

# Universidad de las Ciencias Informáticas

## "Facultad 6"



### **Título: "Aplicación Informática para la gestión de recursos a través del Directorio Activo de la Universidad de las Ciencias Informáticas. "**

Trabajo de Diploma para optar por el título de

Ingeniero Informático

**Autoras:** " Yusimil Dávila Hernández "

" Jany Díaz Rodríguez "

**Tutores:** " Ing. Lissete González Gallo "

" Lic. Omar Mar Cornelio "

**Consultante:** " Ing. Alain Fernández del Toro "

" Febrero, 2010 "



“Cuando se es joven, se crea. Cuando se es inteligente,  
se produce. No se adapta, se innova: la medianía copia;  
la originalidad se atreve.”

José Martí

## DATOS DE CONTACTO

### **Tutores:**

“ Ing. Lissete González Gallo ”  
Universidad de las Ciencias Informáticas, Habana, Cuba.  
Email: [lgallo@uci.cu](mailto:lgallo@uci.cu)

“ Lic.Omar Mar Cornelio ”  
Universidad de las Ciencias Informáticas, Habana, Cuba.  
Email: [omarmar@uci.cu](mailto:omarmar@uci.cu)

### **Consultante:**

“Ing. Alain Fernández del Toro ”  
Universidad de las Ciencias Informáticas, Habana, Cuba.  
Email: [afernandez@uci.cu](mailto:afernandez@uci.cu)

## DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

Yusimil Dávila Hernández

---

Firma del autor

Jany Díaz Rodríguez

---

Firma del autor

Lic. Omar Mar Cornelio

---

Firma del tutor

Ing. Lisete González Gallo

---

Firma del tutor

### AGRADECIMIENTOS

*Esta es quizás la parte más difícil de escribir para nosotras, pues han sido muchas las personas que han contribuido a que este momento llegara.*

*Quisiéramos agradecer su participación, ayuda y comprensión:*

*A nuestros tutores, por tener siempre un tiempo, a pesar de la cantidad de trabajo nunca nos abandonaron y por la entrega de conocimientos que nos han dado.*

*A Aquiles y Nohemi por su ayuda incondicional en el desarrollo de este trabajo.*

*A nuestros compañeros de trabajo por estar siempre en los buenos y malos momentos durante estos años y sobre todo por su apoyo en este momento tan importante para nosotras.*

*A aquellos profesores que fueron los que nos encaminaron y formaron como profesionales.*

*A todos los que en un determinado momento nos preguntaron ¿Y la Tesis?*

*Yusimil:*

*A la Revolución Cubana por darme la posibilidad de llegar hasta aquí.*

*A toda mi familia, sin excepción de ninguno, por apoyarme siempre en todo y por confiar en mí, a mi novio por no dejarme sola.*

*A los amigos que siempre han estado cuando los he necesitado, Mercedes, Julio Villaverde, Juan Carlos, y quienes me han acompañado desde el técnico medio Yurlem, Yadelis, Dunay y Julio y nunca han reclamado nada a cambio.*

*A Omar que me ayudó desde que empecé a trabajar en la UCA, me alentó a estudiar y me ha exigido cada día ser mejor.*

*Jany:*

*A mis padres por siempre confiar en mí y enseñarme a crecer ante las dificultades de la vida.*

*A mi novio por estar siempre a mi lado apoyándome.*

*A mis compañeros de trabajo y amigos, Fernando, Yosnary y Michel que me han dado aliento y fuerzas para seguir adelante.*

## DEDICATORIA

*A mi Mamá, que desde que yo nací soñó con este momento y me formó para hoy cumplírselo.*

*A Minerva que desde que me conoció ha sido como una madre para mí y me ha ayudado a vencer difíciles obstáculos para llegar hasta aquí.*

*A mi tía, mis abuelos y mi padrastro que con amor y orgullo han sido parte importante en mi formación como persona.*

*A mi hermana Yaquelin y mi primo Dennis que siempre han hablado de mí con orgullo lo cual me obligó a seguir adelante y no rendirme.*

*Yusimil*

*A mis padres que siempre me guiaron por el camino correcto y que fueron un ejemplo en mi vida.*

*A mis tíos que con su apoyo moral e incondicional formaron valores que me guiaron hasta aquí.*

*A mis abuelos que con su amor y dedicación hicieron que creyera que todo era posible.*

*Jany*

### RESUMEN

La Universidad de las Ciencias Informáticas (UCI), cuenta con una red interna, de un gran número de computadoras con fines Docentes, Productivos e Investigativos, las cuales se gestionan centralizadamente mediante un Directorio Activo como forma de organizar y administrar los recursos de la red. Los permisos y accesos administrativos son difíciles de adaptar a la estructura organizativa de la red tecnológica de la universidad; por lo que la propuesta está encaminada a desarrollar una para la gestión de usuarios y cuentas de computadora a través del Directorio Activo. A nivel de aplicación se asigna a cada usuario los permisos necesarios para sus funciones delimitando exclusivamente su acceso a los recursos del Directorio Activo adaptado a la estructura de los laboratorios de la UCI según el estándar de codificación establecido como políticas para nombrar las estaciones de trabajo.

Para llevar a cabo este proyecto se utilizaron herramientas y tecnologías que se encuentran establecidas por la Dirección de Informatización y se realizó un estudio de las características que presentan los sistemas existentes para el trabajo con el Directorio Activo desarrollados en el mundo y en Cuba. Siguiendo los pasos que propone la metodología Ágil SCRUM\_XP se elaboró la Documentación Técnica mediante la herramienta de modelado Visual Paradigm, así como la documentación complementaria requerida para describir el funcionamiento del sistema, quedando de este modo conformado para transferirlo mediante su acceso a través de la red publicada como Aplicación Web.

**PALABRAS CLAVE:** cuentas de computadoras, directorio activo, gestión de usuarios, sec-labs-admin, técnico de laboratorio.

## Tabla de contenidos

AGRADECIMIENTOS.....	I
DEDICATORIA .....	II
RESUMEN.....	III
PALABRAS CLAVE:.....	III
Tabla de contenidos.....	IV
INTRODUCCIÓN.....	1
Capítulo1 FUNDAMENTACIÓN TEÓRICA.....	5
Introducción .....	5
1.1 Tendencias Actuales .....	5
1.1.1 ¿Qué son los sistemas de gestión?.....	5
1.1.2 ¿Qué es el Directorio Activo? .....	6
1.1.3 Grupos de seguridad .....	7
1.2 Sistemas existentes para el trabajo con el Directorio Activo.....	7
1.2.1 Active Administrator .....	7
1.2.2 ADManager Plus.....	8
1.2.3 Desktop Authority.....	9
1.3 Estructura de la Dirección de los Laboratorios.....	9
1.3.1 Gestión de usuarios y cuentas de computadoras en la Universidad de las Ciencias Informáticas.....	10
1.3.2 ¿Que es UCILab? .....	10
1.4 ADLDAP .....	11
1.4.1 ADLDAP en la aplicación.....	12
1.5 Políticas de nombres de las PC en la universidad .....	12
1.6 Metodología de desarrollo .....	13
1.6.1 SCRUM.....	13
1.6.2 Extreme Programming (XP).....	13
1.6.3 SCRUM y XP .....	13
1.6.4 RUP .....	14
1.7 Lenguajes, Tecnologías y Herramientas utilizadas .....	15
1.7.1 Lenguaje de programación .....	15
1.7.1.1 ASP.....	15
1.7.1.2 PHP .....	15
1.7.2 Javascript.....	16
1.7.2.1 Ajax.....	17
1.7.3 Herramienta de modelado .....	18
1.7.3.1 Rational Rose .....	18
1.7.3.2 Visual Paradigm.....	18
1.7.4 Gestor de Base de Datos.....	19
1.7.4.1 SQL Server .....	19
1.7.4.2 MySQL.....	19
1.7.5 Servidores Aplicaciones Web .....	20
1.7.5.1 Internet Information Services (IIS) .....	20
1.7.5.2 Apache.....	20
1.7.6 Framework para el desarrollo .....	21
1.8 MECANISMO DE SEGURIDAD .....	22
1.8.1 Cuenta de conexión .....	22



1.9	Pruebas de sistema .....	23
	Conclusiones .....	23
Capítulo 2 CARACTERÍSTICAS DEL SISTEMA .....		24
	Introducción .....	24
2.1	Breve reseña de la aplicación .....	24
2.1.1	Modelo de Dominio .....	25
	Figura 6: Modelo de dominio .....	25
2.2	Requerimientos .....	26
2.2.1	Requisitos funcionales del Sistema .....	26
2.2.2	Requisitos no funcionales .....	27
2.3	Actores del Sistema.....	30
2.4	Descripción de Historial de Usuario .....	31
2.5	Diagrama de Historia de Usuario del Sistema.....	35
	Figura 7: Diagrama historia de Caso de Uso del Sistema.....	35
2.6	Plan de Releases .....	35
	Conclusiones .....	36
Capítulo3 CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA .....		37
	Introducción .....	37
3.1	Plantilla de Modelo del Diseño .....	37
3.2	Descripción del Diagrama de Clases .....	39
3.3	Modelo de Datos .....	40
3.4	Patrones de Diseño .....	41
3.5	Arquitectura del sistema .....	41
3.6	Pantallas Tipos del Sistema .....	42
3.7	Casos de Prueba.....	47
3.7.1	Caso de Prueba del Historial de Usuario: Autenticar usuario .....	48
3.7.2	Caso de Prueba del Historial de Usuario: Administrar.....	49
3.7.3	Caso de Prueba del Usuario: Eliminar PC del dominio .....	50
3.7.4	Caso de Prueba del Historial de Usuario: Generar Reportes .....	51
3.7.5	Caso de prueba del Historial de Usuario: Gestionar Grupos.....	51
3.7.6	Caso de Prueba del Historial de Usuario: Buscar PC en Dominio.....	52
3.7.7	Caso de Prueba del Historial de Usuario: Actualizar usuario de conexión.....	53
3.8	Resultados obtenidos .....	53
	Conclusiones .....	54
CONCLUSIONES .....		55
RECOMENDACIONES .....		56
BIBLIOGRAFÍA .....		57
ANEXOS .....		62
GLOSARIO DE TÉRMINOS .....		74

### INTRODUCCIÓN

La Informática se ha venido desarrollando de manera exponencial a nivel mundial y como parte importante dentro de ella el proceso de desarrollo de software. Teniendo en cuenta que el grueso de las producciones, exportaciones, despliegue de productos de software y soluciones integrales se concentran en los Estados Unidos, Japón y países más avanzados del continente europeo, como consecuencia de la división global del trabajo, algunos otros países en vías de industrialización, ocupan un lugar importante en esta rama. Entre los casos más exitosos se destacan países como la India, Irlanda e Israel. En el contexto latinoamericano países como Brasil, Argentina, Uruguay y Costa Rica también muestran cierto dinamismo en estos procesos, aunque de manera menos importante con respecto a los países antes mencionados. Estas experiencias se concentran algunas en el desarrollo del mercado interno como Brasil y Argentina y otras más vinculadas con las exportaciones e inserción internacional como Uruguay y Costa Rica.

Cuba, a pesar de ser un país subdesarrollado no ha quedado exenta de este desarrollo, y en los últimos años uno de los principales objetivos del estado cubano ha sido el desarrollo de la Industria de Software, no solo con el fin de desarrollar sistemas para la informatización de la sociedad cubana, sino también con el objetivo de insertarse en el mercado internacional. Como parte de este desarrollo en septiembre del 2002 se creó la Universidad de las Ciencias Informáticas (UCI) como un programa de la Revolución, que tiene como tarea fundamental formar y preparar futuros Ingenieros Informáticos que con la aplicación de su conocimiento ayuden al desarrollo de la Industria del Software tanto en el país como en el mercado internacional, produciendo software y brindando servicios informáticos.

Desde la creación de la Universidad existe la dirección de laboratorios la cual tiene como meta brindar un servicio de excelencia a los usuarios que hoy dependen de sus servicios. Los técnicos, en su labor diaria realizan los procesos de instalación de software, configuración y mantenimiento de las computadoras (PC).

Para garantizar la puesta en funcionamiento de las estaciones de trabajo de los laboratorios una vez concluido el proceso de distribución de software, es necesario eliminar las cuentas de computadoras existentes en el directorio activo para garantizar su incorporación como estación de trabajo integrado al servidor de dominio.

El trabajo con el directorio activo es una tarea especializada donde los permisos administrativos son difíciles de gestionar y adaptar a la estructura organizativa implementada en la red tecnológica de la UCI, esto genera fallas de seguridad y permisos excesivos.

Constituye un obstáculo la dependencia de un grupo reducido de administradores encargados de eliminar del dominio las estaciones de trabajo, y que los técnicos, quienes interactúan directamente con esto no puedan tener acceso al trabajo con el directorio activo, lo que dificulta el buen funcionamiento de las estaciones de trabajo en los laboratorios para ofrecer un servicio de excelencia a los públicos internos, viéndose afectado en muchas ocasiones el proceso docente educativo.

Por lo antes expuesto se plantea como **problema científico** de esta investigación: ¿Cómo facilitar la gestión de usuarios y cuentas de computadoras a través del directorio activo para los laboratorios de la Universidad de Ciencias Informáticas?

El **objeto de estudio** es la gestión de recursos en el directorio activo y el **campo de acción**: la gestión de usuarios y cuentas de computadoras en el directorio activo.

Para dar solución al problema científico declarado, esta investigación tiene como **objetivo general**: Desarrollar una aplicación informática para la gestión de usuarios y cuentas de computadora a través del Directorio Activo. A partir de este objetivo general se trazaron los siguientes **objetivos específicos**:

- Analizar la aplicación informática.
- Diseñar la aplicación informática.
- Implementar la aplicación informática.
- Validar la aplicación informática.

Los que se cumplirán a través de las siguientes **tareas de la investigación**:

1. Estudio del arte de la gestión de cuentas de usuarios y computadoras en los directorios activos.
2. Especificación de los requisitos funcionales y no funcionales de la aplicación informática.
3. Modelación de la aplicación informática propuesta.
4. Diseño de las funcionalidades de la aplicación informática.
5. Desarrollo de la aplicación informática.
6. Elaboración y realización de los casos de prueba.

Para llevar a cabo las tareas propuestas y arribar satisfactoriamente al resultado final de la investigación se utilizan los siguientes **métodos científicos**:

**1. Analítico-sintético:** el análisis de diversas metodologías y documentos relacionados con el objeto de estudio de los cuales se extraen los rasgos distintivos más importantes relacionados con la gestión de recursos a través del directorio activo.

**2. Modelación:** este método posibilita modelar el funcionamiento de una aplicación para la Gestión de recursos a través del directorio activo, como parte de una propuesta para dar respuesta a las necesidades del proyecto de Informatización.

**3. Observación:** debido a que se pudo constatar los problemas que existían en la Universidad producto al proceso de Gestión de recursos a través del directorio activo de los Laboratorios, motivándonos a de realizar la investigación.

### **Aportes prácticos esperados de la investigación**

Se espera obtener una aplicación informática para la gestión de usuarios y cuentas de computadora donde los permisos administrativos sean adaptados a la estructura organizativa de la red tecnológica de la UCI, eliminando fallas de seguridad o permisos excesivos para la solución de tareas sencillas. La aplicación propuesta asignará a cada usuario los permisos necesarios para sus funciones delimitando exclusivamente su acceso a los recursos correspondientes a su trabajo diario.

Con la puesta en funcionamiento de la aplicación propuesta, se aumenta el nivel de gestión sobre las funciones establecidas para los técnicos de los laboratorios, ya que todo el proceso de instalación y configuración de las estaciones de trabajo dependería exclusivamente de su desempeño, evidenciando un mejor nivel de satisfacción para los usuarios finales.

El presente trabajo se encuentra estructurado en **3** capítulos, los cuales resumen la siguiente información:

**Capítulo 1. “FUNDAMENTACIÓN TEÓRICA”**, Describe el proceso de gestión de recursos a través del directorio activo en los laboratorios. Se manifiesta el estado del arte de las tecnologías a usar en el desarrollo de la aplicación informática y se exponen la metodología y herramientas a utilizar.

**Capítulo 2. “CARACTERÍSTICAS DEL SISTEMA”**, presenta una descripción de la aplicación informática, con los requerimientos funcionales y no funcionales, así como los diagramas de historiales de usuarios correspondientes.

**Capítulo 3: “CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA”**, se presenta el prototipo no funcional del sistema, se muestra la plantilla de modelo de diseño, además se realiza una breve explicación de los patrones de diseño utilizados y se plantea el estándar de codificación empleado en la implementación.

## Capítulo

# 1

## Capítulo1 FUNDAMENTACIÓN TEÓRICA

### Introducción

Este capítulo es el respaldo teórico de los temas tratados en el informe, necesarios para el entendimiento correcto de la solución planteada. Se describen los conceptos fundamentales asociados al dominio del problema y el objeto de estudio, haciéndose un análisis de los aspectos relacionados con el directorio activo. Se realiza una breve descripción de la gestión de usuarios y cuentas de computadoras de la Universidad de las Ciencias Informáticas. Además se realiza un estudio de las metodologías para el trabajo con el directorio activo. Por último se presenta la fundamentación de las tecnologías utilizadas para el diseño de la aplicación y las propuestas para su implementación y desarrollo.

### 1.1 Tendencias Actuales

Con el surgimiento de internet las Tecnologías de la Información y las Comunicaciones (TIC) han transformado las sociedades y brindado cambios significativos en la estructura de los negocios y en el desarrollo de nuevos servicios, este es el caso del surgimiento de los Sistemas de Gestión.

#### 1.1.1 ¿Qué son los sistemas de gestión?

Un sistema de gestión es una estructura probada para la gestión y mejora continua de las políticas, los procedimientos y procesos de la organización. [1]

Existen diferentes sistemas de gestión, por ejemplo: Sistemas de Gestión de Contenidos, Sistemas de Gestión de Base de Datos, Sistemas de Gestión de la Calidad, Sistemas de Gestión de la Información, Sistemas de Gestión de Reservaciones, y otros, aunque son diferentes presentan un objetivo en común; deben ser implementados en aras de mejorar, organizar y sistematizar los procesos que serán gestionados. Los sistemas de gestión de recursos a través del directorio activo son una herramienta necesaria sobre todo para empresas con grandes redes. La utilización de los mismos facilita en gran medida la labor de los administradores de redes pues les permite asegurarse de que las

infraestructuras críticas de empresas no dejen de funcionar, puesto que son buena parte del motor de su mercado.

### 1.1.2 ¿Qué es el Directorio Activo?

El Directorio Activo (nombre comercial que da Microsoft a un servicio de directorio en una red distribuida) es uno de los puntos de negocio fuertes de Microsoft. Las redes de grandes empresas (compuestas por miles de puestos y decenas de servidores todos Windows) suelen basarse en este servicio que facilita enormemente la labor de un administrador. En ocasiones, Microsoft antepone (comercial y técnicamente) este tipo de sistemas sostenidos por grandes empresas a las necesidades de los clientes individuales. De ahí que a veces precise de tanto tiempo para desarrollar un parche de seguridad efectivo: debe asegurarse de que infraestructuras críticas de empresas no dejen de funcionar, puesto que son buena parte del motor de su mercado. [2]

Windows implementa el concepto de dominio mediante el directorio, que no es más que un almacén de datos jerárquico que guarda información sobre recursos (objetos) en la red. El directorio se implementa como una base de datos optimizada para operaciones de lectura sobre la que se pueden realizar búsquedas de grandes cantidades de información, y con capacidades de exploración (un árbol, básicamente). El Directorio Activo se convierte así en un medio de organizar, administrar y controlar centralizadamente el acceso a los recursos de la red. **(Ver Anexo 1)**

Microsoft, a partir de Windows 2000, decidió seguir estándares para implementar toda la infraestructura que se intercomunica e interactúa para conformar un Directorio Activo. No pudo dejar atrás de golpe lo que venía sosteniendo con Windows NT, pero poco a poco ha ido quedando obsoleto. Entre los estándares que componen un Directorio Activo, y que han ayudado a mejorar su seguridad, destacan:

- DNS (Domain Name System): Windows NT funcionaba básicamente con WINS, una alternativa poco eficiente al DNS. WINS mantenía una tabla con la correspondencia entre direcciones IP y nombres NetBIOS, y la forma de intercambiar esta información entre servidores era realmente poco eficiente.
- LDAP (Lightweight Directory Access Protocol): Protocolo ligero (o compacto) de acceso a directorio. Este es el protocolo mediante el que las aplicaciones acceden y modifican la información existente en el directorio.
- Kerberos: Protocolo utilizado para la autenticación de usuarios y máquinas. Este protocolo es sin duda mucho más eficiente y efectivo que NTLM, usado para autenticar en una red controlada por Windows NT. El servidor de Kerberos coincide con el controlador de dominio.

En ciertas circunstancias bajo un dominio, se sigue usando LM/NTLM o LMv2/NTLMv2 para autenticar. La combinación de los dos primeros sufría enormes problemas de diseño que lo hacían realmente poco fiable. La fuerza bruta funcionaba muy bien sobre ellos. Su evolución, LMv2/NTLMv2, aunque bastante más sofisticada, no puede competir con el estándar Kerberos. Aun así, en Windows 2003 todavía estos protocolos no estaban activados por defecto para ser usados obligatoriamente.

### 1.1.3 Grupos de seguridad

Un grupo es una forma de organizar una serie de usuarios según un uso o propósito específico para realizar determinadas funciones. Los usuarios en la Dirección de Laboratorios se encuentran agrupados por las diferentes áreas respondiendo a la estructura organizacional de la arquitectura de las edificaciones.

Cada área geográfica posee técnicos que se encargan del soporte y mantenimiento de las estaciones de trabajo. Atendiendo a dicha estructura la Dirección de Laboratorios solicitó la creación de grupos en el Directorio Activo para poder asignar a un grupo de usuarios según su área de trabajo, permisos administrativos para poder realizar sus funciones, de esta forma son creados los Sec-labs Admin. (Ver Anexo 2)

## 1.2 Sistemas existentes para el trabajo con el Directorio Activo

A pesar de la importancia del Directorio Activo de Microsoft en las redes corporativas Windows, la funcionalidad que ofrece tiene importantes limitaciones en cuanto a la gestión de políticas de seguridad.

Las siguientes soluciones se integran con Directorio Activo, añadiendo importantes funcionalidades como la delegación de tareas específicas, la posibilidad de crear políticas granulares que se aplican en tiempo real, y por supuesto, la posibilidad de auditar e informar sobre el estado de cumplimiento de las políticas en cada momento.

### 1.2.1 Active Administrator

Active Administrator es una herramienta imprescindible para cualquier administrador responsable de Microsoft Directorio Activo. Es fácil de usar y ofrece amplia funcionalidad para garantizar la máxima coherencia y visibilidad en la aplicación de políticas de grupo.[3]

La administración de Microsoft Active Directory es a menudo una tarea compleja, ya que las herramientas que vienen incluidas en el sistema no otorgan visibilidad suficiente para las



organizaciones y estructuras grandes. Active Administrator va mucho más allá que las propias herramientas de Microsoft, permitiendo una gestión sencilla, informes completos y visibilidad total sobre el sistema.

Active Administrator permite planificar, editar, copiar, restaurar y en general gestionar la configuración de políticas de grupo, dentro de un único interfaz muy sencillo de manejar.

Su funcionalidad "Resultant Set of Policies" (RSOP) permite estudiar escenarios durante la fase de planificación y su capacidad de volver atrás en la aplicación de políticas asegura la máxima seguridad y coherencia en las políticas, desde el primer momento.



Fig. 1. Aplicación para el trabajo con el Directorio Activo: Active Administrator.

### 1.2.2 ADManager Plus

ADManager Plus es una solución web que facilita enormemente la administración de Microsoft Directorio Activo. Permite realizar las tareas más comunes, como las altas y bajas de usuarios y la aplicación de políticas de grupo, a través de un interfaz intuitivo y fácil de aprender. A través de sus informes detallados, ofrece visibilidad completa sobre todos los objetivos en el Directorio Activo. [4]

A través de su interfaz web, los administradores pueden realizar las siguientes tareas:

- Altas y bajas de usuarios.
- Grupos y listas de distribución.
- Delegación de tareas comunes (como la gestión de contraseñas de usuarios).
- Limpieza del AD y borrado de objetos obsoletos.
- Gestión de la configuración de Terminal Services.
- Modificación de atributos por "proceso batch".

ADManager Plus ahorra tiempo y facilita el trabajo, tanto para aquellos técnicos que son nuevos en la gestión de Active Directory, como los Administradores más expertos.



Fig. 2. Aplicación para el trabajo con el Directorio Activo: ADManager Plus.

### 1.2.3 Desktop Authority

Es fácil de instalar, usar y administrar, Desktop Authority ofrece un abanico de prestaciones que no están disponibles ni siquiera en las grandes suites de gestión de configuraciones. Desde control remoto hasta inventario automático de HW y SW, Desktop Authority ofrece un sinfín de funcionalidades que facilitan la labor de los técnicos de informática. Su prestación más destacada es la gestión de políticas de configuración y de seguridad, ya que permite aplicar sus políticas de configuración en tiempo real (las políticas se pueden crear a medida o importar de Microsoft Active Directory). [5]

Se trata de la mejor forma de controlar la configuración de los PCs y servidores en entornos de red. Ningún otro producto permite controlar tantos aspectos de la configuración de Windows, con tanta facilidad y a un precio tan asequible.



Fig. 3. Aplicación para el trabajo con el Directorio Activo: Desktop Authority.

Después de haber realizado un estudio sobre las diferentes soluciones existentes que se integran con el Directorio Activo, se concluye que las herramientas estudiadas cumplen con varias funcionalidades pero de manera independiente, por lo que no existe una herramienta que posibilite crear políticas de seguridad y permita controlar y verificar el cumplimiento de las mismas en cada momento. Además estas herramientas son propietarias.

### 1.3 Estructura de la Dirección de los Laboratorios

La dirección de los laboratorios presenta una estructura organizativa constituida por una dirección general. La misma cuenta con un Subdirector de Formación el cual es el responsable de la formación del personal que labora en esta dirección y un Subdirector de Software que atiende todos los procesos relacionados con la informatización. También se encuentra dividida en diferentes áreas vinculadas cada una a un docente en específico y la Infraestructura Productiva (IP). Cada una de ellas es dirigida por un Jefe de Área el cual es el Especialista Superior responsable del docente. Se encuentra además un Técnico de Hardware, un segundo Jefe de Área con el asistente que atiende toda la documentación y los Jefes de Turnos, que se encuentran distribuidos cuatro por docentes más la Infraestructura Productiva. Estos últimos se subordinan al jefe de su área. Se cuenta además con los técnicos de laboratorios, los cuales juegan un rol fundamental en el control de los mismos.

### 1.3.1 Gestión de usuarios y cuentas de computadoras en la Universidad de las Ciencias Informáticas

El Directorio Activo es un medio de organizar, administrar y gestionar centralizadamente el acceso a los recursos de la red. En la universidad existe un Directorio Activo que es capaz de controlar todos los usuarios y las cuentas de computadoras existentes en la misma, mostrándose nombre, apellidos, usuario, número de solapín, dirección donde trabaja y otros datos, los cuales son insuficientes para lograr una delegación de permisos según la estructura organizacional actual de la universidad. En ésta no se especifica el área donde trabaja (por ejemplo: se obtiene el dato de que es trabajador de la Dirección de Laboratorios pero no en qué docente está ubicado) por tal motivo es necesario crear una base de datos donde se organicen los procesos actuales como el control de los trabajadores especificando área de trabajo, responsabilidad y restricciones de acceso.

El trabajo con el Directorio Activo es una tarea donde los permisos administrativos son difíciles de adaptar a la estructura organizativa de la red tecnológica de la UCI, esto genera fallas de seguridad y permisos excesivos; por lo tanto cuando se formatea una computadora es necesario que el Jefe de Área solicite al Departamento de Informatización de los Laboratorios (personas con los permisos necesarios), la eliminación de la cuenta del DNS para poder incluirla posteriormente en el dominio UCI siguiendo la política de nombre establecida.

En la actualidad el proceso de instalación de las estaciones de trabajo de los laboratorios es una actividad que agrupa varias entidades para su ejecución por lo que se requiere de mucha cohesión para lograr el éxito de esta tarea. Producto de la dinámica de la universidad, el horario de trabajo y otros factores externos y organizativos, se dificulta el proceso de eliminación de la cuenta del DNS, factor que influye en el correcto funcionamiento de las estaciones destinadas a los procesos docentes.

### 1.3.2 ¿Que es UCILab?

La Dirección de Laboratorios tiene como funciones primordiales brindar servicio, soporte e información a la docencia y a la producción. A través de una investigación exhaustiva que se realizó en la Dirección de los Laboratorios se pudo constatar la necesidad de que existiera una aplicación que facilitara y agilizara el trabajo ya que la mayor parte del mismo se realizaba en papel, lo que traía como consecuencia que el flujo de la información no se efectuara con la velocidad requerida, además de lo tedioso que resultaba la búsqueda o solicitud de alguna información. A partir de la situación analizada quedó confirmada la necesidad de confeccionar una aplicación que facilitara el desempeño, control y organización del trabajo en el área de los laboratorios; una aplicación que respondiera a la gestión de

la información y las actividades desarrolladas en los laboratorios como son el control de personal, inventario y reporte entre otros.

El sistema UCILab se encuentra dirigido a los siguientes tipos de audiencia..

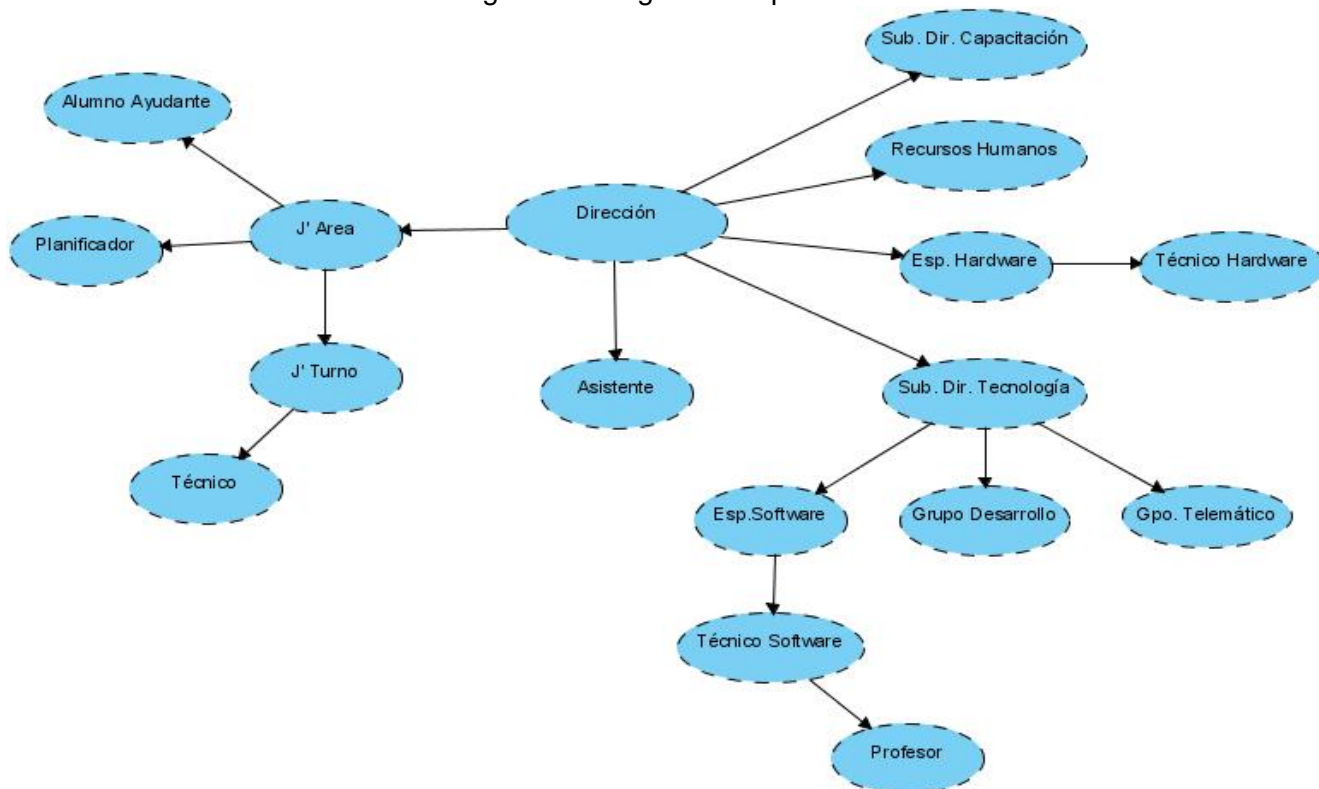


Figura 4. Estructura organizativa de UCILab.

### 1.4 ADLDAP

ADLDAP-LDAP- Autenticación con PHP para Directorio Activo

ADLDAP es una clase de PHP que provee autenticación e integración de LDAP con el Directorio Activo. La integración del Directorio Inteligente Activo con PHP fue el santo grial para muchos de los desarrolladores de redes por un largo tiempo, este proyecto es realmente para ayudar a otros a sobrepasar las mismas barreras que se han experimentado adentrándose en el crucigrama del Directorio Activo LDAP SSL trabajando nativamente en Linux.

Dada la variada naturaleza de las organizaciones y los sitios ADLDAP puede que no sea una solución completa pero puede ser un buen punto de partida. LDAP no es muy amigable a primera vista y es una curva de aprendizaje rápida.

La información que se puede obtener del Directorio Activo es tan usable como se desee hacer. Es necesario llenar toda la información de la cuenta para poder consultar.

Características claves:

- Autenticación de usuarios
- Gestión de grupos
- Gestión de usuarios
- Gestión de contactos
- Creación de buzones de correo
- Licencia

ADLDAP es un software de código abierto y es distribuido bajo, GNU Lesser General Public License [6]

### 1.4.1 ADLDAP en la aplicación

ADLDAP es la librería que se utiliza en la aplicación para conectarse al Directorio Activo, ésta gestiona en el dominio máquinas, grupos y usuarios. Para la realización de este proyecto se incluyeron dos métodos:

EliminarMaquina() que es el que se utiliza para eliminar las máquinas del dominio;

SetBaseDN() que es el que da la base en el Directorio Activo para comenzar la búsqueda desde ahí, ya que la búsqueda se hace similar a un árbol binario y no como las consultas SQL.

### 1.5 Políticas de nombres de las PC en la universidad

En las políticas contenidas en el Plan de Seguridad Informática de la UCI, uno de los aspectos que se trata es el siguiente:

Todas las computadoras deben estar dentro del dominio uci.cu y el nombre que las identifica estará en correspondencia con la localización geográfica o la función a la que se encuentran destinadas.

En el caso de los laboratorios los nombres están definidos por los siguientes parámetros:

LD (laboratorio Docente) 01 (el número del docente) – 101 (número del laboratorio) - 22 (puesto en que se encuentra la PC). (Ver Anexo 3)

### 1.6 Metodología de desarrollo

Se entiende por metodología de desarrollo una colección de documentación formal referente a los procesos, las políticas y los procedimientos que intervienen en el desarrollo del software. En Inglés software development methodology (SDM) o system development life cycle (SDLC). [7]

La finalidad de una metodología de desarrollo es garantizar la eficacia (por ejemplo: cumplir los requisitos iniciales) y la eficiencia (por ejemplo: minimizar las pérdidas de tiempo) en el proceso de generación de software.

#### 1.6.1 SCRUM

SCRUM, más que una metodología de desarrollo de software, es una forma de auto-gestión de los equipos de programadores. Un grupo de programadores deciden cómo hacer sus tareas y cuánto van a tardar en ello. Scrum ayuda a que trabajen todos juntos, en la misma dirección, con un objetivo claro. Scrum permite además seguir de forma clara el avance de las tareas a realizar, de forma que los "jefes" puedan ver día a día cómo progresa el trabajo. Sin embargo, Scrum no es una metodología de desarrollo, puesto que no indica qué se debe hacer para hacer el código. Debería, por tanto, complementarse con alguna otra metodología de desarrollo. Se acopla bien con las metodologías ágiles y en concreto, con la programación extrema. [8]

#### 1.6.2 Extreme Programming (XP)

La Programación Extrema (XP) es una metodología de desarrollo ligera (o ágil) basada en una serie de valores y de prácticas de buenas maneras que persigue el objetivo de aumentar la productividad a la hora de desarrollar programas.

Este modelo de programación se basa en una serie de metodologías de desarrollo de software en la que se da prioridad a los trabajos que dan un resultado directo y que reducen la burocracia que hay alrededor de la programación. [9]

#### 1.6.3 SCRUM y XP

**SCRUM y XP** se acoplan y retroalimentan perfectamente uno dentro del otro. SCRUM es como la interfaz entre el equipo y los clientes y XP entra a funcionar en cómo el equipo debe hacer su trabajo. De esta manera se completan efectivamente, surgiendo así una metodología de procedimiento ágil a seguir SCRUM\_XP.

Esta metodología cuenta con 4 fases: Planificación o Definición esta fase se centra en el que, es donde se identifican los requisitos claves del sistema, se establece la visión y se fijan expectativas. Contando con tres tareas esenciales: ingeniería de sistemas o de información, planificación del proyecto y análisis de los requisitos. La fase de Desarrollo se centra en el Cómo y su propósito es implementar un sistema definiendo como ha de diseñarse las estructuras de datos, como ha de caracterizarse las interfaces y ha de traducirse el diseño en un lenguaje de programación. La Entrega es la fase que tiene como propósito la puesta en operación. La Entrega contiene el cierre del reléase, aquí el sistema está listo para ser liberado y es en la que se realiza integración, la entrega de las pruebas del sistema y la documentación en general. Como última fase Mantenimiento centrada en el cambio que va incorporado a la corrección de errores, a las adaptaciones requeridas a medida que va evolucionando el entorno del software y a cambios debido a las mejoras producidas por los requisitos cambiantes del cliente.

### 1.6.4 RUP

La metodología RUP, llamada así por sus siglas en inglés Rational Unified Process, divide en 4 fases el desarrollo del software:

- Inicio, El Objetivo en esta etapa es determinar la visión del proyecto.
- Elaboración, En esta etapa el objetivo es determinar la arquitectura óptima.
- Construcción, En esta etapa el objetivo es llevar a obtener la capacidad operacional inicial.
- Transmisión, El objetivo es llegar a obtener el release del proyecto.

Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los Objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes.

Vale mencionar que el ciclo de vida que se desarrolla por cada iteración, es llevada bajo dos disciplinas:

- Disciplina de Desarrollo
- Disciplina de Soporte [10]

Como la metodología SCRUM XP es para proyectos cortos y simples, se ha seleccionado para el desarrollo de la investigación.

### **1.7 Lenguajes, Tecnologías y Herramientas utilizadas**

Para la selección del lenguaje de programación, así como la utilización de los diversos componentes; se consultaron diferentes características como portabilidad, multiplataforma y facilidad de uso, resultando seleccionados los que más se acercan a los requerimientos de la plataforma. La creciente informatización de los procesos productivos y sociales provoca que las organizaciones y empresas requieran cada vez más de aplicaciones de software confiables y de calidad tanto en su desarrollo como en su mantenimiento. Es por ello que en los últimos años se han venido publicando estándares, notaciones y procesos de desarrollo de Software que establecen las buenas prácticas para el desarrollo de aplicaciones de software.

La aplicación será desarrollada con las tecnologías establecidas por la Dirección de Informatización.

#### **1.7.1 Lenguaje de programación**

##### **1.7.1.1 ASP**

Active Server Pages (ASP) es una tecnología del lado servidor de Microsoft para páginas Web generadas dinámicamente, que ha sido comercializada como un anexo a Internet Information Server (IIS). ASP realiza muchas tareas apoyándose en objetos que deben ser comprados a ciertas empresas especializadas, o en algunos casos programados por el servidor. La tecnología ASP está estrechamente relacionada con el modelo tecnológico de su fabricante. Es una plataforma que no se ha desarrollado como lo esperaba Microsoft. [11]

##### **1.7.1.2 PHP**

PHP (acrónimo recursivo de "PHP: Hypertext Preprocessor", originado inicialmente del nombre PHP Tools, o Personal Home Page Tools) es un lenguaje del lado del servidor, que se utiliza básicamente para generar páginas dinámicas y que facilita de forma sencilla el acceso a diferentes bases de datos. También se utiliza para generar imágenes, generar pdf. PHP lo desarrollan programadores Web introduciendo nuevas funciones en las versiones mejoradas del lenguaje. [12]

#### **Ventajas de PHP:**

Libre y Abierto (Código fuente disponible, es gratuito).



Multiplataforma: inicialmente fue diseñado para entornos UNIX por lo que ofrece más prestaciones en este sistema operativo, pero es perfectamente compatible con Windows. ASP, solo está orientado a sistemas Windows.

- Soporte para varios servidores Web.
- Fácil acceso a Bases de Datos.
- Mucha documentación (Ejemplos, manuales).
- Presenta una integración perfecta entre Apache-PHP-MySQL.
- Posee una sintaxis bastante clara.
- Fácil aprendizaje.
- Seguro.
- PHP está orientado a objetos.

Como ASP es un lenguaje que ha sido comercializado con IIS (Internet Information Services) se decidió utilizar PHP con Javascript del lado del cliente y un motor AJAX entre el usuario y el servidor porque presenta una compatibilidad total con Apache y permite una interfaz más amigable.

### 1.7.2 Javascript

Javascript es un lenguaje con muchas posibilidades, utilizado para crear pequeños programas que luego son insertados en una página web y en programas más grandes, orientados a objetos mucho más complejos. Con Javascript podemos crear diferentes efectos e interactuar con nuestros usuarios. Este lenguaje posee varias características, entre ellas podemos mencionar que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros. [13]

Javascript es soportado por la mayoría de los navegadores como Internet Explorer, Netscape, Opera, Mozilla Firefox, entre otros.

Con el surgimiento de lenguajes como PHP del lado del servidor y Javascript del lado del cliente, surgió Ajax en acrónimo de (Asynchronous Javascript And XML). El mismo es una técnica para crear aplicaciones web interactivas. Este lenguaje combina varias tecnologías:

HTML y Hojas de Estilos CSS para generar estilos.

Implementaciones ECMAScript, uno de ellos es el lenguaje Javascript.

XMLHttpRequest es una de las funciones más importantes que incluye, que permite intercambiar datos asincrónicamente con el servidor web, puede ser mediante PHP, ASP, entre otros.

Debemos tener en cuenta que aunque Javascript sea soportado en gran cantidad de navegadores nuestros usuarios pueden elegir la opción de Activar/Desactivar el Javascript en los mismos.

### 1.7.2.1 Ajax

Ajax no es una tecnología. Es realmente muchas tecnologías, cada una floreciendo por su propio mérito, uniéndose en poderosas nuevas formas. AJAX incorpora:

- presentación basada en estándares usando XHTML y CSS;
- exhibición e interacción dinámicas usando el Document Object Model;
- Intercambio y manipulación de datos usando XML and XSLT;
- Recuperación de datos asincrónica usando XMLHttpRequest;
- y JavaScript poniendo todo junto.

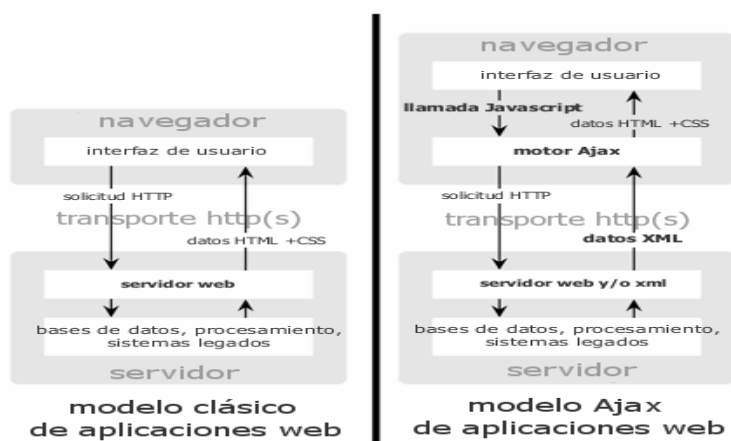


Figura 5: El modelo tradicional para las aplicaciones Web (izq.) comparado con el modelo de AJAX(der.)

Una aplicación AJAX elimina la naturaleza arrancar-frenar- arrancar-frenar de la interacción en la Web introduciendo un intermediario -un motor AJAX- entre el usuario y el servidor. Parecería que sumar una capa a la aplicación la haría menos reactiva, pero la verdad es lo contrario. [14]

En vez de cargar un página Web, al inicio de la sesión, el navegador carga al motor AJAX (escrito en JavaScript y usualmente procesado en un frame oculto). Este motor es el responsable por reenderizar la interfaz que el usuario ve y por comunicarse con el servidor en nombre del usuario. El motor AJAX permite que la interacción del usuario con la aplicación suceda asincrónicamente (independientemente de la comunicación con el servidor). Así el usuario nunca estará mirando una ventana en blanco del navegador y un icono de reloj de arena esperando a que el servidor haga algo.

Cada acción de un usuario que normalmente generaría un requerimiento HTTP toma la forma de un llamado JavaScript al motor AJAX en vez de ese requerimiento. Cualquier respuesta a una acción del usuario que no requiera un viaje de vuelta al servidor (como una simple validación de datos, edición de datos en memoria, incluso algo de navegación) es manejada por su cuenta. Si el motor necesita algo del servidor para responder (sea enviando datos para procesar, cargar código adicional, o recuperando nuevos datos) hace esos pedidos asincrónicamente, usualmente usando XML, sin frenar la interacción del usuario con la aplicación.

### 1.7.3 Herramienta de modelado

El modelado de sistemas software es una técnica para tratar con la complejidad inherente a estos sistemas. El uso de modelos ayuda al ingeniero de software a "visualizar" el sistema a construir. Además, los modelos de un nivel de abstracción mayor pueden utilizarse para la comunicación con el cliente. Por último, las herramientas de modelado pueden ayudar a verificar la corrección del modelo.

Algunas de las herramientas CASE conocidas son el ArgoUML, Rational Rose, Visual Paradigm, Easy CASE, Xcase, CASE Studio 2, CASEWise entre otras. Dentro de las más utilizadas se encuentran el Rational Rose y el Visual Paradigm.

#### 1.7.3.1 Rational Rose

**Rational Rose** es la herramienta CASE desarrollada por los creadores de UML Booch, Rumbaugh y Jacobson, provee un modelado basado en UML y cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases y entregables.[15]

#### 1.7.3.2 Visual Paradigm

**Visual Paradigm** por su parte, es una herramienta CASE que provee soporte para la generación de código, ingeniería inversa para Java. Se integra con Eclipse, Borland JBuilder y Oracle JDeveloper, para soportar las fases de implementación en el desarrollo de software. Tiene dentro de sus características que es portable y posee gran facilidad de uso. [15]

Utiliza UML como lenguaje de modelado y tiene como características:

- Es un producto de calidad.
- Soporta aplicaciones web.

- Se integra con las siguientes herramientas Java: Eclipse/IBM WebSphere, JBuilder, NetBeans IDE, Oracle JDeveloper, BEA Weblogic.
- Es fácil de instalar y actualizar.
- Logra compatibilidad entre ediciones.

Teniendo en cuenta las características antes mencionadas del Visual Paradigm, especialmente su buena integración con PHP, lenguaje que se utilizará para desarrollar la aplicación Web y la ventaja de que presenta una interfaz de usuario de fácil uso y que permite realizar los diagramas y artefactos que se generan durante el desarrollo del software además de realizar un control de las versiones durante todo el ciclo de trabajo, el colectivo de autores decide utilizarla como herramienta de modelado.

### **1.7.4 Gestor de Base de Datos**

#### **1.7.4.1 SQL Server**

SQL Server es un potente motor de bases de datos de alto rendimiento capaz de soportar millones de registros por tabla con una interfase intuitiva. Es una suite de herramientas de gestión integradas que incluye un conjunto de nuevas funcionalidades para el desarrollo, implantación y resolución de problemas de Bases de Datos SQL, así como mejoras a muchas funcionalidades.[16]

Además incorpora un modelo de objetos totalmente programable (SQL-DMO) con el que se puede desarrollar cualquier aplicación que manipule componentes de SQL Server, es decir, hacer aplicación para crear bases de datos, tablas, DTS, backups, entre otros, todo lo que se puede hacer desde el administrador del SQL Server.

#### **1.7.4.2 MySQL**

Existen diferentes arquitecturas para los sistemas de gestión de bases de datos, pero la más extendida, y la que más éxito ha tenido, es la arquitectura relacional MySQL es un servidor de bases de datos relacionales muy rápido y robusto. El mismo proporciona un servidor de base de datos SQL (Structured Query Language) muy rápido, multi-threaded, multi usuario y robusto. El servidor MySQL está diseñado para entornos de producción críticos, con alta carga de trabajo así como para integrarse en software para ser distribuido.[17]

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

Este gestor se creó con la rapidez en mente, de modo que no tiene muchas de las características de los gestores comerciales más importantes, como Oracle, Sybase o SQL Server. No obstante, eso no ha impedido que sea el más indicado para aplicaciones que requieren muchas lecturas y pocas escrituras y no necesiten de características muy avanzadas, como es el caso de las aplicaciones Web y por tanto de la propuesta de solución que se ofrece con este trabajo debido a su gran rapidez y facilidad de uso.

Dado que SQL Server es un sistema gestor de base de datos de software propietario se decidió usar MySQL porque es software libre y se trabaja con facilidad, además tiene una buena integración con PHP.

### **1.7.5 Servidores Aplicaciones Web**

#### **1.7.5.1 Internet Information Services (IIS)**

Internet Information Services (o Server), IIS, es una serie de servicios para los ordenadores que funcionan con Windows. Los servicios que ofrece son: FTP, SMTP, NNTP y HTTP. 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). [18]

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.

#### **1.7.5.2 Apache**

El servidor Apache es un software (libre) servidor HTTP de código abierto para plataformas UNIX, Windows entre otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Apache es de acuerdo al estudio hecho por Netcraft, el servidor WWW más utilizado del momento. Desde su origen ha evolucionado hasta convertirse en uno de los mejores servidores en términos de eficiencia, funcionalidad y velocidad. Apache ha demostrado ser sustancialmente más rápido que muchos otros servidores libres y compite de cerca con los mejores servidores comerciales, es el servidor Web por excelencia.[19]

Entre sus características más sobresalientes están:

Fiabilidad: Alrededor del 90% de los servidores con más alta disponibilidad funcionan con Apache.

Gratuidad: Apache es totalmente gratuito, y se distribuye bajo la licencia Apache Software License, que permite la modificación del código.

**Extensibilidad:** Se pueden añadir módulos para ampliar las ya de por sí amplias capacidades de Apache. Hay una amplia variedad de módulos, que permiten desde generar contenido dinámico (con PHP, Java, Perl, Python), monitorizar el rendimiento del servidor, atender peticiones encriptadas por SSL, hasta crear servidores virtuales por IP o por nombre (varias direcciones Web son manejadas en un mismo servidor) y limitar el ancho de banda para cada uno de ellos. Dichos módulos incluso pueden ser creados por cualquier persona con conocimientos de programación.

Como IIS es un servidor Web propiedad de Microsoft Corporation se decidió usar Apache, que es desarrollado en Software Libre.

### 1.7.6 Framework para el desarrollo

#### **CodeIgniter**

Como cualquier otro framework, CodeIgniter contiene una serie de librerías que sirven para el desarrollo de aplicaciones web y además propone una manera de desarrollarlas que debemos seguir para obtener provecho de la aplicación. Esto marca una manera específica de codificar las páginas web y clasificar sus diferentes scripts, que sirve para que el código esté organizado y sea más fácil de crear y mantener. CodeIgniter implementa el proceso de desarrollo llamado Model View Controller (MVC), que es un estándar de programación de aplicaciones, utilizado tanto para hacer sitios web como programas tradicionales. [20]

Algunos de los puntos más interesantes sobre este framework, sobre todo en comparación con otros productos similares, son los siguientes:

**Versatilidad:** Quizás la característica principal de CodeIgniter, en comparación con otros frameworks PHP. CodeIgniter es capaz de trabajar la mayoría de los entornos o servidores, incluso en sistemas de alojamiento compartido, donde sólo tenemos un acceso por FTP para enviar los archivos al servidor y donde no tenemos acceso a su configuración.

**Compatibilidad:** CodeIgniter, al menos en el momento de escribir este artículo de desarrollo web.com, es compatible con la versión PHP 4, lo que hace que se pueda utilizar en cualquier servidor, incluso en algunos antiguos. Por supuesto, funciona correctamente también en PHP 5.

**Facilidad de instalación:** No es necesario más que una cuenta de FTP para subir CodeIgniter al servidor y su configuración se realiza con apenas la edición de un archivo, donde debemos escribir cosas como el acceso a la base de datos. Durante la configuración no necesitaremos acceso a herramientas como la línea de comandos, que no suelen estar disponibles en todos los alojamientos.

**Flexibilidad:** CodeIgniter es bastante menos rígido que otros frameworks. Define una manera de trabajar específica, pero en muchos de los casos podemos seguirla o no y sus reglas de codificación muchas veces nos las podemos saltar para trabajar como más a gusto encontremos. Algunos módulos como el uso de plantillas son totalmente opcionales. Esto ayuda muchas veces también a que la curva de aprendizaje sea más sencilla al principio.

**Ligereza:** El núcleo de CodeIgniter es bastante ligero, lo que permite que el servidor no se sobrecargue interpretando o ejecutando grandes porciones de código. La mayoría de los módulos o clases que ofrece se pueden cargar de manera opcional, sólo cuando se van a utilizar realmente.

**Documentación tutorializada:** La documentación de CodeIgniter es fácil de seguir y de asimilar, porque está escrita en modo de tutorial. Esto no facilita mucho la referencia rápida, cuando ya sabemos acerca del framework y queremos consultar sobre una función o un método en concreto, pero para iniciarnos sin duda se agradece mucho.

Sin duda, lo más destacable de CodeIgniter es su accesibilidad, ya que se puede utilizar en la mayor gama de entornos.

### 1.8 MECANISMO DE SEGURIDAD

#### 1.8.1 Cuenta de conexión

Para el uso y funcionamiento de la aplicación informática propuesta se requiere de un usuario con los permisos necesarios para escribir en los grupos de administración de los laboratorios (Sec-lab Admin1; Sec-lab Admin2; Sec-lab Admin3; Sec-lab Admin4; Sec-lab Admin5; sec-labs computers; sec-labs managers) y acceso de eliminación de cuentas de computadoras en el DNS. En la actualidad la gestión de estos recursos está centralizada sobre un usuario personal generándose dependencia sobre el mismo.

Para la gestión de los procesos fundamentales de la Dirección de Laboratorios se tiene una cuenta impersonal laboratorios@uci.cu a la que se le asignan los permisos requeridos para el funcionamiento de la aplicación.

A nivel de aplicación se asigna a cada usuario los permisos necesarios para sus funciones, delimitando exclusivamente su acceso a los recursos del Directorio Activo adaptado a la estructura de los laboratorios de la UCI según el estándar de codificación establecido como políticas de nombre de las estaciones de trabajo.

### 1.9 Pruebas de sistema

Según las definiciones de la IEEE/ANSI, las pruebas son una actividad en la cual un sistema o componente es ejecutado bajo unas condiciones o requerimientos especificados, los resultados son observados y registrados, y una evaluación es hecha de algún aspecto del sistema o componente. [21]

Los tipos de pruebas que se consideraron para validar el componente de software incluyen: las pruebas de funcionalidad (seguridad y volumen) y soportabilidad (configuración).

Cada tipo de prueba tiene un objetivo específico y una técnica que lo soporte.

**Prueba de seguridad:** garantiza que los usuarios están restringidos a funciones específicas o su acceso está limitado únicamente a los datos que están autorizados a acceder. Solo aquellos usuarios autorizados a acceder al sistema son capaces de ejecutar las funciones del mismo. Además cumple objetivos específicos de seguridad de cada sistema.

**Prueba de configuración:** se enfoca en evaluar las diferentes variaciones de una aplicación integrada, contra sus requerimientos de configuración. Tiene como meta hacer que la aplicación falle en cumplir sus requerimientos de configuración, de manera que los defectos escondidos sean identificados, analizados, arreglados, y prevenidos en el futuro.

### Conclusiones

En este capítulo se realizó un estudio del estado del arte sobre herramientas existentes para el trabajo con el Directorio Activo. También se estudiaron las metodologías de desarrollo de software. Se seleccionó la metodología, herramientas y tecnologías a utilizar para el desarrollo de la aplicación web, la cual mostrará la dinámica de la aplicación y permitirá que los usuarios puedan realizar sus diversas solicitudes a través de una interfaz amigable.



## Capítulo

# 2

## Capítulo 2 CARACTERÍSTICAS DEL SISTEMA

### Introducción

El siguiente capítulo plantea una descripción del sistema que se pretende desarrollar. Se definen los Requerimientos del mismo, las descripciones de los Historiales de Usuarios, los diagramas de Historiales de Usuarios y los Actores del Sistema. Este proceso de desarrollo se realiza aplicando la metodología ágil Scrum XP.

### 2.1 Breve reseña de la aplicación

El negocio que se estudió, tiene muy bajo nivel de estructuración, con soluciones muy diversas y dispersas, aunque todas llevan el mismo propósito, por lo que no se tiene la posibilidad de identificar los procesos de negocio, las personas que lo inician, quienes son los beneficiados en cada uno de esos procesos, pero sin embargo si podemos identificar al menos eventos, transiciones, “cosas” involucradas a este entorno que no se logra determinar con facilidad y a eso se le llama concepto (objeto) y por tanto lo que se trata de hacer es un modelo de dominio, simplemente reunir todos esos conceptos que están en un entorno de trabajo en un diagrama (diagrama de modelo de dominio).

## 2.1.1 Modelo de Dominio

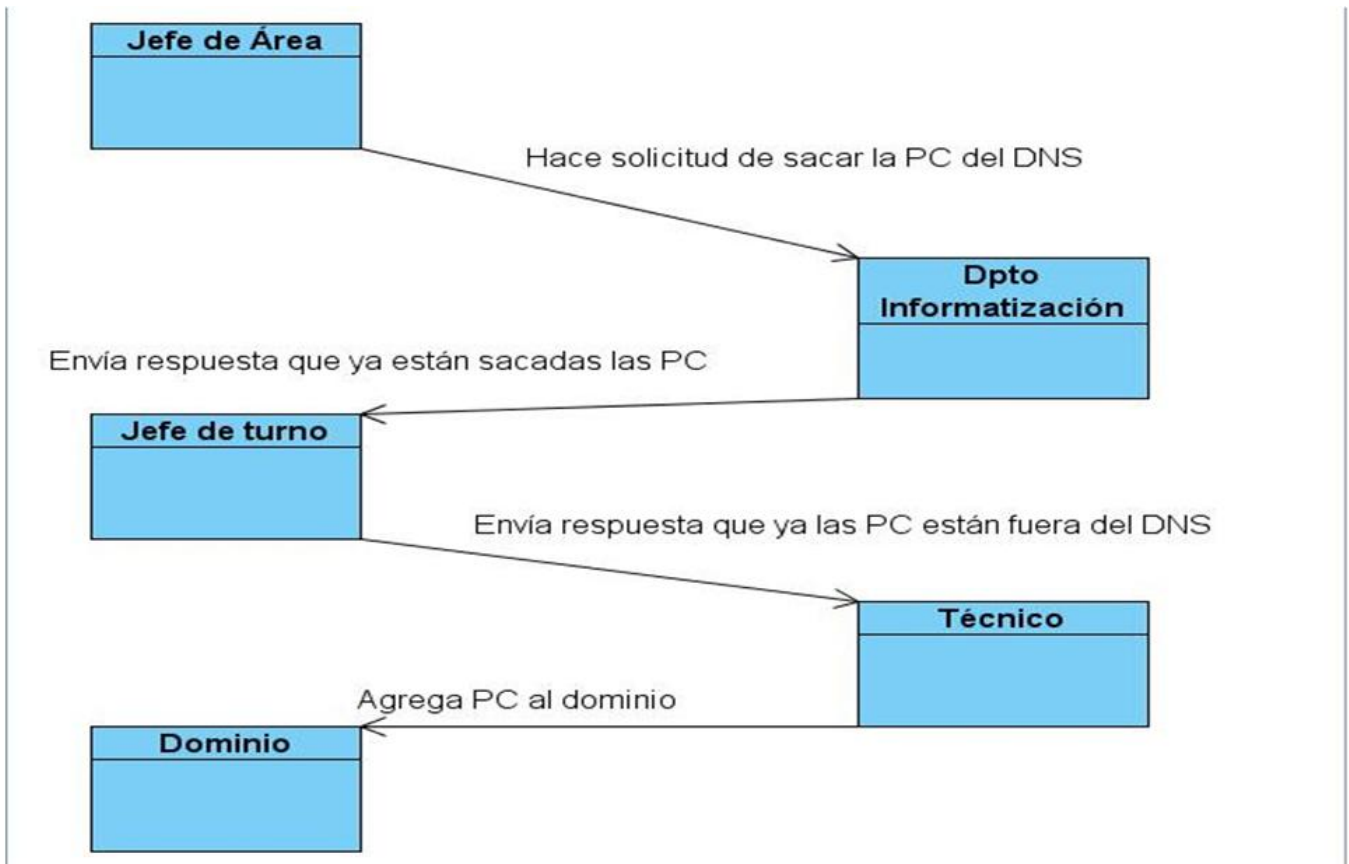


Figura 6: Modelo de dominio

En el modelo de dominio se muestra como se realiza la gestión de usuarios y cuentas de computadoras en la dirección de laboratorios.

El jefe de área que es el máximo responsable del docente le informa al departamento de informatización de la dirección de laboratorios que es el encargado de sistematizar los servicios y de la distribución de software en la dirección de laboratorios el grupo de PC que van a ser instaladas (ya sea por rotura o porque se necesita instalar una nueva imagen) este procede a limpiar los nombres del DNS y le envía respuesta al jefe de turno que es el jefe inmediato de los técnicos para que estos que son los responsables de los laboratorios agreguen la PC al dominio UCI.

## 2.2 Requerimientos

Una colección de requerimientos describe las características o atributos del sistema deseado. Se omite el cómo debe lograrse su implementación, ya que esto debe ser decidido en la etapa de diseño por los diseñadores. [22]

### 2.2.1 Requisitos funcionales del Sistema

Los requisitos funcionales especifican comportamientos particulares de un sistema. La base del requisito es la descripción del comportamiento requerido, que debe ser un claro y legible. El comportamiento descrito puede venir de reglas de organización o de negocio, o puede ser descubierto con sesiones del elicitation con los usuarios, los tenedores de apuestas, y otros expertos dentro de la organización. [23]

La Aplicación debe permitir:

#### **RF 1 Autenticar**

#### **RF 2 Administrar**

- 2.1 Insertar Usuario
- 2.2 Modificar Permisos a los Usuarios
- 2.3 Eliminar Usuario

#### **RF 3 Eliminar Cuenta de Computadora del Dominio**

#### **RF 4 Generar Reporte**

- 4.1 Visualizar PC eliminada por Fecha, Hora, Docente
- 4.2 Visualizar trazas de acción sobre gestión de grupos.

#### **RF 5 Gestionar Grupo**

- 5.1 Insertar miembros a los grupos
- 5.2 Eliminar miembros de los grupos
- 5.3 Buscar miembros en los grupos

#### **RF 6 Buscar PC en Dominio según nombre**

#### **RF 7 Actualizar usuario de conexión**

### 2.2.2 Requisitos no funcionales

Característica requerida del sistema, del proceso de desarrollo, del servicio prestado o de cualquier otro aspecto del desarrollo, que señala una restricción del mismo. [24]

- **Apariencia o interfaz externa**

La interfaz no contiene imágenes para no demorar las respuestas al usuario y está en concordancia al estilo de las aplicaciones de la UCI. El diseño de la interfaz es sencillo, con reconocimiento visual a través de elementos visibles que identifiquen cada una de sus acciones. Es formal, serio, con una navegación sugerente y con una gama cromática respetando pautas, todo esto teniendo en cuenta el fin con el que se desarrolla la aplicación.

- **Usabilidad**

La aplicación puede ser usada por cualquier persona que posea conocimientos básicos en el manejo de la computadora y de un ambiente web en sentido general.

- **Rendimiento**

La disponibilidad de trabajo en red contra el servidor es constante. Se garantiza que la respuesta a solicitudes de los usuarios, sea en un período de tiempo breve (de segundos) para evitar la acumulación de trabajo por parte de los responsables. La aplicación deberá ser lo más estable y confiable posible.

- **Soporte**

Se requiere que el producto reciba mantenimiento ante cualquier fallo que ocurra, para lograr esto la aplicación constará con una documentación apropiada para agilizar su mantenimiento y configuración, de forma tal que el tiempo de mantenimiento sea mínimo en caso de que sea necesario cambiar el equipo de desarrollo.

- **Portabilidad**

El producto es multiplataforma, corre sobre una plataforma web, los servicios en PHP 5 y su sistema de bases de datos en MySQL. Es de fácil mantenimiento.

- **Seguridad**

El sistema se encarga de controlar los diferentes accesos y funcionalidades, identifica al usuario antes de que pueda realizar cualquier acción sobre el sistema. La información será almacenada en bases de datos, dejando registro de toda operación realizada. Garantiza que la información sea vista únicamente por quien tiene derecho a verla. Se usan mecanismos de encriptación (MD5) de los datos que por cuestiones de seguridad no deben viajar al servidor en texto claro, como es el caso de las contraseñas. Se hacen validaciones de la información tanto en el cliente como en el servidor, no obstante los usuarios acceden de manera rápida y operativa a la aplicación sin que los requerimientos de seguridad se conviertan en un retardo para ellos.

- **Restricciones en el diseño y la implementación**

Es una aplicación web desarrollada con la tecnología para creación de páginas web dinámicas PHP 5 y base de datos en MySQL.

- **Legales**

El producto es una aplicación para la gestión de los laboratorios de la UCI, se rige por las normas establecidas por la DIN. Las tecnologías escogidas para el desarrollo de la aplicación, están basadas en la licencia GNU/GPL y la herramienta Visual Paradigm que la universidad cuenta con una licencia.

- **Confiabilidad**

La herramienta de implementación a utilizar tiene soporte para recuperación ante fallos y errores. En caso de alguna dificultad con el funcionamiento del sistema, el tiempo medio de reparación debe ser menor de 24 horas.

La aplicación estará funcionando las 24 horas del día; de esta forma es posible que los usuarios tengan acceso (según sus permisos) en todo momento a la información solicitada.

- **Software**

En los clientes:

- Sistema operativo Linux o Windows.
- Cualquier navegador Web.

En los servidores:

- Sistema Operativo Linux o Unix.
- Sistema Operativo Windows 98 o superior.
- Servidor Web Apache 2.0
- Servidor de Bases de Datos MySQL.
- PHP 5 con servicios web.
- Servidor de Aplicaciones Apache.

- **Hardware**

Para el Servidor:

- Se necesitan como requerimientos mínimos una PC con procesador Pentium II o superior.
- Microprocesador con una velocidad de 3.0 GHz o superior.
- Memoria RAM de 1GB o superior.
- HDD de 80 GB o superior.

Para el cliente:

- Se necesitan como requerimientos mínimos una PC con procesador Pentium II o superior.
- Microprocesador con una velocidad a 1 GHz o superior.
- Memoria RAM de 256 o superior.
- HDD de 20 GB o superior.

- **Disponibilidad**

Se garantiza a los usuarios de la aplicación el acceso a la información solicitada en todo momento (si tiene permiso para ello). La aplicación está diseñada para su funcionamiento constante, la disponibilidad del servicio está condicionada a los recursos y tipo de despliegue en los servidores de la UCI, así como los ciclos de mantenimiento determinados.

- **Restricciones**

Se utiliza Metodología Ágil, SCRUM\_XP ya que permite desarrollar software que funcionen, más que conseguir una buena documentación y que responde a los cambios rápidamente. Como herramienta

de apoyo el Visual Paradigm. Se utiliza como lenguaje de programación PHP 5 y el gestor de base de datos MySQL.

### 2.3 Actores del Sistema

Actor	Descripción
Técnico	Es el que más se beneficia de los resultados de este proceso, ya que puede solucionar la puesta en funcionamiento de las estaciones de trabajo y su incorporación al dominio UCI.
Jefe de Área	Es otro de los Actores que se benefician del proceso, este es el máximo responsable en un Docente de los laboratorios, este hereda todas las funcionalidades del técnico, además puede obtener reportes ya que de esta forma puede conocer cada una de las acciones que realicen los técnicos de su área, también puede insertar o eliminar en dependencia a su necesidad los usuarios a los Sec-Labs-Admin.
Especialista	Este Actor también se beneficia del proceso, este hereda todas las funcionalidades del jefe de área, además es el que tiene todos los privilegios en la aplicación, es el encargado del módulo administrar en el cual puede insertar usuarios nuevos o eliminar alguno existente.

2.4 Descripción de Historial de Usuario

Historial de Usuario: Autenticar

Historia de Usuario	
<b>Número:</b> 01	<b>Nombre de Historia de Usuario:</b> Autenticar.
<b>Modificación de Historia de Usuario Número:</b> ninguna.	
<b>Usuario:</b> Yusimil Dávila Jany Díaz Rodríguez	<b>Iteración Asignada :</b> 1
<b>Prioridad:</b> Alta.	<b>Puntos Estimados:</b> 2 semanas.
<b>Riesgo en Desarrollo:</b> Alta.	<b>Puntos Reales:</b> 1
<b>Descripción:</b> El usuario inserta sus datos (usuario y contraseña), la aplicación verifica que sean correctos y otorga el acceso al sistema con los permisos correspondientes a dicho usuario.	
<b>Observaciones:</b>	

Historial de Usuario: Administrar

Historia de Usuario	
<b>Número:</b> 02	<b>Nombre de Historia de Usuario:</b> Administrar
<b>Modificación de Historia de Usuario Número:</b> ninguna.	
<b>Usuario:</b> Yusimil Dávila Jany Díaz Rodríguez.	<b>Iteración Asignada:</b> 2
<b>Prioridad:</b> Alta	<b>Puntos Estimados:</b> 3
<b>Riesgo en desarrollo:</b> Alta.	<b>Puntos Reales:</b> 2
<b>Descripción:</b> Para Insertar un usuario el Especialista tiene que ingresar el número de solapín, la aplicación verifica en el Directorio Activo y se le asigna un área y un rol. En el caso del modificar y eliminar solo tiene que buscar por cualquiera de los criterios que aparecen, solo puede ser modificada el área y los permisos.	
<b>Observaciones:</b> La aplicación verifica que no exista en la base de dato de UCILab.	



Historial de Usuario: Eliminar PC del dominio

Historia de Usuario	
<b>Número:</b> 03	<b>Nombre de Historia de Usuario:</b> Eliminar PC del dominio
<b>Modificación de Historia de Usuario Número:</b> ninguna.	
<b>Usuario:</b> Yusimil Dávila Jany Díaz Rodríguez.	<b>Iteración Asignada:</b> 3
<b>Prioridad:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en desarrollo:</b> Alta.	<b>Puntos Reales:</b> 1
<b>Descripción:</b> En este caso desde que el usuario se autentica el sistema muestra el docente y los laboratorios a los que tiene permiso, se selecciona el laboratorio y lista las PC que están en el dominio con el nombre correctamente, si desea eliminar alguna, la marca y da al botón " Eliminar "	
<b>Observaciones:</b> Solo se mostrarán las PC que estén en el dominio con el nombre establecido por la dirección de Redes y Seguridad Informática	

Historial de Usuario: Generar Reporte.

Historia de Usuario	
	Nombre de Historia de Usuario: Generar Reporte.
<b>Número:</b> 04	
<b>Modificación de Historia de Usuario Número:</b> ninguna.	
<b>Usuario:</b> Yusimil Dávila Jany Díaz Rodríguez.	<b>Iteración Asignada:</b> 4
<b>Prioridad:</b> Alta	<b>Puntos Estimados:</b> 1.
<b>Riesgo en desarrollo:</b> Alta.	<b>Puntos Reales:</b> 1
<b>Descripción:</b> El caso de uso se inicia cuando el Jefe de Área va a solicitar algún reporte. El registro le muestra la interfaz correspondiente para escoger un criterio de búsqueda. El caso de uso finaliza cuando se emite el resultado de la operación solicitada	
<b>Observaciones:</b>	

Historial de Usuario: Gestionar Grupos

Historia de Usuario	
<b>Número:</b> 05	<b>Nombre de Historia de Usuario:</b> Gestionar Grupos
<b>Modificación de Historia de Usuario Número:</b> ninguna.	
<b>Usuario:</b> Yusimil Dávila Jany Díaz Rodríguez.	<b>Iteración Asignada:</b> 5
<b>Prioridad:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en desarrollo:</b> Medio.	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La aplicación muestra la interfaz grupos que no es mas que donde existen los grupos llamados Sec_Labs Admin por cada una de las área docentes que existen en la dirección, cuando se instalan las PC de un docente se agrega como administrador de las PC y ya todos los usuarios que estén en el grupo son automáticamente administradores de las mismas. Este historial de usuario es para desde la propia aplicación agregar los usuarios a estos grupos o eliminarlos cuando sea necesario.	
<b>Observaciones:</b> La aplicación muestra un buscar usuario para así saber si el usuario está insertado en dichos grupos o no y nos especifica a que grupos este pertenece.	

Historial de Usuario: Buscar PC en Dominio

Historia de Usuario	
<b>Número:</b> 06	<b>Nombre de Historia de Usuario:</b> Buscar PC en Dominio
<b>Modificación de Historia de Usuario Número:</b> ninguna.	
<b>Usuario:</b> Yusimil Dávila Jany Díaz Rodríguez.	<b>Iteración Asignada:</b> 5
<b>Prioridad:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en desarrollo:</b> Medio.	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La aplicación muestra en la interfaz Máquina un “Buscar Máquinas” en el cual el Jefe de Área puede buscar ya sea una máquina específica o un grupo de máquinas con solo parte del nombre de la PC que este desea. Si pone todo el nombre aparecerá específicamente la PC buscada, si introduce parte del nombre aparecerán todas las PC que contienen esta parte en su nombre, se marca y se da al botón eliminar.	

**Observaciones:** En este caso si aparece cualquier PC que este en el dominio uci.cu.

Historial de Usuario: Actualizar Usuario de Conexión

Historia de Usuario	
<b>Número:</b> 07	<b>Nombre de Historia de Usuario:</b> Actualizar Usuario de Conexión
<b>Modificación de Historia de Usuario Número:</b> ninguna.	
<b>Usuario:</b> Yusimil Dávila Jany Díaz Rodríguez.	<b>Iteración Asignada:</b> 5
<b>Prioridad:</b> Alta	<b>Puntos Estimados:</b> 1
<b>Riesgo en desarrollo:</b> Medio.	<b>Puntos Reales:</b> 1
<b>Descripción:</b> La aplicación muestra en el módulo Administración un botón "Usuario de Conexión" a partir del cual muestra un formulario en el cual se especifica el usuario con el cual se va a conectar la aplicación y la contraseña del mismo. Este módulo solo lo podrá ver el especialista.	
<b>Observaciones:</b> Este usuario debe ser un usuario impersonal pero también puede ser el usuario de cualquiera que sea administrador en el Directorio Activo	

2.5 Diagrama de Historia de Usuario del Sistema

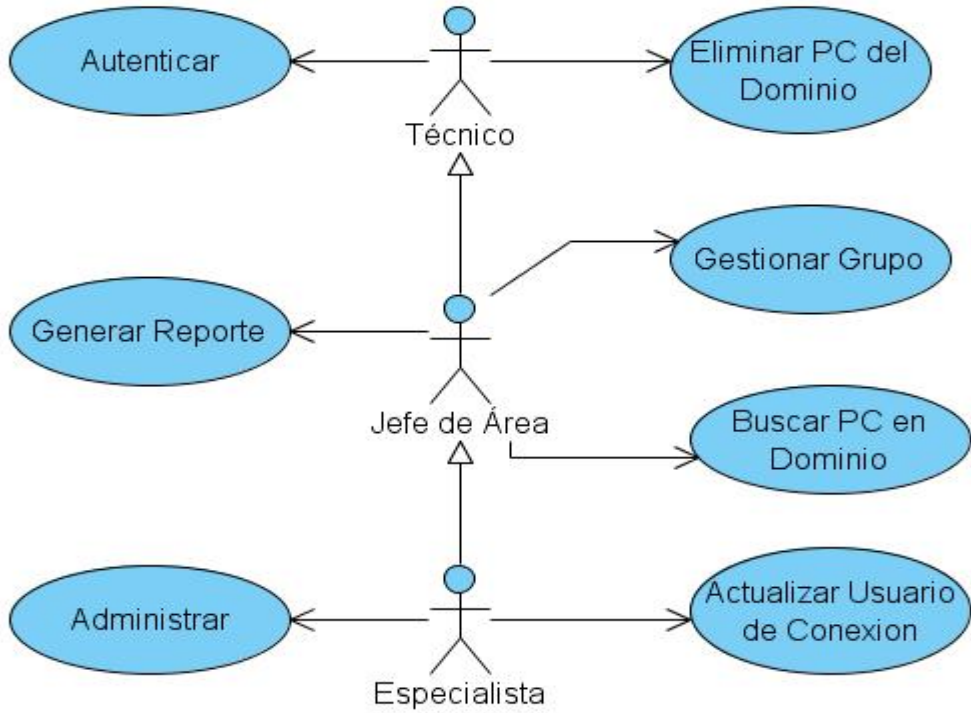


Figura 7: Diagrama historia de Caso de Uso del Sistema.

2.6 Plan de Releases

El Plan de Releases e Iteraciones tiene como entrada la relación de Historias de Usuario definidas previamente. Para colocar una historia en cada iteración se tiene en cuenta la prioridad que definió el cliente para dicha Historia. Como resultado de la priorización de Historias se llegó a la siguiente planificación.

Release	Historias de Usuario	Tiempo Estimado (semanas)
1	01	2
2	02	3
3	03	2
4	04	1
5	05	3
6	06	1
7	07	2

### **Conclusiones**

Dada la propuesta de solución desarrollada en el presente capítulo teniendo en cuenta los Procesos de Negocio y los Requerimientos, se puede comenzar a implementar la para la gestión de recursos a través del Directorio Activo de la Universidad de las Ciencias Informáticas. En el capítulo 3 del presente trabajo de diploma será tratado este tema siguiendo lo marcado por los Historiales de Usuario.

## **Capítulo3 CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA**

### **Introducción**

En el presente capítulo se modelan los artefactos que exige la Metodología Ágil SCRUM\_XP para la fase de desarrollo. Se presenta las pantallas tipos de la aplicación informática que cumple con el manual de pautas de diseño de la Universidad de las Ciencias Informáticas. Se realiza una breve explicación de los patrones de diseño utilizados y se muestra el estándar de codificación empleado en la implementación. Se describe además la arquitectura a tener en cuenta para el diseño del sistema.

### **3.1 Plantilla de Modelo del Diseño**

# Construcción de la Solución Propuesta

## Plantilla Modelo de Diseño

### 1. Diagrama de Paquetes:

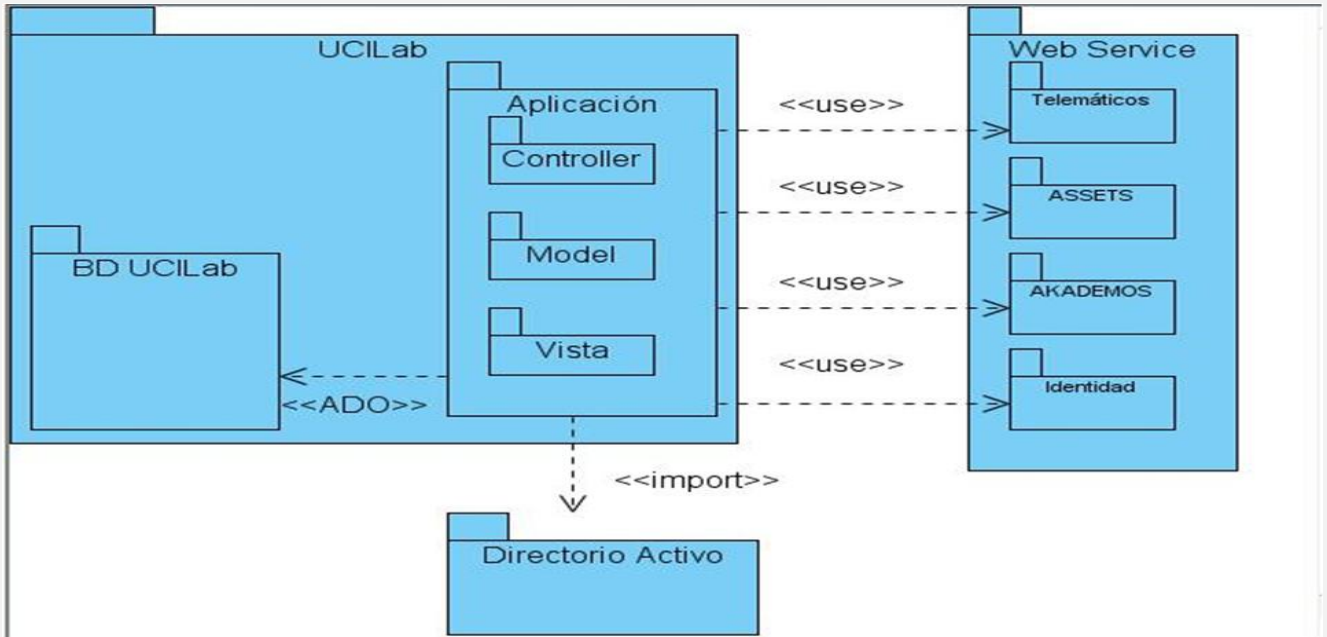


Figura 8: Diagrama de Paquete

### Observaciones:

### 2. Diagrama de Clases:

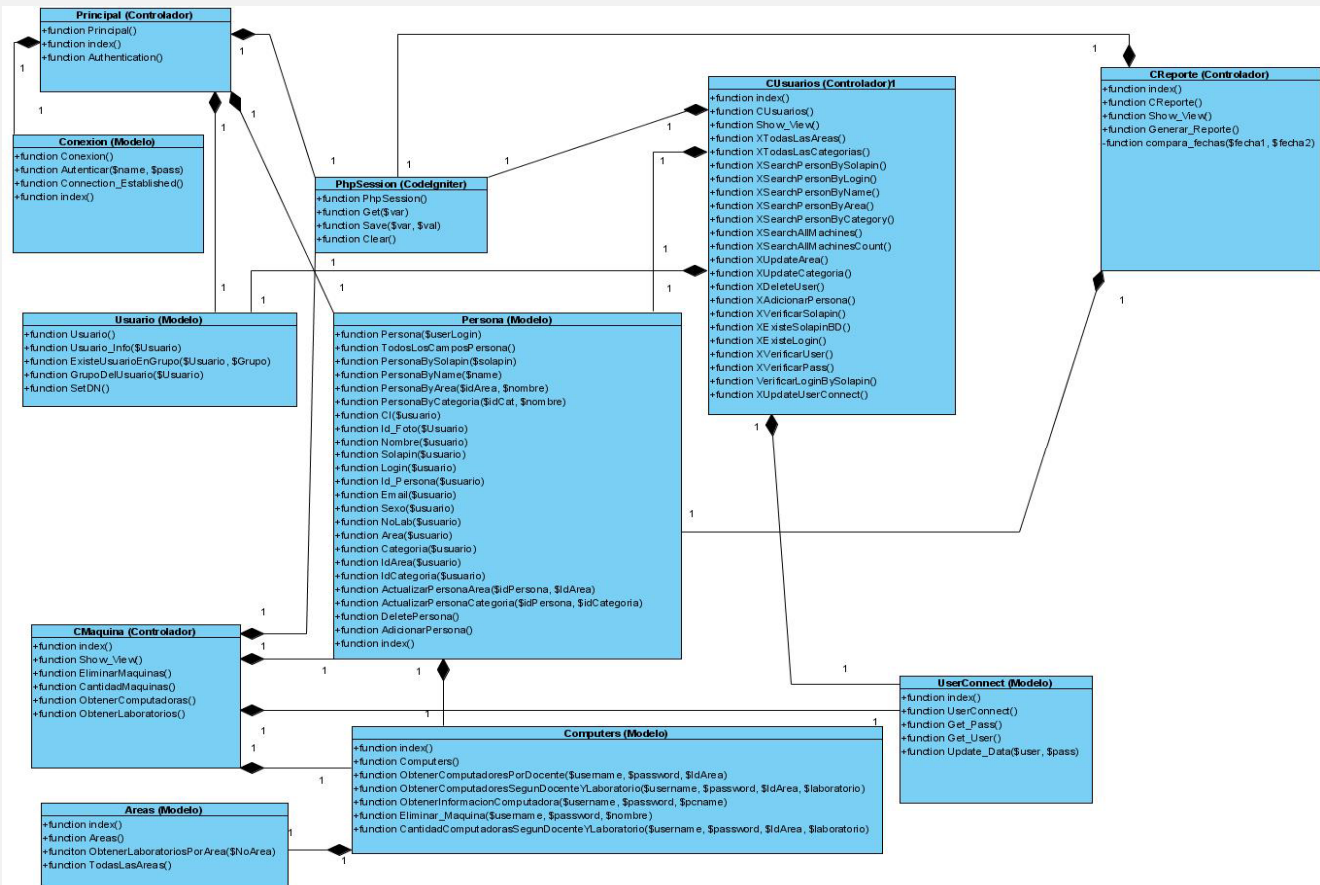


Figura 9: Diagrama de Clases.

### 3.2 Descripción del Diagrama de Clases

La clase **PhpSession** es una clase propia del CodeIgniter, la cual crea variables globales que recorren el sistema completo. Esta clase interactúa con las clases controladoras que son Principal, CMáquina, CReporte y CUsuario a través de una relación de composición.

La clase **Principal** es la clase que controla la autenticación y verifica que el usuario exista en el Directorio Activo. Esta se relaciona mediante composición con las clases Conexión, PhpSession, Persona y Usuario.

La clase **CMáquina** es la clase que controla el módulo máquina.

La clase **CReporte** es la clase que controla el módulo Reporte.

La clase **CUsuarios** es la clase que controla el módulo Usuario.

La clase **CGrupo** administra todo los grupos de administración Sec- Lab- Admins

La clase **Conexion** es la clase que conecta con ADLDAP y verifica si eres usuario. Esta se relaciona mediante composición con la clase Principal.

La clase **Usuario** es la clase que realiza las consultas al dominio UCI. Esta se relaciona mediante composición con las clases CUsuario y Principal.

La clase **Persona** es la clase que gestiona contra la base de datos local. Esta se relaciona mediante composición con las clases Principal, CUsuario y CMáquina.

La clase **Computers** es la clase que gestiona contra el dominio las computadoras. Esta se relaciona mediante composición con la clase Persona y CMáquina.

La clase **UserConnect** es la clase que gestiona la interrelación de la aplicación con el Directorio Activo de la universidad. Esto se logra mediante un usuario impersonal (laboratorios@uci.cu) al que se le asignan los permisos requeridos para el funcionamiento de la aplicación. Esto evita redundancia en la autenticación. Esta se relaciona mediante composición con la clase CUsuario y CMáquina.

La clase **Áreas** es la clase que muestra los docentes, los laboratorios y las computadoras. Esta se relaciona mediante composición con la clase Computers.

La clase **Grupo** es la que hace todas las consultas contra la base de datos y gestiona contra la clase LDAP.



## 3.3 Modelo de Datos

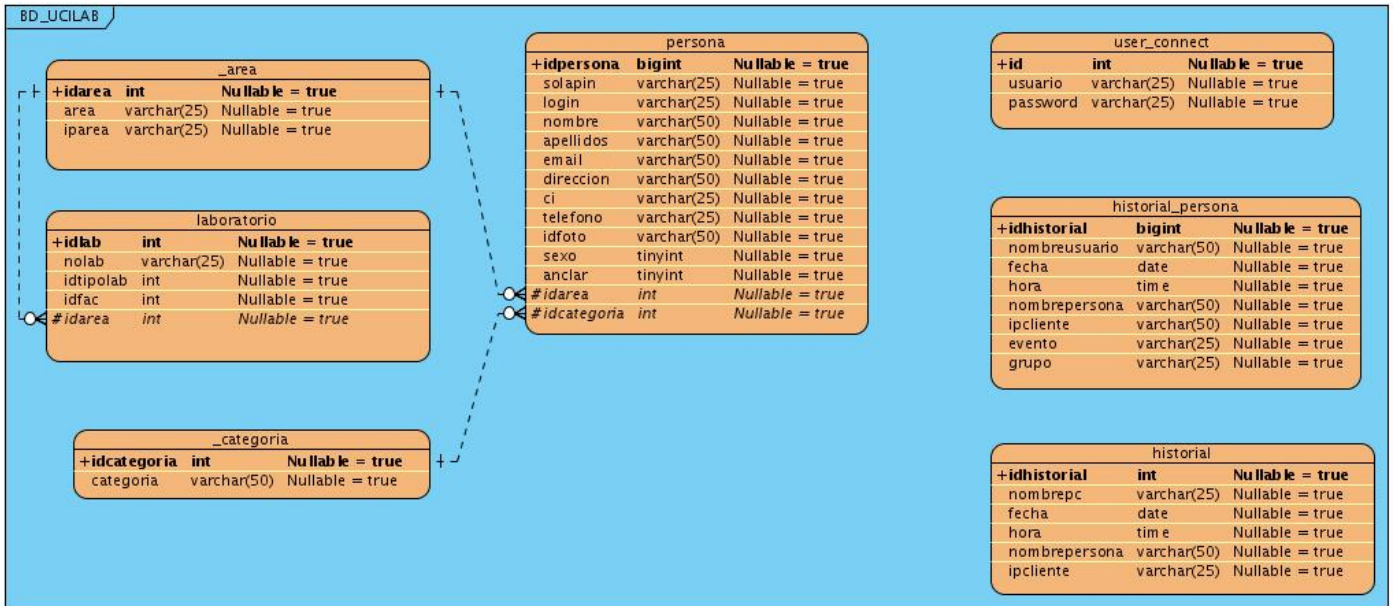


Figura 10: Modelo de Datos.

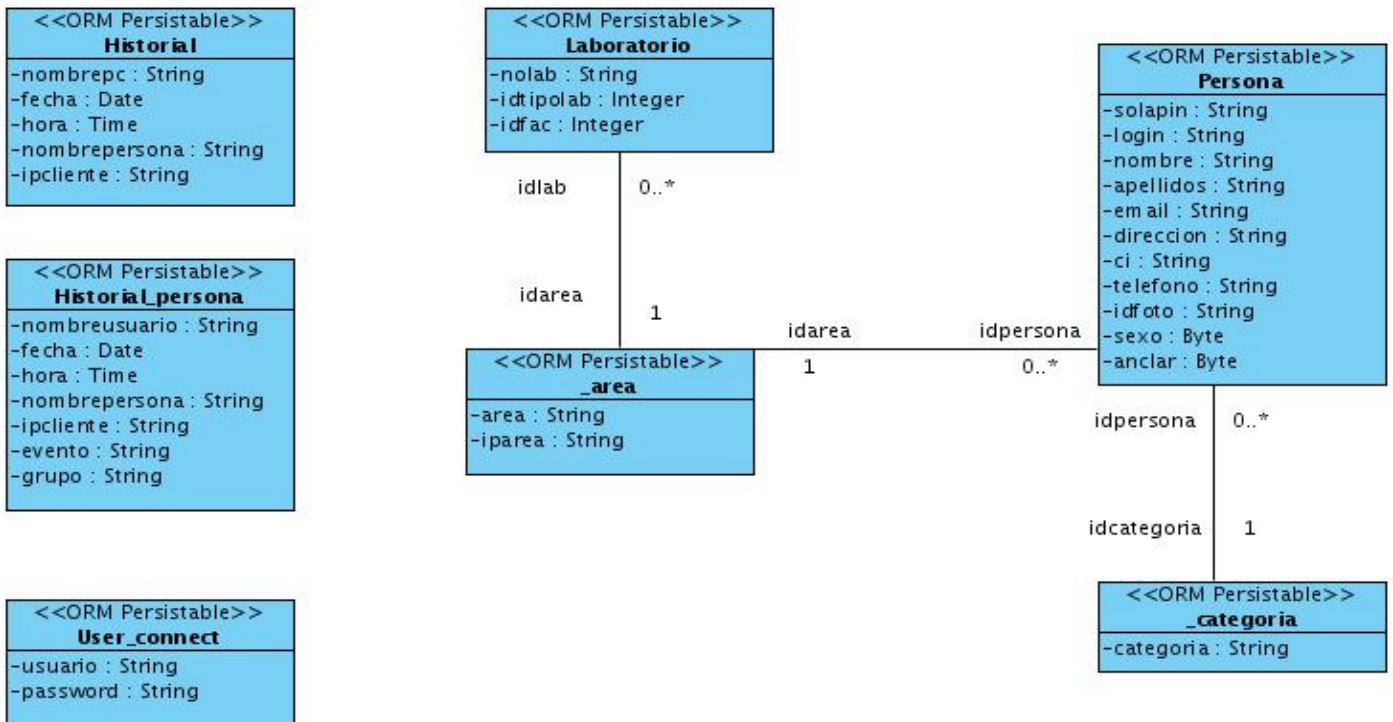


Figura 11: Diagrama de Clases Persistentes.

De la base de datos UCILab se usa una parte del conjunto de tablas que ésta contiene, dentro de ellas se encuentran: Persona, Área, Laboratorio y Categoría.

La tabla Persona registra todos los datos del personal de laboratorios los cuales se obtienen a través de los Web Service que brinda la DIN, la misma no llega a ser una réplica del todo ya que en esta existe más información que no se refleja en dichos servicios como son: las tablas nomencladores, área y categoría.

Las tablas que no están relacionadas son las siguientes:

User\_conect que es la guarda el usuario y contraseña de los usuarios que se conectan a la aplicación.

La contraseña se guarda encriptada por seguridad y la tabla Historial\_persona e Historial son creadas para guardar la información de cada una de las operaciones que se realizan en la aplicación.

### 3.4 Patrones de Diseño

Scrum\_XP propone un diseño simple, para una solución que pueda funcionar y ser implementada, en la cual un diseño de software correcto es aquel que supera todas las pruebas, no tiene lógica duplicada, pone de manifiesto las intenciones importantes de los programadores y tiene el mínimo número de clases y métodos. Por lo que en este apartado no se hará énfasis en los patrones de diseño, pero no pasarán por alto. Reflejando la utilización de los patrones básicos GRASP (General Responsibility Assignment Software Patterns) referentes a cuestiones y aspectos fundamentales del diseño. Los patrones GRASP incluyen al Experto, Creador, Alta Cohesión, Bajo Acoplamiento y Controlador los cuales se verán en cierta medida reflejados.

### 3.5 Arquitectura del sistema

El objetivo principal de la Dirección de Informatización es informatizar todos los procesos internos en cada una de las áreas que rigen la vida de la Universidad, desde la perspectiva de una Ciudad Digital, logrando una total integración de todas las entidades, flujos y/o procesos, basado en el funcionamiento armónico de las tecnologías y los servicios informáticos. Es por este motivo que la arquitectura de la para la gestión de usuarios y cuentas de computadoras a través del Directorio Activo de la Universidad de las Ciencias Informáticas se rige por la arquitectura que define la DIN. Esta arquitectura se basa en un Entorno Descentralizado, Orientado a Servicios, Bajo Acoplamiento entre los servicios y colaboración de estos servicios para dar solución a problemas. Utilización de protocolos estándares, no propietarios, aprobados por la Dirección.

El patrón de diseño Modelo Vista Controlador o MVC describe una forma, muy utilizada en el Web, de organizar el código de una aplicación separando los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

Modelo - Componente encargado del acceso a datos.

Vista - Define la interfaz de usuario, HTML+CSS... enviados en el navegador

Controlador - Responde a eventos y modifica la vista y el modelo. [25]

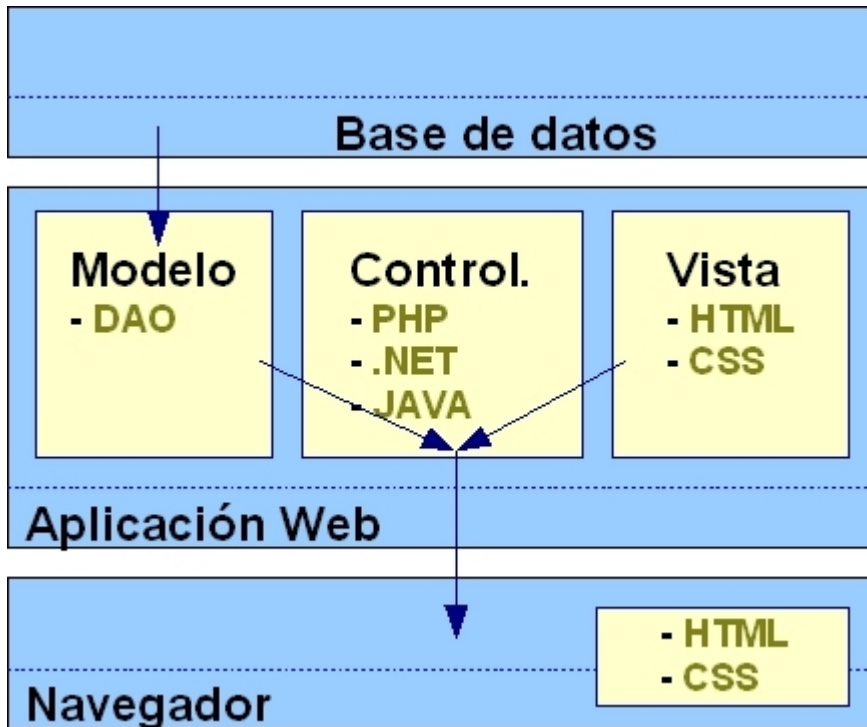


Figura 12: Modelo Vista Controlador

### 3.6 Pantallas Tipos del Sistema

La Pantalla "Autenticar" es la Interfaz principal de la aplicación. En esta es donde el usuario pone su usuario y contraseña para acceder a la misma.

## PANTALLA TIPO GESTIÓN -AUTENTICAR

The image shows a screenshot of a web application interface for authentication. At the top left, there is a header area labeled "PC-Gestor". Below this, the main content area is a light gray rectangle. Inside this rectangle, there is a smaller, centered gray box containing the authentication form. The form consists of two text input fields: the first is labeled "USUARIO" and the second is labeled "CONTRASEÑA". Both input fields have a placeholder text "enter text...". Below the input fields, there is a button labeled "Aceptar".

Figura 13: Pantalla Tipo Autenticar

La Pantalla "Autenticar" muestra dependiendo la categoría del usuario los módulos a los que tiene permiso.

En la pantalla máquina muestra las áreas, laboratorios y las PC, también tiene un buscador que es solo para los especialistas.

## PANTALLA TIPO GESTIÓN -MÓDULO MÁQUINA

The screenshot displays the 'PC-Gestor' application interface. At the top, there is a header with 'PC-Gestor' on the left and 'Nombre y Apellidos-Usuario Area' on the right. Below the header is a navigation menu with four tabs: 'Máquinas', 'Grupos', 'Administración', and 'Reportes'. The 'Máquinas' tab is currently selected.

On the left side, there is a vertical menu with five buttons labeled 'Docente 1' through 'Docente 5'. Below this menu is a search section with the label 'Buscar', a text input field containing 'enter text...', and an 'Aceptar' button.

The main content area shows 'Docente 1-Laboratorio 1' and a 'Paginado' dropdown menu. Below this is a table with the following data:

Número	Selección	Nombre de la PC	Sistema Operativo
1	<input checked="" type="checkbox"/>		
2	<input type="checkbox"/>		
3	<input type="checkbox"/>		
4	<input type="checkbox"/>		
5	<input type="checkbox"/>		
6	<input type="checkbox"/>		
7	<input type="checkbox"/>		
8	<input type="checkbox"/>		
9	<input type="checkbox"/>		

Below the table, there is a 'Cantidad Marcadas' label and an 'Eliminar del Dominio' button.

Figura 14: Pantalla Tipo Máquina

La Pantalla “Máquina ” muestra todas las PC que están en dominio según el laboratorio seleccionado .La tabla muestra la opción de eliminar del dominio, que al seleccionar las PC y pinchar en dicho botón, serán eliminadas del DNS.

La Pantalla “Grupos” le muestra al usuario todos los grupos Sec-Labs-Admin a los cuales este tiene permiso, también tiene un buscador.

## PANTALLA TIPO GESTIÓN -MÓDULO GRUPO

**PC-Gestor** Nombre y Apellidos-Usuario  
Area

Máquinas Grupos Administración Reportes

Sec-Labs-Admin 1 Paginado

Número	Selección	Nombre y Apellidos	Usuario
1	<input checked="" type="checkbox"/>		
2	<input type="checkbox"/>		
3	<input type="checkbox"/>		
4	<input type="checkbox"/>		
5	<input type="checkbox"/>		
6	<input type="checkbox"/>		
7	<input type="checkbox"/>		
8	<input type="checkbox"/>		
9	<input type="checkbox"/>		

Buscar usuarios  
enter text...  
**Aceptar**

Insertar nuevo usuario  
enter text...  
**Insetar**

Eliminar usuarios Seleccionados  
**Eliminar**

Figura 15: Pantalla Tipo Grupos.

La Pantalla “Grupos” es visible solo para los Jefe de Área y el Especialista. Ésta muestra los grupos Sec-Labs-Admin a los cuales este tiene permiso, donde ,dándole clic encima de estos te muestra todos los usuarios que están en dicho grupo , esta te brinda la opción de insertar usuarios nuevos y de eliminar alguno existente y también tiene un buscador.

La Pantalla “Administración” es visible solo para el Especialista. Esta muestra un buscar usuario por diferentes roles, y también tiene un insertar usuario.

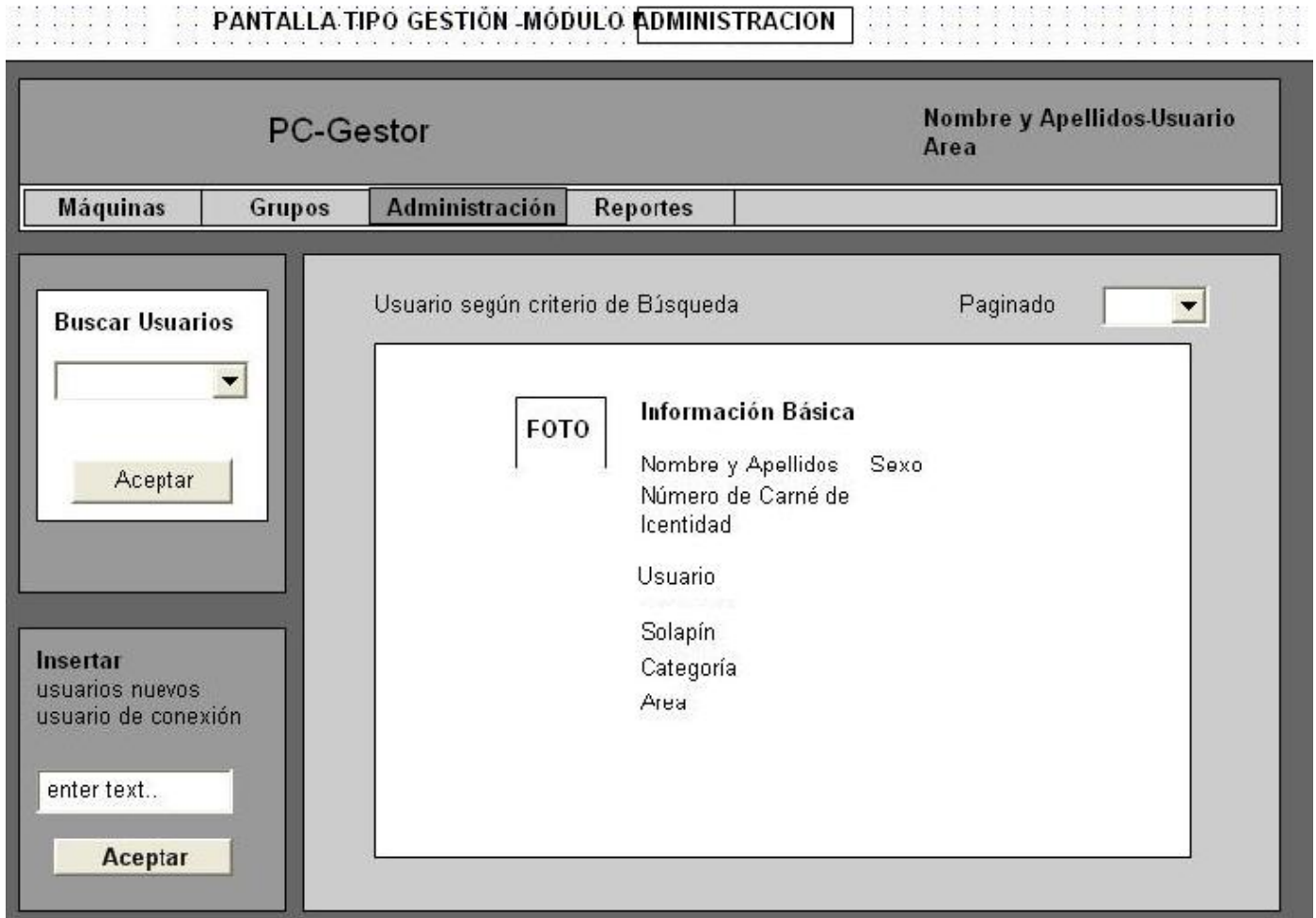


Figura 16: Pantalla Tipo Administración.

En la Pantalla “Administración” el Especialista puede buscar usuarios por los diferentes roles como son solapín, usuario, nombre, área y categoría, los cuales solo podrá modificar el área y la categoría de los mismos, también muestra un insertar usuario para los técnicos que entran nuevos en la dirección donde debe llenar los siguientes campos del formulario como son solapín, área y categoría y al entrar los datos esto lo actualiza en la base de datos.

La Pantalla “Reportes” es visible solo para los Jefe de Área y el Especialista. Esta muestra la interfaz donde puedes escoger los reportes que deseas efectuar ya sea por fecha, hora, docente, usuario, y al dar en el Botón Generar Reporte este te lo muestra.

## PANTALLA TIPO GESTIÓN MÓDULO REPORTES

PC-Gestor Nombre y Apellidos-Usuario  
Area

Máquinas Grupos Administración Reportes

Tipos de Reportes  
Máquinas  
Aceptar

Paginado

**Reporte-Máquinas**  
 Todas Area  Laboratorio   
Fecha Inicio  Final  Hora Inicio  Final

**Reporte-Grupos**  
 Todos Grupo  Evento   
Fecha Inicio  Final  Hora Inicio  Final

Generar Reportes

Figura 17: Pantalla Tipo Reportes.

En la pantalla reporte cuando se genera un reporte por cualquiera de los criterios muestra un formulario con el resultado.

### 3.7 Casos de Prueba

Bajo la metodología Scrum XP, la creación de código es dirigida por las pruebas unitarias. Estas pruebas unitarias son establecidas antes de escribir el código y ejecutadas por el programador constantemente ante cada modificación del sistema. Scrum XP introduce la programación extrema la cual define entre iteración e iteración un conjunto de pruebas de funcionalidad, para poder avanzar a una iteración superior y proponiendo así esta metodología generar como artefacto la Plantilla Caso de Prueba de Aceptación. En la cual se escriben las pruebas realizadas según la Historia de Usuario seleccionado para realizar la comprobación y validar las funcionalidades del sistema.



### 3.7.1 Caso de Prueba del Historial de Usuario: Autenticar usuario

En esta prueba habrá que comprobar que el acceso de los usuarios sea correcto, de manera que solo puedan acceder a los servicios que brinda la aplicación los usuarios previamente autenticados y que en dependencia al tipo de persona solo se le muestren los campos a los que tienen permisos.

<b>Caso de Prueba de Aceptación</b>	
<b>Código Caso de Prueba:0101</b>	<b>Nombre Historial de Usuario:</b> Autenticar.
<b>Nombre de la persona que realiza la prueba:</b> Yusimil Dávila Hernández Jany Díaz Rodríguez.	
<b>Descripción de la Prueba:</b> El usuario, al ejecutar la URL en un navegador Web, le aparecerá la página principal de la aplicación, en la que encontrará un formulario en el que se le solicita el nombre de usuario y la contraseña del dominio UCI. El usuario debe introducir la información requerida en dichos campos y cuando se cumple que el usuario no se encuentra registrado en la aplicación, no tendrá acceso a los servicios que se brindan en la misma. Después de introducir los campos en el formulario el usuario tendrá acceso a los servicios que él tendrá permisos en dependencia al tipo de persona (Especialista, Jefe de Área y Técnico.).	
<b>Condiciones de Ejecución:</b> Ninguna	
<b>Entrada/Pasos de ejecución:</b> El usuario abre la aplicación en un navegador Web. Aparece la interfaz en la que se solicita el nombre de usuario y la contraseña. El usuario (Yusimil Dávila Hernández) introduce dicha información en el campo correspondiente y presiona el botón "Aceptar". La aplicación verifica la información de ambos campos en la base de datos, comprobando así que sean correctas.	
<b>Resultados Esperados:</b> Se espera que el usuario Yusimil Dávila Hernández pueda acceder a los servicios de la aplicación a los que ella tiene permiso.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria	

### 3.7.2 Caso de Prueba del Historial de Usuario: Administrar

Este módulo después de autenticado, solo lo podrán ver los Especialistas.

<b>Caso de Prueba de Aceptación</b>	
<b>Código Caso de Prueba:0202</b>	<b>Nombre Historial de Usuario:</b> Administrar.
<b>Nombre de la persona que realiza la prueba:</b> Yusimil Dávila Hernández Jany Díaz Rodríguez.	
<b>Descripción de la Prueba:</b> El especialista después de autenticarse le aparecerá todos los servicios a los que él tiene acceso y Administrar es uno de ellos, que estará ubicada en la pestaña administración de la aplicación. Al ejecutarla este nos muestra un buscar en el cual podemos hacer la búsqueda por solapín, usuario, nombre, área y categoría, este te muestra los datos de la persona y solo pueden ser modificados los roles área y categoría y nos permite eliminar dicho usuario. Dicha aplicación también nos permitirá insertar usuarios nuevos, el cual nos muestra un formulario en el cual debemos llenar los campos que nos son solicitados y dar en el botón entrar datos.	
<b>Condiciones de Ejecución:</b> Estar autenticado.	
<b>Entrada/Pasos de ejecución:</b> El especialista entra al módulo Administrar, busca el usuario que este quiere hacerle las modificaciones y modifica su área o la categoría. Si desea insertar un nuevo usuario ingresa su número de solapín y usuario, le asigna 1 área y 1 categoría y da al botón "Insertar". Si desea eliminar busca por cualquiera de los criterios y elimina.	
<b>Resultados Esperados:</b> Que el usuario modificado o insertado pueda realizar las operaciones de acuerdo a su nueva categoría. Que el usuario eliminado no acceda a la aplicación.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria	

### 3.7.3 Caso de Prueba del Usuario: Eliminar PC del dominio

En este módulo después de autenticado, este te muestra el docente al cual perteneces y los laboratorios los cuales tienes permiso.

#### Caso de Prueba de Aceptación

**Código Caso de Prueba:**0203

**Nombre Historial de Usuario:** Eliminar PC del dominio

**Nombre de la persona que realiza la prueba:** Yusimil Dávila Hernández  
Jany Díaz Rodríguez.

**Descripción de la Prueba:** El usuario después de autenticarse le aparecerá todos los servicios a los que él tiene acceso y el módulo Máquina es uno de ellos, en este le muestra al usuario los docentes y los laboratorios a los cuales este tiene permiso y de estos laboratorios las PC que se encuentran el dominio uci.cu con el nombre correctamente. Esta nos da la posibilidad de marcar las PC que deseamos eliminar del Directorio activo y pinchar en el botón eliminar.

**Condiciones de Ejecución:** Estar autenticado

**Entrada/Pasos de ejecución:** El usuario escoge el laboratorio, en este le aparecen las PC que se encuentran en el dominio uci.cu con la política de nombres establecidas, marca las PC que va a eliminar y pulsa el botón eliminar.

**Resultados Esperados:** Que se limpie el nombre del DNS y el usuario pueda incluir la PC en el dominio con el nombre correcto.

**Evaluación de la Prueba:** Prueba satisfactoria

## 3.7.4 Caso de Prueba del Historial de Usuario: Generar Reportes

<b>Caso de Prueba de Aceptación</b>	
<b>Código Caso de Prueba:0202</b>	<b>Nombre Historial de Usuario:</b> Generar Reporte
<b>Nombre de la persona que realiza la prueba:</b> Yusimil Dávila Hernández Jany Díaz Rodríguez.	
<b>Descripción de la Prueba:</b> En esta prueba lo que hacemos es generar un reporte por “Máquina eliminada” siguiendo uno o todos de los siguientes criterios (Docente, Laboratorio, Fecha u Hora) y Por “Usuarios” especificando Grupo, Evento, Fecha, Hora.	
<b>Condiciones de Ejecución:</b> Estar autenticado.	
<b>Entrada/Pasos de ejecución:</b> El Jefe de Área escoge los criterios dependiendo del reporte que este desea, le da al botón Generar Reporte y este genera un formulario con todos los datos.	
<b>Resultados Esperados:</b> Poder conocer las operaciones que han realizado los usuarios en la aplicación.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria	

## 3.7.5 Caso de prueba del Historial de Usuario: Gestionar Grupos

En esta prueba después de autenticado, este módulo solo lo podrán ver los jefes de área y el Especialista.

<b>Caso de Prueba de Aceptación</b>	
<b>Código Caso de Prueba:0202</b>	<b>Nombre Historial de Usuario:</b> Gestionar Grupos

**Nombre de la persona que realiza la prueba** Yusimil Dávila Hernández  
Jany Díaz Rodríguez.

**Descripción de la Prueba:** El Jefe de Área después de autenticarse le aparecerá el módulo grupos, este muestra el grupo al cual tiene acceso y al dar clic sobre este te lista todos los nombres y apellidos y usuarios de los técnicos que están contenidos en dicho grupo y te brinda la opción mediante un botón eliminar, de eliminar todos los que fueron seleccionados, además tiene un botón para insertar usuarios nuevos, que aquí inserta el usuario del nuevo técnico del docente y al presionar el botón insertar este lo incluye en el listado donde están los otros técnicos.

**Condiciones de Ejecución:** Estar autenticado

**Entrada/Pasos de ejecución:** El jefe de Área escribe el usuario del nuevo técnico y da en el botón insertar. Para eliminar marca el técnico y da en el botón eliminar.

**Resultados Esperados:** Que el usuario insertado aparezca en el grupo, y el eliminado lo borre del mismo.

**Evaluación de la Prueba:** Prueba satisfactoria

### 3.7.6 Caso de Prueba del Historial de Usuario: Buscar PC en Dominio

#### Caso de Prueba de Aceptación

<b>Código Caso de Prueba:</b> 0202	<b>Nombre Historial de Usuario:</b> Buscar PC en Dominio
------------------------------------	--

**Nombre de la persona que realiza la prueba** Yusimil Dávila Hernández  
Jany Díaz Rodríguez.

**Descripción de la Prueba:** El jefe de área en el buscador especifica todo o parte del nombre y la aplicación le muestra todas las PC que existen en el dominio uci.cu que comienza el nombre de esa forma. El jefe de área puede eliminar una o varias PC

**Condiciones de Ejecución:** Estar autenticado

**Entrada/Pasos de ejecución:** El jefe de Área del docente 4 escribe en el buscador (LD04), le aparecen un listado de todas las PC que su nombre empieza de esta manera. El Jefe de Área decide borrar todas las PC que aunque su nombre empieza con LD04 tiene el

resto mal aprieta la tecla Shift del teclado marca todas las que desea eliminar y pulsa el botón “Eliminar”

**Resultados Esperados:** Que todas las PC marcadas y eliminadas queden fuera del dominio uci.cu

**Evaluación de la Prueba:** Prueba satisfactoria

### 3.7.7 Caso de Prueba del Historial de Usuario: Actualizar usuario de conexión

#### Caso de Prueba de Aceptación

**Código Caso de Prueba:**0202

**Nombre Historial de Usuario:** Actualizar usuario de conexión.

**Nombre de la persona que realiza la prueba** Yusimil Dávila Hernández  
Jany Díaz Rodríguez.

**Descripción de la Prueba:** En esta prueba lo que hacemos es actualizar el usuario de conexión, ya sea porque se cambió la contraseña y hay que actualizarla o porque cambio el usuario

**Condiciones de Ejecución:** Estar autenticado

**Entrada/Pasos de ejecución:** El Especialista entra al módulo Administración en el cual hay un botón “Usuario de Conexión”, este botón lo lleva a una interfaz en la que aparece un formulario y pone usuario, contraseña y confirmación de la misma

**Resultados Esperados:** Que se actualice el usuario de conexión para que la aplicación pueda seguir funcionando

**Evaluación de la Prueba:** Prueba satisfactoria

### 3.8 Resultados obtenidos

En este punto se relacionan los resultados obtenidos hasta el momento por el equipo de desarrollo de la para la gestión de recursos a través del Directorio Activo de la Universidad de las Ciencias Informáticas. Resaltar que posteriormente se pueden esperar muchos más resultados en otras versiones.

## Logros de la aplicación

Entre las principales funcionales que posee la aplicación para la gestión de recursos a través del Directorio Activo de la Universidad de las Ciencias Informáticas se pueden mencionar:

- Es capaz de realizar la Gestión de Usuarios y Cuentas de Computadoras para un Técnico, Jefe de Área o Especialista teniendo en cuenta las diferencias existentes entre ellos en cuanto a categoría.
- Permite tener un mejor control de las máquinas de los laboratorios.
- Permite eliminar, insertar o modificar usuarios a conveniencia y necesidad de la Dirección de Laboratorios para realizar las diferentes operaciones.
- Permite generar reportes para así tener el control de las operaciones realizadas por cada uno de los usuarios de la aplicación.
- Permite insertar o eliminar usuarios a los Sec-Labs admin. garantizando así que los técnicos sean administradores de las PC del docente en que trabaja para garantizar las funciones establecidas para los técnicos de laboratorios.
- Posee interfaz gráfica sencilla y amigable.
- Se encuentra disponible las 24 horas.

## Conclusiones

Se puede concluir que en este capítulo se ha cumplido rigurosamente con todos los artefactos que se generan en esta fase (Desarrollo) de la metodología, por ejemplo: el “El Diagrama de Clases Persistentes” y “Diagrama de Paquetes”. Las pruebas realizadas al sistema implementado siguiendo el Estándar de Codificación han sido satisfactorias.

### CONCLUSIONES

A partir del desarrollo de la aplicación informática para la gestión de recursos a través del Directorio Activo de la Universidad de las Ciencias Informáticas se puede arribar a las siguientes conclusiones:

- ❖ EL estudio realizado detectó la inexistencia de una aplicación informática para la gestión de recursos a través del Directorio Activo de la Universidad de las Ciencias Informáticas lo que constituyó el punto de partida de la investigación.
- ❖ Con el levantamiento de los requisitos funcionales se identificaron las funcionalidades que constituye la base de partida para el diseño de las clases de la aplicación informática para la gestión de recursos a través del Directorio Activo de la Universidad de las Ciencias Informáticas
- ❖ Con el desarrollo de la aplicación se logró un menor tiempo de respuesta ante las afectaciones al servicio por concepto de gestión de recursos a través del Directorio Activo con mayor rapidez y eficiencia, logrando seguridad y protección en consecuencia con el nivel de acceso requerido.



### RECOMENDACIONES

A pesar de haber cumplido los objetivos generales de este trabajo a medida que se ha ido desarrollando el proyecto han surgido ideas que podrían implementarse en un futuro para lograr un sistema más provechoso y efectivo, para lo cual se recomienda:

- ❖ Publicar este documento de trabajo para que sirva de material de estudio a investigaciones futuras.
- ❖ Continuar el desarrollo de este sistema, adicionándole nuevas funcionalidades.
- ❖ Extender el sistema de manera que pueda ser utilizado en las facultades regionales de la universidad.
- ❖ Que se establezcan los Requerimientos No Funcionales de Software con las particularidades necesarias.

## BIBLIOGRAFÍA

1. Ariza Rojas, Maribel y Molina García, Juan Carlos. Introducción y principios básicos del desarrollo de software basado en componentes. Septiembre de 2004. [Citado el: 6 de 1 de 2010].
2. Jacobson, Ivar, Booch, Grady y Rumbaugh, James. El Proceso Unificado de Desarrollo de Software. s.l.: PEARSON EDUCACIÓN S.A, 2000.[Citado el: 15 de 11 de 2009]
3. David Nudelman. - MCSE entre otras cosas [En línea]2009 [Citado el 8 de 11 de 2009.] Disponible en: <http://geeks.ms/blogs/dnudelman/archive/2009/03/03/how-to-documentar-el-directorio-activo.aspx>
4. Desarrollo de software bajo metodologías ágiles (Agile Methods XP) en la práctica [En línea] [Citado el 15 de 11 de 2009.] Disponible en: <http://www.monografias.com/trabajos48/desarrollo-software-agil/desarrollo-software-agil.shtm/>
5. El Directorio Activo[En línea] [Citado el 15 de 11 de 2009.] Disponible en: <http://fferrer.dsic.upv.es/cursos/Windows/Avanzado/ch03.html>
6. El Rincón del Programador [En línea] 2008 [Citado el 7 de 12 de 2009.] Disponible en: <http://www.elrincondelprogramador.com/default.asp?id=29&pag=articulos/leer.asp>
7. Ileana Armenteros Vera1 e Ileana Alfonso Sánchez [En línea] [Citado el 22 de 11 de 2009.] Disponible en: [http://bvs.sld.cu/revistas/aci/vol12\\_2\\_04/aci06204..htm](http://bvs.sld.cu/revistas/aci/vol12_2_04/aci06204..htm)
8. Información general acerca de servicios Web[En línea] [Citado el: 8 de Noviembre de 2009] Disponible en: <http://msdn.microsoft.com/library/spa/default.asp?url=/library/SPA/cpguide/html/cpconwebseviceoverview.asp>
9. Introducción a los Servicios Web en Java. [En línea] [Citado el: 14 de Mayo de 2008] Disponible en: [http://www.programacion.com/tutorial/servic\\_web/](http://www.programacion.com/tutorial/servic_web/)
10. José H. Canós, Patricio Letelier yM<sup>a</sup> Carmen Penadés. Metodologías Ágiles en el Desarrollo de Software [En línea] [Citado el 15 de 11 de 2009.] Disponible en: <http://www.willydev.net/descargas/prev/TodoAgil.pdf>
11. Métodos del desarrollo del sistema de información [En línea] [Citado el 22 de 11 de 2009.] Disponible en: <http://html.rincondelvago.com/metodos-del-desarrollo-del-sistema-de-informacion.html>
12. Metodologías de Desarrollo Software [En línea] [Citado el: 23 de 11 de 2009] Disponible en: <http://www.scribd.com/doc/2050925/metodologias-de-desarrollo-software>

13. Metodologías modernas de desarrollo de Sistemas de Información [En línea] [Citado el 22 de 11 de 2009.] Disponible en:  
<http://www.monografias.com/trabajos12/docmento/docmento.shtml>
14. mundopc.net [En línea]2009 [Citado el 1 de 11 de 2009.] Disponible en:  
<http://mundopc.net/articulos/mitos-y-leyendas-el-directorio-activo-i-conceptos/>
15. Programación [En línea] 2008 [Citado el 5 de 1 de 2010.] Disponible en:  
<http://chuidiang.blogspot.com/2005/12/qu-son-los-patrones-de-diseo-el-patrn.html>
16. Programación Extrema [En línea] Junio del 2004.[Citado el: 23 de 11 de 2009.] Disponible en:  
[www.informatizate.net/articulos/metodologias\\_de\\_desarrollo\\_de\\_software\\_07062004.html](http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html)
17. Rafael Bravo Toledo Gestores personales de bases de datos bibliográficas [En línea] 2007 [Citado el 3 de 12 de 2009.] Disponible en:  
<http://www.infodoctor.org/rafabravo/pbs.htm>
18. Sistemas gestores de bases de datos [En línea] 2007 [Citado el 3 de 12 de 2009.] Disponible en:  
<http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>
19. Sánchez, Jordi. ¿Qué es un framework? [En línea] Septiembre de 2006. [Citado el: 4 de 12 de 2009.] Disponible en:  
<http://jordisan.net/modules/wordpress/2006/que-es-un-framework/>
20. The Apache Software Foundation [En línea] [Citado el: 3 de 12 de 2009.] Disponible en:  
<http://tomcat.apache.org/>
21. Visual Paradigm.[En línea][Citado el: 15 de 11 de 2009] Disponible en:  
[http://www.freedownloadmanager.org/es/downloads/Paradigma\\_Visual\\_para\\_UML\\_%5Bcuenta\\_de\\_Plataforma\\_de\\_Java\\_14715\\_p/](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%5Bcuenta_de_Plataforma_de_Java_14715_p/)
22. WebControl [En línea] 2008 [Citado el 7 de 12 de 2009.] Disponible en:  
<http://soluciones.grupointermark.com/webcontrol/portal.do;jsessionid=020C8036747ED2C768A5E72A1190C3BE?IDM=20&NM=1>
23. W3Schools.com En línea] 2008 [Citado el 5 de 1 de 2010.] Disponible en:  
<http://www.w3schools.com/ajax/default.asp>

### REFERENCIAS BIBLIOGRÁFICAS

1. Grupo BSI México. ¿Qué son los sistemas de gestión? BSI. [En línea] [Citado el: 06 de 11 de 2009.] Disponible en Web:  
<http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion/>.
2. Sabre. Sabre es el proveedor líder en tecnologías, distribución y servicios de mercadeo para la industria de viaje. Sabre Travel Network. [En línea] [Citado el: 06 de 11 de 2009.] Disponible en Web:  
<http://www.sabre.com.mx/acerca/general.htm>
3. ScriptLogic. Administración avanzada de Microsoft Active Directory. [En línea] [Citado el: 10 de 11 de 2009.] Disponible en Web:  
<http://www.ireo.com/productos/active-administrator.html>
4. AdventNet.. Simplifica la administración de Microsoft Directorio Activo. [En línea] [Citado el: 10 de 11 de 2009.] Disponible en Web:  
<http://www.ireo.com/productos/admanager-plus.html>
5. ScriptLogic.. Gestión completa de PCs y servidores. [En línea] [Citado el: 10 de 11 de 2009.] Disponible en Web:  
<http://www.ireo.com/productos/desktop-authority.html>
6. Scott Barnett, Richard Hyland.. adLDAP - LDAP Authentication with PHP for Active Directory. [En línea] [Citado el: 15 de 11 de 2009.] Disponible en Web:  
<http://adldap.sourceforge.net/>
7. Sergi Blanco Cuaresma. Marble Station. [En línea] 2008. [Citado el: 13 de 11 de 2009.]  
<http://www.marblestation.com/?p=644>
8. Ejemplos java y C/linux. [En línea] 2008. [Citado el: 13 de 11 de 2009.]  
<http://www.chuidiang.com/ood/metodologia/scrum.php>
9. Manuel Calero Solís. Una explicación de la programación extrema (XP). [En línea]. [Citado el: 13 de 11 de 2009.]  
<http://www.willydev.net/descargas/prev/ExplicaXp.pdf>
10. María A. Mendoza Sanchez. Metodologías De Desarrollo De Software. [En línea]. [Citado el: 13 de 11 de 2009.]  
[http://www.informatizate.net/articulos/metodologias\\_de\\_desarrollo\\_de\\_software\\_07062004.html](http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html)
11. Infor. [En línea] 1 de abril de 2009[Citado el: 20 de 11 de 2009.].

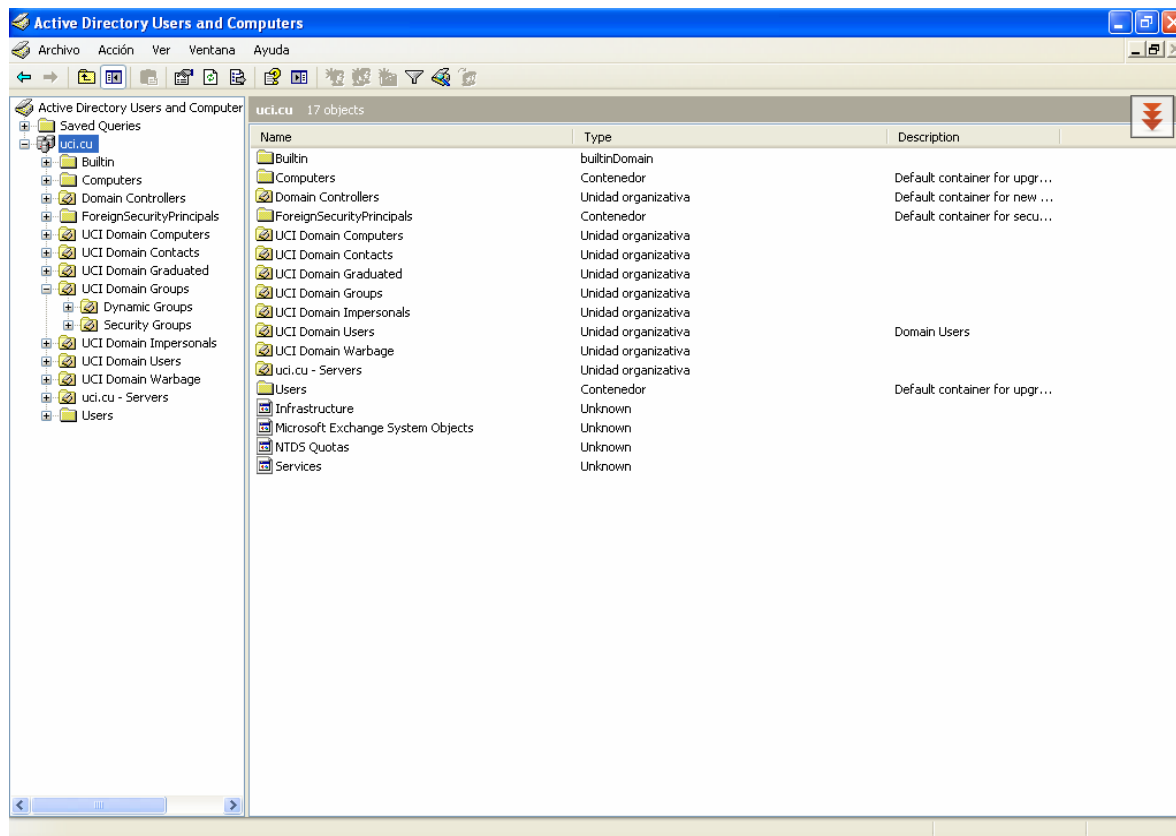
<http://www.infor.com>.

12. FDM. Visual Paradigm for UML (ME). Sitio de descargas de software. [En línea] [Citado el: 15 de 11 de 2009.] Disponible en Web:  
[http://www.freedownloadmanager.org/es/downloads/Paradigma\\_Visual\\_para\\_UML\\_\(M%C3%8D\)\\_14720\\_p/](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_(M%C3%8D)_14720_p/).
13. maestros del web [En línea] 2007. [Citado el: 20 de 11 de 2009.] Disponible en Web:  
<http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>
14. maestros del web [En línea] 2005. [Citado el: 20 de 11 de 2009.] Disponible en Web:  
<http://www.maestrosdelweb.com/editorial/ajax/>
15. Ortiz, Antonio. Error 500. [En línea] 2008. [Citado el: 22 de 11 de 2009.].  
<http://www.error500.net/apache>
16. Rubén Alvarez. DesarrolloWeb.com. [En línea] [Citado el: 16 de 11 de 2009.].  
<http://www.desarrolloweb.com/articulos/262.php>
17. Oracle Corporation and/or its affiliates. MySQL. [En línea] 2009. [Citado el: 17 de 11 de 2009.].  
<http://dev.mysql.com/>
18. Web Services Activity Statement [En línea] [Citado el: 25 de 11 de 2009.] Disponible en:  
<http://www.w3.org/2002/ws/Activity>
19. The Apache Software Foundation. [En línea]2009 [Citado el: 25 de 11 de 2009.] Disponible en:  
<http://httpd.apache.org/>
20. Miguel Angel Alvarez. DesarrolloWeb.com. [En línea]2009 [Citado el: 2 de 12 de 2009.].  
<http://www.desarrolloweb.com/articulos/codeigniter.html>
21. Pruebas [En línea] [Citado el 10 de 12 de 2009.] Disponible en:  
[http://rogeliodavila.com/tcs/TCS%20Notes%20JAVega/Parte\\_15\\_TestBlack.ppt](http://rogeliodavila.com/tcs/TCS%20Notes%20JAVega/Parte_15_TestBlack.ppt)
22. Requerimientos [En línea] [Citado el 18 de 12 de 2009.] Disponible en:  
<http://www.google.com.cu/#hl=es&q=que+es+un+requerimiento&meta=&aq=&oq=que+es+un+requerimiento&fp=bdc9a97574402bb1>
23. WorldLingo [En línea]2009 [Citado el 18 de 12 de 2009.] Disponible en:  
[http://www.worldlingo.com/ma/enwiki/es/Functional\\_requirements/2](http://www.worldlingo.com/ma/enwiki/es/Functional_requirements/2)
24. Iván Garcerant [En línea]2008 [Citado el 18 de 12 de 2009.] Disponible en:  
<http://synergix.wordpress.com/2008/07/07/requisito-funcional-y-no-funcional/>

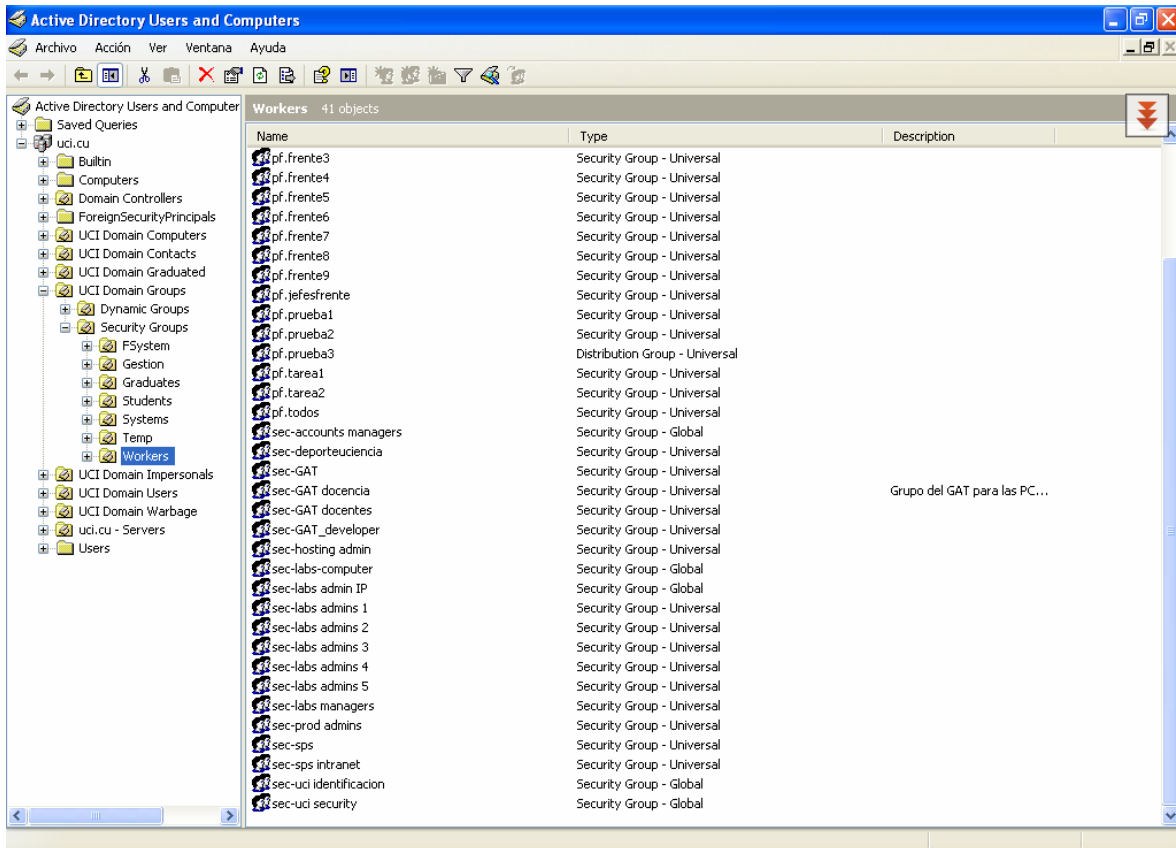
25. programacionweb.net [En línea]2009 [Citado el 22 de 12 de 2009.] Disponible en:  
<http://www.programacionweb.net/articulos/articulo/?num=505>

# ANEXOS

## Anexo 1: Forma organizativa del Directorio Activo

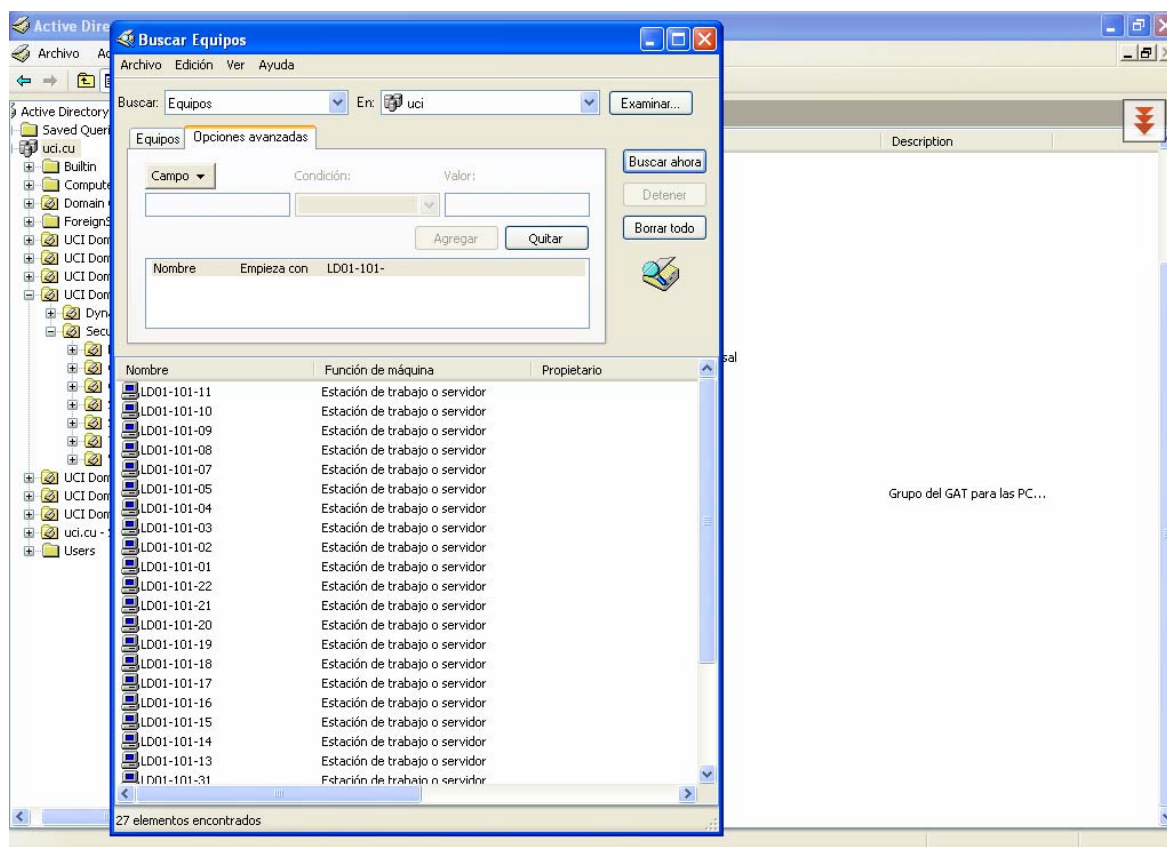


## Anexo 2: Grupos en el Directorio Activo





### Anexo 3: Política de nombre de computadoras.



### Anexo 4: Estándar de Codificación.

Estándar de Codificación que propone el grupo de desarrollo de la librería PEAR.

- Se emplea el Sangrado y la longitud de línea que debe ser utilizando un guión de 4 espacios, sin pestañas. Esto ayuda a evitar problemas con las diferencias, parches y anotaciones.
- Fue usado `<? Php?>` para delimitar el código PHP
- Las estructuras de control empleadas tienen un espacio entre las palabras clave de control y la apertura de paréntesis, para distinguirlos de las llamadas a funciones.

- Las llamadas a las funciones se realizan omitiendo los espacios entre el nombre de la función, la apertura de paréntesis, y el primer parámetro, comas y espacios entre cada parámetro, y ningún espacio entre el último parámetro, el paréntesis de cierre, y el punto y coma.
- Para los comentarios se utiliza el estilo de C (`/* */`) y el estándar de C++ (`//`) los cuales son mas recomendados.
- Las declaraciones de las clases tienen su llave de apertura en una nueva línea.
- Las Funciones Globales utilizadas comienzan con la letra inicial del nombre (tras el prefijo) en minúscula, y cada letra que comienza una nueva "palabra" es mayúscula.
- Los métodos usan la metodología en forma de "camello" en el cual las palabras claves, su inicio va en mayúscula.
- Las variables globales, sus nombres comienzan con un solo carácter seguido por el nombre del paquete y otro subrayado.
- Las constantes verdaderas, falsas y nulas están escritas con minúsculas.

El diseño de la base de datos se rige por las siguientes reglas:

Los nombres son cortos y descriptivos.

- Los nombres con más de una palabra se separan por "\_".
- Para los campos de las tablas el campo carnet de identidad será nombrado como CI.
- Para identificar el campo persona se tomará como identificador único de las personas en la UCI "IdPersona".
- Codelgniter Active Record- permite métodos en cadena.

**Anexo 5: Pruebas de sistema.**

La integridad de los datos se verificó mediante el uso del programa sniffer *Ethereal*<sup>1</sup>. Fue utilizado para "captar" las tramas de datos que son transmitidos entre la aplicación de escritorio y la aplicación de integración.

<b>Caso de Prueba 1</b>	
<b>Entrada:</b>	Uso de <i>Ethereal</i>
<b>Resultado esperado:</b>	La trama que se observa debe estar encriptada.
<b>Resultado de la prueba:</b>	Mediante el uso del mismo no se pudo captar ni interceptar los datos enviados. La trama enviada está encriptada.
<b>Condiciones:</b>	Tener permisos de administración en la PC.

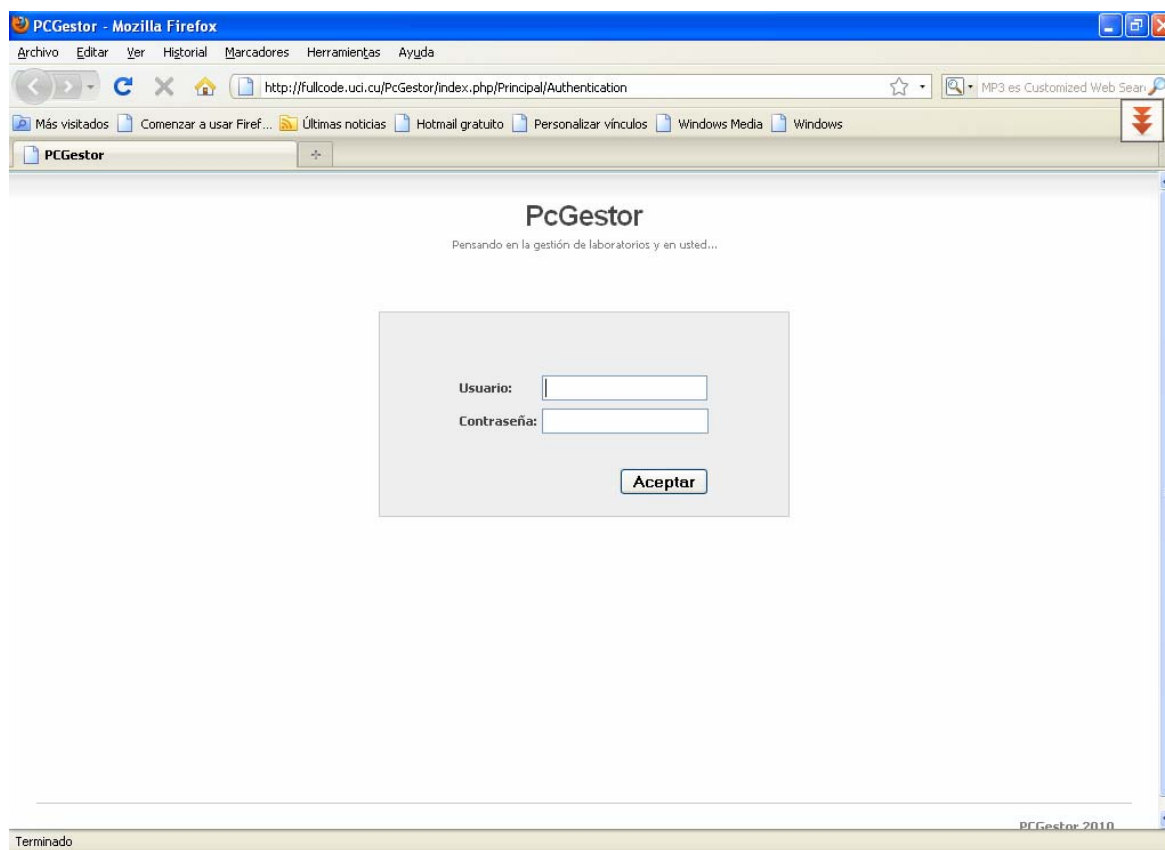
De esta manera, quedó probada la protección contra la alteración o corrupción de los datos.

<b>Caso de Prueba 2</b>	
<b>Entrada:</b>	Inicio de sesión
<b>Resultado esperado:</b>	El sistema verifica la existencia del usuario en la Base de Datos del sistema y su nivel de acceso.
<b>Resultado de la prueba:</b>	No se inicial la aplicación
<b>Condiciones:</b>	Estar insertado y tener los permisos necesarios para la utilización del sistema.

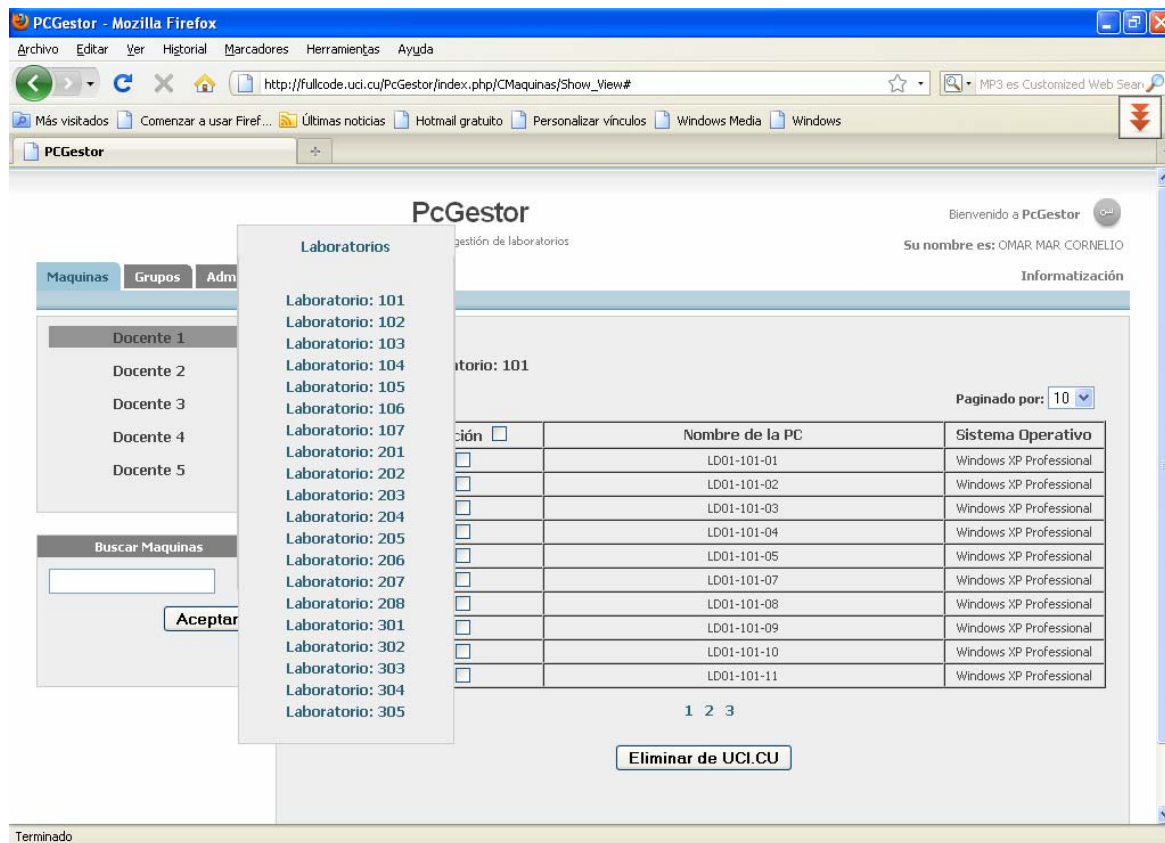
La disponibilidad quedó probada, ya que tienen permiso para acceder al servidor todos aquellos clientes que se encuentren autorizados para trabajar con la aplicación.

**Resultado general de las pruebas de seguridad:** Para cada tipo de usuario conocido y su nivel de acceso las funcionalidades trabajan según lo previsto.

## Anexo 6: Pantalla Autenticar.



Anexo 7: Pantalla Máquina.



## Anexo 8: Pantalla Grupos.

The screenshot shows the PcGestor web application interface. The browser window title is "PcGestor - Mozilla Firefox". The address bar shows the URL: [http://fullcode.uci.cu/PcGestor/index.php/CGrupos/Show\\_View/](http://fullcode.uci.cu/PcGestor/index.php/CGrupos/Show_View/). The page header includes the PcGestor logo, the tagline "Pensando en la gestión de laboratorios", and a welcome message "Bienvenido a PcGestor" with the user name "Su nombre es: OMAR MAR CORNELIO".

The main navigation menu includes "Maquinas", "Grupos", "Administración", and "Reportes". The "Grupos" menu is active, showing a list of groups on the left sidebar:

- Sec - Labs - Admin 1
- Sec - Labs - Admin 2
- Sec - Labs - Admin 3
- Sec - Labs - Admin 4
- Sec - Labs - Admin 5
- Sec - Labs - Computer
- Sec - Labs - Managers

Below the sidebar is a search box labeled "Buscar Usuario en Grupos" with an "Aceptar" button.

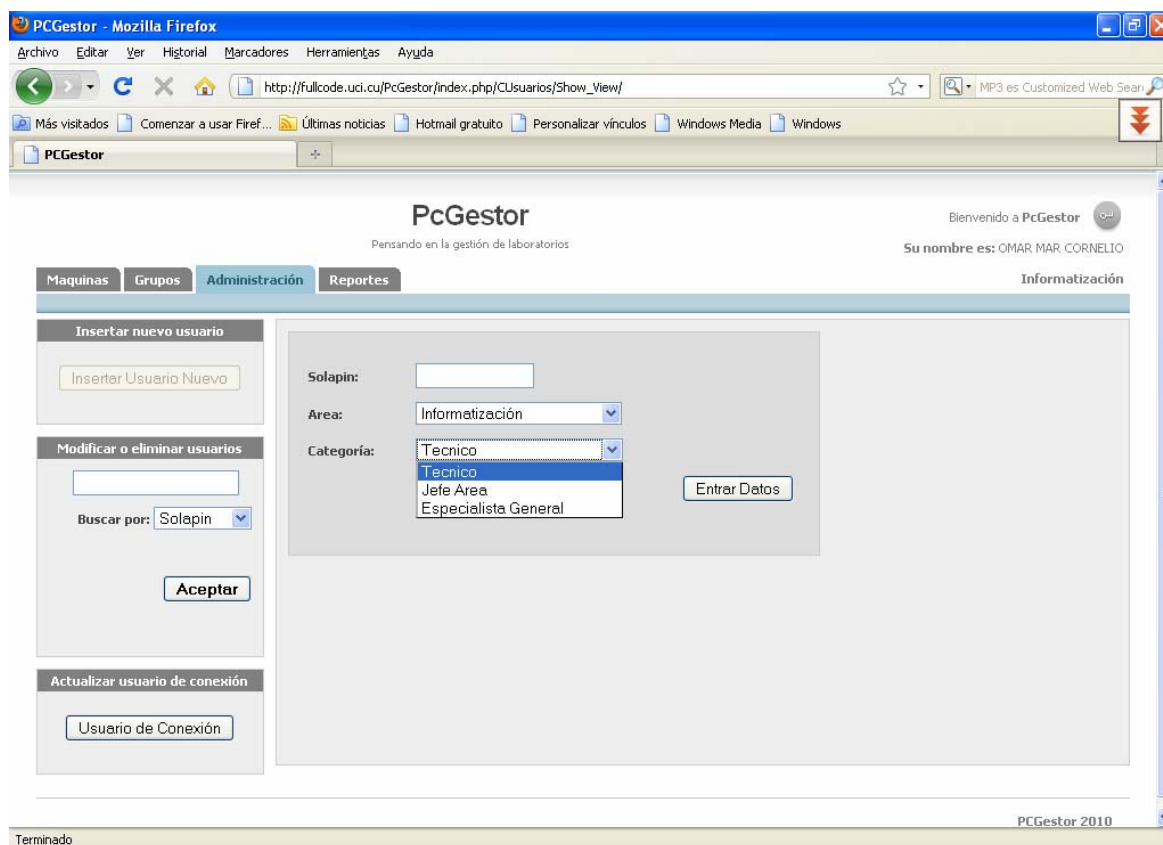
The main content area displays the "Sec - Labs - Admin 1" group. It includes a "Paginado por:" dropdown menu set to "10". Below this is a table of users:

Número	Selección <input type="checkbox"/>	Nombre y Apellidos	Usuario
1	<input type="checkbox"/>	Danally De La Osa Perez	danally
2	<input type="checkbox"/>	Irina Campo Milan	irinac
3	<input type="checkbox"/>	Pedro José Godínez Gómez	pedrojose
4	<input type="checkbox"/>	Linnet Rodriguez Leal	linnet
5	<input type="checkbox"/>	hamed alvelo febles	halvelo
6	<input type="checkbox"/>	Yasmany Gonzalez Díaz	yydiaz
7	<input type="checkbox"/>	Fidel Gonzalez Ramos	fidelgr
8	<input type="checkbox"/>	Enrique Santiago Moreno Mouriz	mouriz
9	<input type="checkbox"/>	Yurien Sosa Pérez	yuriens
10	<input type="checkbox"/>	Jismely Rodriguez Gonzalez	jismely

Below the table are two input fields: "Insertar nuevo usuario:" with an "Insertar" button, and "Eliminar usuarios:" with an "Eliminar" button. A pagination control shows "1 2 3 4 5".

The status bar at the bottom left indicates "Terminado".

## Anexo 8: Pantalla Administración.



## Anexo 9: Pantalla Reporte.

PCGestor - Mozilla Firefox

Archivo Editar Ver Historial Marcadores Herramientas Ayuda

http://fullcode.uci.cu/PCGestor/index.php/CRReporte/Show\_View/

Más visitados Comenzar a usar Firef... Últimas noticias Hotmail gratuito Personalizar vínculos Windows Media Windows

PCGestor Reportes Obtenidos Reportes Obtenidos Reportes Obtenidos

**PcGestor**

Pensando en la gestión de laboratorios

Bienvenido a PcGestor

Su nombre es: OMAR MAR CORNELIO

Maquinas Grupos Administración **Reportes** Informatización

Tipos de Reportes

Reporte #1 "Maquinas"

Todas las **Maquinas Eliminadas** según Area **Docente 1** Laboratorio **-**

Fecha Inicio **2010-02-17** Fecha Final **2010-02-18** Hora Inicio **###** Hora Final **###**

Reporte #2 "Grupos"

Todos los **Usuarios** según Grupo **Sec-Labs-Admin 1** Evento **-**

Fecha Inicio **2010-02-17** Fecha Final **2010-02-18** Hora Inicio **###** Hora Final **###**

Cantidad de Reportes: **2** **Generar Reportes**

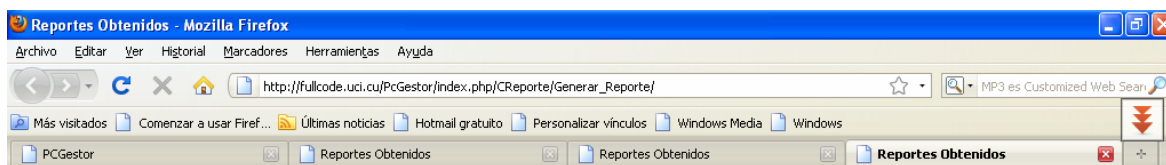
PCGestor 2010

Terminado





## Anexo 10: Resultados de la Búsqueda.



## PcGestor

Pensando en la gestión de laboratorios

## Reporte #1 "Maquinas Eliminadas de UCI.CU"

#	Nombre de la PC	Fecha	Hora	Nombre de la Persona	Ip de la Persona
1	LD01-102-07	2010-02-17	08:46:00	DANIEL FERNANDEZ CAO	10.31.50.20
2	LD01-102-09	2010-02-17	08:46:00	DANIEL FERNANDEZ CAO	10.31.50.20
3	LD01-203-13	2010-02-17	08:46:22	DANIEL FERNANDEZ CAO	10.31.50.20
4	LD01-204-18	2010-02-17	08:46:35	DANIEL FERNANDEZ CAO	10.31.50.20

## Reporte #2 "Acciones de los Usuarios sobre los Grupos"

#	Nombre del Usuario	Fecha	Hora	Nombre de la Persona	Ip de la Persona	Acción Realizada	Grupo
1	yuniorab	2010-02-17	07:55:23	AQUILES PEREZ MIRANDA	10.31.60.114	Eliminación	sec-labs admins 1
2	mayneris	2010-02-17	07:55:23	AQUILES PEREZ MIRANDA	10.31.60.114	Eliminación	sec-labs admins 1

PcGestor 2010

Terminado

### GLOSARIO DE TÉRMINOS

**Active Administrator:** Es una herramienta para el trabajo con el Directorio Activo

**ADLDAP:** Es una clase de PHP que provee autenticación e integración de LDAP con el Directorio Activo.

**ADManager:** Es una herramienta para el trabajo con el Directorio Activo

**Desktop Authority:** Es una herramienta para el trabajo con el Directorio Activo

**DIN:** Dirección de Informatización.

**Directorio Activo:** Es un medio de organizar, administrar y controlar centralizadamente el acceso a los recursos de la red.

**GNU:** Es un acrónimo recursivo que significa GNU No es Unix (GNU is Not Unix). Puesto que en inglés "gnu" (en español "ñu") se pronuncia igual que "new", Richard Stallman recomienda pronunciarlo "guh-noo". En español, se recomienda pronunciarlo ñu como el antílope africano o fonéticamente; por ello, el término mayoritariamente se deletrea (G-N-U).

**GPL:** Es la Licencia pública general de GNU.

**Interfaz:** Es el de mediación, entre hombre y máquina. La interfaz es lo que "media", lo que facilita la comunicación, la interacción, entre dos sistemas de diferente naturaleza, típicamente el ser humano y una máquina como el computador. Esto implica, además, que se trata de un sistema de traducción, ya que los dos "hablan" lenguajes diferentes: verbo-icónico en el caso del hombre y binario en el caso del procesador electrónico.

**Internet:** Es un método de interconexión descentralizada de redes de computadoras implementado en un conjunto de protocolos denominado TCP/IP y garantiza que redes físicas heterogéneas funcionen como una red lógica única, de alcance mundial.

**Linux:** Es la denominación de un sistema operativo tipo Unix (también conocido como GNU/Linux) y el nombre de un núcleo. Es software libre y del desarrollo del código abierto, cuyo código fuente está disponible públicamente, para que cualquier persona pueda libremente usarlo, estudiarlo, redistribuirlo,

comercializarlo y, con los conocimientos informáticos adecuados, modificarlo.

**Microsoft:** Es la compañía que manufactura los sistemas de operación DOS y Windows.

**PC:** Computadoras.

**Procedimiento:** Es la secuencia de acciones concatenadas entre sí, que ordenadas en forma lógica permite cumplir un fin u objetivo predeterminado

**TIC:** Tecnologías de la Informática y las Comunicaciones.

**UCI:** Universidad de Ciencias Informáticas.

**XML:** Son las siglas en inglés de Extensible Markup Language, lenguaje de marcado ampliable o Extensible, desarrollado por el World Wide Web Consortium (W3C). No es más que un conjunto de reglas para definir etiquetas semánticas que nos organizan un documento en diferentes partes. XML es un metalenguaje que define la sintaxis utilizada para definir otros lenguajes de etiquetas estructurados.