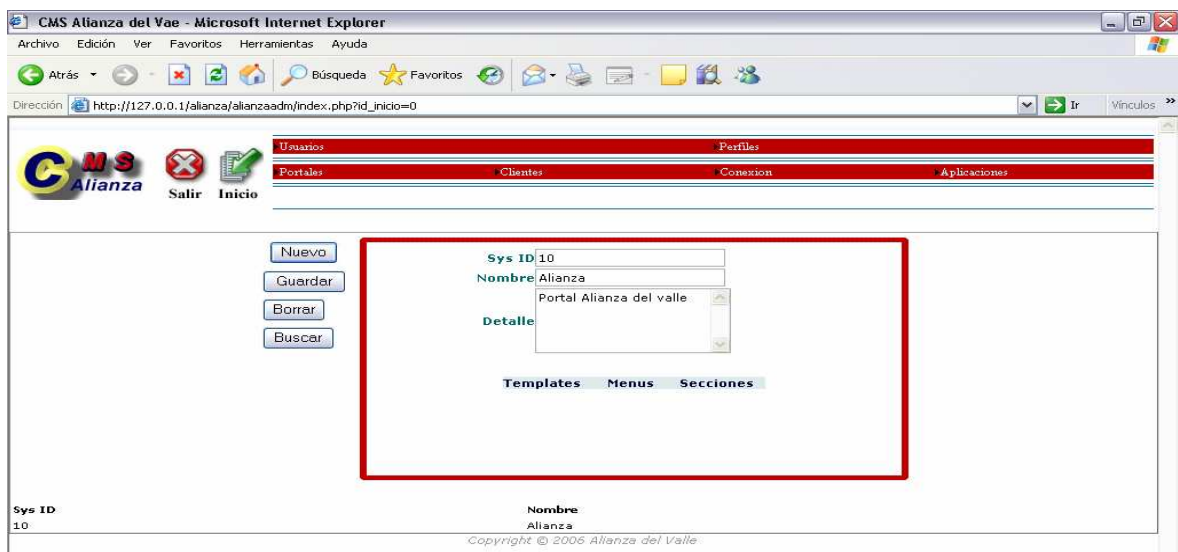
2.5 CONEXIÓN

Mediante el Link ingresamos a la pantalla de conexiones, este realizara el enlace con la base de datos SYBASE, que actualmente maneja la Cooperativa de ahorro y crédito Alianza del Valle, en este momento esta aplicación no se encuentra habilitada ya que la cooperativa no permitió el acceso a su base de datos.



2.6 ADMINISTRACIÓN DE PORTALES

Mediante el Link ingresamos a la pantalla de portales, en este punto cabe recalcar que tendrán acceso el ADMINISTRADOR y el USUARIO del CMS Alianza del Valle.



La administración del portal tiene los siguientes puntos,

Templates: Es la ventana don activamos el template del portal Web.

ID Template	Sistema/Portal	Nombre	Autor	URL	Active
14	Alianza	AlianzaTemplate	Rene Gualotuña	www.alianzadelvalle.com	1

Copyright © 2006 Alianza del Valle

Quiénes Somos
Nuestros Productos
Nuestros Servicios
Beneficios
Agencias
Notialianza
LOGIN

www.alianzadelvalle.fin.ec
BIENVENIDOS
AL PORTAL WEB DE LA
COOPERATIVA DE AHORRO Y CREDITO
Alianza del Valle
Valle de los Chillos,Chauptena,
Antiguavía Conocoto-Amaguaña
Telfs : 2339-432 ; 2332-085
QUITO - ECUADOR

SOLIDEZ
SEGURIDAD
CONFIANZA

Menus: Es la ventana donde se crea los menús para el portal web

CMS Alianza del Vae - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Atrás Búsqueda Favoritos

Dirección http://127.0.0.1/alianza/alianzaadm/index.php?table=ali_pmenu&sessid=cmsalianza1159739656&lstab=10&campo=sys_id&obp=num

Usozios Perfiles
Portales Clientes Conexion Aplicaciones

Salir Inicio

Nuevo Guardar Borrar Buscar

ID
 Sistema/Portal
 Menu nombre
 Style
 Tipo menu
 Activo

ID	Sistema/Portal	Menu nombre	Style	Tipo menu	Activo
29	Alianza	Notalianza	txtmenu	1	1
28	Alianza	AServicios	txtmenu	1	1
27	Alianza	Beneficios	txtmenu	1	1
26	Alianza	Nproductos	txtmenu	1	1
25	Alianza	Qsomos	txtmenu	1	1
24	Alianza	prindpal	txtmenu	2	1
21	Alianza	NServicios	txtmenu	1	1

Copyright © 2006 Alianza del Valle

Listo Internet

Inicio Inte... Sin tbul... Reprod... Macrom... Macrom... TESIS MANUA... 04:55 p.m.

Inicio Mapa Contáctanos <http://www.allanzadelvalle.fin.ec>

Alianza del Valle Ltda.
Cooperativa de Ahorro y Crédito

Su cooperativa amiga!

Quienes Somos
 Nuestros Productos
 Nuestros Servicios
 Beneficios
 Agencias
 Notalianza
 LOGIN

Mision, Vision // Principios De Gestion //

QUIENES SOMOS

Internet

Inicio... 04:47 p.m.

Secciones: Es la ventana donde se crea las secciones del menú para el portal web y este a su vez tiene las siguientes partes:

ID seccion	Sistema/Portal	Etiqueta
40	Alianza	Menu Notialianza
39	Alianza	Menu Agencias
38	Alianza	Menu Beneficios
37	Alianza	Menu Nservicios
36	Alianza	Menu Nproductos
35	Alianza	Menu Qsomos
34	Alianza	HomeAlianza
33	Alianza	Servicios

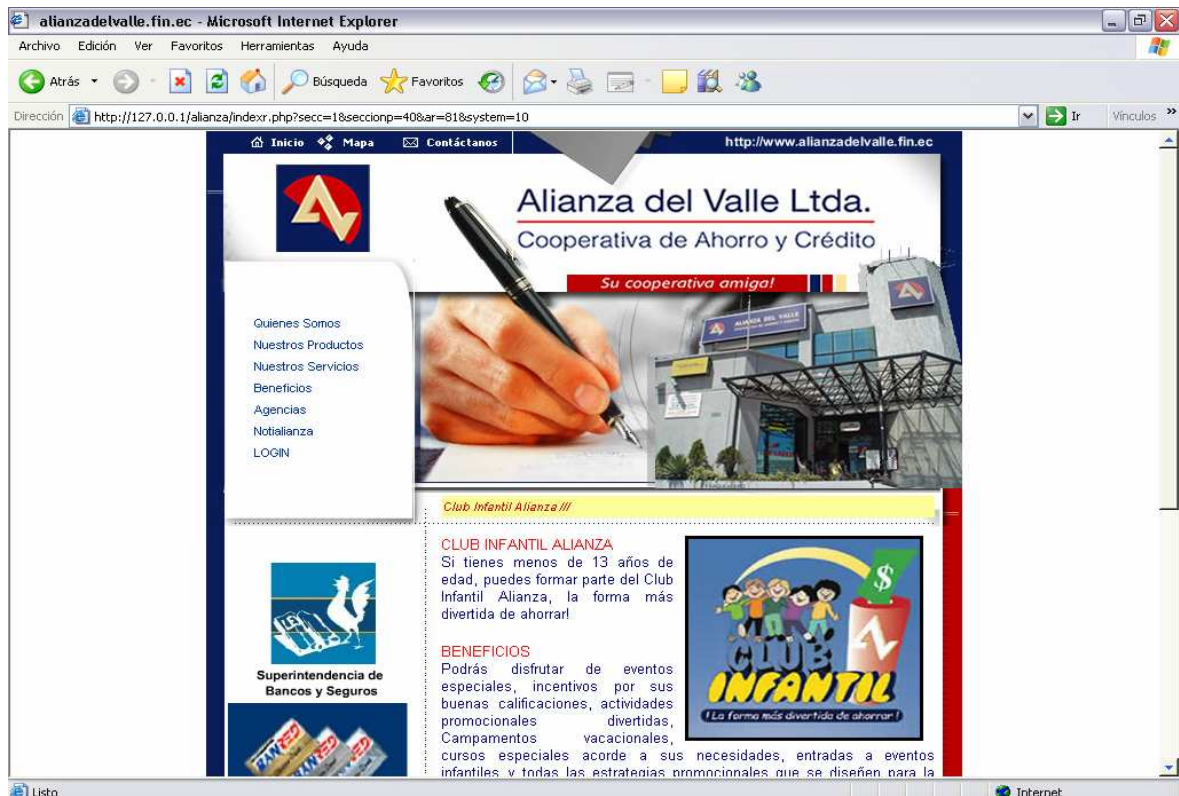
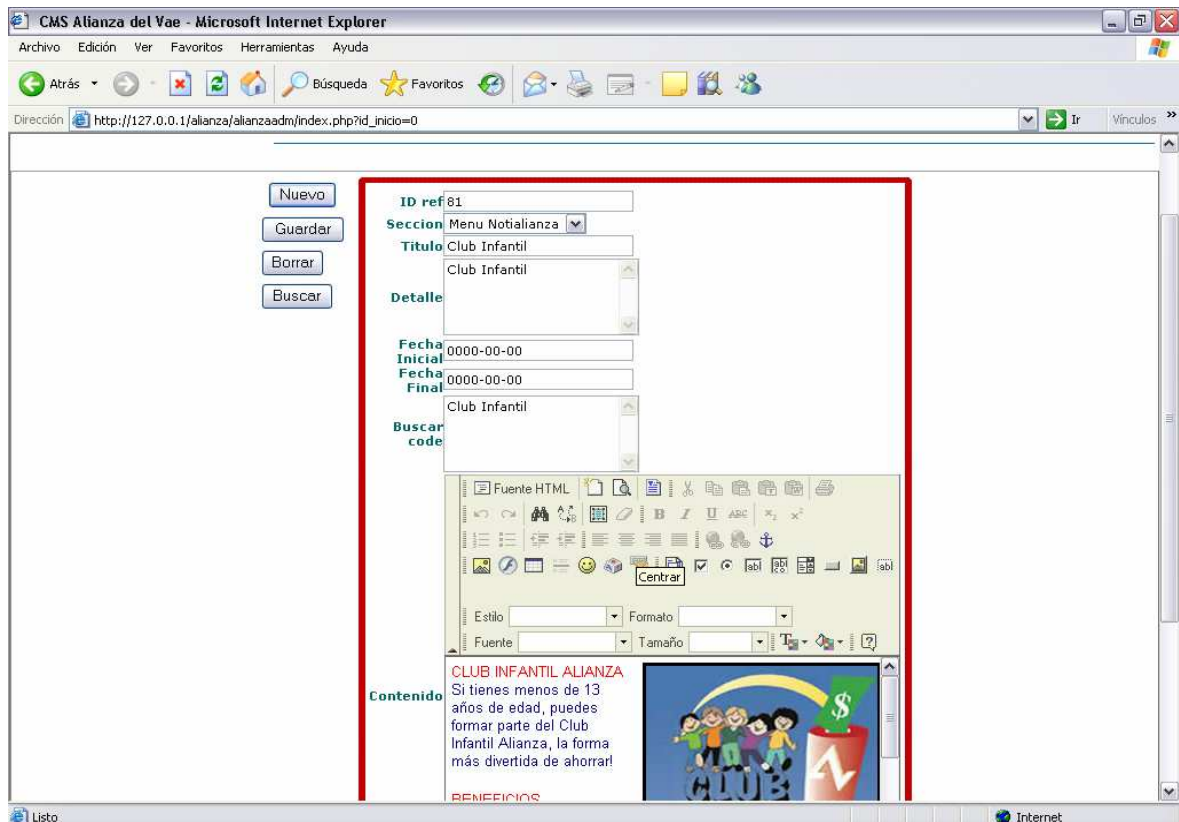
Copyright © 2006 Alianza del Valle

Partes de la sección: Es la ventana determina las secciones que tendrán los menus.

ID	Seccion	Tipo	Menu
39	HomeAlianza	art	29
38	HomeAlianza	men	24

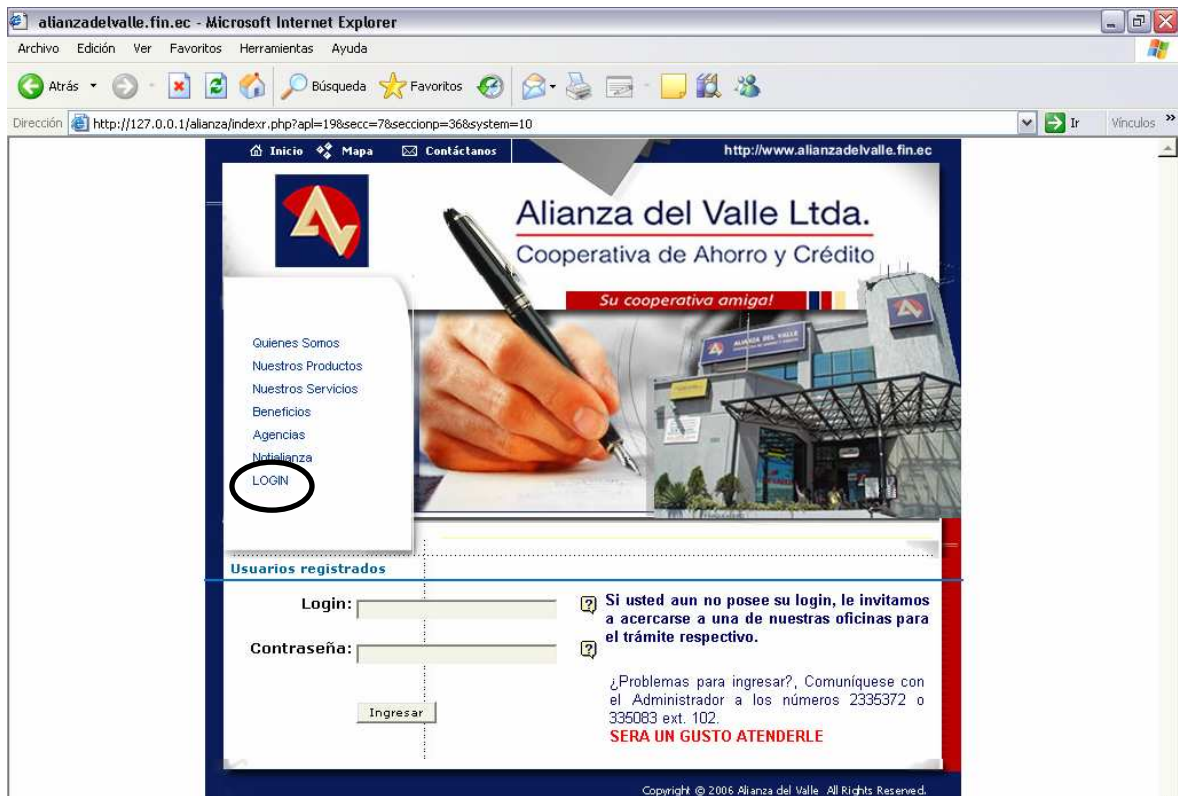
Copyright © 2006 Alianza del Valle

Contenido Portal por Sección : Es la ventana donde se crea el contenido de cada sección, mostrada en el portal.

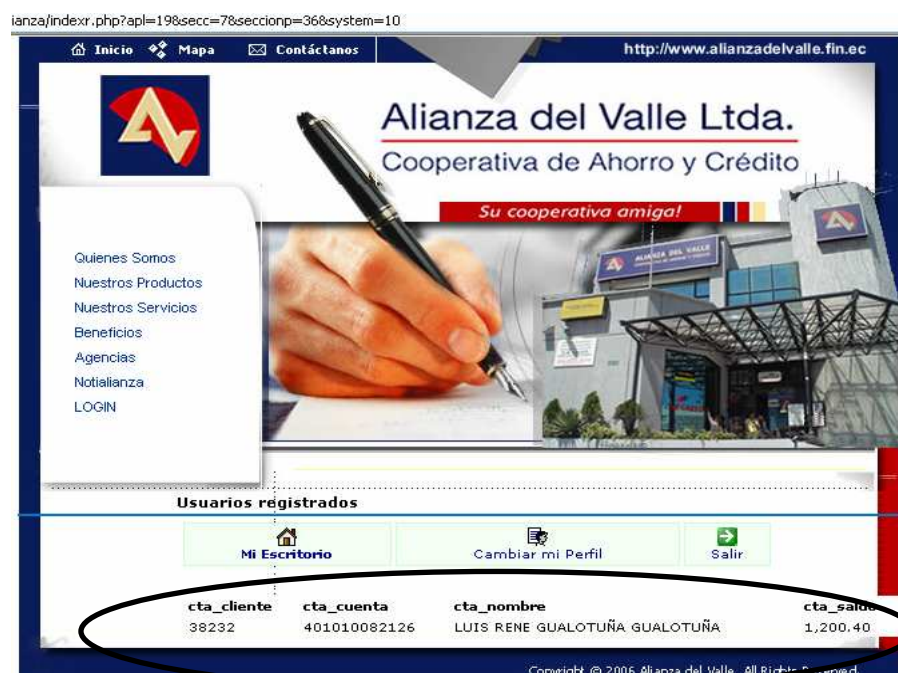


2.6 CONSULTA DE SALDOS

Mediante el Link LOGIN ingresamos a la pantalla de consulta de saldos, para el mismo necesitamos la cedula de identidad y su contraseña.



Una vez validado su información, nos muestra el estado de nuestra cuenta.



Con el link Cambiar mi perfil, se puede modificar la información del cliente

Usuarios registrados

Mi Escritorio **Cambiar mi Perfil** Salir

Cambiando el perfil de LUIS RENE GUALOTUÑA GUALOTUÑA

Login (Cédula de Identidad o RUC) 1716194046

Nombre o Razón Social: LUIS RENE GUALOTUÑA GUALOTUÑA

Dirección: AMAGUAÑA

Ciudad: QUITO

Teléfono: 2879509

Email: simarene@hotmail.com

Password: *****

Confirmación Password: *****

Actualizar Datos

La opción salir nos permite cerrar nuestra sesión de consulta de saldos.

alianzadelvalle.fin.ec - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Búsqueda Favoritos

Dirección http://127.0.0.1/alianza/indexr.php?system=10&sessid=8&apl=19&seccapl=2&secc=7

Inicio Mapa Contáctanos http://www.alianzadelvalle.fin.ec

Alianza del Valle Ltda.
Cooperativa de Ahorro y Crédito

Su cooperativa amiga!

Quiénes Somos
Nuestros Productos
Nuestros Servicios
Beneficios
Agencias
Notialianza
LOGIN

Usuarios registrados

Gracias por su apoyo es un gusto tenerlo en nuestro equipo.
Usted ha salido del sistema

Inicio

Copyright © 2006 Alianza del Valle All Rights Reserved.

Inicio Int... Sin títul... Reprod... Macrom... Macrom... TESIS MANUA... 05:13 p.m.