



Universidad de las Ciencias
Informáticas

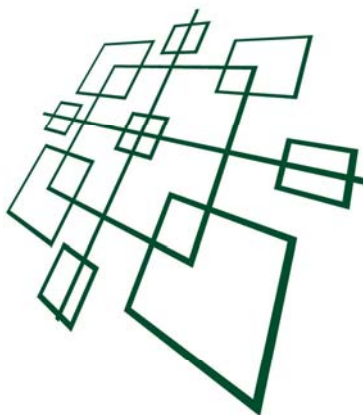
Dirección de Informatización



ESTUDIO DE ALTERNATIVAS PARA LA MIGRACIÓN DEL SISTEMA AUTOMATIZADO PARA LA GESTIÓN ACADÉMICA “AKADEMOS” A SOFTWARE LIBRE

Trabajo de diploma para optar por el título de:

**Ingeniero en
Ciencias Informáticas**



**Autores: Frank Benavides Dalmendray
Darien Cepero Rojas**

Tutor: Ing. Yunier Saborit Ramírez

Ciudad de la Habana, Mayo 2007

A:

Mi Papá, mi Mamá y a mi Familia.

Frank

A:

Mi Papá, mi Mamá y a mi Familia.

Darien

Agradecimientos comunes

A la revolución por darnos la oportunidad de forjarnos como ingenieros en la primera universidad surgida al calor de la batalla de ideas.

A nuestro tutor Yunier Saborit por su apoyo en nuestro trabajo de tesis.

A Emil Lima Valdés, por su apoyo incondicional y su gran aporte a nuestra formación como profesionales.

A Rosmel por su ayuda en la revisión del documento y la migración del sistema.

A Orlando y Yenier por su ayuda en la migración del sistema.

A los profesores de nuestra facultad por su contribución a nuestra formación.

A Maria Antonia, Maye, Fifi, Mayte, Vicky y al personal de secretaría que tanta ayuda e ideas nos han dado.

A los miembros del grupo de desarrollo de Akademos: Camejo, Andry, Dianly (El loco), Roberton, Griselita, Norges, Elisabel, Noel, Olivia, Molina, Yaniris, Joe, Pilciner, Acralys, Yudeisy, Catherine, Alexey.

A Nilia Ana, Dayany, Liuris, Keyttia y Manuel Alejandro por su ayuda en la conformación de este documento.

A las comunidades de desarrollo de software libre por la información y la ayuda que nos brindaron en todo momento.

Agradezco a:

Mi papá y a mi mamá por ser mi mayor ejemplo en todos los aspectos de la vida y por estar siempre dispuestos a ayudarme y apoyarme; si no fuera por ellos no sería ni la mitad de lo mucho o lo poco que soy hoy en día.

Mis abuelos y a mi hermano por poder contar siempre con ellos y brindarme su apoyo incondicional.

Maidely, “bonita mía”, por ser una de las personas más especiales y más importantes que han aparecido en mi vida.

Emil, quién supo ayudarme en mi formación como estudiante y profesional.

Camejo, por ser el amigo incondicional que todos desean tener y por enseñarme que aún existen personas desinteresadas, honestas, firmes y de buen corazón; y a Yaniris por ser como Camejo, sin necesidad de ser su reflejo... ellos me entienden.

Loryen, por haberme enseñado tanto y haber compartido una parte de su vida conmigo.

Pepe, mi amigo, mi compañero de trabajo y de lucha en el proyecto.

Mayuli, por ser una amiga leal y estar siempre dispuesta a ayudarme.

Todo el que, de una forma u otra, puso su granito de arena para poder cumplir esta meta.

Frank

Agradezco a:

Mis padres, por el amor que me han entregado y la confianza que han depositado en mí.

Mi Abuela Maria, por el cariño que me ha dado y por lo mucho que me ha malcriado.

Mi tía Gloria y mi tío Omar por su apoyo y por estar a mi lado en los momentos más difíciles y más alegres.

Mi amigo y hermano Emil Lima por su ayuda en todo momento.

Frank, mi amigo y mi compañero de tesis, por su ayuda incondicional desde los primeros días en al UCI, por su apoyo en todo momento.

Mis compañeros (a) de apartamento y de aula por el apoyo que me han brindado y darme lo oportunidad de formar parte de un colectivo tan grandioso.

Toda mi familia, mis tíos y mis tías y mis primos.

Ana por su paciencia y su dedicación en estos días de tanto trabajo e insomnio.

Mis vecinos y amigos por la confianza que han depositado en mí.

Todo el que ha puesto una gota de esperanza y empeño en hacer realidad este sueño.

Darien

Declaración de Autoría

Declaramos que somos únicos autores de este trabajo y autorizamos a la Dirección de Informatización de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmamos la presente a los ____ días del mes de Mayo del año 2007.

Frank Benavides Dalmendray

Darien Cepero Rojas

Yunier Saborit Ramírez

Opinión del tutor del trabajo de diploma

Título: **Estudio de alternativas para la migración del Sistema Automatizado para la Gestión Académica de la Universidad de las Ciencias Informáticas – “Akademos” a Software Libre.**

Autor(es): Frank Benavides Dalmendray; Darien Cepero Rojas.

El tutor del presente Trabajo de Diploma considera que durante su ejecución el estudiante mostró las cualidades que a continuación se detallan.

Por todo lo anteriormente expresado considero que los estudiantes están aptos para ejercer como Ingenieros en Ciencias Informáticas; y propongo que se le otorgue al Trabajo de Diploma la calificación de ____puntos.

____ días del mes de Mayo del año 2007.

Yunier Saborit Ramírez

Resumen

La Universidad de las Ciencias Informáticas, recinto donde se concentra la mayor comunidad informática de Cuba, juega un papel importante en el desarrollo de la informatización de la sociedad cubana. Este centro no está ajeno a las políticas de migración a software libre trazadas por la Oficina de Informatización del Ministerio de la Informática y las Comunicaciones, al encontrarse en un proceso de migración a software libre de todos los sistemas elaborados y asesorados por la Dirección de Informatización de dicho centro.

El presente trabajo, cuyo título es **Estudio de alternativas para la migración del Sistema Automatizado para la Gestión Académica de la Universidad de las Ciencias Informáticas – “Akademos” a Software Libre**, presenta una caracterización de dicho sistema y sus principales prestaciones, la gestión de la seguridad del mismo, se detalla la arquitectura bajo la cual se rigió su elaboración así como la tecnología usada durante su construcción y las herramientas empleadas en su creación. Este trabajo constituye el resultado de una investigación sobre las tecnologías y herramientas que en el mundo del software libre pueden ser empleadas para la migración de aplicaciones Web construidas bajo la plataforma **.NET** y el gestor de base de datos **Microsoft SQL Server 2000**, arrojando una estrategia para la migración de Akademos en el menor tiempo y al menor costo posible.

Este documento puede ser fuente de consulta para la migración de sistemas con similares características, en diverso entornos de trabajo.

Abstract

The University of Informatics Sciences, enclosure where the biggest informatics science community of Cuba concentrates itself, plays an important role in the development of the computerization of Cuban society. This center is not unaware to the policies of migration to free software established by the Office of Computerization of the Ministry of Informatics and Communications, while being in a process of migration to free software of all systems elaborated and advised by the Direction of Computerization of this center.

This paper, which title is **Study of alternatives for the migration of the Automated System for the Academic Management of the University of Informatics Sciences - “Akademos” to Free Software**, presents a characterization of that system and its main benefits, the management of its security, a detailed explanation of the architecture under which its elaboration was governed, as well as the technology used during its construction and the tools used in its creation. This work constitutes the result of an investigation on the technologies and tools that, in the world of free software, can be used for the migration of Web applications constructed under the platform .NET and the Microsoft SQL Server 2000 database manager, establishing a strategy for the migration of Akademos in the shortest time and to the lowest cost possible.

This document can be of great help for the migration of systems with similar characteristics, in diverse work environment.

Índice:

Introducción:	1
Capítulo I: Sistema Automatizado para la Gestión Académica – Akademos	4
1.1 Introducción	4
1.2 La Gestión Académica	4
1.3 Akademos	4
1.4 Principales características.....	5
1.5 Arquitectura de Akademos	5
1.6 Módulos de Akademos.....	6
1.7 Seguridad de Akademos	9
1.8 Aporte social y valoración económica que tiene Akademos en la actualidad.....	10
1.9 Descripción de herramientas empleadas en el desarrollo de Akademos	12
1.9.1 SQL Server	13
1.9.2 Plataforma .NET.....	13
1.9.3 Web Services XML.....	14
1.9.4 Internet Information Services (IIS).....	14
1.10 Alternativas existentes para la migración de sistemas con las características de Akademos a Software Libre	15
1.11 Conclusiones.....	16
Capítulo II: Estudio para la migración	17
2.1 Introducción.....	17
2.2 Posición de Cuba ante el uso y la migración a Software Libre.....	17
2.3 Razones y necesidades para hacer la migración de Akademos	20
2.4 Plataformas Software Libre. Propuesta para la migración.....	23
2.4.1 Caracterización de las Plataformas Web Libres (PWL) más difundidas.....	23
2.4.2 Selección de la Plataforma Web Libre para hacer la migración.	26
2.5 Servidores Web Libres. Propuesta para la migración.....	32

2.5.1	Apache	32
2.5.2	Características más importantes del Servidor Web Apache 2	35
2.5.3	El Servidor Web que se propone para la migración	38
2.6	Servidores de Base de Datos Libres. Propuesta para la migración	39
2.6.1	La Base de Datos de Akademos	39
2.6.2	PostgreSQL.....	40
2.6.3	MySQL	43
2.6.4	El gestor de base de datos que se propone para la migración	46
2.7	Conclusiones.....	47
Capítulo III: Desarrollo de la migración		48
3.1	Introducción.....	48
3.2	Desarrollo de la migración de la capa de presentación	50
3.3	Desarrollo de la migración de la capa de negocio.....	53
3.4	Desarrollo de la migración de la capa de datos.....	57
3.5	Desarrollo de la migración de los Web Services XML.....	61
3.6	Conclusiones.....	61
Conclusiones generales		63
Recomendaciones		64
Bibliografía citada.....		65
Bibliografía consultada		68
Anexos.....		70

Introducción:

La Dirección de Informatización de la Universidad de las Ciencias Informáticas, en la actualidad, asume como una de sus prioridades la migración de todos los sistemas de gestión que permiten la automatización de los procesos de este centro de estudios. La comunidad universitaria dispone de un sistema que ha ayudado a descentralizar los procesos de gestión académica, Akademos, el cual ha sido desarrollado con herramientas privativas y en entornos de desarrollo que no cumplen con las políticas de migración y utilización de software libre de nuestro país. Promoción de la utilización de software libre, así como la investigación y la búsqueda de alternativas para la migración son las metas trazadas por esta dirección y la razón de ser de este trabajo.

Tema:

Estudio de alternativas para la migración del Sistema Automatizado para la Gestión Académica de la Universidad de las Ciencias Informáticas – “Akademos” a Software Libre.

Situación Problemática:

Existen Sistemas Web que han sido desarrollados utilizando la plataforma .NET y el gestor de base de datos SQL Server dada la rapidez y perfección con que son terminados. Dicho proceso no es costeable por Cuba, además que no cumple con las nuevas políticas de migración a Software Libre asumidas por la dirección del país.

La dirección de informatización de la Universidad de las Ciencias Informáticas (UCI) desarrolló el Sistema Automatizado para la Gestión Académica – “Akademos” bajo estas mismas condiciones; cumpliendo con todas las exigencias de la Universidad. Este sistema no cumple con las nuevas políticas de migración a Software Libre y, sin embargo, resulta imprescindible para el buen funcionamiento del proceso docente de la Universidad. Por otra parte no existe una vía

establecida para la migración de sistemas Web de este tipo a Software Libre en la Universidad de las Ciencias Informáticas.

Problema Científico:

¿Cuáles son los pasos y alternativas a seguir para migrar el Sistema Automatizado para la Gestión Académica de la Universidad de las Ciencias Informáticas – “Akademos” a Software Libre?

Objeto de estudio:

Pasos y alternativas para la migración de sistemas Web desarrollados utilizando la plataforma .NET y el gestor de base de datos SQL Server a Software Libre.

Objetivo General:

Proponer una estrategia de migración a Software Libre de Akademos, la cual no ponga en riesgo sus prestaciones y la integración con el resto de los sistemas de la Dirección de Informatización de la Universidad de las Ciencias Informáticas, además de permitir la migración en el menor tiempo y con el menor costo, posibles.

Objetivos Específicos:

1. Llevar a cabo un análisis del estado actual del Sistema Automatizado para la Gestión Académica – “Akademos”.

2. Realizar un estudio de pasos y alternativas existentes para la migración de sistemas Web desarrollados sobre la plataforma .NET y el gestor de base de datos SQL Server a Software Libre.
3. Hacer un análisis sobre las plataformas Software Libre, los servidores Web y los gestores de base de datos Software Libre posibles para la migración del sistema.
4. Seleccionar la plataforma, dentro de las estudiadas, para la migración del sistema.
5. Seleccionar el Servidor Web, dentro de los estudiados, para la migración del sistema.
6. Seleccionar el gestor de base datos, dentro de los estudiados, para la migración del sistema.
7. Desarrollo de la Migración.

Hipótesis:

La elaboración de una estrategia para la migración a Software Libre del Sistema Automatizado para la Gestión Académica - “Akademos” en la Universidad de las Ciencias Informáticas permitirá ajustarse a las nuevas políticas de migración a Software Libre de Cuba y, por tanto, la posibilidad real de su utilización en el futuro.

Capítulo I: Sistema Automatizado para la Gestión Académica – Akademos

1.1 Introducción

Con este capítulo, se pretende mostrar al lector una panorámica del sistema que será objeto de migración¹, sus principales características, la arquitectura bajo la cual fue desarrollado y los módulos que lo componen. Además se expondrá una breve descripción de la tecnología empleada para el desarrollo del sistema y de las herramientas utilizadas durante la implementación del mismo.

1.2 La Gestión Académica

La gestión académica en una universidad es un proceso mediante el cual se controla, organiza y dirigen todas las actividades propias del proceso docente; entre las que se encuentran la matrícula de los estudiantes, la gestión del plan de estudio y el control de la asistencia y evaluaciones de los estudiantes. En esta tarea intervienen directivos, personal de secretaría y profesores.

1.3 Akademos

Akademos es un sistema automatizado para la gestión académica desarrollado por un equipo conformado por diplomantes provenientes del Plan CUJAE, estudiantes de la Universidad de las Ciencias Informáticas (UCI), bajo el liderazgo del Ing. Emil Lima Valdés y la Dirección de Informatización de la UCI. Su elaboración comienza en los primeros meses del año 2004 y las primeras versiones del Módulo de Matrícula y Plan de Estudio [Ver Epígrafe 1.6] son liberadas en octubre del propio año. Akademos se ha desarrollado como una aplicación Web haciendo uso de

¹ (migration) proceso por el cual las aplicaciones y los datos existentes trabajan en una computadora o sistema operativo diferente.

la tecnología ASP .NET y los XML WebServices [Ver Epígrafe 1.9] para garantizar su interoperatividad con otras aplicaciones. El sistema permite la participación activa de directivos, personal de secretaría, profesores y estudiantes en los procesos de gestión académica, agilizando así los mecanismos propios de la gestión académica y disminuyendo las posibles ocurrencias de errores.

1.4 Principales características

Akademos es un sistema informático que se diseñó teniendo en cuenta los siguientes principios:

- El dinamismo del proceso de gestión académica constituye la principal fuente de riesgos para un sistema que intente automatizarlo.
- Un sistema que automatice la gestión académica debe lograr que todos los involucrados (directivos, personal de secretaría, profesores y estudiantes) tengan un papel activo en el proceso.
- El plan de estudio es la entidad fundamental del proceso de gestión académica y rige todos sus subprocesos (matrícula, control, planificación, etc.)

Akademos está dividido en siete módulos [Ver Epígrafe 1.6], los cuales abarcan todos los procesos involucrados en la gestión docente. Estos módulos interactúan entre sí para llevar a cabo cada una de las tareas que automatiza el sistema.

1.5 Arquitectura de Akademos

Para la construcción de Akademos era necesario recurrir a una tecnología que garantizara flexibilidad en la distribución de nuevas versiones, a la vez que permitiera el acceso al sistema a un gran número de usuarios. Según estos requerimientos se optó por la utilización de la

tecnología Web la cual brinda estas y otras bondades. Específicamente se decidió el empleo de la tecnología ASP .NET, la cual facilita la creación de páginas Web complejas, mediante la utilización de un modelo de programación completamente orientado a objetos y el acceso al Framework .NET, el cual cuenta con una gran cantidad de clases reutilizables.

Además, se ha hecho un uso intensivo de los XML WebServices, lo que permitió la integración exitosa con otros sistemas de la Universidad.

1.6 Módulos de Akademos

Akademos está dividido en siete módulos siendo el módulo de Plan de Estudio la entidad fundamental del sistema, como se muestra en la siguiente figura.

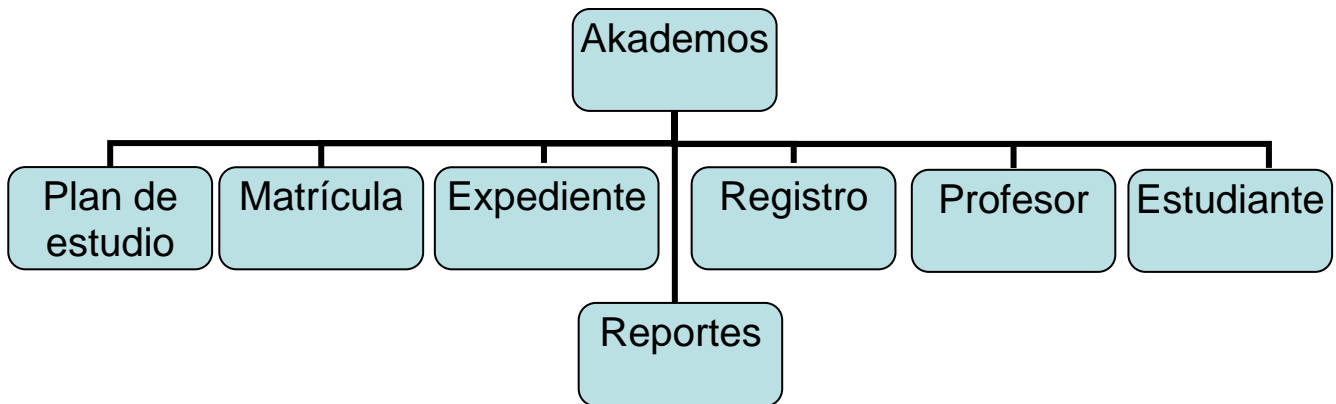


Figura 1.1 Módulos de Akademos.

Módulo de Plan de Estudio

El plan de estudio es la entidad fundamental del sistema. Este módulo permite la definición de los diferentes perfiles y disciplinas en las que se agrupan las asignaturas. Con la definición de varios

planes de estudios, por los docentes responsables, es posible realizar la gestión de diferentes especialidades o carreras.

Módulo de Matrícula

El objetivo fundamental de este módulo es el control de los datos de los estudiantes, así como la gestión de los movimientos a que son sometidos los estudiantes en su paso por la universidad. En los centros que aceptan cada año miles de estudiantes, la tarea de realizar la matrícula de estos puede convertirse en una labor compleja y lenta, si no se involucra a una gran cantidad de personas en esta tarea. El módulo de matrícula permite la descentralización del proceso de matrícula en un conjunto de personas tan grande como se decida por la dirección del centro de estudio.

Módulo de Expediente

El módulo de expediente es un repositorio digital de los documentos de los estudiantes. Los expedientes pueden almacenar documentos basados en plantillas, así como otros de libre formato o generados por el propio sistema. Cada vez que se crea un documento este es almacenado junto con otros datos, como su fecha de creación y el autor, para facilitar su posterior localización. Toda esta información constituye el grueso de la historia de la universidad.

Módulo de Registro del Profesor

El módulo de registro es el encargado de permitir el control docente de los estudiantes. Uno de los principios sobre los que se basó el diseño de Akademos es que la información sea introducida al sistema por los mismos que la originan previendo de esta manera la ocurrencia de errores. Son los profesores los que aplican las diferentes formas de control del avance de los estudiantes, de ahí que Akademos les permita directamente introducir los resultados de todas las evaluaciones

que apliquen así como mantener actualizado el registro de asistencia de todos los estudiantes en las distintas asignaturas.

Módulo de Profesor

El objetivo de este módulo es permitir la asignación de los profesores a las diferentes estructuras administrativas. Este módulo es controlado por el personal de secretaría el cual asigna a los profesores con diferentes niveles de acceso sobre las estructuras² en las cuales mantendrá actualizado el registro académico de los estudiantes. Además permite establecer un balance de la carga docente de los profesores, así como un mejor control de los profesores por facultad y grupo.

Módulo de Estudiante

El módulo de estudiante está diseñado específicamente para que el estudiante pueda consultar sus evaluaciones en las distintas asignaturas por él vencidas, así como su desempeño en las asignaturas del período docente en curso. Además permite el acceso a su expediente y con esto a documentos como la Hoja de Matrícula entre otros. Por otra parte brinda la posibilidad de consultar la trayectoria docente del resto de los estudiantes del centro de estudios.

Módulo de Reportes

La generación de reportes es quizás la tarea fundamental de todo sistema de gestión de datos. Este módulo provee a los usuarios un conjunto de herramientas, las cuales permiten el diseño y publicación de nuevos reportes. Con esta funcionalidad el usuario puede diseñar reportes que

² Estructuras administrativas de la institución, Eje Facultad 1 y Facultad 2 que son estructuras de tipo facultad y 1501 y 2304 que son estructuras de tipo Grupo.

involucren diferentes aristas de los estudiantes, pueden ser las notas y la proveniencia social, convirtiéndose en una poderosa herramienta para el estudio de la comunidad estudiantil.

1.7 Seguridad de Akademos

Akademos tiene tres niveles de seguridad:

- Seguridad de los datos:
 - o Para el almacenamiento de los datos se ha optado por la utilización de SQL Server 2000, el cual es capaz de garantizar la seguridad tanto lógica como de integridad de los datos almacenados en la base de datos³.

- Seguridad de los recursos lógicos:
 - o Se deja al sistema operativo, Windows en este caso, la responsabilidad de restringir el acceso a cada uno de los recursos lógicos de la aplicación como páginas Web, Servicios Web, archivos XML, etc.

- Seguridad de los elementos específicos del negocio que automatiza la aplicación:
 - o Akademos implementa varios niveles de acceso para restringir las acciones que puede realizar un usuario determinado.
 - o Akademos lleva un control sobre las acciones que realizan los usuarios en el sistema. De esta forma es posible detectar comportamientos indebidos y ayudar a los desarrolladores a mejorar la usabilidad del sistema. Por otro lado, es posible

³ (database) un archivo compuesto de registros donde cada uno contiene campos junto con un conjunto de operaciones para realizar búsquedas, ordenaciones, reordenaciones y otras funciones.

realizar un monitoreo de las incidencias en tiempo real si se utiliza el “Monitor de incidencias”, el cual es una aplicación de escritorio, desarrollada para Akademos en la plataforma .NET y que informa a los administradores de cada uno de los eventos que ocurren.

1.8 Aporte social y valoración económica que tiene Akademos en la actualidad

Aporte social

- Mejor control e incidencia en el aprendizaje de los estudiantes:
 - o Akademos permite que las notas sean introducidas apenas concluyan las evaluaciones. Con esto se pueden hacer estudios en menos tiempo y con mayor profundidad gracias a lo cual se podrán tomar decisiones más acertadas y con mayores posibilidades de influir en la mejora del desempeño docente de los estudiantes.

- Mejor control de las habilidades de los educandos para su ubicación en proyectos productivos:
 - o Akademos permite mantener actualizado el expediente digital de los estudiantes con toda la información relevante independientemente de su contenido y forma. Este expediente puede contener, por ejemplo, todas las evaluaciones que ha recibido un estudiante por su trabajo en cada uno de los proyectos en que ha participado, todos los cursos que ha vencido o todos los diplomas y reconocimientos de carácter docente e investigativo que ha ganado. Con esto los jefes de proyectos pueden realizar selecciones más acertadas de los estudiantes de la universidad para participar en sus proyectos.

- Posibilidad de realización de investigaciones sociales.
 - o La masa estudiantil de la Universidad de las Ciencias Informáticas es en sí misma una población representativa de la masa estudiantil cubana dada su variedad en términos de proveniencia geográfica, social y académica. Es por eso que mediante la utilización de Akademos pueden realizarse análisis importantes dado que este permite generar reportes basados en la combinación de múltiples variables.

Valoración económica

- Ahorro de insumos de oficinas
 - o Aún cuando existan restricciones legales en cuanto a la documentación dura que se maneja en la gestión académica, la posibilidad de que tanto profesores como estudiantes, directivos y personal de secretaría, puedan consultar en línea y desde cualquier punto de la intranet universitaria, toda la información que se gestiona en Akademos, conlleva a que se logre un ahorro considerable de materiales de impresión (papel, toner de impresoras, etc).
- Ahorro por compra de un sistema similar en el extranjero.
 - o Para el desarrollo de Akademos se estudiaron un conjunto de sistemas que tenían objetivos similares. Uno de los sistemas estudiados es Ágora el cual se comercializa en varias variantes [1]. La variante que incluye un conjunto de funcionalidades similares a Akademos tiene un precio de 3160€.

1.9 Descripción de herramientas empleadas en el desarrollo de Akademos

La necesidad de un sistema que permitiera llevar a cabo la automatización de los procesos de la gestión académica, la no existencia de un sistema capaz de adaptarse a las condiciones de la Universidad de las Ciencias Informáticas, de enfrentar los cambios de manera natural y con el menor tiempo y al menor costo posible, dan lugar al surgimiento del Sistema Automatizado para la Gestión Académica – “Akademos” como un proyecto de la Dirección de Informatización de dicha universidad en el año 2004. Akademos tenía que ser creado en el menor tiempo posible, ser un sistema que se integrase con el resto de los sistemas de la Dirección de Informatización y un sistema que permitiese la gestión académica logrando que todos los involucrados (directivos, personal de secretaría, profesores y estudiantes) tuviesen un papel activo en el proceso.

Producto de las necesidades expuestas, la poca experiencia en múltiples plataformas en la comunidad de desarrolladores de una Universidad con apenas unos dieciocho meses de iniciada y unido a esto, la complejidad que trae consigo un sistema de tal envergadura fue decidido, por la Dirección de Informatización en conjunto con la dirección del proyecto, que Akademos sería un sistema con las siguientes características:

- Un sistema Web distribuido.
- Para el almacenamiento de los datos se optó por la utilización de SQL Server 2000.
- Plataforma de desarrollo .NET.
- Para el intercambio con otras aplicaciones se optó por la utilización de Servicios WEB XML.
- Como servidor Web se optó por IIS (Internet Information Service).

A continuación haremos una descripción de la plataforma, el gestor de base de datos, los Servicios WEB XML, el servidor Web y las herramientas empleadas durante el desarrollo de Akademos.

1.9.1 SQL Server

SQL Server es un Sistema de Gestión de Base de Datos⁴ Relacional⁵ con todas las características que tienen estas aplicaciones; un servidor disponible en varias ediciones en el que es posible utilizar un lenguaje, Transact-SQL, para programar procedimientos almacenados y desencadenadores, aparte de poder definirse tablas, índices, vistas, etc.

Microsoft SQL Server 2000

Microsoft SQL Server 2000 es un sistema cliente/servidor de administración de base de datos relacional, diseñado para soportar un alto volumen de base de datos incluyendo procesamiento de transacciones en línea de alto rendimiento (online transaction processing - OLTP), almacenamiento de datos (data warehousing) y aplicaciones de comercio electrónico (e-commerce). Microsoft SQL Server 2000, como producto de Microsoft, es privativo y su precio asciende a más de 250 dólares por procesador.

1.9.2 Plataforma .NET

.NET es una plataforma de software que conecta información, sistemas, personas y dispositivos. La plataforma .NET conecta una gran variedad de tecnologías de uso personal y de negocios, de teléfonos celulares a servidores corporativos, permitiendo el acceso a información importante, donde y cuando se necesiten.

Desarrollado con base en los estándares de Servicios Web XML, .NET permite que los sistemas y aplicaciones, ya sea nuevos o existentes, conecten sus datos y transacciones

⁴ (database management system) una interfaz software entre la base de datos y el usuario. Un sistema de gestión de base de datos maneja las solicitudes de los usuarios para realizar acciones de base de datos y permite cumplir con los requisitos de control de la seguridad e integridad de los datos.

⁵ (relational database) base de datos que almacena la información en tablas, filas y columnas de datos, y efectúa las búsquedas usando datos en columnas específicas de una tabla para localizar datos adicionales en otra tabla.

independientemente del sistema operativo, tipo de computadora o dispositivo móvil que se utilice, o del lenguaje de programación empleados para crearlo. [2]

1.9.3 Web Services XML

Los Servicios Web son la más innovadora tecnología para los negocios en la Web. Los Servicios Web XML utilizan tecnologías programables y reutilizables que aprovechan la flexibilidad de Internet. Con ellos es posible tener una infinidad de aplicaciones conectados en red, ya sea que se ejecuten en diferentes plataformas, proporcionando información a todos sus clientes, socios de negocios y empleados.

Los Servicios Web tienen como base un conjunto de estándares abiertos, incluyendo XML⁶, SOAP⁷, WSDL⁸ y UDDI⁹, los cuales son controlados por el World Wide Web Consortium (W3C¹⁰). Trabajar con .NET significa usar protocolos abiertos que unen sistemas y aplicaciones existentes, permitiendo aprovechar mejor todos los beneficios que ofrecen. La información en su empresa aparece como una entidad única, integrada y fácil de compartir con otras empresas. [3]

1.9.4 Internet Information Services (IIS)

Internet Information Services (o Server), IIS, es una serie de servicios para los ordenadores que funcionan con Windows. Originalmente era parte del Option Pack para Windows NT. Luego fue

⁶ Sigla en inglés de eXtensible Markup Language («lenguaje de marcas extensible»), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C).

⁷ Siglas de Simple Object Access Protocol. Es un protocolo estándar creado por Microsoft, IBM y otros, está actualmente bajo el auspicio de la W3C que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. SOAP es uno de los protocolos utilizados en los servicios Web.

⁸ Siglas de Web Services Description Language, un formato XML que se utiliza para describir servicios Web.

⁹ Siglas del catálogo de negocios de Internet denominado Universal Description, Discovery and Integration. El registro en el catálogo se hace en XML. UDDI es una iniciativa industrial abierta (sufragada por la OASIS) entroncada en el contexto de los servicios Web.

¹⁰ Comité que promulga estándares con sedes en Estados Unidos, Europa y Japón. W3C se dedica, en parte, a estimular el desarrollo de estándares abiertos para la Web, como por ejemplo los lenguajes HTML y XML, promocionar la interoperabilidad y en conseguir que la Web logre todo su potencial.

integrado en otros sistemas operativos de Microsoft destinados a ofrecer servicios, como Windows 2000 o Windows Server 2003. Windows XP Profesional incluye una versión limitada de IIS. Los servicios que ofrece son: FTP¹¹, SMTP¹², NNTP¹³, HTTP¹⁴ y HTTPS¹⁵. Este servicio convierte a un computador en un servidor de Internet o Intranet es decir que en las computadoras que tienen este servicio instalado se pueden publicar páginas Web tanto local como remotamente (servidor Web).

El servidor Web se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas, por ejemplo Microsoft incluye los de Active Server Pages (ASP) y ASP.NET. También pueden ser incluidos los de otros fabricantes, como PHP o Perl. [4].

IIS es privativo y se ejecuta solo en sistemas operativos Windows cuyo precio asciende a más de 700 dólares por procesador.

1.10 Alternativas existentes para la migración de sistemas con las características de Akademos a Software Libre

La búsqueda de alternativas para hacer la migración de sistemas con características similares a las de Akademos, es uno de los principales objetivos de este trabajo. En este epígrafe se presentará el resultado de la misma.

¹¹ Es un protocolo de transferencia de ficheros entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor.

¹² Protocolo simple de transferencia de correo electrónico. Protocolo de red basado en texto utilizado para el intercambio de mensajes de correo electrónico entre computadoras o distintos dispositivos.

¹³ Es una aplicación de Internet que consiste en un protocolo usado para la lectura y publicación de artículos de noticias.

¹⁴ El protocolo de transferencia de hipertexto. Es el protocolo usado en cada transacción de la Web.

¹⁵ Es la versión segura del protocolo HTTP. El sistema HTTPS utiliza un cifrado basado en las Secure Socket Layers (SSL) para crear un canal cifrado (cuyo nivel de cifrado depende del servidor remoto y del navegador utilizado por el cliente) más apropiado para el tráfico de información sensible que el protocolo HTTP. Cabe mencionar que el uso del protocolo HTTPS no impide que se pueda utilizar HTTP.

La migración de sistemas a software libre es una problemática que se puede apreciar con frecuencia en grupos de desarrollo y soporte de software. Las ventajas que puede traer consigo desarrollar sistemas sobre software, plataformas y gestores de bases de datos puramente privativos se ven deterioradas con el paso del tiempo. Son más de una las instituciones, los países y grupos de desarrollo que, después de un largo período de trabajo y acumulación de experiencias, se lanzan a la migración a software libre de sus sistemas.

La UCI en la actualidad da sus primeros pasos a la migración de sus sistemas de gestión interna a alternativas libres, pues la experiencia acumulada es muy escasa y no se cuenta con la documentación de los aislados trabajos que se ha realizado en este sentido.

Desafortunadamente no existe una metodología, una estrategia o un conjunto de reglas para hacer la migración a software libre de un sistema con características semejantes a las de Akademos; son varias las plataformas o lenguajes libres, los gestores de bases de datos y los servidores Web con los que se le pudiese dar solución a la migración de Akademos, razón por la cual se le ha dedicado un capítulo a este tema.

1.11 Conclusiones

Hasta este punto se ha presentado una descripción del sistema que será sujeto a la migración, exponiendo un resumen de sus características esenciales, los módulos que lo componen, su arquitectura y la gestión de la seguridad del mismo. También se abordaron las condiciones bajo la cual surgió el proyecto y las tecnologías empleadas para su desarrollo, así como el gestor de base datos y el servidor Web utilizado para el desarrollo y explotación del mismo.

Capítulo II: Estudio para la migración

2.1 Introducción

En este capítulo se pretende hacer un estudio de las plataformas software libre con las que se pudiese hacer la migración de Akademos, así como el servidor Web y el gestor de base de datos que se pudiera emplear para lograrla. Otro objetivo de este capítulo es dejar bien claro las herramientas que se emplearán en la migración, dejando sentadas las bases para dar paso al Capítulo III donde se propondrá una estrategia de migración.

2.2 Posición de Cuba ante el uso y la migración a Software Libre

El férreo bloqueo económico sostenido por el gobierno de los Estados Unidos sobre Cuba por más de 45 años, impide el uso y pago de licencias de software producido por compañías estadounidense tan prestigiosas como IBM, Oracle, Adobe, Sun Microsystems, McAfee Computer Virus Software y Microsoft compañía líder mundial en la producción de software puramente privativo y anti-ético como la suelen llamar prestigiosos autores de la rama. Durante todo estos años son más de una las artimañas, mentiras y los obstáculos impuesto por el gobierno de los Estados Unidos para impedir que el desarrollo tecnológico llegue a nuestro país.

Son numerosos los ejemplos y muchos los artículos que tocan muy de cerca este tema. Estadistas y medios de prensa en más de una ocasión resaltan hechos tan significativos en este sentido, como que cada Byte que llega a nuestro territorio cuesta más de lo normal, pues los equipos de cómputo se encarecen hasta un 30 por ciento por encima de su precio al no poder adquirirse directamente de mano de sus productores y tener que pagar grandes sumas por comprarlos a intermediarios. Microsoft, en las licencias de sus productos, consigna que estos no pueden ser exportados a Cuba, ni a mediadores que comercialicen con Cuba o sencillamente a quienes ellos sospechen que lo hacen. [5]

Pero estos asuntos legales no son el único motivo para emerger de las turbulentas aguas del Software Privativo y ser Cuba uno más de los tantos países en dar los primeros pasos a la migración a software libre de sus sistemas de administración pública, educativos y de gestión hospitalaria. El simple hecho de hacer uso de Software Privativo va más allá de problemas legales, su uso puede verse hasta como un problema de seguridad nacional. En ocasión de la Feria Informática de la Habana 2007 el ministro de la Informática y las Comunicaciones, Ramiro Valdés Menéndez, llamó a los sistemas informáticos del mundo como un "campo de batalla" donde Cuba está luchando contra el imperialismo.

Esto no es solo una idea, no es un sueño, ni una utopía, Cuba ya es parte del Movimiento de Software Libre Mundial, desarrolla sus propias estrategias a la par que sigue las experiencias de países como Alemania, Argentina, Brasil, Chile, China, España, Francia, México, República Dominicana y Venezuela. Estos últimos años han sido de gran avance para el desarrollo de la comunidad de software libre en nuestro país, son muchos los esfuerzos y el interés por los organismos centrales del estado y sus dependencias en aras de apoyar este proceso de migración. Los más de 600 Joven Club de Computación también se convertirán paulatinamente en espacios para el desarrollo y consolidación de la comunidad de Software Libre cubana. Como parte de esta estrategia son muchos los recursos que el estado cubano ha puesto a disposición de la formación de profesionales capacitados en esta rama con el objetivo de alcanzar la independencia en este terreno, estrategia que está puesta en marcha desde principios de este siglo.

En la ciudad de la Habana, en el mes de Junio del año 2001, se firma la llamada Declaración de La Habana sobre el uso de Software Libre y Abierto. Esta declaración se firmó en ocasión de celebrarse el 1er Taller Latinoamericano para la Promoción del uso de Software Libre en la Educación, Ciencia, Cultura y Áreas Sociales; representantes cubanos unidos a representantes del área como Uruguay, Brasil, Ecuador y Colombia acordaron lo siguiente:[6]

- Recomendar a los gobiernos de la región el establecimiento de políticas que favorezcan el uso de software libres y abiertos en los servicios públicos.
- Recomendar a la comunidad internacional, la celebración, cada año, del día mundial del software libre, preferiblemente el 5 de octubre por ser en esa fecha cuando se dio a conocer la creación del núcleo del sistema operativo GNU/Linux, como expresión de un trabajo colectivo acumulado durante varios años para lograr este objetivo.
- Incrementar la divulgación del uso del software libre en nuestra región, mediante estrategias nacionales, regionales y locales adecuadamente concebidas.
- Desarrollar programas de estudios que incrementen la enseñanza de software libre y abierto en las universidades y otros centros docentes de nuestros países.
- Incluir los aspectos de software libre en las políticas que se tracen para favorecer la construcción de un mundo sin exclusión social y con igualdad de acceso a los programas tecnológicos y a la información.
- Promover iniciativas para hacer llegar al usuario final técnicas, herramientas y metodologías que les permitan utilizar software de código abierto en la solución de sus problemas.
- Promover la presencia de libros y artículos sobre software libre en revistas, sitios Web y otras ediciones de carácter científico.

Pero el uso de software propietario es más que un problema económico, legal y de seguridad, ya que además de hacernos dependientes de la potencia enemiga que más nos ataca, podemos decir que tampoco nos enriquece tecnológicamente, puesto que no nos permite estudiarlo, ni modificarlo, o mejorarlo, ni distribuirlo; por lo cual promueve el consumismo y el individualismo, y no permite desarrollar nuestra propia identidad y cultura. [7]

La posición de Cuba queda bien clara, promover una cultura de libertad e independencia tecnológica es una tarea de todos y que mejor forma de dar nuestro aporte a este proceso que no sea revolucionar nuestro entorno y hacerlo a la vez que promovemos el uso del Software Libre.

2.3 Razones y necesidades para hacer la migración de Akademos

La posición de Cuba ante el uso del software privativo está bien clara y es esta política la principal razón para hacer la migración. Pero surgen varias interrogantes:

- ¿Es necesario hacer la migración de Akademos?
- ¿Es factible esta migración?

Para dar respuesta a la primera interrogante se deben tener en cuenta los siguientes elementos que involucran a Akademos:

- Sistema Operativo y Servidor Web.
- Gestor de base de datos.

En el Capítulo I se hace alusión a las características de Akademos; su plataforma de desarrollo exigía para su construcción y puesta en marcha que se usase, como servidor Web, Microsoft Internet Information Server (IIS) en su versión 6, ejecutándose sobre Windows Server 2003. En la actualidad Akademos comparte alojamiento con otros sistemas que brindan servicios a la comunidad universitarias [Ver Anexo 1], esto trae consigo que en horarios picos el sistema se torne algo lento. Los términos legales y de licencias sobre el uso de software privativo se trataron en el epígrafe anterior, pero es bueno señalar que este sistema operativo costaría, en caso de que se pagase por su licencia, más de 700 dólares por procesador.

Alejándonos de los costos por conceptos de software y licencias sería bueno analizar también el precio que indirectamente estamos pagando por concepto de seguridad y para profundizar más, valdría la pena hacernos las siguientes preguntas. ¿Qué tan seguro es Windows y su servidor Web? Esta pregunta es muy frecuente en las comunidades de desarrollo y usuarios avanzados de este sistema operativo; son varios los artículos y autores que tratan este tema. Expertos en la

materia aseguran que este sistema operativo no es tan seguro como lo hacen creer sus proveedores, es imposible confiar en su seguridad cuando no podemos ver ni modificar a nuestras necesidades su código fuente. Los datos recogidos por Netcraft¹⁶ muestran que servidores Windows son reiniciados con mucha más frecuencia que máquinas que ejecutan servidores libres, esto es debido entre otras razones a los problemas de seguridad que con frecuencia presenta Windows. Microsoft pone a disposición de sus usuarios actualizaciones de seguridad con gran frecuencia. Estas actualizaciones son parches a fallos del sistema operativo que en muchas ocasiones son informados por los propios clientes y es común que estos parches tarden en aparecer hasta 3 meses después de reportados. Aproximadamente la mitad de estos fallos son considerados como críticos y pueden poner en riesgo millones de dólares en dependencia del uso que se le esté dando al sistema.

Si problema es ejecutar a Akademos sobre un servidor Windows es también un problema que su gestor de base de datos sea parte de la gama de productos privativos de Microsoft. SQL Server 2000, con un precio de más de 250 dólares por procesador, es el gestor que da soporte actualmente al flujo de información que gestiona Akademos. Este servidor, al igual que el servidor Web, también brinda servicios a otro conjunto de sistemas utilizados por la comunidad universitaria [Ver Anexo 2]; esta también es una causa de que en horarios picos se pueda apreciar una considerable lentitud en el sistema. Como los productos de Microsoft solo se pueden ejecutar sobre Windows, esto trae consigo que los riesgos de seguridad a los que está sometido el sistema operativo también afecten al gestor de base de datos y pongan en riesgo toda la información gestionada por el sistema.

No se puede negar que Windows 2003 Server y SQL Server 2000 son bastantes estables y en los tres años que se han empleado para la gestión académica de la universidad no han reportado problemas, ni ocasionado pérdidas de información alguna; pero lo que está claro, después de

¹⁶ Compañía dedicada al análisis y encuestas de servidores Web conectados a Internet

conocer la posición de nuestro país ante el uso y migración a software libre, es que no se debe utilizar SQL Server 2000 ejecutándose sobre un servidor Windows 2003 Server existiendo prestigiosos gestores y sistemas operativos libres que brindarían las mismas prestaciones.

La factibilidad de hacer la migración del sistema puede considerarse como uno de los objetivos que se propone en este trabajo y después de un estudio de las posibles Plataformas, Servidores Web y Gestores de Base de Datos se podrá llegar a una conclusión más certera de cuan factible es hacer la migración, que herramientas utilizar, que vía sería la más rápida y menos costosa. Pero desde este momento se pueden dejar bien claro algunos beneficios tangibles e intangibles que traerían consigo la migración. El beneficio tangible más importante asociado a la migración, es la supresión de las caras licencias de software que, aunque hoy por hoy no se pagan, sí hay que tenerlas presentes si se quiere exportar el sistema a otro país. Son varios los beneficios intangibles asociados a la migración por lo cual es preferible enumerarlos:

- Alinear el desarrollo tecnológico de un sistema clave, que soporta procesos base de un centro de alta importancia estratégica, con las políticas tecnológicas de nuestro país.
- Alcanzar un conocimiento teórico y práctico en materia de esfuerzo, de selección de herramientas y de alternativas tecnológicas que permitan allanar el camino a futuras iniciativas de migración de sistemas similares hacia software libre.
- Lograr que Akademos disfrute de todos los beneficios inherentes al software libre.

Akademos tiene más de 2 años de prestación estable de servicios a la comunidad universitaria y nunca se había hablado de una posible migración. Es bueno dejar bien claro el por qué ahora es un buen momento para asumir esta tarea.

- El sistema se encuentra en un estado de estabilidad, sus módulos más críticos están cumplimentados en más de un 95% para una primera versión.

- Akademos cuenta con un equipo de desarrollo preparado para asumir en paralelo el mantenimiento del sistema en su estado actual y la nueva versión sobre una plataforma libre.
- Se prevé que Akademos se despliegue en otros centros educacionales que exigen el cumplimiento de las políticas sobre el empleo de software libre en los sistemas de gestión, por lo que se necesita probar y desarrollar la versión libre antes que se complete su despliegue.
- La migración de Cuba hacia el software libre se encuentra en una etapa de pura investigación y determinación de vías y metodologías, por lo tanto se necesitan proyectos que sean pioneros en este sentido.

2.4 Plataformas Software Libre. Propuesta para la migración

2.4.1 Caracterización de las Plataformas Web Libres (PWL) más difundidas

Para el desarrollo de aplicaciones Web existen varias plataformas y lenguajes de desarrollo en software libre. Entre estas sobresalen tres en particular Mono, Java y PHP de las que a continuación se hará una breve descripción.

PHP

PHP (acrónimo de "PHP: Hypertext Preprocessor") es un lenguaje interpretado de alto nivel especialmente diseñado para el desarrollo Web y que puede ser embebido en páginas HTML, ejecutándose en el servidor. Cuenta con algunos entornos de desarrollo Web libres entre los que se destaca Quanta que fue diseñado especialmente para el proyecto KDE¹⁷. Presenta muchas

¹⁷ K Desktop Environment es un entorno de escritorio gráfico e infraestructura de desarrollo para sistemas Unix y, en particular, Linux.

ventajas entre las que se encuentra el Modo de Diseño de Página Visual (VPL) “WYSIWYG¹⁸”, la de visualizar el código fuente o ambas vistas a la vez.

JAVA

Plataforma Java Edición Empresarial (Java Platform Enterprise Edition) o Java EE es una plataforma de programación para desarrollar y ejecutar aplicaciones en lenguaje Java con arquitectura de n niveles, basándose en la utilización de módulos de software que se ejecutan sobre un servidor de aplicaciones.

Existen un grupo de frameworks¹⁹ que cumplen con los estándares establecidos por el Java EE y que facilitan el desarrollo de aplicaciones Web, entre los que sobresalen tres que pueden ser utilizados de forma conjunta para este fin Struts, Spring e Hibernate.

Struts es uno de los frameworks más difundidos, con más tiempo en el mercado y con mejores prestaciones en el desarrollo de aplicaciones Web bajo el patrón Modelo Vista Controlador (MVC²⁰).

Spring entra dentro de la clasificación de peso ligero por su bajo consumo de recursos. Hace uso de beans²¹ para su funcionamiento y se integra de forma muy eficaz con otros frameworks como Struts e Hibernate. Su integración con Hibernate permite hacer un uso más cómodo de éste,

¹⁸ acrónimo de *What You See Is What You Get* (lo que usted ve es lo que usted consigue) le permite a un usuario visualizar un documento según aparecerá en el producto final, y editar directamente el texto, gráficos u otros elementos dentro de esa vista.

¹⁹ Término utilizado para referirse a un conjunto de librerías o clases que se utilizan para implementar la estructura básica de una aplicación o sistema.

²⁰ Patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones Web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página, el controlador es el Sistema de Gestión de Base de Datos y el modelo es el modelo de datos.

²¹ Un JavaBean o bean es un componente de software que se puede reutilizar y que puede ser manipulado visualmente por una herramienta de programación en lenguaje Java

abstrayendo al programador de tareas repetitivas como pueden ser el abrir y cerrar una Session Hibernate entre otras.

Finalmente se encuentra el framework Hibernate. El mismo simplifica enormemente el trabajo con la base de datos, ya que permite realizar un mapeo entre la base de datos relacional y el modelo de objetos de la aplicación a través de archivos xml con lo que se pueden utilizar todas las ventajas propias de la programación orientada a objetos. Este framework está ampliamente difundido y es muy eficiente.

Por otra parte se encuentra el Eclipse, un entorno de desarrollo integrado (IDE²²) de altas prestaciones que permite la incorporación de módulos, entre los que está Web Tool Platform (WTP²³) con el que aumentan las posibilidades en el diseño y creación de páginas Web.

MONO

Mono es un proyecto de código abierto promovido por Novell²⁴, con el objetivo de crear un grupo de herramientas libres, basadas en GNU/Linux y compatibles con la tecnología .NET²⁵. Mono provee el software necesario para desarrollar y ejecutar aplicaciones cliente y servidor en Linux, Solaris, Mac OS X, Windows y Unix, funcionando en varias arquitecturas: x86, PowerPC, S390, SPARC, HPPA, StrongARM. Es compatible con múltiples lenguajes entre los que están C#, Java, Boo, Temerle, Visual Basic.NET y Python [8]. Entre los entornos de desarrollo integrado (IDE) que utiliza Mono está Monodevelop en su versión 0.13.1.

²² (integrated development environment) conjunto de herramientas integradas para el desarrollo de software. Las herramientas se ejecutan normalmente a través de una interfaz de usuario y constan de un compilador, un editor y un depurador, entre otras.

²³ Conjunto de APIs para el desarrollo de aplicaciones en Java EE

²⁴ Novell, Inc. es una compañía de origen estadounidense dedicada al software, específicamente en el área de sistemas operativos de redes, como Novell Netware y Linux, entre otras ramas de la tecnología.

²⁵ .NET es el proyecto de Microsoft de una nueva plataforma de desarrollo de software con énfasis en la transparencia de redes, que tenga independencia de la plataforma y que permita un rápido desarrollo de aplicaciones.

2.4.2 Selección de la Plataforma Web Libre para hacer la migración.

Akademos, como ya se analizó en el Capítulo I, ha sido desarrollado utilizando la tecnología ASP.NET, lenguaje C# y el Framework .NET en su versión 1.1. Es un sistema que se encuentra en etapas muy avanzadas de desarrollo, con versiones estables de todos sus módulos, los cuales son utilizados activamente por la Universidad de las Ciencias Informáticas. Está basado en arquitectura de tres capas (interfaz, negocio, acceso a datos) y de sus módulos, el más sencillo, Profesor, tiene en la capa de interfaz no menos de diez páginas Web y diez controles de usuario²⁶, en la de negocio tres clases de lógica del negocio de ese módulo y en la de acceso a datos cinco clases; mientras que uno de los más complejos, Plan de Estudio, tiene cerca de sesenta páginas Web y veinte controles de usuario, en la capa de negocio hay once clases y en la de acceso a datos treinta y ocho clases. Tomando en cuenta lo antes explicado podemos concluir que a la hora de escoger o determinar cual podría ser la plataforma más conveniente para migrar el sistema a software libre se debe partir del análisis de aquellas que permitan aprovechar al máximo todas las funcionalidades ya desarrolladas, necesitando para ello reprogramar o modificar lo menos posible lo que ya está implementado; lo que provocaría dedicar menos tiempo y recursos para este propósito.

De las plataformas y lenguajes de desarrollo en software libre presentados PHP es uno de los más rápidos, más livianos y que menos tiempo de desarrollo consume. Es sencillo de utilizar, se encuentra ampliamente difundido, existe gran cantidad de documentación y están formados muchos grupos y foros de debate que facilitan todo el proceso de desarrollo. Existen varios IDE de desarrollo no propietario, siendo Quanta uno de los que más prestaciones y facilidades brinda. Sin embargo tiene algunas desventajas como son que al escoger esta alternativa habría que comenzar prácticamente desde cero el desarrollo del sistema puesto que la semejanza o compatibilidad entre C# en .NET y PHP es prácticamente nula. Sería necesario capacitar a todo

²⁶ Permite encapsular varios controles con su lógica asociada en un solo control de usuario, que luego podrá ser utilizado en más de una página Web.

el equipo de desarrollo o tal vez formar un nuevo equipo que tuviera conocimientos de este lenguaje. Además mientras el nuevo sistema fuera desarrollado, tendría que seguir funcionando y prestando servicios la versión actual de Akademos desarrollada en .NET, por lo que sería preciso dejar una parte del grupo actual de desarrollo dando soporte al mismo y eso significaría más tiempo y recursos.

La segunda plataforma analizada, Java EE se encuentra también muy difundida, existe mucha documentación, son muchos los frameworks y plugins²⁷ desarrollados, existe un IDE muy potente y altamente desarrollado que se ajusta prácticamente a todas las necesidades y son muchos los grupos de debate y foros de discusión que le dan soporte. Aún así presenta una serie de inconvenientes como son el hecho de que el proceso de desarrollo es un poco más lento y complicado. La similitud entre los lenguajes de programación C# y Java es bastante alta pero aún así por cuestiones de arquitectura y formas de implementar y desarrollar no se puede aprovechar en gran medida la versión de Akademos en .NET, principalmente se perdería casi totalmente la capa de interfaz lo cual sería bastante costoso en tiempo y recursos. Por supuesto nuevamente sería preciso un proceso de capacitación y preparación de todo el equipo de desarrollo en la nueva plataforma, así como tendría que dejarse una parte del equipo dando soporte a la versión de Akademos en uso.

Finalmente se encuentra Mono que tiene entre sus inconvenientes que es un proyecto que se encuentra en desarrollo, el IDE de desarrollo se encuentra bastante atrasado y carece de muchas facilidades y ventajas, no existe mucha documentación sobre el mismo y no son muchos los grupos de debate y los foros de discusión. Sin embargo es cien por ciento compatible con el lenguaje C#, ya que es compatible con múltiples lenguajes; incorpora o admite el uso de una gran parte de los controles visuales existentes en la plataforma .NET, como son los tan fundamentales

²⁷Programa que interactúa con una aplicación central o cliente (un navegador Web, un cliente de correo o un IDE de desarrollo, por ejemplo) para proveer ciertas funcionalidades muy específicas.

e imprescindibles controles de usuario y los datagrid²⁸ entre otros; permite el uso total de las funcionalidades brindadas por el .NET framework 1.1, llegando al punto de poder ejecutar ensamblados producidos por el Visual Studio sin necesidad de recompilar. A pesar de la juventud del proyecto, a la que hacíamos referencia anteriormente, existen en la actualidad muchas empresas y grupos de desarrollo que han implementado sistemas y aplicaciones utilizando parcial o totalmente Mono y algunos frameworks desarrollados para el mismo.

Tabla 2.1: Sistemas y aplicaciones desarrolladas utilizando Mono. [9]

Aplicaciones comerciales	Sistema o Aplicación	Descripción	Empresa
	Plasma	Sistema para la administración de código fuente	Codice Software
	Chrome Compiler	Compilador comercial de Pascal	RemObjects
	Unity	Editor de juegos de 3D	Otee
	GMOVIL	Herramienta para administrar procesos comunes para los puntos de venta de Vodaphone	Advanced Software Production Line
Aplicaciones Software Libre	Last-Exit	Reproductor basado en Gnome.	
	Galaxium	Cliente de mensajería instantánea	
	Gib	Herramienta para generar temas de iconos para Gnome	
	MonoUML	Herramienta CASE ²⁹	
	LAT	Herramienta administrativa de LDAP	

²⁸ El control de servidor Web Datagrid es una cuadrícula de múltiples columnas, enlazada a datos.

²⁹ Ingeniería de Software Asistida por Computadora (Computer Aided Software Engineering), técnica para el uso de computadoras en una o más fases del ciclo de vida del software.

Existen además componentes y librerías desarrolladas haciendo un uso parcial o total de Mono.

Tabla 2.2: Componentes y librerías desarrolladas utilizando Mono. [10]

	Nombre	Descripción
Librerías	GtkSharp	Librería que permite desarrollar aplicaciones graficas nativas para Gnome utilizando Mono.
	Windows.Forms	Parte de la distribución estándar de Mono. Es un trabajo en progreso para implementar las API ³⁰ System.Windows.Form de Microsoft.
	wxNet	Librería para el desarrollo de aplicaciones para wxWindows
Componentes	fyiReporting	Incluye un diseñador de reportes con soporte para las especificaciones RDL. Este software incluye componentes reusables y controles para Windows.Forms
	ReportMan	Motor para la creación de reportes usando xml para su almacenamiento
	FlexCelNet	Es un paquete para la generación de reportes en Visual Studio .NET 2003 y 2005 en archivos Excel y PDF.

Luego de realizar un análisis con más profundidad y de presentar las ventajas y desventajas propias de cada una de las plataformas y lenguajes de programación en estudio, podemos concluir que Mono es de todos el más indicado para llevar a cabo la migración; pues no pondría en peligro las prestaciones y servicios que brinda actualmente Akademos, permitiría llevar a cabo la migración en el menor tiempo posible y con el mínimo consumo de recursos y brindaría la posibilidad de reutilizar al máximo todas las clases, páginas Web del sistema, controles de usuario y componentes. Para lograr esto el tiempo de capacitación sería mínimo pues ya el

³⁰ Interfaz de Programación de Aplicaciones (Application Programming Interface), Interfaz de código fuente que un sistema o librería de programa provee para dar soporte a las solicitudes de los servicios hechos por un programa de computadora.

equipo de desarrollo de Akademos cuenta con madurez y experiencia en el desarrollo de aplicaciones .Net.

Estado actual

Mono está en la actualidad en un proceso acelerado de desarrollo y mejoras, impulsado por Novell, una empresa de prestigio y con grandes perspectivas en el mundo del software. Esta empresa estableció acuerdos de intercambio tecnológico y de cooperación con Microsoft, con los cuales muchos clientes más conservadores, que nunca estuvieron dispuestos a cambiar a Linux han comenzado a considerarlo con más seriedad. Mono se encuentra en su versión 1.2.3 y ya se trabaja en el desarrollo de la versión 2.0 que ofrecerá soporte total para ASP.NET 2.0 y ADO.NET 2.0.

Este proyecto ha desencadenado un gran interés en el desarrollo de componentes basados en C#, librerías y frameworks para Mono. Muchos de ellos han sido desarrollados por el mismo equipo de Mono. Entre ellos sobresalen Tao framework para el trabajo con extensiones OpenGL, Mono.Directory.LDAP³¹ y Novell.Directory.LDAP para el acceso a LDAP, Mono.Data para el trabajo con base datos como PostgreSQL, MySQL, Firebird, SQLite, Microsoft SQL Server, Oracle y fuentes de datos ODBC, Mono.Http para el trabajo con manejadores http y Mono.XML para el trabajo con XML.

El Monodevelop, es un IDE de desarrollo libre para GNOME que hace uso de Mono, diseñado inicialmente para el desarrollo con C# y otros lenguajes de .NET y encontrándose en estos momentos en un proceso de desarrollo y mejoras para lograr llegar a la versión 1.0. La arquitectura de Monodevelop se divide en tres capas, la capa principal (Core layer) que provee

³¹ Protocolo Ligero de Acceso a Directorio (Lightweight Directory Access Protocol), protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

los servicios para las aplicaciones, la capa de Proyectos (Projects layer) que implementa el modelo de objetos del proyecto y la capa Superior que es en general el propio IDE.

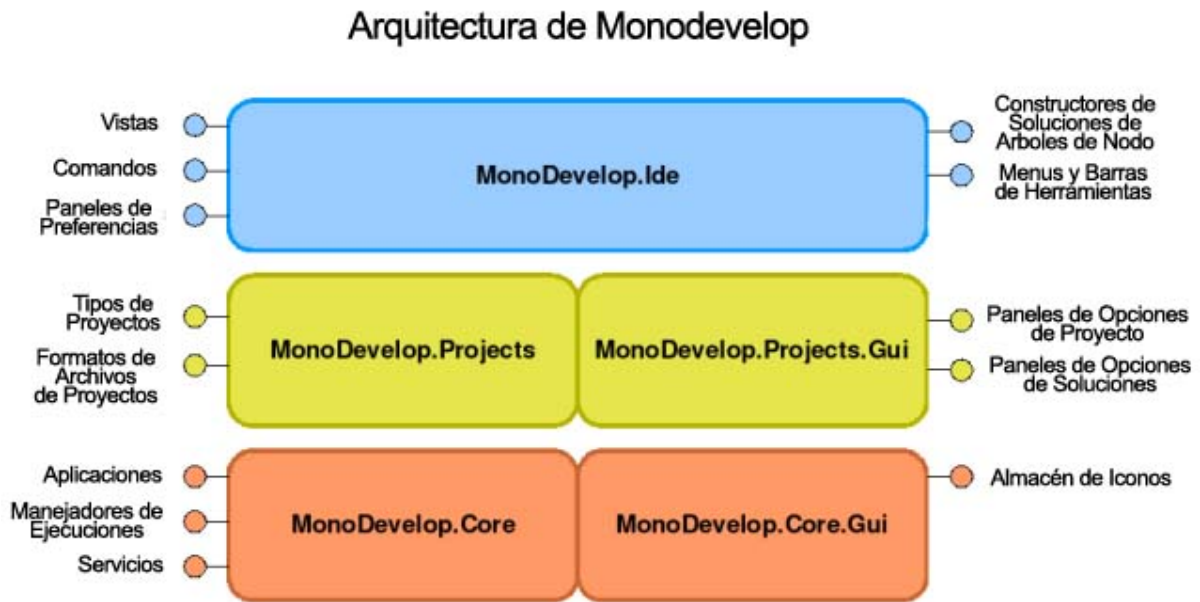


Figura 2.1: Arquitectura de Monodevelop [11].

Reusabilidad

Uno de los aspectos fundamentales en la selección de la plataforma de desarrollo es el grado de reusabilidad que ésta permita.

Para el desarrollo de Akademos se aprovecharon las ventajas propias del Framework .NET en su versión 1.1. Las funcionalidades que se utilizan de este framework ya están desarrolladas en Mono, agrupadas en los frameworks `Novell.Directory.LDAP`, `Mono.Data` y `Mono.XML` entre otros, haciendo que sea casi nulo el código a reprogramar para este fin. Otro de los aspectos positivos que brinda el usar Mono es que permite emplear el mismo lenguaje utilizado para programar todas las clases del sistema, C#.

Experiencia en CUBA y en la UCI en el desarrollo con la Plataforma Web Libre seleccionada

En la actualidad existe muy poca experiencia en Cuba en el desarrollo de sistemas informáticos utilizando la plataforma Mono. Sin embargo en la Universidad de las Ciencias Informáticas una de sus diez facultades, la facultad número diez específicamente, está incursionando con fuerza en este mundo. Esta facultad centra su plan de estudio en el uso por completo de software libre, siendo especie de un gran laboratorio del que se esperan grandes resultados a nivel nacional. En los dos semestres del primer año, del curso regular diurno 2006 - 2007 se imparten las asignaturas Introducción a la Programación y Programación 1 utilizando Mono y el Monodevelop como IDE de desarrollo, por lo que posiblemente sean sus más de doscientos cincuenta estudiantes y profesores el grupo más grande a nivel nacional de programadores que dan los primeros pasos en este sentido. Unido a esta gran comunidad de estudiantes y profesores de la facultad diez, un grupo de desarrollo de la UCI, que informatiza los procesos de Registro y Notaria de la República Bolivariana de Venezuela, está haciendo una fuerte incursión en el empleo de Mono como plataforma libre para el desarrollo de los servicios Web que forman parte de la arquitectura del sistema en desarrollo.

2.5 Servidores Web Libres. Propuesta para la migración

2.5.1 Apache

Ya seleccionada la plataforma en la que se realizará la migración, se debe seleccionar el servidor Web Software Libre (SWSL) que se encargará de dar respuesta a las peticiones http hechas desde los clientes del sistema. Son más de uno los SWSL con los que se cuenta en la actualidad tales como Apache, Lighttpd, Cherokee, Nanoweb, que son los más difundidos. Independientemente de la cantidad de servidores, las prestaciones y la disponibilidad de los mismos es difícil dudar ante la utilización del Apache; no es un secreto que el servidor Web SW más popular y difundido en el mundo es el Apache, fue considerado por mucho tiempo el

proyecto punta de lanza del movimiento de Software Libre en el mundo, por su ayuda en el impulso del mismo. Es una suerte que se pueda contar con un SW tan ligero, altamente configurable y a su vez libre como lo es este. Apache es catalogado como el SW número uno, más del 58 % de los SW utilizados en el mundo son Servidores Apaches, de acuerdo con una encuesta dada a conocer por la compañía Netcraft. Esta compañía muestra periódicamente estadísticas de los SW más utilizados, el siguiente gráfico muestra las estadísticas publicadas correspondientes al mes de Febrero del 2007.

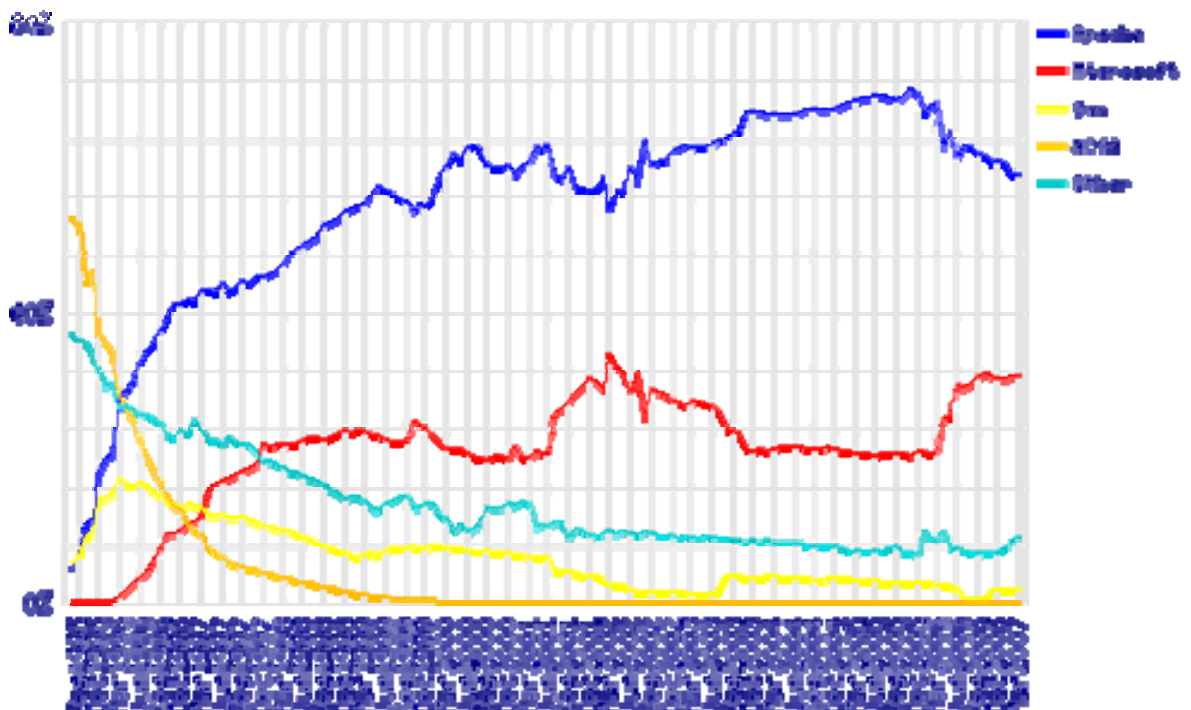


Figura 2.2. Estadísticas de los SW más populares. [12]

Tabla 2.3. Estadísticas de los SW más populares. [12]

Servidor Web	Enero 2007	(%)	Febrero 2007	(%)	Cambio
Apache	64312083	60.17	63869543	58.7	-1.47
Microsoft	32898421	30.78	33833566	31.09	0.31
Sun	1749026	1.64	1845584	1.7	0.06
Zeus	551767	0.52	530596	0.49	-0.03

Ventajas al utilizar el servidor Web Apache

A continuación se muestran algunos de los aspectos que hacen de Apache un servidor Web tan popular y a su vez el más utilizado en el mundo. [13]

- **Apache es un servidor altamente configurable de diseño modular**

Las prestaciones de Apache son muy sencillas de ampliar en dependencia de las necesidades de quien lo utilice. Sus creadores aseguran que con un nivel aceptable en los lenguajes de programación C o Perl es posible desarrollar un módulo para que realce una función deseada. Esto trae consigo que son muchos los módulos para apaches que han sido desarrollados a lo largo de su historia.

- **Apache es una tecnología gratuita de código fuente abierto**

Apache va mas allá de ser un software gratuito, su código fuente es abierto, este código fuente se puede usar, modificar y distribuir de forma gratuita. Todo lo relacionado con las licencias de apache se puede ampliar en la siguiente dirección: <http://www.apache.org/licenses/>.

- **Apache funciona en Linux, en otros sistemas de Unix y Windows**

A pesar de que muchos autores catalogan a Apache como uno de los mayores logros del mundo del Software Libre, este no sólo corre en sistemas operativos de esta línea. Apache aunque se considera que es más estable y eficiente cuando se ejecuta sobre sistema Linux, también se puede ejecutar sobre Solaris y Mac OS. Muchos sistemas que corren sobre Windows utilizan como servidor Web a Apache 2.0 en vez de Microsoft Internet Information Server (IIS).

2.5.2 Características más importantes del Servidor Web Apache 2

Apache 2 presenta muchas otras características, entre ellas un elaborado índice de directorios, un directorio de alias, negociación de contenidos, informe de errores HTTP configurable, ejecución SetUID³² de programas CGI³³, gestión de recursos para procesos hijos, integración de imágenes del lado del servidor, reescritura de las URL, comprobación de la ortografía de las URL y manuales online. Otras características esenciales de Apache 2 se muestran a continuación y fueron tomadas de la Biblia de Apache 2. [13]

- **Soporte del último protocolo HTTP 1.1**

Apache es uno de los primeros servidores Web en integrar el protocolo HTTP 1.1. Es totalmente compatible con el nuevo estándar HTTP 1.1 y al mismo tiempo sigue siendo compatible con HTTP 1.0. Apache está preparado para todas las novedades del nuevo protocolo. Por ejemplo, antes de HTTP 1.1, un navegador Web tenía que esperar una respuesta del servidor Web antes de poder emitir otra petición. Con el surgimiento de HTTP 1.1, esto ha dejado de ser así. Un navegador Web puede enviar solicitudes en

³² Setuid y setgid son términos de Unix, abreviaturas para "Set User ID" y "Set Group ID", respectivamente. Setuid, también llamado a veces "suid", y setgid son permisos de acceso que pueden asignarse a archivos o directorios en un sistema operativo basado en Unix.

³³ El CGI por sus siglas en inglés 'Common Gateway Interface' es de las primeras formas de programación Web dinámica. Es utilizado comúnmente para contadores, base de datos, motores de búsqueda, formularios, generadores de e-mail automático y mapas de imágenes, juegos en línea y otros.

paralelo, las cuales ahorran ancho de banda dejando de transmitir las cabeceras HTTP en cada solicitud. De algún modo esto ofrece un estímulo del lado del usuario final porque los archivos solicitados en paralelo aparecerán antes en el navegador.

- **Sencillo, con la configuración basada en un poderoso archivo**

El servidor Apache no posee una interfaz de usuario gráfica para su administración. Se trata de un sencillo archivo de configuración llamado *httpd.config* para algunas distribuciones, en otras como Debian y Ubuntu se llama *apache2.config* y que se puede utilizar para configurar Apache. Únicamente necesita un editor de texto para su configuración. Sin embargo, es lo suficientemente flexible para permitirle repartir la configuración de su host virtual en múltiples archivos para no sobrecargar un único archivo *httpd.config* con toda la gestión de las múltiples configuraciones de servidores virtuales.

- **Soporte para CGI (Common Gateway Interface)**

Apache soporta CGI utilizando los módulos *mod_cgi* y *mod_cgid*. Es compatible con CGI y aporta características extendidas como personalización de las variables de entorno y soporte de reparación de errores o debugging, que son difíciles de encontrar en otros servidores Web.

- **Soporte de FastCGI**

No todo el mundo escribe sus CGI en Perl. ¿Cómo pueden hacer sus aplicaciones CGI más rápidas? Apache también tiene una solución para esto. Utilizando el módulo *mod_fcgi* para implementar un entorno FastCGI dentro de Apache y logrando hacer que las aplicaciones FastCGI arranquen rápidamente.

- **Soporte de host virtuales**

Apache es además uno de los primeros servidores Web en soportar tanto host basados en IP como host virtuales.

- **Soporte de autenticación http**

Apache soporta autenticación básica basada en la Web. Está también preparado para autenticación por resumen, que es algo que los navegadores Web populares ya han implementado. Apache puede implementar autenticación básica utilizando tanto archivos estándar de contraseña como los DBM, llamadas a SQL o llamadas a programas externos de autenticación.

- **Soporte de scripts PHP**

Este lenguaje de script ha comenzado a ser muy utilizado y Apache ofrece un amplio soporte de PHP utilizando el módulo *mod_php*.

- **Soporte de servlets de Java**

Los servlets de Java y las Java Server Pages (JSP) se están convirtiendo en algo muy común en los sitios Web dinámicos. Puede ejecutar servlets de Java utilizando el premiado entorno Tomcat con Apache.

- **Servidor proxy integrado**

Puede convertir Apache en un servidor proxy caché. Sin embargo, la implementación actual del módulo opcional de proxy no soporta HTTP proxy o el último protocolo HTTP 1.1.

- **Estado del servidor y adaptación de registros**

Apache le da una gran flexibilidad en el registro y la monitorización del estado del servidor. El estado del servidor puede monitorizarse mediante un navegador Web. Además, puede adaptar sus archivos de registro a su gusto.

- **Soporte de Server Side Includes (SSI)**

Apache ofrece un conjunto de Server Side Includes que añaden una gran flexibilidad para el desarrollador del sitio Web.

- **Soporte de Secured Socket Layer (SSL)**

Puede crear fácilmente un sitio Web SSL utilizando OpenSSL y el módulo *mod_ssl* de Apache.

- **Soporte para ASP.NET**

La gran libertad de Apache y su amplia comunidad de desarrollo han hecho posible correr las populares aplicaciones desarrolladas en ASP .NET en entornos distintos de Windows y con solo Microsoft Internet Information Server (IIS). En este gran avance ha jugado un papel fundamental el proyecto Mono, con la creación del módulo *mod_mono*.

2.5.3 El Servidor Web que se propone para la migración

Téngase en cuenta que para la migración del sistema, el servidor Web es un componente muy importante. Se necesita un servidor Web cuyas características y prestaciones no sólo se asemejen a las del Microsoft Internet Information Server IIS, si no que de ser posible tenga un rendimiento mayor, sea escalable y capaz de manejar la concurrencia de múltiples usuarios, además de permitir ejecutar aplicaciones ASP .NET. Las características y la madurez de Apache 2 asociado a sus prestaciones y su escalabilidad, lo hacen el servidor Web candidato y propuesto para hacer la migración del sistema. El solo hecho de que sea el servidor Web más difundido en el mundo, que sea software libre y que tenga soporte para correr aplicaciones ASP .NET y todo esto asociado al conjunto de características expuestas con anterioridad lo hacen ser el servidor que se propone para la migración del sistema.

2.6 Servidores de Base de Datos Libres. Propuesta para la migración

2.6.1 La Base de Datos de Akademos

En el Capítulo I se hizo referencia al gestor de base de datos empleado en el desarrollo y la puesta en funcionamiento de Akademos. Microsoft Sql Server 2000 es un completo gestor de base de datos relacional y forma parte de la gama de productos privativos de Microsoft, capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea y ha sido la fuente de almacenamiento de toda la información generada como resultado de los procesos gestionados por Akademos.

Muchas de las tablas de esta base de datos cuentan con más de 323 000 registros relacionados con información de otras tablas, en la actualidad el fichero de datos (MDF)³⁴ de esta base de datos tiene un tamaño de 390 mb, el fichero de los log (LDF)³⁵ llega a crecer hasta 1 GB en un período de tiempo promedio que no excede los 30 días. Estos ficheros crecen como resultado del flujo de información generada por los procesos que el sistema automatiza. La tabla 2.4 muestra algunos detalles de esta base de datos.

Tabla 2.4. Detalles de la base de datos de Akademos.

	Tablas	Vistas	Procedimientos almacenados	Funciones	Diagramas
No.	144	124	52	13	8

En el ámbito de Software Libre son varios los gestores de base de datos con los que se pueden contar, pero dentro de este entorno hay dos gestores que sobresalen por su eficiencia y

³⁴ (Master Database File) sus siglas en inglés, traducido al español, archivo principal de la base de datos. Este fichero contiene toda la información de la base de datos.

³⁵ (Log Database File) sus siglas en inglés, traducido al español, archivo de los log de la base de datos.

capacidad de funcionamiento, proporcionando soluciones a miles de personas, de forma totalmente gratuita, son estos PostgreSQL y MySQL. A continuación se hará una caracterización de ambos gestores donde se expondrán las características más sobresalientes de cada uno, concluyendo con la selección de uno de estos para enfrentar la migración de la base de datos actual de Akademos.

2.6.2 PostgreSQL

Es un sistema de gestión de base de datos objeto-relacional (ORDBMS) resultado de varios años de trabajo y evolución del proyecto Ingres en la Universidad de Berkeley. PostgreSQL se distribuye bajo la licencia BSD³⁶, y utiliza el lenguaje SQL92/SQL99³⁷, ha sido propulsor y creador de muchos conceptos que existen en el modelo objeto-relacional que se incluyen en sistemas gestores de base de datos puramente comerciales y más difundidos en este ámbito. Ya se encuentra liberada la versión 8.2 de este producto.

Características como la orientación a objeto, los tipos de datos, el empleo de funciones, restricciones, triggers, reglas e integridad transaccional, hacen de PostgreSQL un gestor de base de datos objeto-relacional.

Características de PostgreSQL

PostgreSQL cumple completamente el ACID, esta es la propiedad que debe cumplir una base de datos para realizar transacciones seguras. ACID es un acrónimo de Atomicity, Consistency, Isolation and Durability: Durabilidad, Aislamiento, Consistencia e Indivisibilidad en español. PostgreSQL no solo se ejecuta en entornos libres, sino que corre en casi todos los principales sistemas operativos: Linux, Unix, BSDs, Mac OS, Beos, Windows, etc.

³⁶ La licencia BSD es la licencia de software otorgada principalmente para los sistemas BSD (Berkeley Software Distribution). Pertenece al grupo de licencias de software Libre.

³⁷ Son estándares del lenguaje sql.

En contraste a muchos sistemas de base de datos comerciales, es extremadamente común que compañías reporten que PostgreSQL nunca ha presentado caídas en varios años de operación de alta actividad. Por su arquitectura de diseño, escala muy bien al aumentar el número de CPUs y la cantidad de RAM. Soporta transacciones y, desde la versión 7.0, llaves foráneas (con comprobaciones de integridad referencial), además también presenta soporte para triggers y procedimientos almacenado. Soporta los tipos de datos bases y también datos de tipos: fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP), cadenas de bits, etc. Además permite la creación de tipos de datos propios.

PostgreSQL cuenta con una gran comunidad de desarrollo distribuida en todo el mundo. Esto hace posible contar con una amplia documentación pública, organizada, libre y en diferentes idiomas.

El código fuente está disponible para todos sin costo alguno. Si se necesita extender o personalizar PostgreSQL de alguna manera, esto se puede hacer con un mínimo esfuerzo sin costos adicionales en dependencia de los conocimientos y experiencia del desarrollador. Esto es complementado por la comunidad de profesionales y entusiastas de PostgreSQL alrededor del mundo.

PostgreSQL usa una estrategia de almacenamiento de filas llamada MVCC³⁸ esto le proporciona mejor respuesta en ambientes de grandes volúmenes de datos; los principales proveedores de sistemas de base de datos comerciales usan también esta tecnología, por las mismas razones. PostgreSQL disfruta del soporte de numerosas compañías, las cuales auspician desarrolladores y proveen recursos de hosting. Además existe una numerosa comunidad de compañías que ofrecen soporte de PostgreSQL, desde individuos hasta compañías multinacionales. [14]

³⁸ Multi-Version Concurrency Control es una técnica avanzada para mejorar las prestaciones de una base de datos en un entorno multiusuario.

PostgreSQL 8.1

Los nuevos release por lo general son más estables que las versiones anteriores y se basan en la corrección de errores presentes en las precedentes, pero para lograr la estabilidad se necesita de un tiempo después de ser liberada la versión. PostgreSQL; ya cuenta con varios release de su versión 8.2, pero en este trabajo se hará una caracterización de la versión 8.1.

En este release PostgreSQL además de mantener sus funcionalidades básicas, incluidas en versiones anteriores, incorpora nuevas características avanzadas que lo hacen un gestor más profesional, robusto y estable.

Con el uso de roles, ahora el gestor simplifica el manejo de grandes cantidades de usuarios con esquemas complejos de privilegios superpuestos. Las funciones ahora soportan parámetros de entrada (IN), salida (OUT) y entrada/salida (INOUT), lo cual mejora sustancialmente el soporte de lógica compleja y aplicaciones J2EE y .NET. Esta versión implementa “*Compromiso en Dos Fases*” (Two-Phase Commit, 2PC), esta es una característica muy necesaria para aplicaciones WAN y centros de cómputo heterogéneos ya que permite transacciones conformes con ACID a través de servidores distribuidos.

No solo son estas las mejoras que tiene PostgreSQL en su versión 8.1, su gestor de memoria ha sido mejorado de manera que escala casi linealmente con el número de procesadores. Esto conlleva mejoras significativas de rendimiento en sistemas de 8 bits, 16 bits, dual-core y multi-core.

La utilización de Recorridos de Mapas de Bits mediante el cual los índices son convertidos a mapas de bits en memoria cuando es apropiado, otorgando hasta veinte veces más rendimiento en consultas complejas para tablas muy grandes siendo esta otra de las mejoras de esta versión.

Esto ayuda a simplificar la administración de base de datos reduciendo significativamente la necesidad de índices multicolumna.

El Particionamiento de Tablas es otro de los aportes de este release, mediante esta técnica el optimizador de consultas es capaz de evitar recorrer secciones completas de tablas grandes, a través de una técnica conocida como Exclusión por Restricciones. Similar a las características de Particionado de Tablas de otros sistemas gestores de datos, esta característica mejora tanto el rendimiento como la gestión de datos para tablas de varios gigabytes.

Bloqueos Compartidos de Registros, este modelo de bloqueos de PostgreSQL, en esta versión, soporta niveles de concurrencia aún mayores, a través de la adición de candados compartidos a nivel de registro para llaves foráneas. Estos candados compartidos mejorarán el rendimiento de inserción y actualización para muchas aplicaciones OLTP³⁹ de gran concurrencia. [15]

2.6.3 MySQL

A diferencia de PostgreSQL, MySQL es un sistema de gestión de base de datos relacional (RDBMS) y utiliza el lenguaje SQL92/SQL99. MySQL es una idea originaria de la empresa MySQL AB con sede inicial en Suecia, en la actualidad se considera como el gestor de base de datos más popular y más difundido en el mundo del software libre. MySQL usa la licencia GPL⁴⁰, para definir qué es lo que se puede y no se puede hacer con el software para diferentes situaciones. Sin embargo, MySQL también se distribuye bajo una versión comercial para poder integrar este gestor con un software propietario.

³⁹ OnLine Transaction Processing es un tipo de procesamiento de transacciones a través de una red de computadoras

⁴⁰ La GNU GPL (General Public License o licencia pública general) es una licencia creada por la Free Software Foundation a mediados de los 80, y está orientada principalmente a proteger la libre distribución, modificación y uso de software

Características de MySQL

MySQL, al igual que PostgreSQL, cumple con el ACID que es la propiedad que debe cumplir un gestor de base de datos para hacer transacciones seguras como se explicó con anterioridad. MySQL está desarrollado en C y en C++ lo que lo hace verdaderamente liviano y muy rápido. Al igual que PostgreSQL, se puede ejecutar en diferentes plataformas tales como Linux (SuSE y Red Hat), FreeBSD, Sun Solaris (Versiones 8 y 9) y Windows. MySQL hace un uso más fácil de múltiples CPUs, si están disponibles, gracias a que hace un uso más completo de los multi hilos que crea.

Para hacer más eficiente el gestor, MySQL implementa un sistema de reserva de memoria muy rápido basado en hilos y Joins usando un multi-join de un paso optimizado y tablas hash en memoria, que son usadas como tablas temporales. Las funciones SQL están implementadas usando una librería altamente optimizada lo que les permite ser tan rápidas como sea posible. Normalmente no hay reserva de memoria tras toda la inicialización para consultas. El servidor está disponible como un software independiente para utilizarse en un entorno de red cliente/servidor.

MySQL posee un sistema de privilegios y contraseñas que es muy flexible y seguro, el cual permite la verificación basada en el host. Para garantizar la seguridad de las contraseñas todo el tráfico de contraseñas es encriptado cuando se conecta un cliente con un servidor.

Al igual que PostgreSQL, MySQL cuenta con una gran comunidad de desarrollo distribuida en todos los continentes y en varios idiomas que da soporte a través de artículos, noticias y listas de correo. La documentación de este gestor es muy abundante y está disponible de forma libre en el propio sitio de MySQL: <http://www.mysql.com> .

MySQL 5.0

En esta versión de MySQL se incluyen una serie de transformaciones y mejoras que hacen de este sistema un gestor más profesional. A continuación se muestran algunas de las características más esenciales de este nuevo release. Un resultado importante de estos cambios es que MySQL proporciona un mejor cumplimiento del estándar SQL.

En esta nueva versión de MySQL se soportan cursores simples dentro de procedimientos y funciones almacenadas. La sintaxis es la de SQL empotrado. Los cursores no son sensibles, son de sólo lectura, y no permiten scrolling.

Otra de las inclusiones de este release es el MySQL Instance Manager (IM). Este es el demonio que corre en el puerto TCP/IP, el cual provee monitoreo y administración de las instancias del servidor de datos MySQL. MySQL Instance Manager está disponible para sistemas operativos basados en Unix. MySQL Instance Manager se incluye en las distribuciones de MySQL desde la versión 5.0.3, y puede usarse en lugar del script *mysqld_safe* para arrancar y parar MySQL Server, incluso desde una máquina remota.

MySQL 5.0 introduce matemáticas de precisión, esto es, tratamiento numérico que proporciona resultados más precisos y más control sobre valores no válidos que en versiones anteriores.

Otras nuevas funcionalidades que presenta esta versión son los procedimientos almacenados y las funciones, un procedimiento almacenado es un conjunto de comandos SQL que pueden almacenarse en el servidor. Una vez que se hace, los clientes no necesitan relanzar los comandos individuales pero pueden en su lugar referirse al procedimiento almacenado. Esto trae consigo mejoras de rendimiento pues se envía menos información entre el cliente y el servidor.

Las vistas (incluyendo vistas actualizables) fueron introducidas en esta versión del servidor de base de datos MySQL, así como el empleo de triggers y los tipos de datos (BIT y VARCHAR). [16]

2.6.4 El gestor de base de datos que se propone para la migración

Son varias las características de PostgreSQL 8.1 y MySQL 5.0, así como lo son sus ventajas, desventajas y prestaciones, lo cual hace un poco difícil la tarea de seleccionar el servidor de base de datos que se propondrá para la migración del sistema. Es difícil hacer una comparativa entre ambos gestores, para hacerla hay que basarse en las prestaciones que ambos tienen en común, esto resulta algo verdaderamente engorroso puesto que ambos gestores poseen distintas implementaciones.

Para la migración del sistema se necesita un gestor de base de datos cuyas características y prestaciones se ajusten en gran medida a las siguientes condiciones.

- Altos niveles de confiabilidad y escalabilidad.
- Sea estable durante el manejo de grandes volúmenes de información.
- Soporte altos niveles de concurrencias.
- Soporte integridad referencial.
- Soporte procedimientos almacenados o funciones y vistas.

Estas cinco características son las más críticas y son las que debe presentar el gestor que se emplee para la migración. De los dos gestores analizados con anterioridad el gestor, cuyas características son las más semejantes a las necesidades que debe cumplir el gestor que se propondrá para la migración, es PostgreSQL en su versión 8.1. El hecho de que sea un sistema de gestión de base de datos objeto-relacional, que soporte la integridad referencial y que en ambientes en que se gestionen grandes volúmenes de datos su capacidad operacional se

comporte más estable que MySQL, hace que, junto al resto de características anteriormente expuestas en este epígrafe, sea más apropiado para la migración que MySQL.

Por lo tanto, proponemos el uso del gestor PostgreSQL 8.1 y su ejecución sobre un sistema operativo software libre para hacer un uso más óptimo de sus capacidades en el manejo de grandes volúmenes de información.

2.7 Conclusiones

En este capítulo se ha sentado las bases que darán soporte al desarrollo de la migración, la selección de la plataforma, el servidor Web y el gestor de base de datos que cumple con las políticas cubanas respecto al uso de software libre. El empleo de Mono como plataforma de desarrollo, Apache 2 como servidor Web y PostgreSQL en su versión 8.1 para la migración de Akademos a software libre darán paso al desarrollo de la migración propuesta. Los estudios realizados en este capítulo pueden servir de referencia para instituciones o grupos de desarrollo cuyo objetivo sea migrar sistemas con características similares a Akademos.

Capítulo III: Desarrollo de la migración

3.1 Introducción

En el Capítulo II se realizó un estudio de las plataformas de desarrollo libres que se pudiesen emplear para la migración del sistema, así como los gestores de Bases de Datos y los Servidores Web que cumplen con las políticas del uso del software libre en Cuba. Como resultado de este estudio, se propuso para realizar la migración del sistema, como plataforma de desarrollo, a Mono, apoyado en el servidor Web Apache 2 para dar alojamiento al mismo, y como gestor de bases de datos se propuso a PostgreSQL en su versión 8.1.

En este capítulo se tomará como muestra un módulo del sistema, el cual será sometido al proceso de migración. Como se explicó con anterioridad, Akademos ha sido desarrollado bajo la arquitectura de tres capas y para lograr un mejor desarrollo de la migración se decidió dividir este proceso en tres etapas en base de la arquitectura de Akademos. Vale aclarar que el hecho de dividirlo en tres etapas no quiere decir que el proceso necesite ejecutarse en esta secuencia, ha sido simplemente una forma de agrupar aspectos relevantes de la migración tomando en cuenta la relación de estos con cada una de las tres capas.

El módulo piloto

Como ya se comentó, para hacer más demostrativo el desarrollo de la migración se tomará como objeto de estudio un módulo de Akademos, este módulo será el módulo de Matrícula. Aunque en el Capítulo I se da una panorámica de este módulo, en la tabla 3.1 se recogen algunos detalles de este módulo.

Tabla 3.1: Componentes del Módulo de Matrícula.

Módulo	Interfaz			Negocio	Acc. Datos
	Clases Ayuda Interfaz (CAI)	Páginas	Controles de Usuarios (CU)		
Matrícula	8	37	21	7	16

El módulo de matrícula está completado en su totalidad para una primera versión, actualmente es el módulo más estable del sistema y el menos propenso a cambios sistemáticos o a largo plazo, ha sido empleado para la matrícula masiva de la UCI durante los cursos académicos (2004-2005), (2005-2006) y (2006-2007), y recientemente fue empleado en las tres facultades regionales con que cuenta dicha universidad para la realización del mismo proceso. Gracias a la estabilidad y madurez de este módulo, asociado a la experiencia del grupo de desarrollo de Akademos en procesos que engloban a grandes cantidades de actores, este curso se logró hacer la matrícula de más de 2500 estudiantes de manera simultánea desde aproximadamente 200 puestos de matrícula, distribuidos en 10 laboratorios de la universidad en apenas 72 horas de trabajo. El grupo de desarrollo de Akademos, a petición de la Dirección de Informatización de la UCI, se ha trazado como meta hacer este proceso para el curso (2007-2008) con este módulo ejecutándose sobre Mono, empleando a Apache 2 como servidor Web y utilizando a PostgreSQL en su versión 8.1 como gestor de bases de datos.

Monodevelop como IDE de Desarrollo.

En el epígrafe 2.4.2 del Capítulo II se hace una descripción del Monodevelop y su estado actual. Si cierto es que Monodevelop no cuenta en la actualidad con una version 1.0 y no se pueden comparar sus prestaciones con las del Visual Studio, es bueno reconocer que su version actual

es bastante estable en comparación con versiones anteriores las cuales presentaban varios problemas, además, a nivel de laboratorio ha permitido al grupo de desarrollo de Akademos hacer la migración de este módulo de prueba sin presentar grandes complicaciones. Aunque en un epígrafe de este capítulo se hará alusión a la nueva definición del esquema de seguridad que se propone para la migración, se puede adelantar que este esquema fue desarrollado, probado y puesto en marcha utilizando como IDE de desarrollo el Monodevelop. El Anexo 3 muestra el Monodevelop en su modo operativo durante la compilación de las clases que conforman el nuevo esquema de seguridad de Akademos.

3.2 Desarrollo de la migración de la capa de presentación

Cumpliendo con los objetivos de este capítulo y tomando como módulo piloto para la realización de la migración al módulo de Matrícula, veremos dos puntos importantes que involucran a la capa de presentación dentro del desarrollo de la migración.

- Configuración del servidor Web Apache 2 para dar alojamiento al Sistema.
- Mejora de la interfaz del módulo de Matrícula para que sea compatible con diferentes navegadores.

Configuración del servidor Web Apache 2 para dar alojamiento al Sistema.

En el epígrafe 2.5 del Capítulo II se da como propuesta para la migración del sistema que el mismo se ejecute sobre el servidor Web Apache 2. Dada la selección de Mono como plataforma de desarrollo, nuestro servidor Web tiene que brindar soporte para aplicaciones Web ASP .NET. Apache 2, para incorporar esta funcionalidad, hace uso de un módulo especial desarrollado por el equipo de Mono, el módulo Mod_Mono. Este módulo transfiere las solicitudes de páginas ASP .NET a un programa externo, Mod_Mono_Server, el que se encarga de procesar las solicitudes.

La comunicación entre el módulo de Apache 2 y Mod_Mono_Server se establece a través de un socket Unix o socket TCP.[17]

Para las pruebas reales realizadas en la migración del sistema, se utilizó Ubuntu⁴¹ en su versión 6.10 conocida como “Edgy Eft”. Fue necesario instalar el paquete Mono y el paquete Monodevelop con todas sus dependencias, culminando así la correcta instalación del entorno de programación de Mono. Para lograr el soporte nativo de las páginas ASP .NET bajo Apache 2, como primer paso se instaló el Apache 2. Luego se procedió a instalar los paquetes mono-xsp⁴², libapache2-mod-mono⁴³ y todas sus dependencias; y aunque no es obligatorio, es recomendable instalar el archivo asp.net-examples, el que consiste en un conjunto de archivos de ejemplo de ASP .NET que nos servirán para comprobar el correcto funcionamiento del módulo Mod_Mono.

Mejora de la interfaz del módulo de Matrícula para que sea compatible con diferentes navegadores.

Este módulo, como el resto de los módulos de Akademos, fue desarrollado fundamentalmente para su utilización con el Microsoft Internet Explorer⁴⁴, siendo este navegador la principal herramienta para llevar a cabo las pruebas del sistema. Lamentablemente muchos controles y estilos desarrollados con el Visual Studio .NET 2003 no son cien por ciento compatibles con otros navegadores existentes que son muy populares y tienen la ventaja de ser multiplataforma, entre los que se destacan el Mozilla Firefox⁴⁵, Netscape⁴⁶ y el Opera⁴⁷.

⁴¹ Sistema Operativo, basado en Debian GNU/Linux, especial para computadoras de escritorio y servidores. La versión 6.10 fue lanzada en octubre del 2006, con mejoras en la interfaz y en el sistema de arranque entre otras.

⁴² Servidor Web sencillo para ejecutar aplicaciones ASP .NET escrito en C#.

⁴³ Permite ejecutar páginas ASP .NET en UNIX con Apache 2 y Mono

⁴⁴ Navegador gráfico propietario para la Web desarrollado por Microsoft e incluido en su línea de sistemas operativos desde 1995. Se encuentra actualmente en su versión 7.

⁴⁵ Navegador gráfico software libre para la Web desarrollado fundamentalmente por la corporación Mozilla. Es una navegador multiplataforma. Actualmente su versión estable es la 2.0.0.3.

⁴⁶ Navegador gráfico para la Web desarrollado por la empresa Netscape Communications. Se encuentra actualmente en su versión 8.1.

Para lograr mayor compatibilidad con los navegadores de Internet es recomendable seguir las especificaciones establecidas por la W3C, consorcio internacional que produce estándares para la Web. Gracias a estos estándares establecidos por este consorcio se evita que cualquier fabricante se pueda adueñar del monopolio de la Web, facilitando así la creación de navegadores y herramientas de desarrollo Web por cualquier empresa, grande o pequeña. El html⁴⁸ de las páginas Web creadas utilizando el IDE de desarrollo Visual Studio .NET 2003 no cumple a cabalidad, en algunos casos, con las reglas establecidas en esos estándares.

Para lograr aumentar la compatibilidad del Módulo de Matrícula fue necesario modificar el archivo de configuración *web.config*, archivo en el que se definen aspectos de seguridad, de compilación y tratamiento de errores, entre otros. A este archivo se le tuvo que adicionar una sección [Ver Anexo 4], sección que este archivo admite para facilitar la identificación del navegador que se utiliza en el cliente, permitiendo así diferenciar entre navegadores como Netscape, Mozilla – Firefox, Konqueror, Opera y el Internet Explorer. Esto es necesario hacerlo para cada proyecto o sitio Web que exista en el servidor o de lo contrario se puede modificar, de los archivos de configuración del Framework de .NET, el archivo *machine.conf*, el cual se centra en aspectos relacionados con la configuración de todos los proyectos Web que se ejecutan en el servidor. Además fue necesario hacer cambios directos [Ver Anexo 5] en el html de algunas páginas que presentaban algunos atributos que no son del todo compatibles con los navegadores de Internet más populares.

⁴⁷ Navegador gráfico propietario para la Web desarrollado por la empresa de software Opera. Es un navegador multiplataforma. Actualmente su versión estable es la 9.20.

⁴⁸ Lenguaje de Etiquetado de Hipertexto (HyperText Markup Language). Es un lenguaje común que se utiliza para la publicación de hipertexto en la Web.

3.3 Desarrollo de la migración de la capa de negocio

La seguridad es un punto clave para los desarrolladores y arquitectos de sistemas. Las aplicaciones que gestionan y almacenan información sensible deben adoptar medidas de seguridad frente a ataques de personas que intenten apropiarse de dicha información o simplemente alterar el estado de la misma. Un sistema distribuido con más de doce mil usuarios y que gestiona toda la información generada como resultado de los procesos docentes de un centro de estudio debe tener un robusto diseño de seguridad para garantizar la integridad de dicha información.

ASP .NET implementa la autenticación utilizando proveedores para la misma; se trata de módulos de código que comprueban las credenciales e implementan otras funcionalidades de seguridad como, por ejemplo, la generación de cookies⁴⁹. ASP .NET admite los siguientes proveedores de autenticación: [18]

- **Autenticación por formularios.** El uso de este proveedor permite que las solicitudes sin autenticación se redirijan a un formulario HTML especificado utilizando el redireccionamiento del lado del cliente. A continuación, el usuario podrá proporcionar las credenciales de inicio de sesión y volver a enviar el formulario al servidor. Si la aplicación autentica la solicitud (empleando la lógica específica de la aplicación), ASP .NET envía un cookie que contenga las credenciales o una clave para volver a adquirir la identidad del cliente. Las solicitudes posteriores se envían con el cookie en el encabezado de las mismas, lo que significa que son innecesarias futuras autenticaciones.

⁴⁹ Es un fragmento de información que se almacena en el disco duro del visitante de una páginaWeb a través de su navegador, a petición del servidor de la página. Esta información puede ser luego recuperada por el servidor en posteriores visitas.

- **Autenticación de Pasaporte.** Se trata de un servicio de autenticación centralizado que proporciona Microsoft y que ofrece un medio único de inicio de sesión, así como servicios de suscripción para los sitios de participación. ASP .NET, junto al kit de desarrollo de software (SDK) de Microsoft Pasaporte, facilita una funcionalidad similar a los usuarios de Passport y de autenticación por formularios.
- **Autenticación de Windows.** Este proveedor hace uso de las capacidades de autenticación de Microsoft Internet Information Server (IIS). Una vez que IIS completa su autenticación, ASP .NET utiliza el identificador del usuario autenticado para autorizar el acceso.

El esquema de seguridad es el componente de Akademos que más se vio afectado durante el proceso de migración de la capa de negocio. Antes de entrar en detalles sobre los cambios a los que fue sometido el mismo, sería bueno hacer una breve caracterización de su estado actual y bajo qué circunstancias fue construido. Akademos como ya se ha comentado fue concebido para ejecutarse sobre IIS pues su plataforma de desarrollo así lo requería. El uso de este servidor Web trae consigo la factibilidad de usar el proveedor de seguridad de *Autenticación de Windows*, anteriormente mencionado, lo cual es una gran ventaja cuando el sistema se ejecuta en servidor Windows y se cuenta con un LDAP, ventaja que se puede explotar en la UCI. El esquema de seguridad del sistema tiene una fuerte dependencia del uso del LDAP y mediante él se controlan todos los usuarios y grupos de dominio que tienen permisos en el sistema. Para el manejo de los permisos los componentes de seguridad de Akademos hacen uso del usuario activo en el sistema dejando el modo de autenticación al IIS. La información que se necesite sobre el usuario activo se obtiene del LDAP a través de la clase CUsuarioActivo que pertenece al namespace (espacio de nombre) *UtilesSeguridad*. Para una mejor comprensión del modelo de Autenticación de Akademos [Ver Anexo 6].

El flujo de la seguridad en un sistema que implemente arquitectura de tres capas debe persistir entre todos sus componentes ya que no es una buena práctica centrar la seguridad en sólo una de las capas y muchos autores recomiendan hacer énfasis en la seguridad desde la interfaz hasta la capa más baja, la de acceso a datos, sin descuidar los riesgos en la capa de negocio.

El esquema de seguridad de Akademos ha sufrido algunos cambios pues inicialmente estaba concebido para ser utilizado bajo *Autenticación de Windows*. Con el uso de Apache 2 como servidor Web y el sistema ejecutándose sobre un servidor GNU/Linux esta ventaja que proporciona el IIS no existe. En este trabajo los componentes de seguridad del sistema fueron rediseñados para la utilización de autenticación mixta o sea *Autenticación de Windows* o *Autenticación por formularios*, esto permite que la seguridad de Akademos sea transparente al servidor Web que se utilice para dar alojamiento al sistema.

Este nuevo esquema de seguridad fue construido con el objetivo de permitir la migración del sistema en el menor tiempo posible sin que se viesen afectadas las restantes capas de su arquitectura. Las clases que conforman el namespace *UtilesSeguridad* en su nuevo diseño quedan recogidas de manera detallada en el Anexo 7 y el Anexo 8. Este conjunto de clases son las encargadas de dar soporte al esquema de seguridad mixta que se propone para la migración del sistema [Ver Anexo 9]. El uso del tipo de seguridad se define en el fichero de configuración *constantes.xml*⁵⁰ ubicado en la capa de negocio.

Para hacer uso de este modelo se definieron cinco nuevas constantes en el fichero de configuración *constantes.xml*, se realizó un cambio en la capa de presentación de todos los módulos, dos cambios en la clase *CPermiso*, un cambio en la clase *CIncidencia* y se creó un proyecto Web sencillo con una página para la autenticación de los usuarios.

⁵⁰ Este fichero de configuración es utilizado en la capa de negocio del sistema para definir un conjunto de constantes. Ejemplo (cadenas de conexión a la BD, ubicación de la carpeta de imágenes, dirección del repositorio de fotos, etc.)

Las nuevas constantes se describen a continuación en la tabla 3.2.

Tabla 3.2. Constantes para el nuevo esquema de seguridad

Nombre	Valor	Descripción
LDAP	LDAP://UCI.CU	Nombre del LDAP donde será implantado el sistema.
UsuarioLDAP	CN=ad search,OU=Systems,OU=UCI Domain Impersonals,DC=uci, DC=cu	Usuario que se utiliza para encuestar el LDAP.
PassUserLDAP	*****	Contraseña del usuario que se utiliza para encuestar el LDAP.
Autentication_Mod	Forms / Windows	Modo de Autenticación.
Nomb_Cookie	AuthCookie	Nombre de la Cookie que se creará si la autenticación es por formulario.

Estos valores de las constantes son los valores utilizados durante la etapa de desarrollo y despliegue en la UCI.

Aunque en este capítulo se dedica un epígrafe a los cambios que sufre la capa de presentación durante la migración, hay un cambio producto del rediseño del esquema de seguridad del sistema. Este cambio surge de la necesidad de controlar los roles al que pertenece el usuario activo en el sistema. Anteriormente se utilizaba el método *IsInRole ()* que es un método de la clase interfaz *Iprincipal* del namespace `System.Security.Principal` que pertenece a la biblioteca de clases de Microsoft .NET Framework y ahora se propone la utilización del método *IsInRole ()* de la clase *CInfoUsuarioActivo* perteneciente al nuevo esquema de seguridad. Este método es utilizado con frecuencia desde los controles de usuario `CUTareas.ascx`, `CUTareasH.ascx` y `CUMenu.ascx` para la gestión de vínculos a interfaces del sistema a las que

solo puede acceder uno o varios roles. Esto mismo pasa con los métodos *EnRolAD()* y *Nombre()*, que se utilizan en *Cpermiso* y *Cincidencia* y que fueron cambiados por *UsuarioEstaEnRol()* y *ObtenerLoguinUA()*, respectivamente.

Como último cambio, a raíz del rediseño del esquema de seguridad surgió un nuevo módulo el cual será el encargado de permitir la autenticación de los usuarios al entrar al sistema. Este módulo es muy sencillo, sólo cuenta con una página y un control de usuario donde los usuarios del sistema introducen sus credenciales de domino (usuario y contraseña) las cuales serán validadas contra el LDAP [Ver Anexo 10]. El intercambio de información entre los clientes y este módulo será a través del protocolo https para garantizar la integridad de las credenciales del usuario del sistema.

3.4 Desarrollo de la migración de la capa de datos

Ya migrada la capa de presentación y la capa de negocio solo restaría desarrollar la migración de la capa más baja del sistema, la capa de datos. Esta capa se divide en dos componentes esenciales: la base de datos del sistema y las clases de acceso a datos. La arquitectura de Akademos es un factor que ayudó mucho en el desarrollo de la migración, por su bajo acoplamiento y la reutilización de componentes, que han sido factores de gran ventaja y se explotaron al máximo durante el proceso de migración.

De los dos componentes esenciales de esta capa, primeramente se realizó la migración de la base de datos. Aunque PostgreSQL en gran medida cumple con mucha de las características de SQL, el equipo de desarrollo de Akademos enfrentó algunos contratiempos a la hora de realizar la migración del gestor. Antes de detallar estos contratiempos y la solución a los mismos, se detallará como se logró migrar la base de datos a un servidor PostgreSQL 8.1. Para la migración de la base de datos se empleó una técnica muy conocida y empleada para migración de SQL a

PostgreSQL. Para este proceso, se utilizó una Herramienta Case⁵¹ muy empleada para el diseño y modelación de bases de datos, el ER/Studio en su version 6.0.1. Con el ER/Studio se le realizó la ingeniería inversa a la base de datos física de Akademos para obtener el modelo lógico. Debido a que el ER/Studio no permite generar para PostgreSQL se aprovechó las similitudes que este gestor presenta con MySQL permitiendo utilizar el modelo lógico anteriormente obtenido de la ingeniería inversa para la generación de un modelo físico para MySQL. El script generado para MySQL no difiere mucho de lo que se necesita para generar la base de datos en PostgreSQL. Antes de correr este script se realizaron algunos cambios en el mismo que se muestra en la siguiente tabla.

Tabla 3.1: Transformaciones para PostgreSQL en script generado para MySQL.

MySQL	PostgreSQL
autoincrement	serial
datetime	date
tinyint	smallint
longtext	text

Con este script se generó la nueva base de datos de Akademos en un servido PostgreSQL. Posteriormente, se fueron migrando los datos de todas las tablas de manera jerárquica atendiendo a la integridad referencial entre las mismas. Para hacer esta migración se utilizó el Microsoft SQL Enterprise Manager⁵². El módulo de expediente de Akademos da la posibilidad de que la información que se almacene en el mismo sea mediante documentos de tres diferentes formatos; uno de estos formatos de almacenamiento es el de documentos basados en plantillas [Ver Epígrafe 1.6]. Esta funcionalidad del sistema da la posibilidad de que el usuario defina en

⁵¹

(Computer Aided Software Engineering), Ingeniería de Software Asistida por Ordenador son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero.

⁵² Es una consola de gestión y motorización 32-bit visual basada en Windows. Es un punto de configuración y gestión de base de datos remotas y locales.

una plantilla XHTML⁵³ un documento que posteriormente formará parte del expediente del estudiante, estas plantillas a nivel de bases de datos será una tabla contenedora de toda la información que en su diseño el usuario del sistema defina [Ver Anexo 11]. Aquí surge la primera problemática, la libertad de que el usuario pueda definir los campos que conforman la plantilla provoca que en ocasiones los nombres de estos campos sean compuestos y separados por espacios o contengan signos de puntuación Ej. (P Nombre o P. Nombre) que son una misma manera de referirse al campo Primer Nombre. El uso de nombres compuestos separados por espacios o por signos de puntuación en los campos de las plantillas, dio lugar a una problemática a la hora de migrar la información en tablas asociadas a plantillas como [Hoja de Matrícula] y [Hoja de Prematrícula], pues durante la migración el *driver* (controlador) que permite la el intercambio de información entre ambos gestores Sql y PostgreSQL, asume como campos independientes los nombres que conforman el nombre compuesto separado por espacios o signos de puntuación. En el caso de P. Nombre el driver asume que P es un campo de la tabla y Nombre es otro campo. La solución dada a este problema fue renombrar los campos cuyos nombres contuviesen espacios o signos de puntuación en las tablas de la base de datos Sql, hacer la migración de la información y posteriormente regresar a su estado inicial los nombres de los campos que fueron renombrados.

El diseño inicial de la base de datos del sistema proponía evitar en gran medida la utilización de procedimientos almacenados, con el objetivo de hacer más sencilla y menos costosa la migración a un gestor que no soportase tales funcionalidades, en su lugar se utilizaban consultas parametrizadas como una técnica de seguridad para evitar que se ejecutasen consultas que pudiesen poner en riesgo la integridad de la información. Aunque esta decisión de no utilizar procedimientos almacenados es algo polémica desde distintos puntos de vista, no es objetivo de este trabajo hacer una crítica del diseño de la base de dato. Haciendo hincapié en el problema

⁵³ Acrónimo inglés de eXtensible Hypertext Markup Language (lenguaje extensible de marcado de hipertexto), es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas web. XHTML es la versión XML de HTML, por lo que tiene, básicamente, las mismas funcionalidades, pero cumple las especificaciones, más estrictas, de XML

real, independientemente de la propuesta de diseño inicial de la base de datos, con el tiempo y la maduración del equipo de desarrollo del sistema fue inevitable que se comenzasen a utilizar a pequeña escala los procedimientos almacenados; las grandes búsquedas y recuperación de información se vieron favorecidas en rapidez de ejecución gracias al empleo de esta funcionalidad que nos brindaba SQL. Este es el segundo contratiempo al que se enfrentó el equipo de desarrollo de Akademos durante la migración de la base de datos, pues los procedimientos implementados en SQL no podían ser exportados a PostgreSQL de forma automática. Como solución a este problema no quedó otra alternativa que reprogramar estos procedimientos almacenados como funciones en PostgreSQL en sus diferentes lenguajes (*plpgsql*, *sql*, *internal*, *c*).

Akademos cuenta con una única clase encargada de interactuar con la base de datos, esta clase nombrada CEntidad contiene un conjunto de métodos encargados de realizar operaciones de insertar, modificar, eliminar y obtener información de una tabla de la base de datos, así como la ejecución de procedimientos almacenados [Ver Anexo 12]. De esta clase hereda otra clase por cada tabla de la base de datos. CEntidad es el segundo componente de la capa de acceso a datos que sufrió modificaciones durante el proceso de migración; esta clase fue construida inicialmente para interactuar con un gestor de base de datos Sql por lo que fue reprogramada para interactuar con un gestor PostgreSQL. Los cambios realizados en la reprogramación de esta clase fueron mínimos. Dicha clase cuenta con tres métodos únicos que son los encargados de ejecutar las sentencias Sql y los procedimientos almacenados. El primero de estos métodos es *Ejecutar()* que recibe como parámetro un dato de tipo cadena que se ejecutará como una consulta Sql; este método no retorna ningún valor. El segundo es una sobrecarga a este *Ejecutar()* que recibe un dato de tipo cadena que representa el nombre de un procedimiento almacenado y otro dato de tipo cadena que representa los parámetros que se le pasan al mismo,

devuelve un DataSet⁵⁴. El tercer método es *EjecutarSeleccion()* este método recibe como parámetro un dato de tipo cadena que se corresponde a una sentencia Sql que se ejecutará, este método devuelve un DataSet. Estos tres métodos utilizaban objetos del namespace System.Data.SqlClient que pertenece a la biblioteca de clases de Microsoft .NET Framework y para lograr la comunicación con la base de datos PostgreSQL se utilizó el namespace Npgsql.

Npgsql

Npgsql es el proveedor de datos de PostgreSQL para la plataforma .NET. Permite que cualquier programa desarrollado para el framework de .NET pueda acceder el servidor de base de datos Postgresql. Está implementando completamente en código C #. Funciona con Postgresql 7 y versiones superiores. [19]

3.5 Desarrollo de la migración de los Web Services XML

Akados puede considerarse como el repositorio digital de toda la información de los estudiantes de la UCI. Muchos sistemas que brindan servicios a la comunidad universitaria se nutren de la información alojada en Akados. Durante el diseño de Akados se propuso para su integración con estos sistemas, la utilización de los Web Services XML. Estos Servicios Web también fueron migrados a software libre sin ningún contratiempo.

3.6 Conclusiones

En este capítulo se ha realizado el desarrollo de la migración a software libre del módulo de matrícula de Akados, utilizando la tecnología, el servidor Web y el gestor de base de datos propuesto para la migración del sistema como resultado del estudio realizado durante la elaboración del Capítulo II. Además se ha empleado para el desarrollo de la migración de este

⁵⁴ Un DataSet guarda información en un entorno desconectado. Después de que usted establece una conexión con una Base de Datos entonces puede acceder a sus datos. Prácticamente un DataSet viene a ser una caché de memoria interna de datos recuperados de un origen de datos.

módulo piloto el Monodevelop como IDE de desarrollo. Este capítulo ha permitido demostrar la factibilidad de la propuesta para la migración realizada en el Capítulo II.

Conclusiones generales

A partir de la investigación realizada se arriba a las siguientes conclusiones:

- En la UCI y en Cuba no existe una estrategia de migración que se adecue a las necesidades de migración del Sistema Automatizado para la Gestión Académica – “Akademos”.
- La estrategia de migración propuesta plantea la utilización de Mono como plataforma de desarrollo, Apache 2 como servidor Web y PostgreSQL en su versión 8.1 como gestor de base de datos.
- La estrategia de migración propuesta no pone en riesgo ninguna de las funcionalidades del Sistema Automatizado para la Gestión Académica - “Akademos”, ni la integración de este con el resto de los sistemas de la UCI.
- La utilización de la arquitectura de tres capas (presentación, negocio y acceso a datos) y de una clase intermedia entre las clases de la última capa y la base de datos permite una migración más transparente, con menor consumo de recursos y de tiempo.
- Este trabajo puede servir de referencia para grupos de desarrollo que intenten asumir la migración de sistemas con características similares a Akademos.

Recomendaciones

- Realizar un plan para hacer la migración de todos los módulos de Akademos, tomando como punto de partida los módulos más utilizados por la comunidad universitaria.
- Realizar la matrícula del curso 2007-2008 con el módulo de matrícula migrado a software libre.

Bibliografía citada

- [1] KHERIAN SOFT. *Agora-Software de gestión de academias y centros de formación* Última actualización: 2006. [Consultado el: 10 de noviembre de 2006]. Disponible en: http://www.kherian.com/agora_asistente_ediciones.htm.
- [2] MICROSOFT. *Introducción a Microsoft .Net, Que es .Net* Última actualización: 2003. [Consultado el: noviembre de 2006]. Disponible en: <http://www.microsoft.com/latam/net/introduccion/quees.asp>.
- [3] MICROSOFT. *Introducción a Microsoft .Net, Que son los Web Services XML* Última actualización: 2003. [Consultado el: noviembre de 2006]. Disponible en: <http://www.microsoft.com/latam/net/introduccion/webservices.asp>.
- [4] WIKIMEDIA FOUNDATION. *Internet Information Services* Wikipedia Última actualización: 17 April de 2007. [Consultado el: noviembre de 2006]. Disponible en: <http://es.wikipedia.org/wiki/IIS>.
- [5] CUBA VS BLOQUEO. *Muros virtuales (I)* Última actualización: 13 octubre 2005. [Consultado el: 4 abril 2007 de 2007]. Disponible en: <http://www.cubavsbloqueo.cu/Default.aspx?tabid=1085>.
- [6] BRASIL, PSL. *Declaración de La Habana sobre el uso de Software Libre y Abierto* Última actualización: 8 de febrero de 2004. [Consultado el: 7 abril de 2007]. Disponible en: <http://www.softwarelivre.org/articles/43>.
- [7] Meneses Abad Abel, Soto López Nilet, Rodríguez Montero Rafael. *Informatización Para la Sociedad Cubana desde la perspectiva ideológica del uso de las plataformas Software*

Libre. 15 p.

- [8] MONO. *Mono* [Página Web]. [Consultado el: 25 de marzo de 2007]. Disponible en:
http://www.mono-project.com/Main_Page.
- [9] MONO. *Software* [Página Web]. [Consultado el: 25 de marzo de 2007]. Disponible en:
<http://www.mono-project.com/Software>.
- [10] MONO. *Gui Toolkits* [Página Web]. [Consultado el: 25 de marzo de 2007]. Disponible en:
http://www.mono-project.com/Gui_Toolkits.
- [11] MONO. *Architecture Overview* [Página Web]. [Consultado el: 26 de marzo de 2007].
Disponible en: http://www.monodevelop.com/Architecture_Overview.
- [12] NETCRAFT. *February 2007 Web Server Survey* Última actualización: febrero de 2007.
[Consultado el: febrero de 2007]. Disponible en:
http://news.netcraft.com/archives/2007/02/02/february_2007_web_server_survey.html.
- [13] KABIR, M. J. *La Biblia de Servidor Apache 2*. Madrid: Anaya Multimedia, 2003. 864 p.
- [14] TIENDALINUX. *Ventajas de PostgreSQL* Última actualización: 31 de abril 2003.
[Consultado el: abril de 2007]. Disponible en:
http://soporte.tiendalinux.com/portal/Portfolio/postgresql_ventajas_html.
- [15] TIENDALINUX. *PostgreSQL 8.1 Lanzado* Última actualización: 9 de noviembre de 2005.
[Consultado el: abril de 2007]. Disponible en:
http://soporte.tiendalinux.com/portal/News/news_item.2005-11-09.4671516085.

- [16] AB, M. *MySQL 5.0 Reference Manual* Última actualización: 4 de marzo de 2007.
[Consultado el: abril de 2007]. Disponible en:
<http://dev.mysql.com/doc/refman/5.0/es/index.html>.
- [17] MONO. *Mod_mono* [Página Web]. [Consultado el: 12 de abril de 2007]. Disponible en:
http://www.mono-project.com/Mod_mono.
- [18] MICROSOFT. *Autenticación en ASP .NET: Directrices de seguridad para .NET* Última actualización: 13 de enero de 2003. [Consultado el: marzo de 2007]. Disponible en:
<http://www.microsoft.com/spanish/msdn/articulos/archivo/261001/voices/authaspdotnet.asp>.
- [19] Npgsql. *What is Npgsql* Última actualización: 8 de octubre de 2006. [Consultado el: abril de 2007]. Disponible en: <http://npgsql.projects.postgresql.org>.

Bibliografía consultada

1. *Consortio World Wide Web Wikipedia* Última actualización: 27 de abril de 2007 [Consultado el: 20 de abril de 2007]. Disponible en: <http://www.w3c.es>.
2. KABIR, M. J. *La Biblia de Servidor Apache 2*. Madrid: Anaya Multimedia, 2003. 864 p.
3. KONQUEROR.ORG. *Konqueror - Web Browser* [Página Web]. [Consultado el: 20 de abril de 2007]. Disponible en: <http://konqueror.org>.
4. LEIVA ZUÑIGA YURISLEIDYS, R. R. D. *Linux y el Software Libre, una alternativa real y económica para la UCI*. UCI: 2006. 11 p.
5. MUNZENMAYER, V. *Soporte para páginas ASP.NET en Apache 2* Última actualización: 24 de julio 2006. [Consultado el: diciembre de 2006]. Disponible en: <http://vmunzenmayer.blogspot.com/2006/07/soporte-para-paginas-aspnet-en-apache2.html>.
6. NETSCAPE-COMMUNICATIONS. *Netscape* [Página Web]. [Consultado el: 20 de abril de 2007]. Disponible en: <http://browser.netscape.com/ns8/product/default.jsp>.
7. NIEL M. BORNSTEIN, E. D. *Mono: A Developer's Notebook*. O'Reilly, 2004. 302 p.
8. PECOS, N. *Comparativa MySQL vs PostGreSQL* Última actualización: 6 de julio 2002. [Consultado el: marzo de 2007]. Disponible en: <http://www.netpecos.org/docs/>.
9. RIORDAN, R. M. *Aprenda Programación en Microsoft SQL Server 2000 ya*. Madrid: McGraw-Hill, 2001. 642 p.

10. WIKIMEDIA FOUNDATION, I. *Application framework* Wikipedia Última actualización: 18 abril 2007. [Consultado el: 7 de abril de 2007]. Disponible en:
http://en.wikipedia.org/wiki/Application_framework.

11. WIKIMEDIA FOUNDATION, I. *Internet Explorer* Wikipedia Última actualización: 29 abril de 2007. [Consultado el: 14 de abril de 2007]. Disponible en:
http://en.wikipedia.org/wiki/Internet_explorer.

Anexos

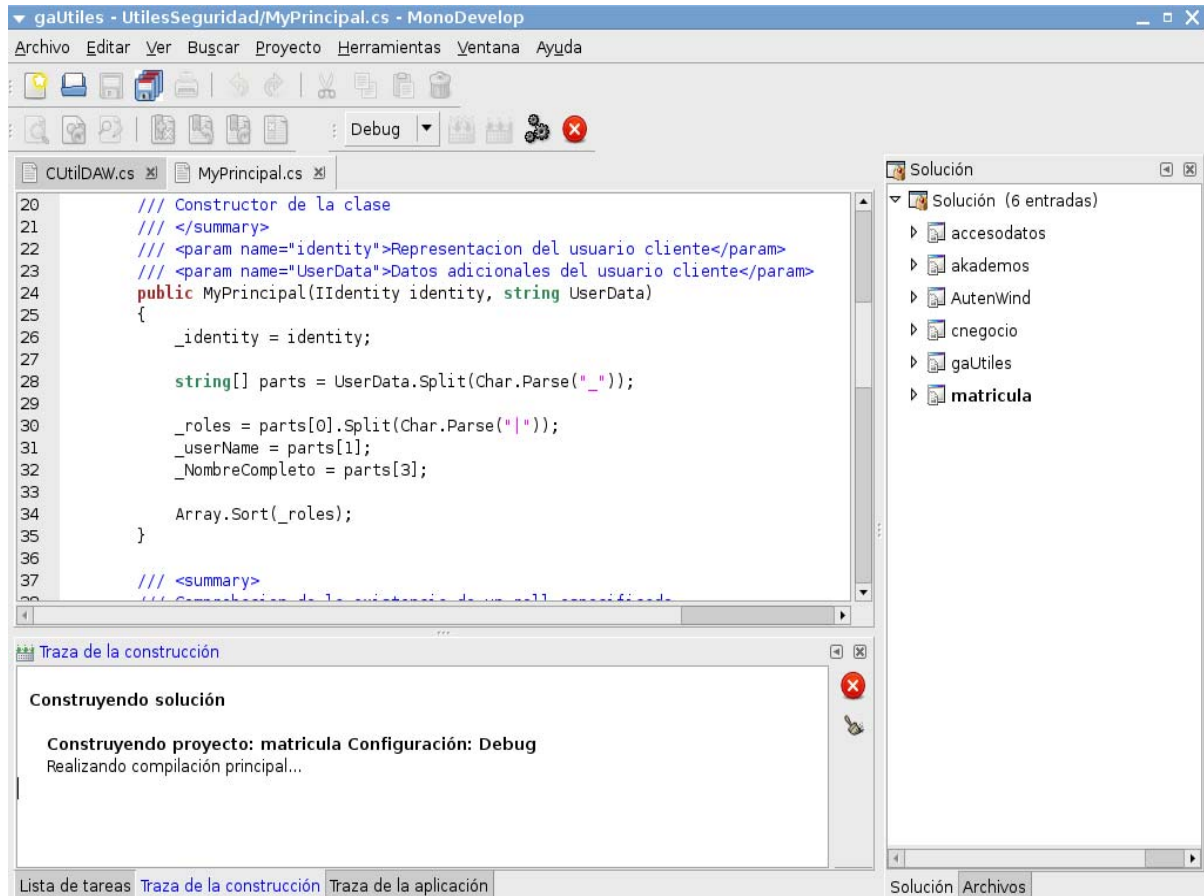
Anexo 1: Sistemas que comparten alojamiento en el IIS con Akademos

Sistemas	Descripción
Alojamiento	Sistema para la gestión de alojamiento.
Béisbol	Sistema para mostrar estadísticas de la serie nacional de Béisbol.
Ciudadano	Servicios Web que permiten el intercambio de información con la base de datos de Ciudadano.
Comedor	Sistema para la gestión de los comedores.
Directorio	Directorio de la Intranet universitaria
Identificación	Sistema de identificación.
Inventario	Sistema para el control de inventario
Relaciones Internacionales	Sistema de gestión de relaciones internacionales
Seguridad y Protección	Sistema de seguridad y protección
Trabajadores	Sistema para la gestión de trabajadores y tercerizados
Transporte	Sistema para la gestión de Transporte
Ucidrez	Portal del ajedrez en la UCI

Anexo 2: Base de datos que coexisten en el servidor SQL Server 2000 con Akademos

Base de datos	Descripción
Almacén (transporte)	Base de datos del sistema de transporte.
Alojamiento	Base de datos del sistema de alojamiento.
Béisbol	Base de datos del sistema de béisbol.
Ciudadano	Base de datos Ciudadano.
Comedor Externos	Bese de datos del sistema de gestión de los comedores.
Estado Acceso	
Fénix (Transporte)	Base de datos del sistema de transporte.
Históricos (Transporte)	Base de datos del sistema de transporte.
Identificación	Base de datos del sistema de identificación.
Inventario	Base de datos del sistema de inventarios.
Registro (Transporte)	Base de datos del sistema de transporte.
Relaciones Internacionales	Base de datos del sistema de seguridad y protección.
Renta (Transporte)	Base de datos del sistema de transporte.
Residencia	Base de datos del sistema de residencia.
Trabajadores	Base de datos del sistema de trabajadores y tercerizados.
Transporte	Base de datos del sistema de transporte.
Ucicomedor	Bese de datos del sistema de gestión de los comedores.
Ucidrez	Base de datos del portal del ajedrez en la UCI
Visitas (Seguridad y Protección)	Base de datos del sistema de seguridad y protección.

Anexo 3: Monodevelop durante la depuración del nuevo esquema de seguridad



Anexo 4: Sección de Identificación de Navegador de Internet

```
<browserCaps>
```

```
<!-- GECKO Based Browsers (Netscape 6+, Mozilla/Firefox, ...) -->
<case match="^Mozilla/5\.0 \([^\)]*\) (Gecko/[-\d]+)(?'VendorProductToken'
(?'type'^[\d]*)([\d]*)(?'version'(?'major'\d+)(?'minor'\.\d+)(?'letters'lw*))?)">
    browser=Gecko
```

```
<filter>
```

```
    <case match="(Gecko/[-\d]+)(?'VendorProductToken'
(?'type'^[\d]*)([\d]*)(?'version'(?'major'\d+)(?'minor'\.\d+)(?'letters'lw*)))">
        type=${type}
    </case>
    <case> <!-- plain Mozilla if no VendorProductToken found -->
        type=Mozilla
    </case>
</filter>
```

```
frames=true
```

```
    tables=true
    cookies=true
    javascript=true
    javaapplets=true
    ecmaScriptVersion=1.5
    w3cdomVersion=1.0
    css1=true
    css2=true
    xml=true
    tagwriter=System.Web.UI.HtmlTextWriter
```

```
<case match="rv:(?'version'(?'major'\d+)(?'minor'\.\d+)(?'letters'lw*))">
    version=${version}
    majorVersion=0${major}
    minorVersion=0${minor}
    <case match="^b" with="${letters}">
        beta=true
    </case>
</case>
</case>
```

```
<!-- AppleWebKit Based Browsers (Safari...) -->
```

```
<case match="AppleWebKit/(?'version'(?'major'\d?)(?'minor'\d{2})(?'letters'lw*)?)">
```

```
    browser=AppleWebKit
```

```
        version=${version}
        majorVersion=0${major}
        minorVersion=0.${minor}
        frames=true
        tables=true
        cookies=true
        javascript=true
        javaapplets=true
        ecmaScriptVersion=1.5
        w3cdomVersion=1.0
        css1=true
        css2=true
        xml=true
        tagwriter=System.Web.UI.HtmlTextWriter
```

```

<case match="AppleWebKit/(?'version'(?'major'\d+)(?'minor'\d+)(?'letters'lw*))(.*)?(?'type'[^'\d]*).*( |$)">
    type=${type}
</case>
</case>

<!-- Konqueror /-->
<case match=".[K|k]onqueror/(?'version'(?'major'\d+)(?'minor'(\.[\d])*)(?'letters'[^;]*));\s+(?'platform'[^;\\]*)?(:\\)">
browser=Konqueror
    version=${version}
    majorversion=0${major}
    minorversion=0${minor}
    platform=${platform}
    type=Konqueror
    frames=true
    tables=true
    cookies=true
    javascript=true
    javaapplets=true
    ecmaascriptversion=1.5
    w3cdomversion=1.0
    css1=true
    css2=true
    xml=true
    tagwriter=System.Web.UI.HtmlTextWriter
</case>

<!-- Opera /-->
<case match="Opera[ /](?'version'(?'major'\d+)(?'minor'(\.'minorint'\d+)(?'letters'lw*))">
    <filter match="[7-9]" with="${major}">
        tagwriter=System.Web.UI.HtmlTextWriter
    </filter>
    <filter>
<case match="7" with="${major}">
    <filter>
        <case match="[5-9]" with="${minorint}">
            ecmaascriptversion=1.5
        </case>
        <case>
            ecmaascriptversion=1.4
        </case>
    </filter>
    </case>

<case match="[8-9]" with="${major}">
    ecmaascriptversion=1.5
    </case>
</filter>
</case>

</browserCaps>

```

Anexo 5: Algunos Ejemplos de código html cambiado

Cambiar: <body bottomMargin="0" leftMargin="0" topMargin="0" rightMargin="0">

Por: <body style="margin: 0">

Cambiar: <TABLE id="Table1" height="25" cellSpacing="0" cellPadding="0" width="100%" border="0">

Por: <TABLE id="Table1" style="height:25; width:100%" border="0" cellSpacing="0" cellPadding="0">

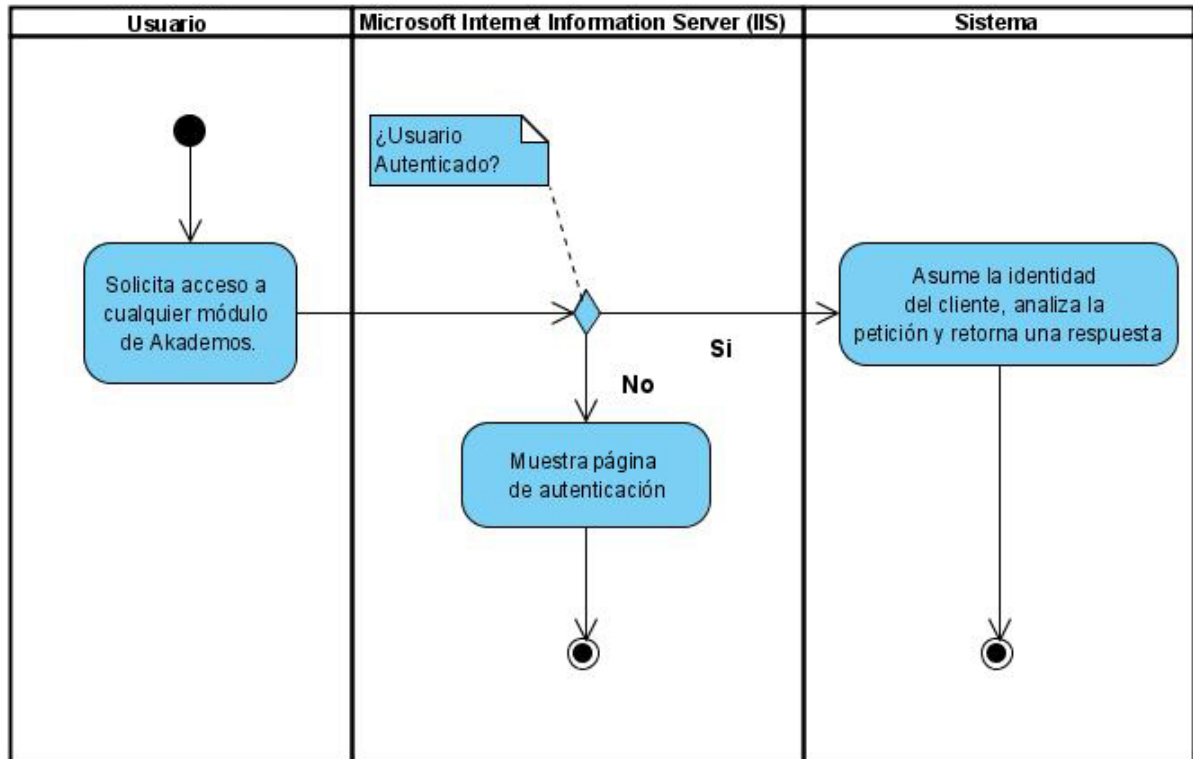
Cambiar: <TD background="../akademos/imagenes/imagen.jpg">

Por: <TD style="background-image:url('../akademos/imagenes/imagen.jpg)">

Cambiar: bgColor="darkslategray"

Por: bgColor="#2F4F4F"

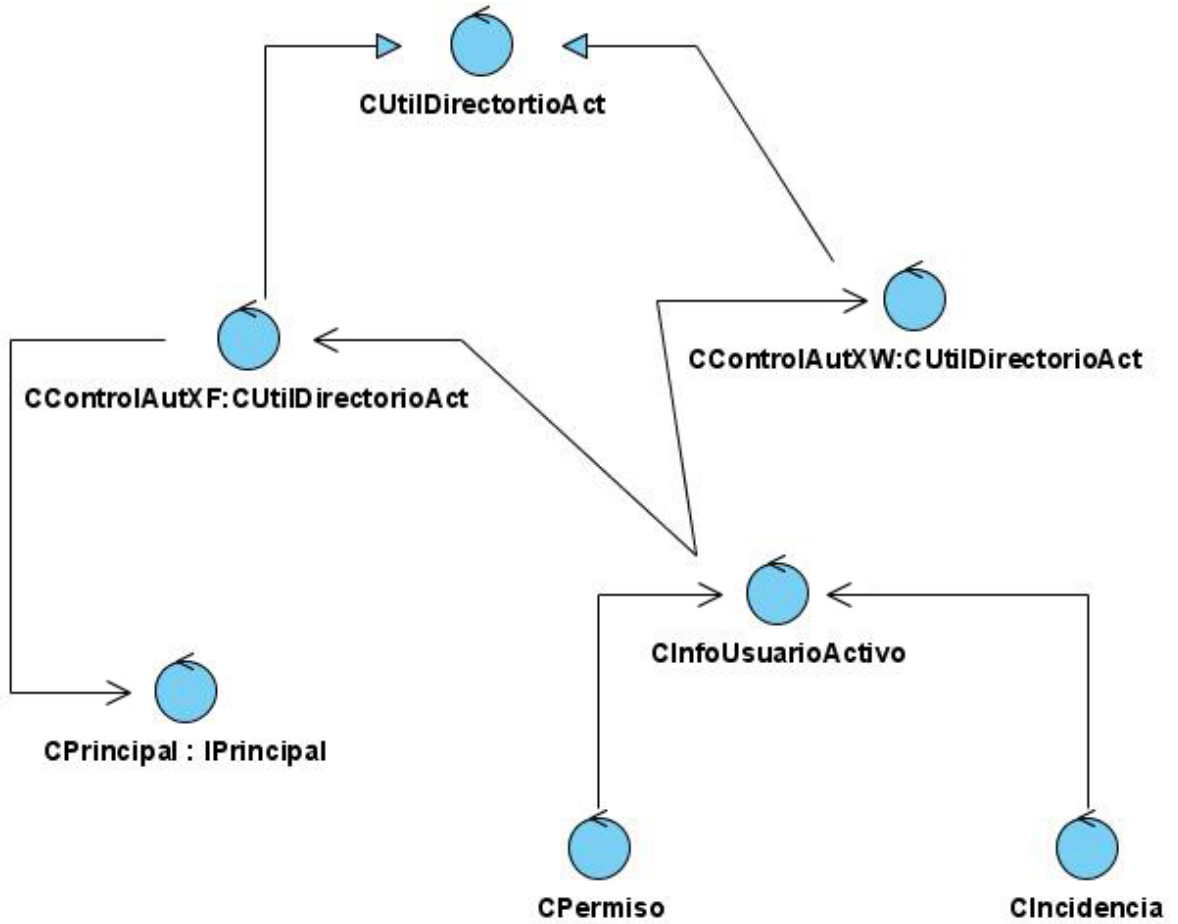
Anexo 6: Modelo de autenticación de Akademos usando el proveedor de seguridad de Autenticación de Windows



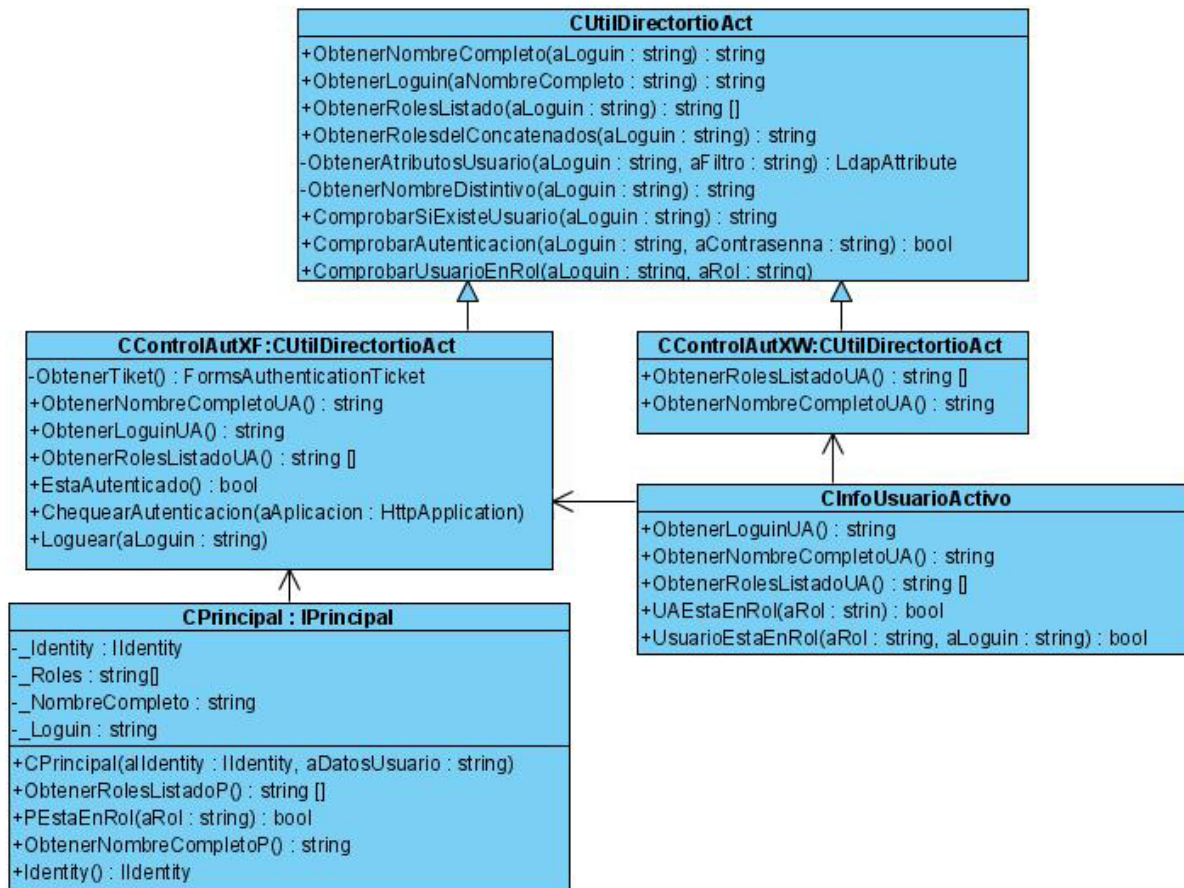
Anexo 7: Conjunto de clases de nuevo esquema de seguridad propuesto para la migración de Akademos

Clase	Descripción	Dependen-cias
CUtilDirectorioAct	Esta clase es la encargada de interactuar con el directorio activo, a través de ella se recupera toda la información que se necesite de los usuarios del LDAP.	Novell.Directory.LDAP (dll)
CControlAutXF	Esta clase hereda de CUtilDirectorioAct, se emplea para la gestión de autenticación por formulario. El manejo de las cookie y el chequeo continuo de la validez de las credenciales del usuario activo son sus funcionalidades fundamentales.	CPrincipal CUtilDirectorioAct
CControlAutXW	Esta clase hereda de CUtilDirectorioAct, gestiona la información del usuario activo cuando el modo de autenticación es Autenticación por Windows.	CUtilDirectorioAct
CPrincipal	Esta clase hereda de IPrincippal y su función es crear un objeto IPrincippal que será utilizado por CControlAutXF.	IPrincipal
CInfoUsuarioActivo	Esta clase es la encargada de manipular toda la información que se necesite desde cualquier artefacto de la arquitectura del sistema sobre el usuario activo en el mismo.	CControlAutXF CControlAutXW
CPermiso	Esta clase es la encargada de manipular todos los permisos que gestiona el sistema (asignación, comprobación, activación, etc.)	CInfoUsuarioActivo
CIncidencia	Esta clase se encarga de manipular todas las incidencias que se generan en el sistema	CInfoUsuarioActivo

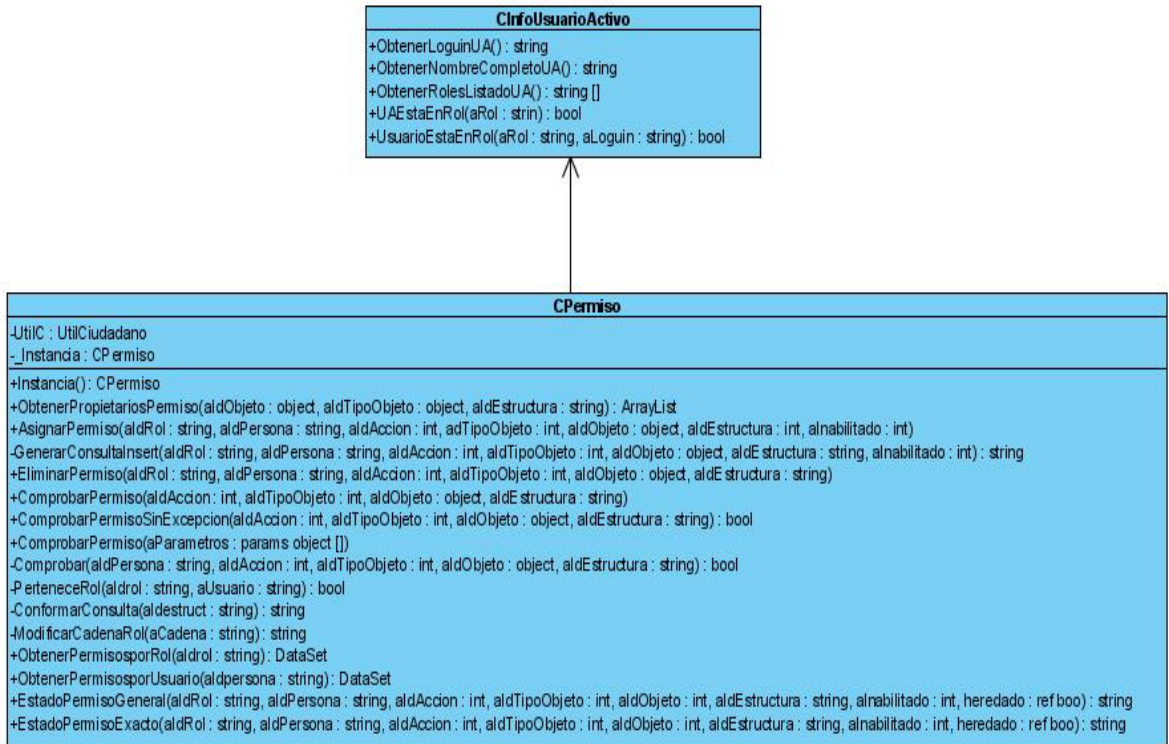
Anexo 8: Diagrama de clases para la seguridad mixta

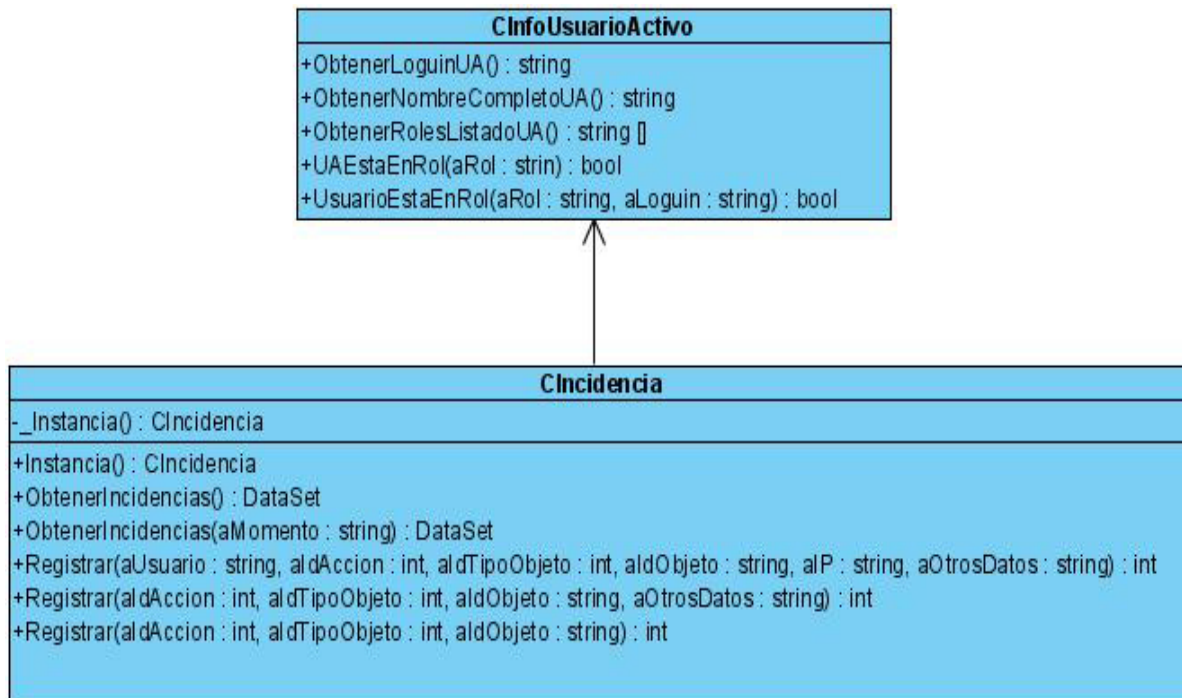


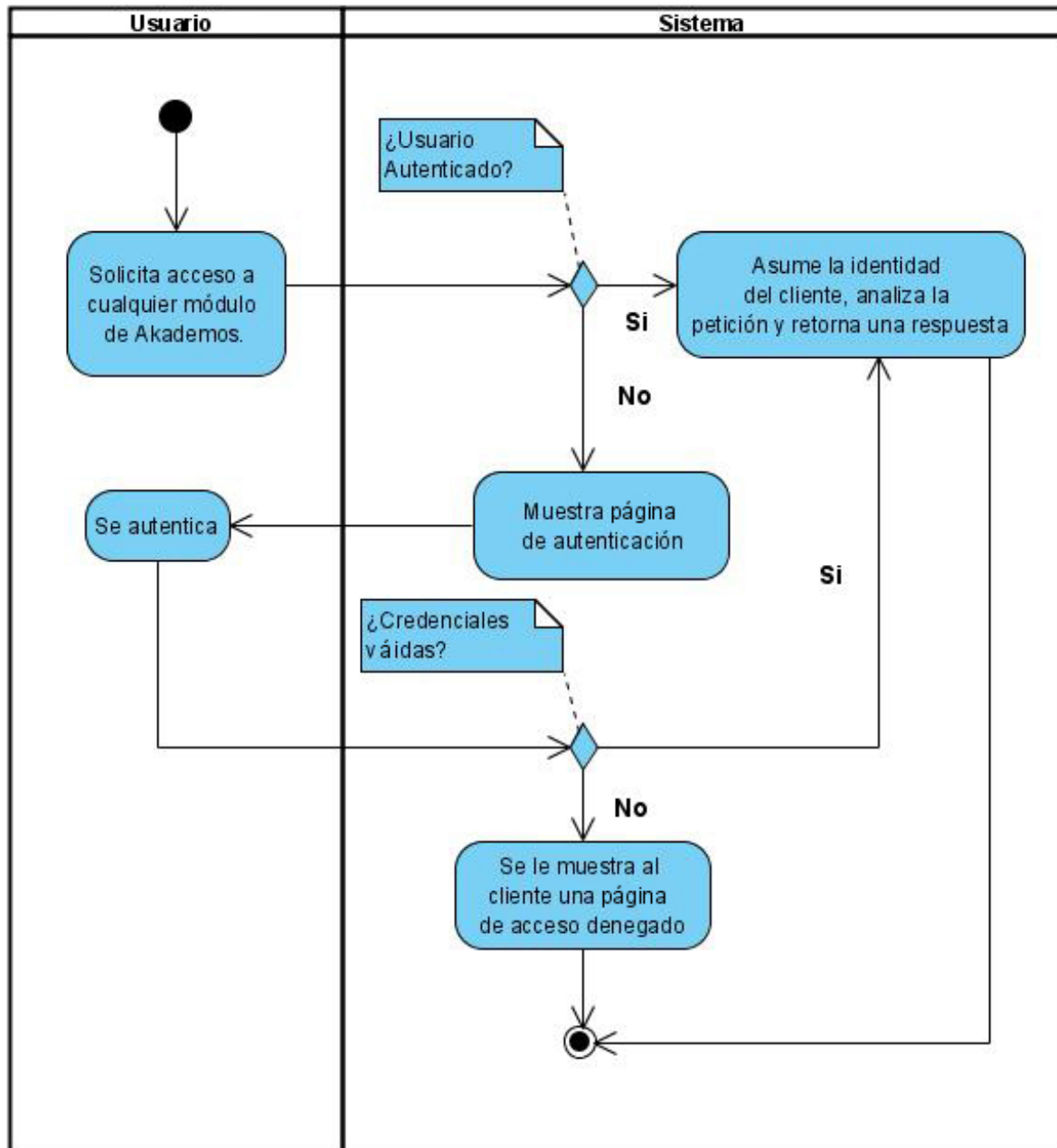
Anexo 8: Diagrama de clases para la autenticación mixta (Continuación)



Anexo 8: Diagrama de clases para la autenticación mixta (Continuación). Relación con la clase CPermiso



Anexo 8: Diagrama de clases para la autenticación mixta (Continuación). Relación con la clase CIncidencia

Anexo 9: Modelo de autenticación de Akademos usando el proveedor de seguridad de Autenticación por formularios

Anexo 10: Página de autenticación de Akademos



Entre aquí usando su nombre y contraseña del dominio uci.cu:
(Las Cookies deben estar habilitadas en su navegador)

Usuario:

Contraseña:

Entrar

Anexo 11: Ejemplo de plantilla definida por el usuario del sistema. Plantilla Hoja de Prematrícula

Hoja de Prematrícula

Datos Personales

IdIngreso:
Carné de Identidad:
Primer Nombre:
Segundo Nombre:
Primer Apellido:
Segundo Apellido:
Sexo:
Nombre del padre:
Nombre de la madre:

Residencia Permanente:
Número:
Apartamento:
Entre:
Localidad:
Provincia:
Municipio:

Datos de Procedencia

Nombre del centro:
Índice Académico:
Nota de Matemática:
Vía de Ingreso:
Tipo de Centro:
Asignatura de concurso:
Observaciones:

Anexo 12: Clase CEntidad, sus atributos y métodos

CEntidad
-FCadenaConexion : string
-Esquema : CEsquema
-NombreTabla : string
+CEntidad()
+CEntidad(aNombreTabla : string)
-SQLSeleccion(aCampos : string, aExterna : bool) : string
+Seleccionar() : DataSet
+Seleccionar(aCampos : string, aExterna : bool) : DataSet
+Seleccionar(aCampos : string) : DataSet
+Seleccionar(aCampos : string, aCondicion : string, aExterna : bool) : DataSet
+Seleccionar(aCampos : string, aCondicion : string) : DataSet
+Insertar(aArgs : params object [])
+Modificar(aArgs : params object [])
+Actualizar(aArgs : params object [])
+Eliminar()
+Eliminar(aCondicion : string)
+Ejecutar(aSql : string)
+Ejecutar(aNombrePA : string, aParametros : string) : DataSet
+EjecutarSeleccion(aSql : string) : DataSet
+CondicionBusqueda(aCampos : string [], aPatron : string) : string
+Existen() : bool
+Existen(aCondicion : string) : bool

