

**CONTROL Y REGISTRO DE ENTRADA Y SALIDA DE EQUIPOS Y DE  
USUARIOS EN INSTITUCIONES ACADÉMICAS.**

**Joaquín Leonardo Flores Rodríguez**

**Michael Enrique Moreno Pulido**

**CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS**

**FACULTAD DE INGENIERÍA**

**DEPARTAMENTO DE INFORMÁTICA, REDES Y ELECTRÓNICA PROGRAMA  
DE TECNOLOGÍA EN INFORMÁTICA**

**BOGOTA**

**I-2015**

**Documento A**

**CONTROL Y REGISTRO DE ENTRADA Y SALIDA DE EQUIPOS Y DE  
USUARIOS EN INSTITUCIONES ACADÉMICAS.**

**Joaquín Leonardo Flores Rodríguez**

**Michael Enrique Moreno Pulido**

**Trabajo de grado**

**Para optar al título de**

**Tecnólogo en Informática**

**CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS**

**FACULTAD DE INGENIERÍA**

**DEPARTAMENTO DE INFORMÁTICA, REDES Y ELECTRÓNICA PROGRAMA  
DE TECNOLOGÍA EN INFORMÁTICA**

**BOGOTA**

**I-2015**

**Documento A**

**Nota de aceptación**

---

---

---

---

---

---

---

**Firma del presidente del jurado**

---

**Firma del jurado**

---

**Firma del jurado**

**Bogotá D.C. Enero de 2015**

## DEDICATORIA

### **Michael:**

Este trabajo de tesis va única y exclusivamente con todo mi cariño para esas personas que me apoyaron durante tanto tiempo, a esas personas que nunca dudaron de mí y que me motivaron durante todo este largo trayecto, a ellos mi más sincero agradecimiento. Con todo mi corazón.

- Mi Mama, Beatriz Pulido Quintero
- Mi Papa, Cenen Moreno Runza
- Mi Hermana, Jessica Johana Pulido
- Mi Hermana, Yuri Pulido

### **Joaquín:**

Agradezco a mis padres Joaquin Florez y Sara Rodriguez por brindarme su apoyo y buena energía para el desarrollo y culminación del trabajo de grado a si también a los docentes que me acompañaron a lo largo de mi carrera y que por el conocimiento brindado me permitieron presentar este trabajo.

## AGRADECIMIENTOS

Este trabajo de tesis primeramente nos gustaría agradecer a todas las personas que nos apoyaron durante este proceso educativo.

A Dios por darnos esta oportunidad de vivir y ayudarnos a cumplir una de las metas propuestas.

Agradecer también a la Corporación Universitaria Minuto De Dios por darnos la oportunidad de estudiar y ser unos excelentes profesionales.

A Nuestra familia que con tanto esfuerzo nos apoyó para realizar este sueño y hacerlo realidad.

**A los profesores**, que nos brindaron un poco de su conocimiento durante toda la carrera, en especial a:

- Ing. Sócrates Amador Rojas
- Ing. Luis Eduardo Pérez Peregrino
- Ing. Gonzalo Martin Rodríguez Carrillo
- Ing. Miguel Hernández

Y por último a nuestros jefes de trabajo, Ing. Fernando De La Peña e Ing. Gustavo Cárdenas que nos apoyaron y nos motivaron hasta el último momento de nuestra carrera profesional.

## TABLA DE CONTENIDO

<b>INTRODUCCION</b>	<b>PÁG</b>
1.1 Título del proyecto .....	11
1.2 Planteamiento del Problema .....	12
1.3 Alcances y Justificación .....	13
1.4 Objetivos .....	14
➤ Objetivos Principales.....	14
➤ Objetivos Específicos .....	14
<b>INGENIERIA DEL PROYECTO</b>	
2.1 Metodología y Modelo de desarrollo .....	15
<b>ANALISIS Y DISEÑO</b>	
3.1 Definición de Requerimientos .....	19
➤ Requerimientos Funcionales.....	19
➤ Requerimientos No Funcionales .....	20
3.2 Descripción del sistema propuesto.....	21
➤ Objetivos Principales.....	22
3.3 Diseño del sistema propuesto– .....	25
➤ Diagramas Estructurales .....	25
➤ Diagramas Dinámicos .....	26
<b>DESARROLLO</b>	
4.1 Especificaciones Técnicas .....	49
➤ Software .....	49
➤ Hardware.....	49

5. GLOSARIO.....	50
6. CONCLUSIONES.....	51
7. BIBLIOGRAFIA.....	52
8. MANUALES	
8.1 MANUAL DEL SISTEMA	
8.2 MANUAL DEL USUARIO	

### LISTA DE GRÁFICOS ESPECIALES

	Pág.
<b>Gráfica No 1. Metodología de desarrollo XP.....</b>	<b>16</b>
<b>Gráfica No 2. Modelo Vista Controlador.....</b>	<b>21</b>
<b>Gráfica No 3. Login Usuario .....</b>	<b>22</b>
<b>Gráfica No 4. Interfaz administrador.....</b>	<b>23</b>
<b>Gráfica No 5. Interfaz Encargado .....</b>	<b>24</b>
<b>Gráfica No 6. Diagrama de clases.....</b>	<b>26</b>
<b>Gráfica No 7. Diagrama de paquetes .....</b>	<b>27</b>
<b>Gráfica No 8. . Diagrama Casos de uso Administrador .....</b>	<b>37</b>
<b>Gráfica No 9. Diagrama Casos de uso del Encargado .....</b>	<b>44</b>
<b>Gráfica No 10. Diagrama Casos de uso del Sistema.....</b>	<b>48</b>

## RESUMEN

Últimamente se ha venido dando gran importancia al uso de las nuevas tecnologías en los lugares de aprendizaje han habilitado las salas de cómputo en los establecimientos educativos con el fin de brindar las herramientas necesarias para el aprendizaje autónomo y de calidad del estudiante. Teniendo en cuenta esto, se presentara un software que permite gestionar un control sobre las diferentes aulas que las universidad minuto de Dios brinda para el uso de los estudiantes, así como también controlar aulas de cómputo que están destinadas para la enseñanza. Con este software se pretende controlar y gestionar el uso de los equipos por parte de los estudiantes, como también generar reportes para los administradores de las salas los cuales les permitirá tomar decisiones acerca de los equipos y su disponibilidad. Al desarrollar un aplicativo de software el proceso de desarrollo consta de diferentes partes donde se analiza desde los requerimientos necesarios para que el software realice las tareas o procesos esperados así como se pueda adaptar a un desarrollo de software previo, también se analizan los requerimientos de software y hardware que se requieren para la ejecución del mismo. Así que para este software se ha hecho un levantamiento de información para generar los diferentes diagramas con los que nos apoyaremos para el desarrollo así como también, una investigación sobre la metodología ágil XP que es la que se aplicó al proyecto y el modelo de desarrollo MVC.



## ABSTRACT

Lately, it has been attaching great importance in the use of new technologies in the places of, enabled romos of computing in the educational institutions for provide the tolos necessary for independent learning and quality for student. Given this, wants present a software for managing control over different classrooms in the educational institution Minuto de Dios provide for the use of students, designed for teaching. This software provide to cotrol and manage the use of computers by students, as well generate reports for managers of the rooms and make decisions about of equipment and availability.

Developing this application of software consists of different parts which is analyzed the requirements for the software perform tasks or processes for can adapt, also. Software requirements andhardware required for the implementation of the same are also analyzed.

So, for this software, with survey of information , generate the different diagrams with which we rely for development for a well investigation into XP (Extreming Programming). A metodology agile and the development model MVC.

## INTRODUCCION

Este trabajo tiene como objetivo, realizar un software para el control y registro de alumnos y equipos en las diferentes salas de cómputo en distintas instalaciones de educación (Colegios, Universidades, entre otros), para el préstamo de estos, además de llevar un mejor control sobre el uso de dichas salas y/o equipos.

La metodología de desarrollo que utilizaremos será XP (*eXtreme Programming*), ya que se esta metodología se diferencia de las otras principalmente por su énfasis en la adaptabilidad que en la previsibilidad y ya que es una metodología orientada a objetos se utilizara el patrón de diseño MVC (Modelo, Vista, Controlador).

Para el desarrollo de este proyecto se van a utilizar los diagramas correspondientes a UML que nos van a ayudar a estructurar los componentes del sistema así como su comportamiento e interacción.

La razón primordial por la cual se escogió este desarrollo, es porque en los diferentes lugares como las instituciones académicas que poseen aulas y/o equipos de cómputo para el préstamo de estos, en la gran mayoría el control que se tiene sobre ingresos de los estudiantes y de equipos es muy poco o nulo dando cabida a inconvenientes como la perdida de componentes de los equipos, perdida de objetos olvidados como son las memorias flash, así como CD y/o USB.

**Control y Registro de entrada y salida de equipos y de usuarios en instituciones académicas.**

## 1.2 PLANTEAMIENTO DEL PROBLEMA

Así como en varias universidades piensan en la comodidad de los usuarios, otorgándoles facilidad de acceso a la información y a la obtención de conocimiento para servir como herramienta de desarrollo para la formación académica de los estudiantes, cada institución educativa otorga aulas o salas de computo, las cuales esten disponibles para que los estudiantes puedan hacer uso de ellas, sin embargo, el uso y el cuidado que se les da a los equipos en estas salas en algunas ocasiones no es el necesario o no es el adecuado.

También en algunas instituciones no académicas se brinda la posibilidad a los empleados de acceder a estas salas y el uso que se les da no es el adecuado por parte de estos ya sea utilizándolas para descargar aplicativos o haciendo uso de aplicaciones y/o paginas prohibidas por parte de la empresa o la institución la cual se presta el servicio.

En algunos casos las organizaciones tienen 2 o hasta 3 aplicativos diferentes para el control de sus equipos haciendo así un trabajo más complejo para los administradores o encargados de dichas salas.

Haciendo una revisión de los problemas más frecuentes en algunas salas de cómputo de diferentes universidades de Bogotá encontramos los siguientes problemas:

- Olvido de memorias USB y CD's por parte del usuario.
- Robo de componentes de hardware como teclados o mouse.
- Utilización de software no permitido.

- El control de los equipos por parte de los encargados de las aulas es muy tedioso (gran número de equipos)

### **1.3 ALCANCE Y JUSTIFICACIÓN**

Los alcances que se desarrollaran con esta aplicación serán, la creación y administración de encargados de sala así como la asignación de salas por encargado, también la administración de las diferentes salas así como los equipos en ella, permitiendo hacer un inventario de sala así como de equipos.

Por otra parte el sistema va a realizar funciones simples como son crear usuarios, borrarlos, modificarlos, consultar la información de cada sala de cómputo, esto con el fin de que el administrador principal pueda tener información de cada una de las diferentes salas. Además de esto el software permitirá a los docentes actualizar su información personal así como llevar una agenda de sus actividades pendientes.

También el software será capaz de realizar múltiples tareas desde el bloqueo del equipo, bloqueo de programas e instalaciones de aplicativos en red.

Todo esto con una interfaz sencilla y completa que permitirá la recreación de cómo están los equipos en las aulas y en el momento de su uso cuáles son sus estados actuales y por qué usuarios están siendo usados en el momento.

## 1.4 OBJETIVOS

Los siguientes objetivos planteados a continuación nos darán a conocer las soluciones de la problemática que este proyecto desea resolver.

### ➤ **Objetivo General.**

Desarrollar una aplicación la cual permita gestionar salas de cómputo y usuarios que hacen uso de estas mismas, permitiendo un mayor control y una mejor administración sobre cada una de estas aulas.

### **Objetivos Específicos.**

Los objetivos específicos del proyecto, son los siguientes:

- 1) Gestionar la disposición de los equipos por sala así como las personas que estarán encargadas de estas mismas.
- 2) Generar reportes acerca del uso de los equipos y los usuarios registrados en el sistema.
- 3) Administrar y registrar los usuarios que harán uso de estas salas así como el equipo que usan y el tiempo que duraran.
- 4) Asignación de roles para el acceso al sistema y sus funcionalidades.

## 2. INGENIERIA DEL PROYECTO

En la ingeniería del proyecto se dará una descripción sobre el diseño del software, donde abarcará los datos, interfaces, arquitecturas y sus componentes a nivel de diseño.

### 2.1 Modelo De Desarrollo

El modelo de desarrollo que se utilizara para este proyecto será el método de programación extrema (Xp) ya que esta se fija en tres capas que agrupan las doce prácticas básicas de la programación extrema.

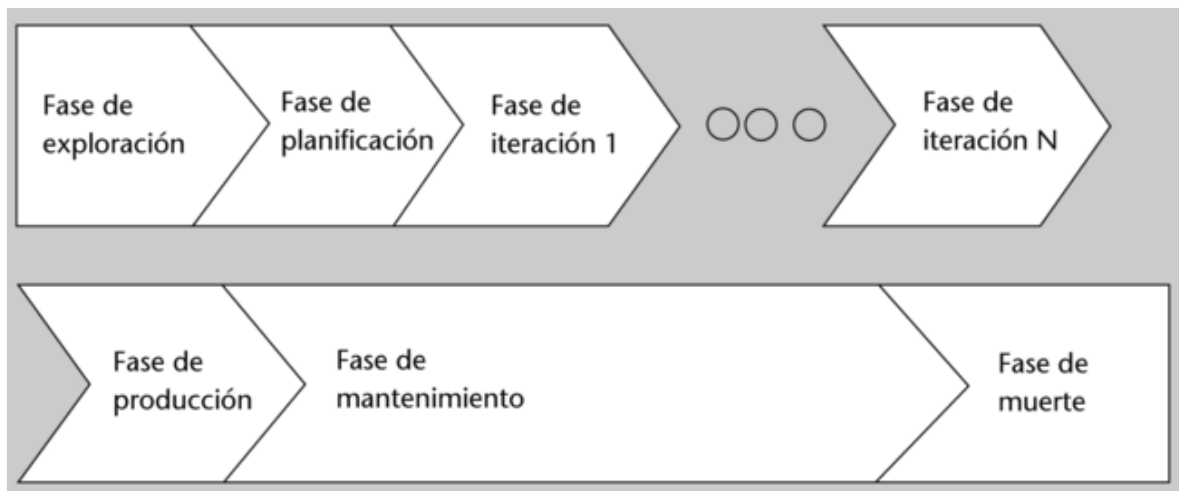
- 1) Metodología de programación: diseño sencillo, test, refactorización y codificación con estándares.
- 2) Metodología de equipo: propiedad colectiva del código, programación en parejas, integración continua, cuarenta horas semanales y metáfora del negocio.
- 3) Metodología de procesos: cliente in situ, entregas frecuentes y planificación del juego.

El punto de partida de la metodología XP son las variables que utiliza para cada proyecto: coste (la inversión económica y en recursos), tiempo (el tiempo empleado, determinado por la fecha de entrega final), calidad (del código y del aplicativo desarrollado) y alcance (Conjunto de funcionalidades).

De estas cuatro variables, sólo tres podrán ser fijadas por el cliente y/o por el jefe de proyectos, la cuarta es responsabilidad del equipo de Desarrollo y se establecerá en función de las otras tres.

A continuación se presentara el ciclo de vida de la programación extrema.

- 1) Fase de exploración
- 2) Fase de planificación
- 3) Fase de iteraciones
- 4) Fase de producción
- 5) Fase de mantenimiento
- 6) Fase de muerte del proyecto



**Gráfica No 1: Metodología de desarrollo XP.**

### **Fase de exploración**

La fase de exploración es la primera fase del ciclo de vida de la metodología

XP, en la cual se desarrollan los siguientes procesos:

- 1) Las historias de usuario.
- 2) El spike arquitectónico.
- 3) La metáfora del negocio.



Todo comienza con las "historias de usuario". En esta fase los usuarios plantean a grandes rasgos las funcionalidades que desean obtener del aplicativo.

Las historias de usuario tienen el mismo propósito que los casos de uso, salvo en un punto crucial; las escriben los usuarios y no el analista. Han de ser descripciones cortas y escritas en el lenguaje del usuario sin terminología técnica.

Estas garantizan que dichas historias se han comprendido y se han implementado correctamente

### **Fase de planificación**

El resultado ha de ser una planificación (recordemos que siempre flexible) del proyecto.

El procedimiento es el siguiente:

- El cliente entrega al equipo de desarrollo las historias del usuario que ha confeccionado, pero priorizándolas de mayor a menor importancia.
- El equipo de desarrollo las estudia y estima el costo de implementación.
- Una vez se tiene la lista de historias priorizadas junto con su costo de implementación, se procede a convocar la reunión del plan de entregas.

### **Fase de iteraciones**

Como se ha dividido el proyecto en iteraciones, esta fase se repetirá tantas veces como iteraciones sean necesarias. Generalmente, cada iteración suele ser de dos a tres semanas.

El Plan de iteración se desarrolla de la siguiente manera:

- Se recogen las historias de usuario asignadas a esta iteración.
- Se detallan las tareas a realizar por cada historia de usuario.
- Se priorizan las tareas que más valor darán al negocio, intentando que se finalicen historias de usuario lo antes posible.
- Se reparten las primeras tareas al equipo de desarrollo y el resto se deja en una cola de tareas sin asignar de dónde se irán asignando a medida de que el proyecto avance.
- Se convocan reuniones de seguimiento diarias para diagnosticar el retrasando en las estimaciones o el adelantando a ellas y así poder desechar o incorporar historias de usuario.

### **Fase de producción**

Llegamos a esta fase al alcanzar la primera versión que el usuario final decida que puede ponerse en producción.

En la etapa de producción se realizan también iteraciones como en la anterior etapa, pero el ritmo de éstas ya no es de dos a tres semanas, sino mensuales.

Esta fase se mantiene hasta que se realice la última entrega, con la que finalizamos el ámbito del aplicativo y pasamos al mantenimiento del mismo.

Durante la fase de producción, el ritmo de desarrollo decae debido a que el equipo debe solventar las incidencias de los usuarios. Es por esto por lo que a veces es necesario incorporar nuevo personal al equipo.

### **3. ANALISIS Y DISEÑO**

El análisis y diseño en el desarrollo de un sistema de software consiste en un proceso en el cual se define la estructura del aplicativo y su forma de desarrollo. Según el análisis y diseño los aspectos a tratar son los siguientes:

#### **3.1 DEFINICION DE REQUERIMIENTOS**

A continuación se enumeran los requerimientos funcionales y no funcionales que tendrá el software.

##### **➤ Requerimientos Funcionales:**

Los requerimientos funcionales de un sistema describen lo que el sistema debe hacer. Estos requerimientos dependen del tipo de software que se desarrolle, de los posibles usuarios del software y del enfoque general tomado por la organización al redactar requerimientos. A continuación se fijaron los siguientes requerimientos funcionales:

- Registrar encargado sala
- Habilitar/deshabilitar encargado sala
- Registrar docente
- Registrar alumno
- Actualizar docente
- Actualizar alumno
- Consultar usuario “Docente, alumno”

- Listar equipos
- Bloquear equipo
- Desbloquear equipo
- Bloquear programas
- Ver pantalla remota
- Generar un informe del uso de los equipos
- Generar informes de los equipos

➤ **Requerimientos No Funcionales:**

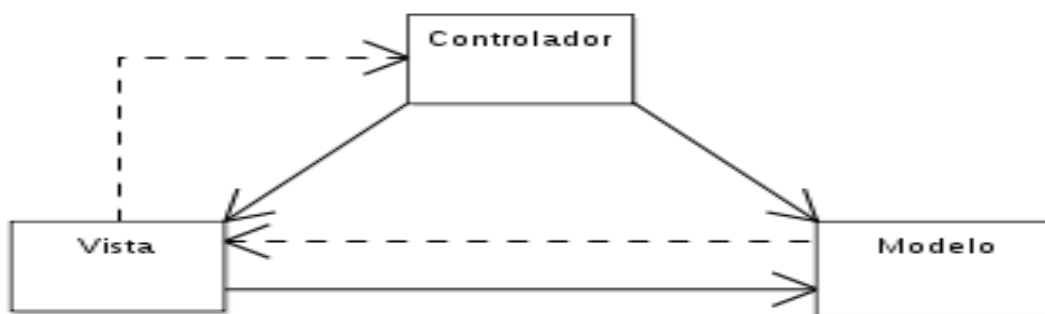
Los requerimientos no funcionales, como su nombre sugiere, son aquellos requerimientos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de éste como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento. A continuación se fijaron los siguientes requerimientos no funcionales:

- Comunicación con base de datos de la institución.
- Detectar componentes del pc
- Envió pantalla remota
- Recibir comandos del servidor
- Gestión de privilegios
- Gestión y control de privilegios según se hayan registrado en el sistema.
- Comunicación con entre el aplicativo cliente y servidor.

- Registro del tiempo de uso de los equipos.
- Registro de usuario por equipo.
- Bloquear programas no permitidos

### 3.2 DESCRIPCION DEL SISTEMA PROPUESTO

El patrón de diseño que se va a aplicar para este proyecto es el de modelo, vista, controlador (MVC), ya que este sistema de arquitectura de software permite estructurar y separar los diferentes componentes de la aplicación en distintas capas según su funcionalidad dentro de la aplicación “Objetos, Vistas, Control”. Este patrón propone 3 componentes de vital importancia los cuales son, Modelo (Estructura de los datos), Vista (Interfaz de usuario) y Controlador (lógica del control), es decir, tanto en el modelo como en el control se definen los componentes para la representación de la información y en la vista se encarga de la interacción con el usuario (En el grafico No 2. Modelo Vista Controlador). El desarrollo del MVC se trabajara con los siguientes lenguajes de desarrollo, los cuales son Visual Basic .NET y SQL SERVER 2008.



**Gráfica No 2: Modelo Vista Controlador**

**Nota:** las líneas solidas indican una asociación directa, y las punteadas una indirecta.

A continuación explicaremos cada patrón para tener una mejor idea de cómo va hacer el funcionamiento de la aplicación a desarrollar:

**Modelo:** Siendo este módulo la representación de la información con la cual el sistema opera, se incluirá la bases de datos con su respectiva conexión, así como también su procesamiento, validación, asociación y cualquier otra tarea respectiva.

Toda esta información se envía a la “Vista” para ser mostrada al usuario.

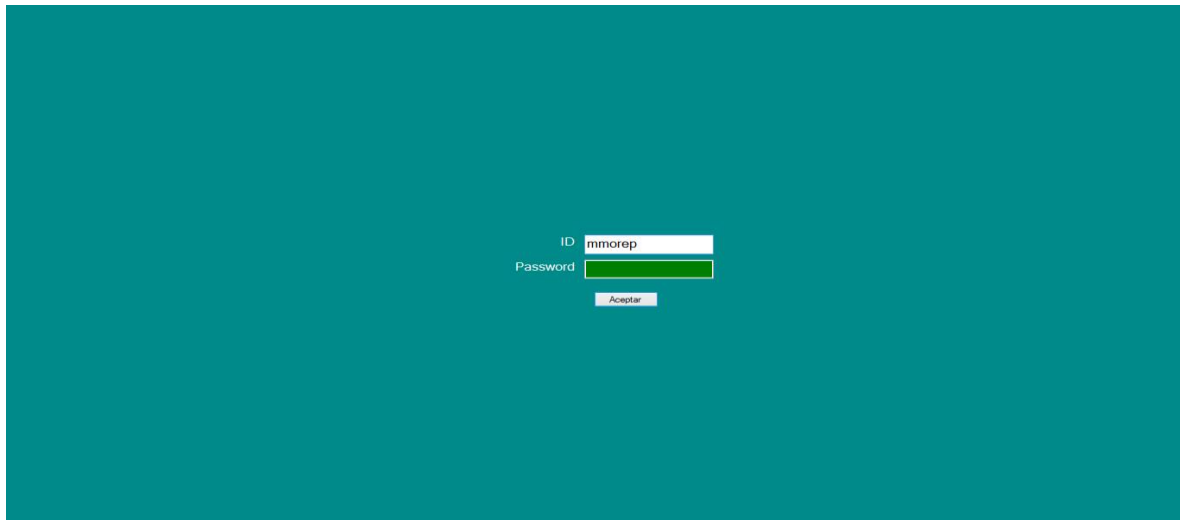
El acceso o manipulación a dicha información llegan al “modelo” desde el “controlador” del sistema.

**Vista:** En este módulo se incluirá la parte visual del sistema para que el usuario pueda interactuar con la información que se muestra proveniente del “Modelo”.

**Controlador:** El controlador es el que administra la información gestionando las peticiones del usuario y así mostrando la información requerida ayudada de la “Vista” y el “Modelo”. Este controlador siempre va a estar a la espera de alguna petición por parte del usuario, una vez procesada esta petición se envía los datos a la capa modelo para que esta se muestre en la capa vista.

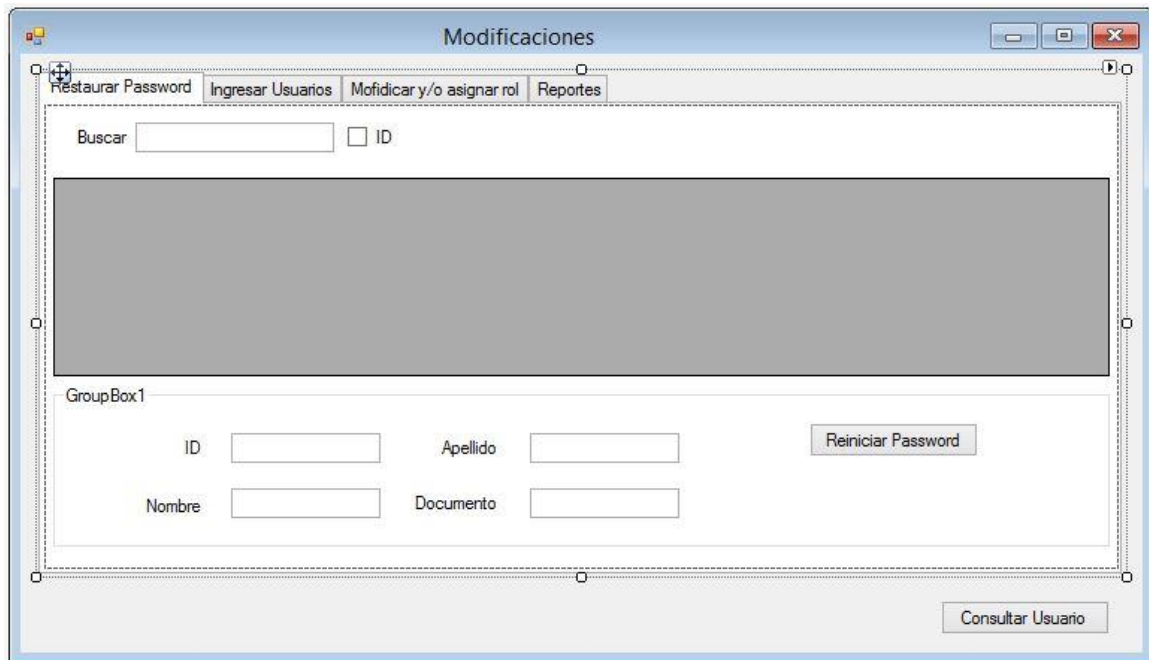
Para llevar a cabo nuestro objetivo que es realizar un software que controle y registre la entrada de equipos y de usuarios a instalaciones académicas se toman en cuenta los módulos vistos anteriormente. Este proyecto tiene como finalidad proporcionar un software que el facilite el trabajo tanto al administrador de salas de computo como al encargado, pudiendo brindar un excelente servicio a todos nuestros usuarios y obteniendo un mejor control de cualquier sala de computo.

Este software orientado principalmente al control de salas de computo, busca obtener una presentación sencilla pero a la vez llamativa en donde se podrá acceder al Login inicial para poder usar un equipo de cualquier sala de cómputo.

A screenshot of a user login interface. The background is a solid teal color. In the center, there is a white rectangular area containing the login form. The form has two input fields: the first is labeled 'ID' and contains the text 'mmorep'; the second is labeled 'Password' and is filled with a solid green color. Below the password field is a small button labeled 'Aceptar'.

### **Gráfica No 3: Login usuario**

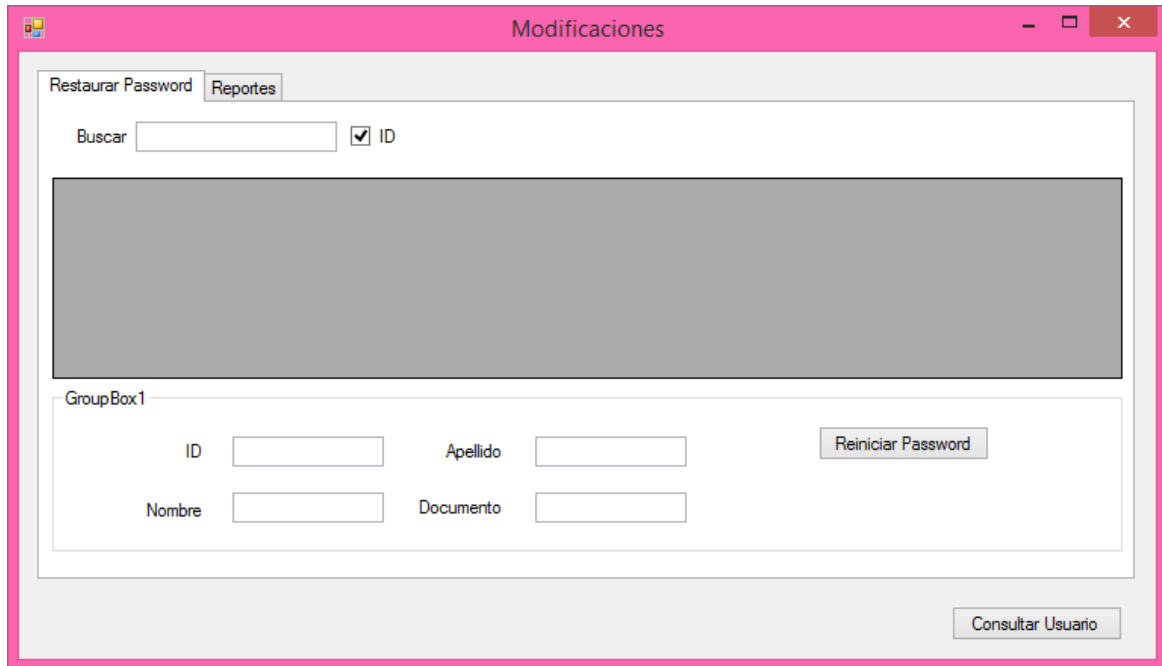
Como se puede observar en la gráfica No 3. (Login usuario), el usuario deberá ingresar su número de ID y su contraseña registrada en la base de datos de la aplicación, para poder hacer uso del equipo. Una vez ingresado el ID y contraseña el sistema le permitirá hacer uso del equipo. En caso de que el usuario olvide su ID o su contraseña, el encargado de la sala podrá consultar información del estudiante por medio de su número de documento en la base de datos y así poder reiniciar la contraseña.



**Gráfica No 4: Interfaz Administrador.**

Como se observa en la gráfica No 4. (Interfaz Administrador), al ingresar al sistema, este reconocerá si es administrador o un encargado, después el usuario podrá crear informes, ver información de los usuarios en las cuales también puede crear a los encargados de sala, modificación de privilegios y podrá configurar las salas dependiendo de su nivel de privilegios.





**Gráfica No 5: Interfaz Encargado.**

En la gráfica anterior se observa la interfaz del encargado, el cual en ella podrá consultar usuarios, configurar algunas cosas (dependiendo de los privilegios que el administrador le haya otorgado) de las salas respectivas y poder controlar las salas de cómputo.

### 3.3 Diseño del sistema propuesto

El diseño de la aplicación va a estar soportado bajo el lenguaje unificado de modelado (UML), ya que en este lenguaje encontramos la posibilidad de visualizar, especificar, construir y documentar un sistema.

En este lenguaje de modelado se catalogan dos tipos de diagramas los cuales son:

- **Diagramas estructurales:** Son los que se encargan de definir la estructura de un sistema dando a conocer clases, objetos, entidades, entre otras.
- **Diagramas dinámicos:** Son los que se encargan de mostrar el comportamiento del sistema.

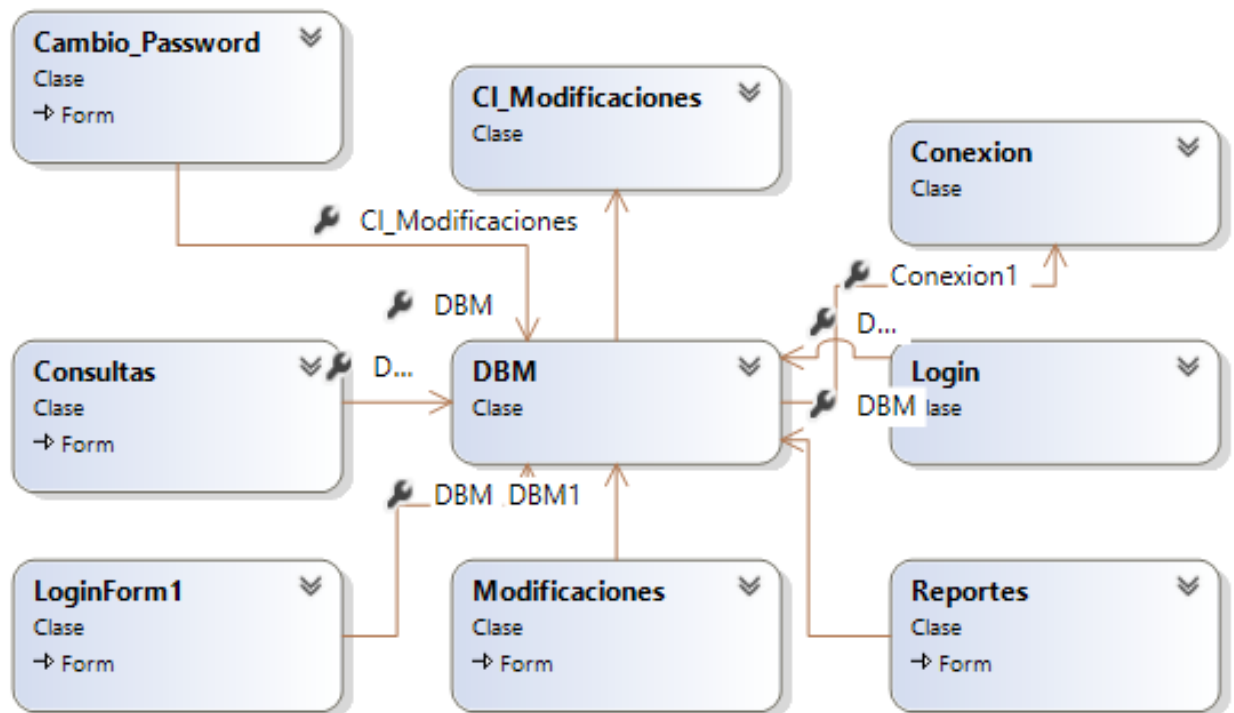
#### ➤ **Diagramas Estructurales:**

Para la definición de la estructura del software se ha optado por la realización de los siguientes diagramas estructurales:

- **Diagrama de clases:** Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. (Ver Gráfica No 5).

A continuación se presentara el diagrama de clases el cual visualizara como va a estar conformado el sistema dando a conocer las clases, atributos y/o métodos que se incluyen dentro del proyecto.

## Diagrama de clases

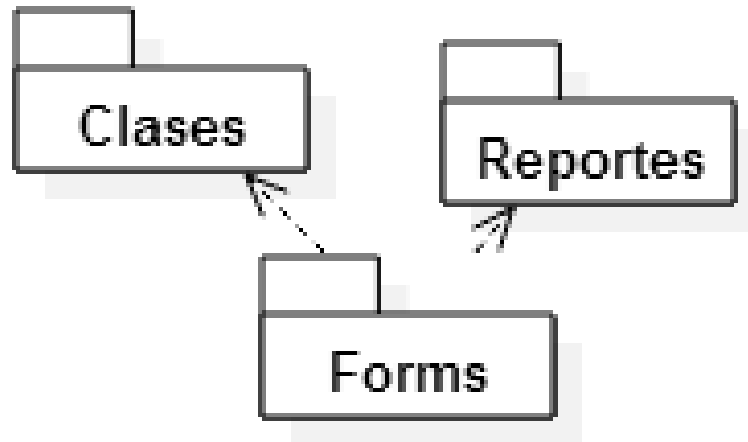


**Grafico No 6. Diagrama de clases**

En la gráfica No. 6 (Diagrama de clases) se puede observar que tenemos 10 clases las cuales cada una depende de la otra haciendo cualquier clase de vital importancia para cualquiera. Se puede observar que la clase servidor siempre va a estar en contacto con su Control manteniendo una comunicación frecuente con la base de datos, la cual nos va a suministrar la información que necesitamos en cualquier momento de diferentes usuarios.

- **Diagrama de paquetes:** un diagrama de paquetes muestra cómo un sistema está dividido en agrupaciones lógicas mostrando las dependencias entre esas agrupaciones. Dado que normalmente un paquete está pensado como un directorio, los diagramas de paquetes suministran una descomposición de la jerarquía lógica de un sistema.

## Diagrama de paquetes



**Gráfica No. 7. Diagrama de paquetes**

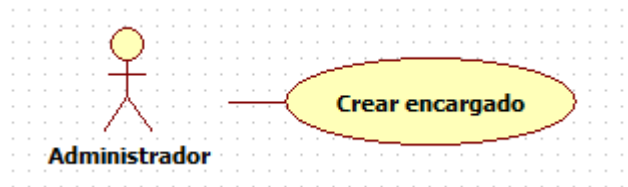
En la gráfica No. 7 (Diagrama de paquetes) Se reflejar la organización de los paquetes mostrando las dependencias de cada una de ellas.

### ➤ Diagramas Dinámicos

Para la definición del comportamiento del software se ha optado por la realización de los siguientes diagramas Dinámicos:

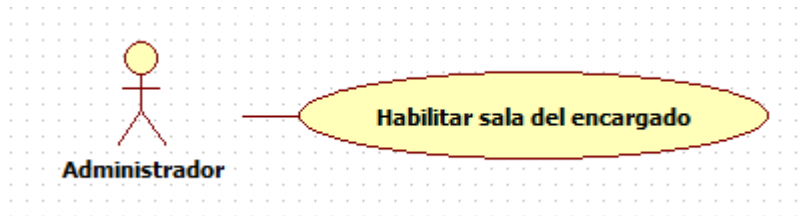
- **Diagrama de casos de uso:** un diagrama de casos de uso es una especie de diagrama de comportamiento UML mejorado. El Lenguaje de Modelado Unificado (UML), define una notación gráfica para representar casos de uso llamada modelo de casos de uso.

A continuación representaremos los casos de uso de cada uno de los roles del sistema, con el fin de poder ilustrar de una mejor forma la funcionalidad de cada uno de los usuarios.



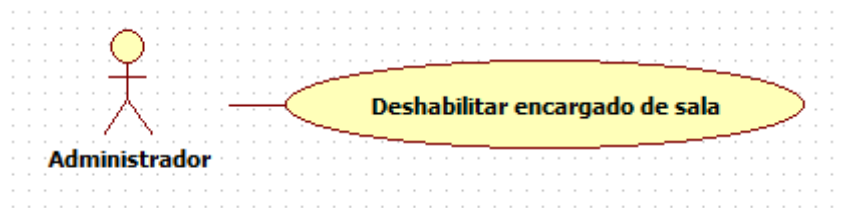
Nombre:	Crear encargado de sala	
Actor:	Administrador	
Descripción:	Crea un usuario con capacidades de administrar salas y docentes y alumnos	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresar Login	1. Valida
	2. Ingresar a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona crear encargado sala	3. Carga formulario de ingreso de datos
	4. Ingresar datos de la persona	4. Registra datos
		5. Registra usuario como encargado sala
Alternativa:	Ingresar Login	Valida
	1. Ingresar a modulo administrativo	1. Carga modulo administrativo
	2. Selecciona crear encargado sala	2. Carga formulario de ingreso de datos
	3. Selecciona un usuario registrado	3. Muestra datos usuario seleccionado
	4. Asigna usuario como encargado	4. Registra usuario como encargado sala
Precondición:	Estar en la base de datos de la institución y/o empresa	
Poscondición:		
Presunción:	El Usuario queda registrado como encargado sala	

### Caso de uso No. 1: Administrador (Crear encargado de salas)



Nombre:	Habilitar encargado de sala	
Actor:	Administrador	
Descripción:	Activa a un encargado de sala en el sistema	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona lista de encargados	3. Carga lista de encargados
	4. Selecciona encargado	4. Muestra información de encargado
	5. Habilita al encargado	5. Cambia el estado del encargado a activo
Alternativa:	Ingresar LogIn	Valida
	1. Ingresar a modulo administrativo	1. Carga modulo administrativo
	2. Selecciona lista de encargados	2. Carga lista de encargados
	3. Busca encargado por nombre o id	3. Muestra coincidencias de búsqueda
	4. Selecciona encargado	4. Muestra información de encargado
	5. Habilita al encargado	5. Cambia el estado del encargado activo
Precondición:	La persona ya está registrada y está asignada como encargado	
Presunción:	El encargado está habilitado	

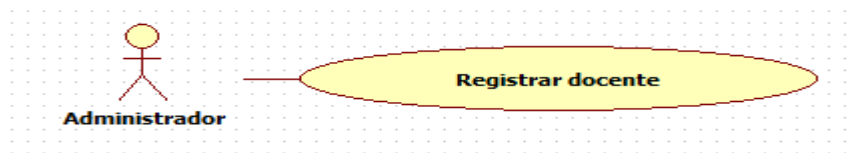
**Caso de uso No. 2: Administrador (Habilitar encargado de sala)**



Nombre:	Deshabilitar encargado sala	
Actor:	Administrador	
Descripción:	Desactiva a un encargado de sala en el sistema	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingres a LogIn	1. Valida
	2. Ingres a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona lista de encargados	3. Carga lista de encargados
	4. Selecciona encargado	4. Muestra información de encargado
	5. Deshabilita al encargado	5. Cambia el estado del encargado a inactivo
Alternativa:	1. Ingres a LogIn	1. Valida
	2. Ingres a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona lista de encargados	3. Carga lista de encargados
	4. Busca encargado por nombre o id	4. Muestra coincidencias de búsqueda
	5. Selecciona encargado	5. Muestra información de encargado
	6. Deshabilita al encargado	6. Cambia el estado del encargado a inactivo
Precondición:	La persona ya está registrada y está asignada como encargado	
Presunción:	El encargado esta deshabilitado	

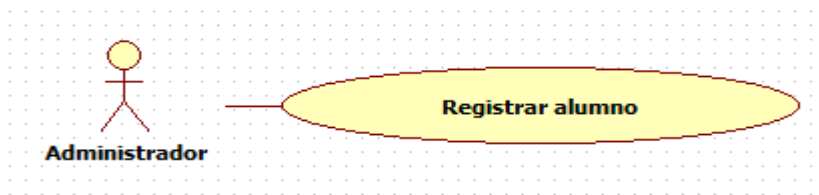
### Caso de uso No. 3: Administrador (Deshabilitar encargado de salas)





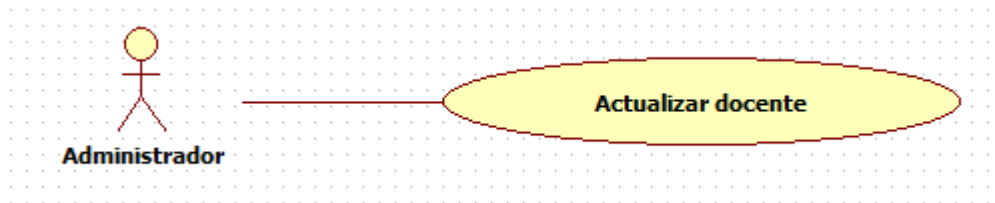
Nombre:	Registrar docente	
Actor:	Administrador	
Descripción:	Registra un docente en la base de datos	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona registrar usuario	3. Carga formulario de ingreso de datos
	4. Ingresar datos de usuario	
	5. Le asigna el rol de docente al usuario	4. Registra al usuario como docente
Alternativa:	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona registrar usuario	3. Carga formulario de ingreso de datos
	4. Busca usuario por nombre o id	4. Muestra coincidencias de búsqueda
	5. Selecciona usuario	5. Muestra información de usuario
	6. Le asigna el rol de docente al usuario	6. Cambia el rol del usuario por el de docente
Precondición:		
Presunción:	Se agrega un nuevo docente a la base de datos	

#### **Caso de uso No. 4: Administrador (Registrar docente)**



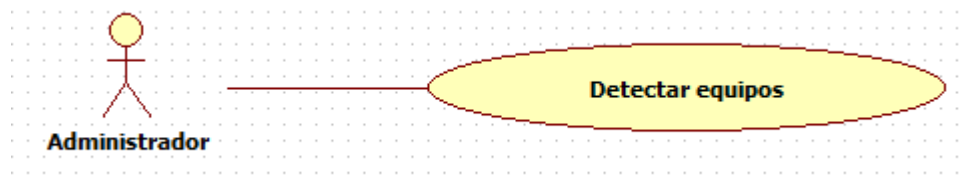
Nombre:	Registrar alumno	
Actor:	Administrador	
Descripción:	Registra un alumno en la base de datos	
Flujo Principal:	1. Evento Actor	1. Evento Sistema
	2. Ingresa LogIn	2. Valida
	3. Ingresa a modulo administrativo	3. Carga modulo administrativo
	4. Selecciona registrar usuario	4. Carga formulario de ingreso de datos
	5. Ingresa datos de usuario	
	6. Le asigna el rol de alumno al usuario	6. Registra al usuario como alumno
Alternativa:	1. Ingresa LogIn	1. Valida
	2. Ingresa a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona registrar usuario	3. Carga formulario de ingreso de datos
	4. Busca usuario por nombre o id	4. Muestra coincidencias de búsqueda
	5. Selecciona usuario	5. Muestra información de usuario
	6. Le asigna el rol de alumno al usuario	6. Cambia el rol del docente por el de alumno
Precondición:		
Presunción:	Se agrega un nuevo docente a la base de datos	

### Caso de uso No. 5: Administrador (Registrar Alumno)



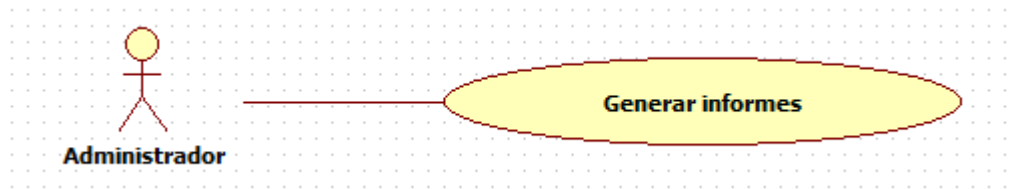
Nombre:	Actualizar docente	
Actor:	Administrador, Encargado Sala	
Descripción:	Actualiza a un docente en la base de datos	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona registrar usuario	3. Carga formulario de ingreso de datos
	4. Busca docente por nombre o id	4. Muestra coincidencias de búsqueda
	5. Selecciona usuario	5. Muestra información de usuario
	6. Actualiza o agrega información	6. Guarda Cambios
Alternativa:	Evento Actor	Evento Sistema
	1. Ingresar Login	1. Valida
	2. Ingresar a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona botón usuario	3. Carga formulario de ingreso de datos
	4. Ingresar ID o CC	4. Muestra información
	5. Actualizar información	5. Guardar cambios
Precondición:	Debe pertenecer a la bases de datos.	
Presunción:	Se actualizo información del docente	

### Caso de uso No. 6: Administrador (Actualizar Docente)



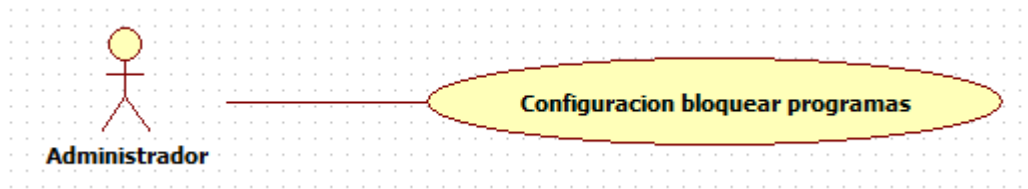
Nombre:	Detectar Equipos	
Actor:	Administrados	
Descripción:	Detecta los equipos que estén conectado en red	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresa LogIn	1. Valida
	2. Ingresa a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona configuración salas	3. Carga configuración salas
	4. Selecciona detectar equipos	4. Muestra equipos encontrados en red
	5. Configura equipos	
Alternativa:	1. Ingresa LogIn	1. Valida
	2. Ingresa a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona configuración salas	3. Carga configuración salas
	4. Selecciona detectar equipos	4. Muestra equipos encontrados en red
	5. Guarda equipos	
Precondición:	Debe pertenecer a la misma red.	
Poscondición:		
Presunción:	Muestra los equipos detectados por el sistema.	

### Caso de uso No. 7: Administrador (Detectar equipos)



Nombre:	Generar Reportes	
Actor:	Administrador	
Descripción:	Crea y visualiza los diferentes reportes del sistema.	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona reportes	3. Carga lista de reportes disponibles
	4. Selecciona reporte a generar	4. Genera reporte
	5. Visualiza/imprime reporte	
Alternativa:	Evento Actor	Evento sistema
	1. Ingresar al módulo administrativo	1. Valida
	2. Selecciona Generar y/o crear informes	2. Carga lista de reportes disponibles
	3. Selecciona reporte a generar	3. Genera reporte
	4. Imprime/Guarda/Visualiza el informe	4. Lo guarda en carpeta determinada por defecto
Precondición:	Debió haberse logeado	
Presunción:	Genero un informe	

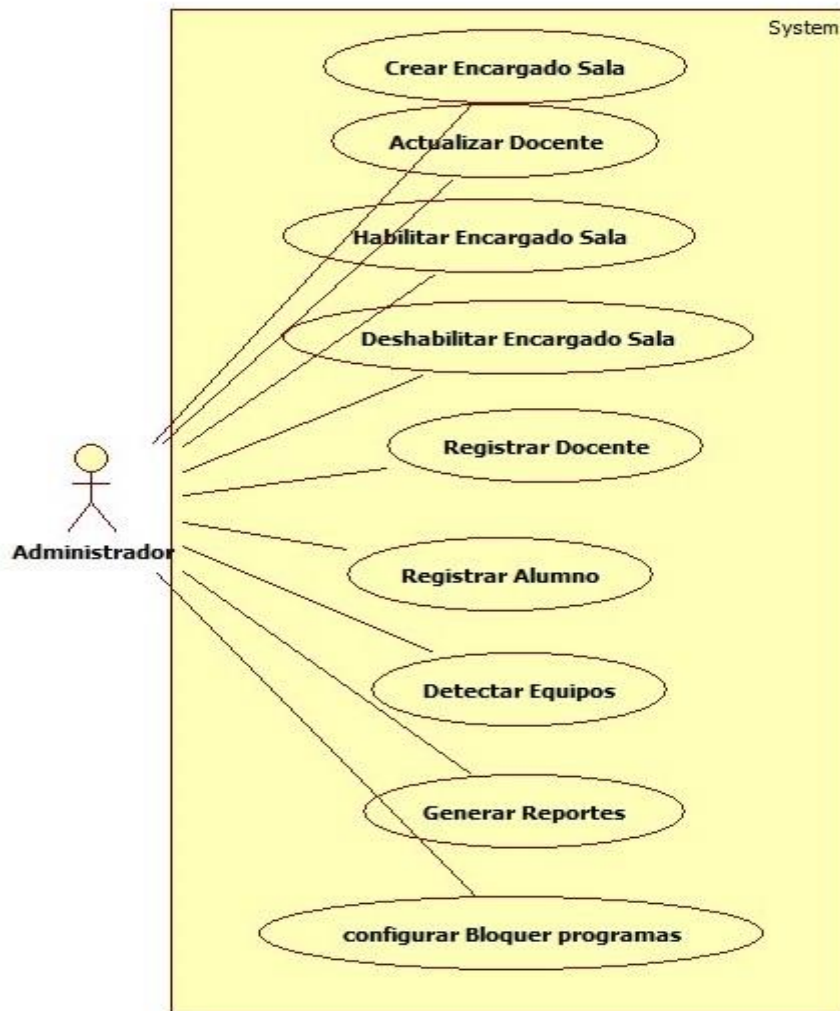
### Caso de uso No. 8: Administrador (Generar informes)



Nombre:	Configuración bloquear programas	
Actor:	Administrador	
Descripción:	Detecta los equipos que estén conectado en red	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresa Login	1. Valida
	2. Ingresa a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona configuración bloqueo	3. Carga configuración bloqueo
	4. Agrega nuevos procesos a la lista de bloqueo	4. Guarda cambios
	5. Configura equipos	
Alternativa:	1. Ingresa Login	1. Valida
	2. Ingresa a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona configuración bloqueo	3. Carga configuración bloqueo
	4. Selecciona una configuración predeterminada	4. Guarda cambios
Precondición:	Los equipos deben estar conectados a la misma red	
Poscondición:		
Presunción:	Visualiza todos los equipos en red	

### Caso de uso No. 9: Administrador (Configurar bloqueo de programas)

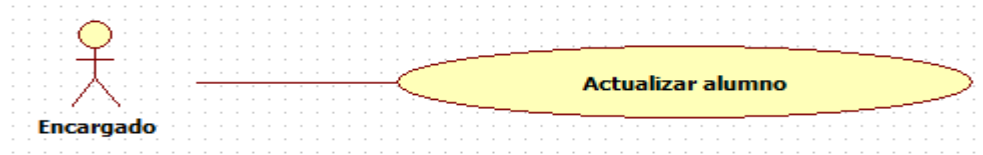
## Diagrama de casos de uso del Administrador.



**Gráfica No 5. Diagrama Casos de uso Administrador.**

Como podemos ver, describimos minuciosamente cada uno de las funcionalidades que podrá hacer el Administrador para un mayor entendimiento de este.

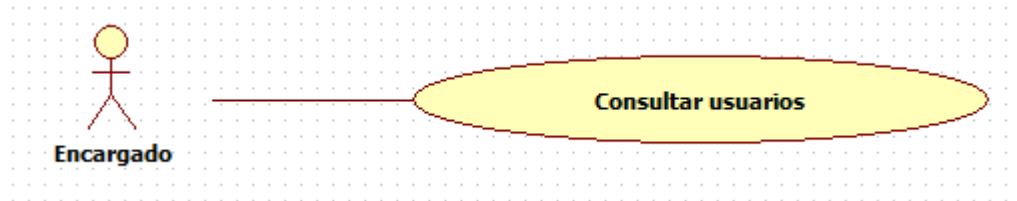
## CASOS DE USO ENCARGADO DE SALA



Nombre:	Actualizar Alumno	
Actor:	Encargado	
Descripción:	Actualiza a un alumno en la base de datos	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona registrar usuario	3. Carga formulario de ingreso de datos
	4. Busca alumno por nombre o id	4. Muestra coincidencias de búsqueda
	5. Selecciona usuario	5. Muestra información de usuario
	6. Actualiza o agrega información	6. Guarda cambios
Alternativa:	Evento Actor	Evento sistema
	1. Ingresar login	1. Valida
	2. Ingresar a modulo administrativo	2. Carga modulo administrativo
	3. Selecciona usuario	3. Carga formulario de ingreso de datos
	4. Busca al alumno por ID o CC	4. Muestra información del alumno
	5. Actualiza información	5. Guarda cambios
Precondición:	Debe estar en la base de datos de la institución y/o empresa	
Presunción:	Se actualizo información del alumno	

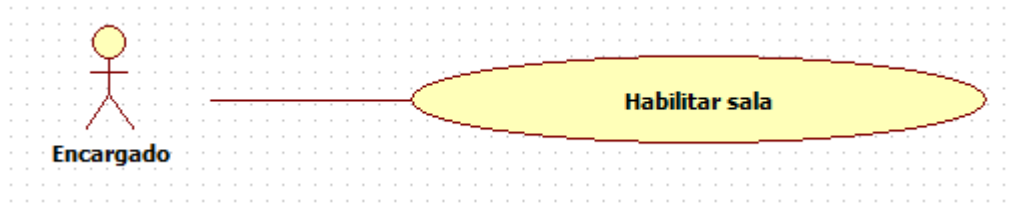
### Caso de uso No. 1: Encargado (Actualizar Alumno)





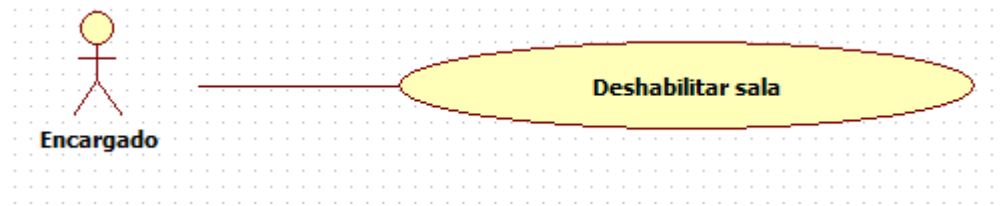
Nombre:	Consultar Usuarios	
Actor:	Encargado	
Descripción:	Visualiza la información de un usuario	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo Administrativo	2. Carga Modulo Administrativo
	3. Selecciona Registrar usuario	3. Carga Formulario de ingreso de datos
	4. Busca usuario por nombre o id	4. Muestra coincidencias de búsqueda
	5. Selecciona usuario	5. Muestra información de usuario
Alternativa:	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo Administrativo	2. Carga Modulo Administrativo
	3. Selecciona Registrar usuario	3. Carga Formulario de ingreso de datos
	4. Busca usuarios por rol	4. Muestra coincidencias de búsqueda
	5. Selecciona usuario	5. Muestra información de usuario
Precondición:	Debe estar en la base de datos de la institución y/o empresa	
Poscondición:		
Presunción:	Se muestra la información del usuario	

## Caso de uso No. 2: Encargado (Consultar Usuarios)



Nombre:	Habilitar Sala	
Actor:	Encargado	
Descripción:	Habilita todos los equipos de determinada sala	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo equipos	2. Carga modulo equipos
	3. Selecciona sala a habilitar	3. Muestra información de la sala
	4. Habilita sala	4. Habilita la sala
Alternativa:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo Equipos	2. Carga modulo equipos
	3. Busca sala	3. Muestra coincidencias de búsqueda
	4. Selecciona sala a habilitar	4. Muestra información de la sala
	5. Habilita sala	5. Habilita la sala
Precondición:	La sala esta deshabilitada	
Poscondición:		
Presunción:	La sala está habilitada para su uso	

### Caso de uso No. 3: Encargado (Habilitar Sala)



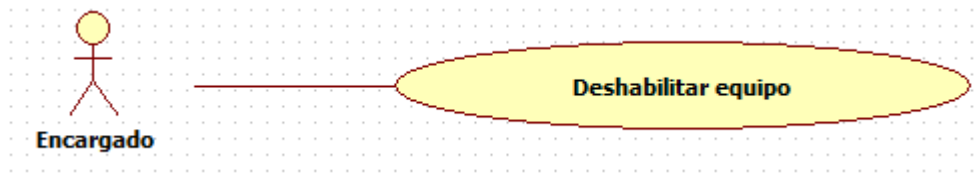
Nombre:	deshabilitar Sala	
Actor:	Encargado	
Descripción:	Deshabilita el uso de los equipos en determinada sala	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo equipos	2. Carga modulo equipos
	3. Selecciona sala a deshabilitar	3. Muestra información de la sala
	4. deshabilita sala	4. deshabilita la sala
Alternativa:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo equipos	2. Carga modulo equipos
	3. Busca sala	3. Muestra coincidencias de búsqueda
	4. Selecciona sala a deshabilitar	4. Muestra información de la sala
	5. Deshabilita sala	5. Deshabilita la sala
Precondición:	La sala está habilitada	
Poscondición:		
Presunción:	La sala ya no está disponible para su uso	

**Caso de uso No. 4 Encargado (Deshabilitar Sala)**



Nombre:	Habilitar Equipo	
Actor:	Encargado	
Descripción:	Habilita el uso de determinado equipo en una sala	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo equipos	2. Carga modulo equipos
	3. Selecciona sala	3. Muestra información de la sala
	4. Selecciona equipo a habilitar	4. Muestra información equipo
	5. Habilita equipo	5. Habilita el equipo
Alternativa:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo equipos	2. Carga modulo equipos
	3. Busca equipo por sala	3. Muestra coincidencias de búsqueda
	4. Selecciona equipo a habilitar	4. Muestra información del equipo
	5. Habilita equipo	5. Habilita el equipo
Precondición:	El equipo esta deshabilitado	
Presunción:	El equipo está listo para usarse	

### Caso de uso No. 5 Encargado (Habilitar Sala)



Nombre:	Deshabilitar Equipo	
Actor:	Encargado	
Descripción:	Deshabilita el uso de determinado equipo en una sala	
Flujo Principal:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo equipos	2. Carga modulo equipos
	3. Selecciona sala	3. Muestra información de la sala
	4. Selecciona equipo a deshabilitar	4. Muestra información equipo
	5. Deshabilita equipo	5. Deshabilita el equipo
Alternativa:	Evento Actor	Evento Sistema
	1. Ingresar LogIn	1. Valida
	2. Ingresar a modulo equipos	2. Carga modulo equipos
	3. Busca equipo por sala	3. Muestra coincidencias de búsqueda
	4. Selecciona equipo a deshabilitar	4. Muestra información del equipo
	5. deshabilita equipo	5. deshabilita el equipo
Precondición :	El equipo está habilitado	
Presunción:	El equipo no se puede usar	

### Caso de uso No. 6: Encargado (Deshabilitar equipo)

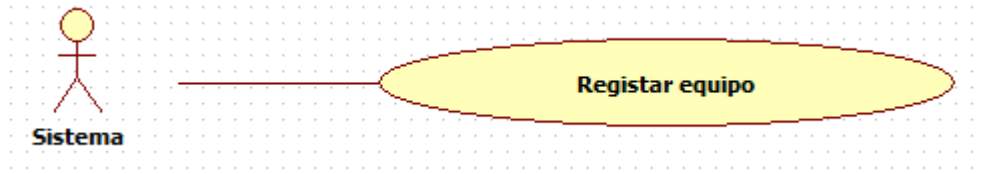
### Diagrama de casos de uso del Encargado.



**Gráfica No 6. Diagrama Casos de uso del Encargado.**

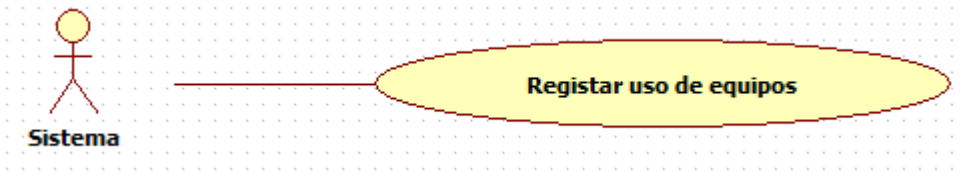
Como podemos ver, describimos minuciosamente cada uno de las funcionalidades que podrá hacer el encargado de cada sala para un mayor entendimiento de este.

## CASOS DE USO SISTEMA



Nombre:	Registrar Equipos	
Actor:	Sistema	
Descripción:	Registra en la bd los equipos que tengan la aplicación cliente	
Flujo Principal:	Evento Actor	Evento Sistema
Alternativa:		
Precondición:	La aplicación cliente debe estar en el equipo.	
Poscondición:		
Presunción:	El equipo se registra en la base de datos.	

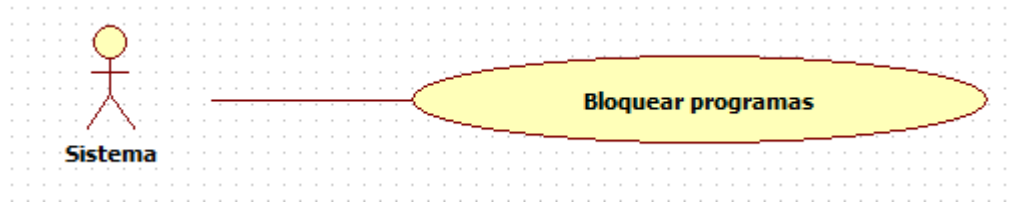
### Caso de uso No. 1: Sistema (Registrar equipos)



Nombre:	Registrar uso equipo	
Actor:	Sistema	
Descripción:	Registra el tiempo de uso del equipo	
Flujo Principal:	Evento Usuario	Evento Sistema
	1. Usuario desbloquea equipo	1. Equipo a la espera de usuario
	2. Usuario hace uso de equipo	
	3. Usuario Cierra terminal	3. Equipo registra tiempo de uso y usuario
Alternativa:		
Precondición:	La aplicación cliente debe estar en el equipo.	
Poscondición:		
Presunción:	Se registra el tiempo de uso del equipo en la base de datos	

**Caso de uso No. 2: Sistema (Registrar uso de equipos)**

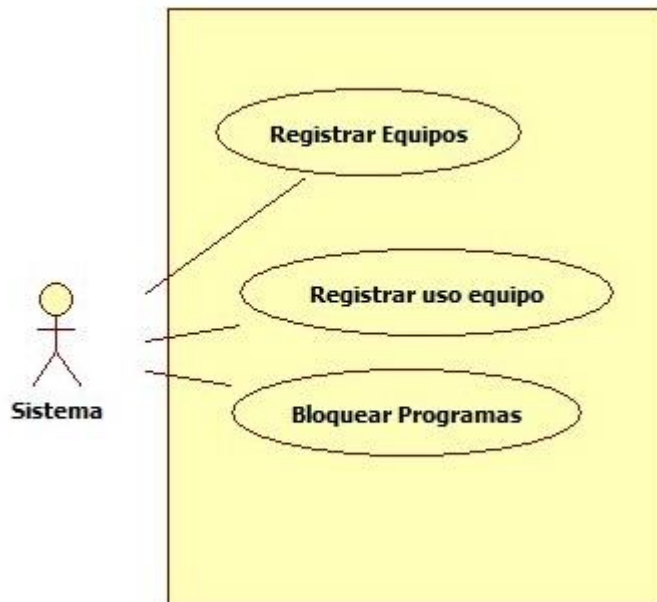




Nombre:	Bloquear Programas	
Actor:	Sistema	
Descripción:	Bloquea la ejecución de programas no permitidos	
Flujo Principal:	Evento Actor	Evento Sistema
	1.Monitorea procesos activos	
	2.Detecta proceso no permitido	
	3.Cierra proceso	
	4.Informa al usuario	
Alternativa:		
Precondición:	Se tiene la lista de programas a bloquear.	
Poscondición:		
Presunción:	Se bloquean los programas que están en la lista de bloqueo.	

**Caso de uso No. 3: Sistema (Bloquear programa)**

### Diagrama de casos de uso del Sistema.



Gráfica No 7. Diagrama Casos de uso del En Sistema.

Como podemos ver, describimos minuciosamente cada uno de las funcionalidades que hará el sistema

- **Diagrama de flujo o de Actividades:** Es la representación gráfica del algoritmo o proceso. Se utiliza en disciplinas como programación, economía, procesos industriales y psicología cognitiva.

A continuación se puede observar el diagrama de flujo, con el fin de poder ilustrar de una mejor forma la funcionalidad de cada uno de los usuarios.

## 4. DESARROLLO

Durante esta etapa se busca definir los requerimientos técnicos a nivel de software y hardware para que la aplicación tenga un óptimo rendimiento.

### 4.1 Especificaciones Técnicas

Durante la etapa de desarrollo se busca definir los requerimientos técnicos a nivel de software y hardware de la aplicación con el fin de generar un listado con las características necesarias de un equipo a nivel de lógico y físico para que este pueda ejecutar sin ningún problema la aplicación desarrollada.

#### ➤ **Software**

Las especificaciones de software necesarias para la debida ejecución de la aplicación son:

Sistema Operativo Windows Xp SP3 en adelante.

Motor base de datos sql server "Equipo Servidor"

Net Framework 4.5.

#### ➤ **Hardware**

Las especificaciones de hardware necesarias para la debida ejecución de la aplicación son:

Intel Pentium Dual Core 2.5GHZ minima

5MB en espacio de disco duro

512mb de Memoria RAM mínima

Disponer de conexión LAN entre los equipos a trabajar

Un equipo que funcione como servidor de la base de datos

## 5. GLOSARIO

- Casos de uso: un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema. Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas
- Componente: Paquete de clases donde interactúan unas con otras.
- Metodología: conjunto de procedimientos racionales utilizados para alcanzar una gama de objetivos
- Modelo ER: El modelo de datos más extendido es el denominado ENTIDAD/RELACIÓN (E/R) En el modelo E/R se parte de una situación real a partir de la cual se definen entidades y relaciones entre dichas entidades.
- Spikes: programas de prueba para reducir riesgos.
- UML: Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más Conocido y utilizado en la actualidad.

## 6. CONCLUSIONES

Este trabajo permitió mejorar nuestras habilidades en el proceso de creación de software, en la recolección de información, definición de requerimientos y el manejo de la metodología de desarrollo.

También permitió desarrollar más nuestra habilidad al momento de diseñar y aplicar un software que cumpla ciertos criterios de funcionamiento y seguridad.

A la vez el interés por la investigación creció durante el desarrollo de este sistema, tomando en cuenta varios puntos de vista sobre cómo desarrollar una metodología ágil y cómo se puede sacar ventaja de ella. Gracias a toda esta información recolectada durante el transcurso, se realizó un análisis profundo al proyecto el cual, gracias a esto nos dimos cuenta de varias falencias que teníamos en este y así los pudimos resolver sin ningún inconveniente.

Todos estos factores nos ayudaron a ampliar mucho más nuestro conocimiento, sobre todo, lo que vimos durante el transcurso de nuestra carrera.

Gracias a todo lo visto durante nuestra carrera profesional, decidimos hacer un software, el cual ayudara tanto a nuestros administradores como a nuestros encargados de salas de cómputo, liberando cargas de trabajo y haciendo mucho más eficiente el desempeño de estas personas.

## BIBLIOGRAFIA

- James Rumbaugh; Ivar Jacobson y Grady Booch. (2000). El proceso unificado del desarrollo de software. Eu: Addison-wesley.
- Jorge Fernandez Gonzales (2013). Introducción a las metodologías ágiles. España: OUC
- Análisis y diseño de sistemas -software-Ingeniería de sistemas: análisis y diseño del software. [Libro en línea de Kendall y Kendall,]. Recuperada Octubre 18, 2013, de <http://unefazuliasistemas.files.wordpress.com/2011/04/analisis-y-disenio-de-sistemas-kendall-kendall>
- Introducción de UML. Recuperado Diciembre 20, 2013, en [http://omg.org/gettingstarted/what\\_is\\_uml.htm](http://omg.org/gettingstarted/what_is_uml.htm)
- Metodología Ágil XP. Recuperada mayo 15, 2014, de <http://eisc.univalle.edu.co/materias/WWW/material/lecturas/xp.pdf>
- Bryant, P. (1999). Biodiversity and Conservation. Recuperada Noviembre de 15 , 2013, de <http://darwin.bio.uci.edu/~sustain/bio65/Titlpage.htm>
- Extreme Programming Explained: Embrace Change [libro de Kent Beck], Recuperada 12 abril, 2014

# **8.1 Manual del sistema**

## **Contenido.**

<b>Objetivo del sistema .....</b>	<b>57</b>
<b>Desarrollo del sistema .....</b>	<b>58</b>
<b>Diagrama de clases .....</b>	<b>58</b>
<b>Base de datos.....</b>	<b>59</b>
<b>Diagrama de paquetes .....</b>	<b>60</b>



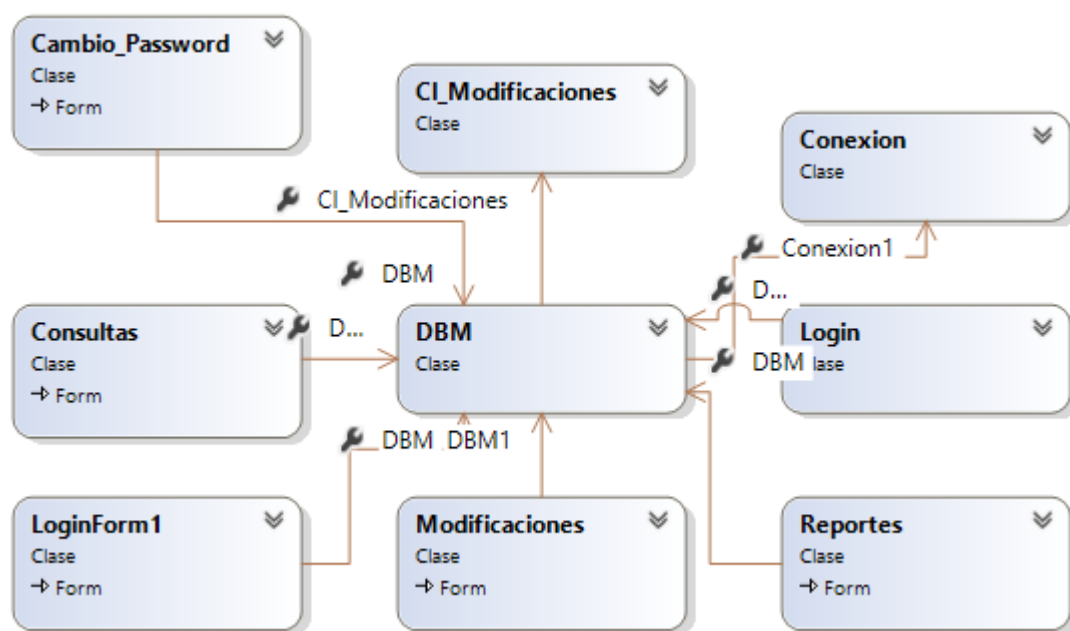
## **Objetivo del sistema**

Este proyecto tiene como objetivo disminuir el rango de inseguridad que se presentan en las salas de computo de las instituciones académicas, ya que algunas de estas instituciones, se hacen prestamos de equipos y los usuarios que manejan dichos equipos les dan un mal uso, produciendo pérdidas para la institución y/o empresa. Para ello se desarrolló un software el cual supla ese problema, ayudando al usuario a supervisar y/o controlar todos los problemas provocados por aquellos usuarios que le dan mal uso a estos equipos, minimizando las perdidas teniendo una mejor seguridad en la institución. Para brindar un alto nivel en la seguridad, por parte del usuario final se tendrá que proveer información la cual lo autentique que es miembro de la organización. A continuación de ello podrá usar el equipo con total libertad sin infringir las leyes de la organización y manteniendo información vital de quien está usando el equipo (motivo de seguridad), después de la inactividad del usuario el equipo se bloqueará, haciendo que el próximo usuario tenga que suministrar de nuevo la información requerida por el sistema para poder hacer uso de este.

## Desarrollo del sistema

En este capítulo del manual, encontraremos cómo se desarrolla el sistema, veremos el paso a paso de como inicia el sistema, como se desenlaza y finalmente como termina, para esto se dotara de gráficos en modo de diagramas para la ejemplificación del sistema.

Para comenzar veremos el diagrama de clases, el cual nos describirá la estructura de un sistema mostrando sus clases, atributos, objetos entre otros.



Gráfica No 1. Diagrama de clases.

Para el desarrollo de esta aplicación se utilizó el lenguaje .NET, utilizando su propio IDE y compilador Visual Basic .NET. Como Podemos apreciar (Gráfica No.1 Diagrama de clases), el control es de vital importancia en este proyecto, ya que este mantiene comunicación con el servidor, la base de datos y los actores. La clase llamada “Control” se encarga del control del servidor el cual provee los servicios del otro nodo llamado en este caso “cliente”. Dicho “cliente” provee información mediante un formulario el cual el usuario deberá llenar obligatoriamente para poder hacer uso de un equipo, después de haber llenado los datos necesarios, el sistema los valida en la base de datos de la organización respectiva y determinará si podrá usarlo o no. Una vez vista la estructura de nuestro sistema, mostraremos la composición lógica de la base de datos.

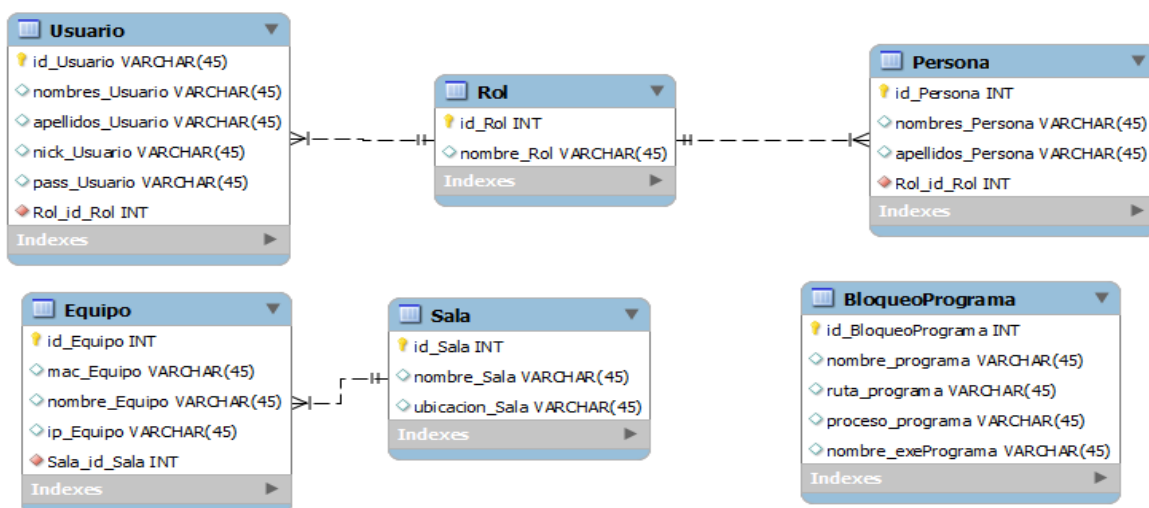
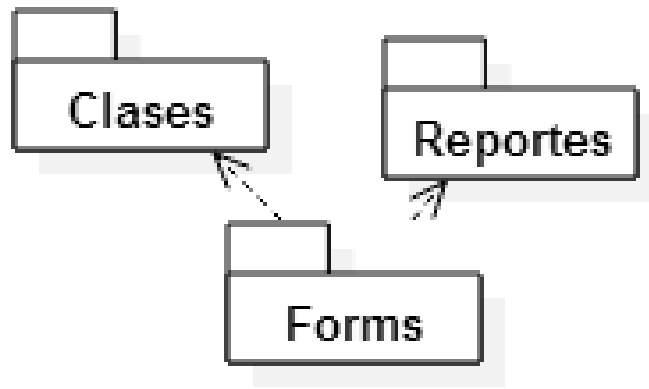


Gráfico No 2. Base de datos.

## Diagrama de paquetes



Gráfica No. 2 Diagrama de paquetes

En la gráfica No. 7 (Diagrama de paquetes) podemos reflejar la organización de los paquetes mostrando las dependencias de cada una de estos componentes.

# **8.1 Manual del Administrador**

## **Contenido.**

<b>Objetivo del sistema .....</b>	<b>63</b>
<b>Desarrollo del sistema .....</b>	<b>64</b>
<b>Diagrama de clases .....</b>	<b>64</b>
<b>Base de datos.....</b>	<b>65</b>
<b>Diagrama de paquetes .....</b>	<b>66</b>
<b>Descripción del sistema .....</b>	<b>67</b>
<b>Base de Datos .....</b>	<b>68</b>

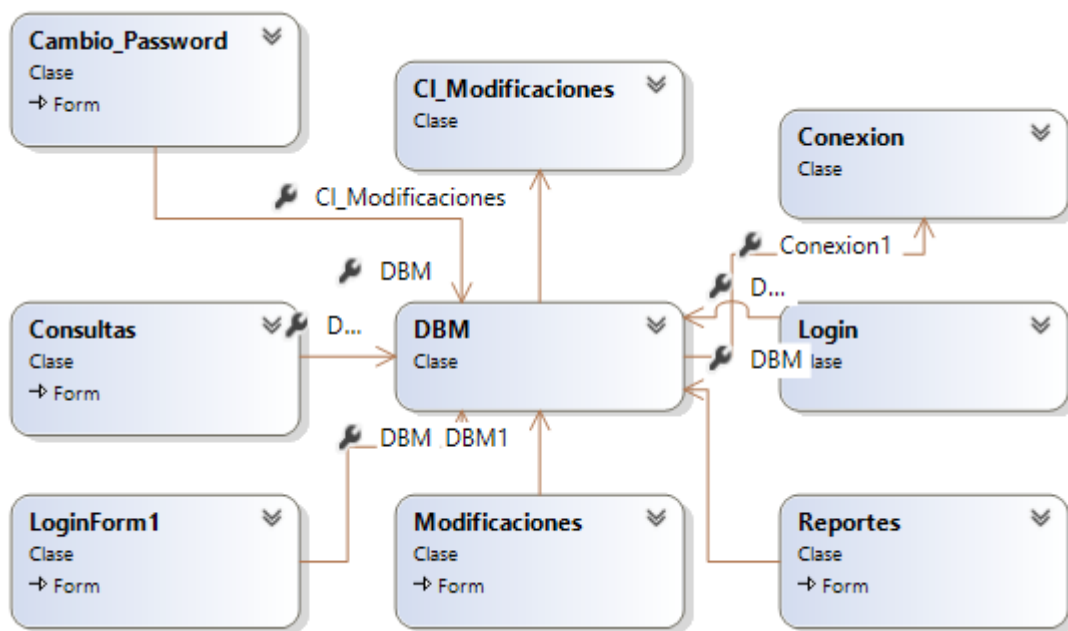
## **Objetivo del sistema**

Este proyecto tiene como objetivo disminuir el rango de inseguridad que se presentan en las salas de computo de las instituciones académicas, ya que algunas de estas instituciones, se hacen prestamos de equipos y los usuarios que manejan dichos equipos les dan un mal uso, produciendo pérdidas para la institución y/o empresa. Para ello se desarrolló un software el cual supla ese problema, ayudando al usuario a supervisar y/o controlar todos los problemas provocados por aquellos usuarios que le dan mal uso a estos equipos, minimizando las perdidas teniendo una mejor seguridad en la institución. Para brindar un alto nivel en la seguridad, por parte del usuario final se tendrá que proveer información la cual lo autentique que es miembro de la organización. A continuación de ello podrá usar el equipo con total libertad sin infringir las leyes de la organización y manteniendo información vital de quien está usando el equipo (motivo de seguridad), después de la inactividad del usuario el equipo se bloqueará, haciendo que el próximo usuario tenga que suministrar de nuevo la información requerida por el sistema para poder hacer uso de este.

## Desarrollo del sistema

En este capítulo del manual, encontraremos cómo se desarrolla el sistema, veremos el paso a paso de como inicia el sistema, como se desenlaza y finalmente como termina, para esto se dotara de gráficos en modo de diagramas para la ejemplificación del sistema.

Para comenzar veremos el diagrama de clases, el cual nos describirá la estructura de un sistema mostrando sus clases, atributos, objetos entre otros.



Gráfica No 1. Diagrama de clases.

Para el desarrollo de esta aplicación se utilizó el lenguaje .NET, utilizando su propio IDE y compilador Visual Basic .NET.



Como Podemos apreciar (Gráfica No.1 Diagrama de clases), el control es de vital importancia en este proyecto, ya que este mantiene comunicación con el servidor, la base de datos y los actores. La clase llamada “Control” se encarga del control del servidor el cual provee los servicios del otro nodo llamado en este caso “cliente”. Dicho “cliente” provee información mediante un formulario el cual el usuario deberá llenar obligatoriamente para poder hacer uso de un equipo, después de haber llenado los datos necesarios, el sistema los valida en la base de datos de la organización respectiva y determinará si podrá usarlo o no.

Una vez vista la estructura de nuestro sistema, mostraremos la composición lógica de la base de datos.

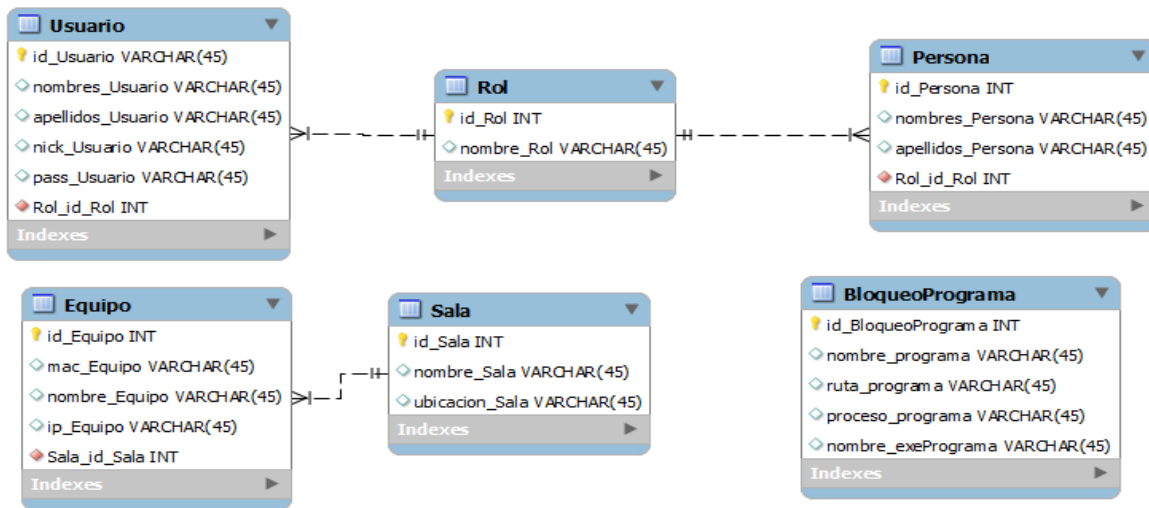
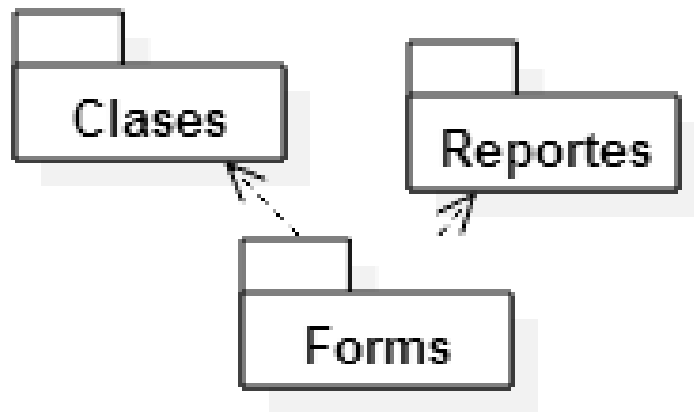


Gráfico No 2. Base de datos.

## Diagrama de paquetes



Gráfica No. 2 Diagrama de paquetes

En la gráfica No. 7 (Diagrama de paquetes) podemos reflejar la organización de los paquetes mostrando las dependencias de cada una de estos componentes.

## **Descripción del sistema**

El sistema cuenta con dos aplicaciones un cliente y otro servidor;

La aplicación cliente se encargara de guardar las características del pc donde esta esté instalada así como, el bloqueo de los equipos, software y el conteo del tiempo de uso del equipo por parte del usuario.

La aplicación servidor será el medio con el cual se enviaran los comandos de bloqueo de equipo y de software por parte del encargado de la sala o del administrador así como la gestión y administración de los usuarios.

La comunicación de las aplicaciones cliente servidor se realiza por el protocolo tcp/ip ya que es un protocolo orientado a la mantención de la conexión.

## Base de datos

Para el acceso y manipulación de los datos de la base de datos el aplicativo se apoya en el uso de dos usuarios diferentes para cada tipo de aplicación (cliente/servidor) de tal manera que los permisos de modificación y acceso estén acordes a las necesidades de la aplicación y el del administrador para tal caso se tiene que para el acceso de la aplicación administrador se realiza por medio del usuario admin y contraseña admin, y el acceso de la aplicación cliente usa el usuario equipo y contraseña equipo.

### Gestión de Usuarios

Hay tres tipos de usuarios en el sistema los cuales son:

**Estudiante:** es el nivel básico de acceso el cual permite registrar el uso dado a los equipos por parte del estudiante así como la visualización de la pantalla de forma remota del mismo.

**Encargado Sala:** es el usuario encargado de gestionar otros usuarios "Estudiantes" así como del control de los equipos en las diferentes salas.

**Administrador:** es el usuario encargado de gestionar a los usuarios encargados sala así como generar los diferentes reportes que posee la aplicación.

Una vez vista la estructura de nuestro sistema, mostraremos la composición lógica de la base de datos.