



***FACULTAD DE CIENCIAS DE LA  
COMPUTACION  
BUAP***

***Desarrollo de un Sistema para el  
Servicio de Medios Audiovisuales***

***Tesis que presenta:***

***Julieta Barragán Acuña***

***Para obtener el grado de:***

***Licenciatura en Ciencias de la  
Computación***

***Asesor:***

***M.C. Alma Delia Ambrosio Vázquez***

***Heróica Puebla de Zaragoza Octubre 2012***



## **INDICE**

<b>Introducción</b>	<b>3</b>
<b>Capítulo I: Marco Teórico</b>	
<b>1.1 Bases de Datos</b>	<b>3</b>
<b>1.1.1 Normalización</b>	<b>4</b>
<b>1.1.2 Diagrama entidad-relación</b>	<b>7</b>
<b>1.2 Ingeniería del Software:</b>	
<b>1.2.1 Paradigma orientado a objetos</b>	<b>11</b>
<b>1.2.2 Metodología de desarrollo RUP</b>	<b>11</b>
<b>1.2.3 Modelado de Lenguaje Unificado UML</b>	<b>13</b>
<b>1.2.4 Seguridad en el Sistema</b>	<b>24</b>
<b>Capítulo II: Estado o Campo del Arte</b>	
<b>2.1 Estado del Arte</b>	<b>27</b>
<b>Capítulo III: Diseño del Sistema</b>	
<b>3.1 Descripción de la información</b>	<b>31</b>
<b>3.2 Representación del flujo de información</b>	<b>37</b>
<b>3.3 Partición Funcional</b>	<b>47</b>
<b>Capítulo IV: Implementación y Pruebas</b>	
<b>4.1 Implementación y pruebas del sistema</b>	<b>52</b>
<b>Conclusión</b>	<b>59</b>
<b>Perspectivas</b>	<b>59</b>
<b>Bibliografía</b>	<b>60</b>
<b>Glosario</b>	<b>61</b>



## **INTRODUCCION**

*Debido al rápido desarrollo tecnológico el uso de la tecnología se ha hecho realidad en prácticamente cualquier ámbito de la vida diaria, por lo que es necesario para los usuarios de la tecnología contar con la forma de acceder a todos los procedimientos del uso de la tecnología con la mayor simplicidad y efectividad posible, tener a la mano las consecuentes formas de administrar esta tecnología vía web con la finalidad de facilitar a los usuarios consulta de inventarios, reservado, acceso y disposición de equipo audiovisual, servicios y catering, que garantice disponibilidad de estos equipos, áreas y servicios en la fecha y hora requerida.*

*Sin embargo, estas metas llevan consigo las consecuentes tareas de llevar un control del inventario de equipo, préstamos, devoluciones, reservado de salas para conferencias, servicios de alimentos, más los imprevistos que puedan ocurrir con el equipo, requiere ser información precisa y oportuna; debido a que el equipo se puede volver indispensable para impartir una conferencia.*

*Ante la diversidad y modalidades de infraestructura a reservar, es necesario realizar estos procesos vía internet para agilizar el reservado, préstamo y devolución de la infraestructura, adicionalmente tener una base de datos actualizada sobre la infraestructura disponible y en buen estado para tenerla en calidad de préstamo.*

*El objetivo del sistema, es facilitar a los usuarios el reservado, acceso y disposición de equipo audiovisual, infraestructura y servicios, haciendo uso de las Tecnologías de la Información.*

*El sistema del servicio de medios audiovisuales fue desarrollado usando la metodología RUP, UML y programado en PHP y MySql.*

## **Capítulo I. Marco Teórico**

### **1.1 Base de Datos**

Un Sistema de Bases de Datos es un conjunto de registros, datos y relaciones que refleja las necesidades de información de una empresa. Estos registros datos y relaciones se agrupan en unidades independientes que satisfacen los requerimientos de información de la empresa. Para un conjunto de aplicaciones que cumplen con algún criterio dado de similitud. Cada una de estas unidades es una base de datos. [8]

Es claro que, una empresa puede necesitar una o más Bases de Datos según la clase de información que requiera.

Una base de datos consiste en una colección de datos relacionados entre sí, que contiene información acerca de una empresa determinada. Toda base de datos se diseña, se construye y prueba con datos para un propósito específico. Esta dirigida a un grupo de usuarios y tiene ciertas aplicaciones preconcebidas que interesan a dichos usuarios.



Las ventajas de utilizar un sistema de bases de datos, es que una base de datos es compacta, rápida, menos laboriosa y actual. Ofreciendo a la empresa un control centralizado de la información.

Existe una amplia base teórica de las bases de datos relacionales. Esta base teórica ayuda al diseño de las bases de datos relacionales y al procesamiento eficiente de las peticiones de los usuarios de información de la base de datos.

Una base de datos relacional consiste en un conjunto de tablas, a cada una de las cuales se le asigna un nombre exclusivo. Cada fila de la tabla representa una relación entre un conjunto de valores. El usuario del sistema de bases de datos puede consultar esas tablas, insertar nuevas tuplas, borrar tuplas y modificar las tuplas.

Existen dos métodos convencionales que son de relevante importancia para el diseño de bases de datos: el modelo entidad-relación y la normalización.

### **1.1.1 Normalización**

La teoría de la normalización ayuda a reconocer las cualidades no deseadas en un archivo y la forma de corregirlas. [14]

Con el procedimiento de normalización, un archivo conceptual se representa como una tabla de dos dimensiones llamada relación: La forma más simple para representar datos mediante una tabla.

El método basado en la normalización supone que el análisis de sistemas produce una lista de los campos de datos de la aplicación y las relaciones entre ellas para que posteriormente sea el proceso de normalización el que separe los campos que identifican identidades de aquellos que sólo las describen.

La meta final del proceso de normalización es la agrupación de todos los atributos (o campos) de una base de datos en relaciones adecuadas para que la base se pueda almacenar con el mínimo de datos redundantes. El propósito de este proceso es quitar las cualidades indeseables de una relación que puedan causar anomalías en el almacenamiento cuando se efectúen operaciones de actualización en la base de datos.

El proceso de normalización empieza con la combinación de todos los datos de la base en una relación, la que a su vez se descompone en dos o más relaciones más pequeñas. Se efectúan descomposiciones sucesivas de las relaciones intermedias hasta que todas las relaciones obtenidas pertenecen a la cuarta forma normal (4FN).

Antes de describir el proceso de la normalización, se debe descubrir la manera de determinar la forma normal de una relación a partir de su relación de atributos conocida como "dependencia funcional".

#### **DEPENDENCIA FUNCIONAL.**



El análisis de una relación de dependencia funcional entre los campos de una relación permite clasificar la relación en una de las cuatro formas normales. El concepto de dependencia funcional (DF), se tomó de las matemáticas elementales. Se dice que Y es función de X,  $Y = f(X)$ , si el valor de Y está siempre determinado por el valor de X. Si se aplica la misma terminología a una relación, la dependencia funcional entre los atributos A y B en una relación se define como sigue: El atributo A es funcionalmente dependiente del atributo B si el valor de A esta determinado por el valor de B. Tal dependencia se simboliza:



Esta representación más simple es útil cuando el diagrama de dependencia funcional contiene una gran cantidad de atributos.

La clave primaria de una relación puede ser una clave compuesta (o concatenada) que consta de más de un campo. Por lo tanto, un atributo puede ser funcionalmente dependiente de un grupo de atributos. Un campo en una clave principal compuesta se llama atributo principal. Cualquier campo que no forme parte de la clave principal se llama no-clave.

El propósito de establecer un diagrama de dependencia funcional para una relación es usar un método gráfico para determinar la forma normal de la relación. En las secciones siguientes, se verán ejemplos de las cuatro formas normales y técnicas para normalizar relaciones que previenen anomalías del almacenamiento causadas por relaciones indeseables entre los atributos.

### **PRIMERA FORMA NORMAL (1FN).**

Una relación está en la primera forma normal (o 1FN) si todos los campos en cada registro contienen un solo valor tomado de sus dominios respectivos. El dominio de un campo es el rango de valores permitidos para el campo.

#### **Normalización y verificación de la relación 1FN.**

Como se describió anteriormente, las anomalías de almacenamiento se pueden atribuir a la presencia de uno o más campos no-clave que no son total y funcionalmente dependientes de la clave principal. Afortunadamente, las anomalías de almacenamiento de la relación 1FN se pueden eliminar con el siguiente procedimiento:

- i) Quitar de la relación 1FN todos los campos no-clave que no sean totalmente dependientes de la clave primaria.
- ii) Guardar los campos no-clave que fueron quitados en relaciones nuevas y adecuadas.

### **SEGUNDA FORMA NORMAL (2FN).**



primaria con menos de tres atributos principales es automáticamente 4FN. Una dependencia multivalor no trivial también es llamada dependencia unión-binaria no trivial.

Sin normalizar	Datos almacenados redundantemente en un archivo no plano.
1a. forma normal	La relación tiene registros por separado para cada valor en cada campo del registro o cada campo de un registro contiene un solo valor.
2a. forma normal	Cada clave depende total y funcionalmente de su clave principal.
3a. forma normal	Ninguna no-clave depende indirectamente de su clave principal, y cada determinante es clave aspirante.
BFCN	La relación no contiene claves aspirantes sobrepuestas.
4a. Forma normal	La relación no contiene dependencias multivalores no triviales.

**Figura:** NIVELES DE NORMALIZACIÓN.

### **1.1.2 DIAGRAMA ENTIDAD RELACION**

El modelo de datos entidad-relación (E-R) está basado en una percepción del mundo real que consta de un conjunto de objetos básicos llamados entidades y de relaciones entre estos objetos. Una entidad es una <<cosa>> u <<objeto>> en el mundo real que es distinguible de otros objetos. Por ejemplo para el caso que nos ocupa un inmueble es una entidad. Las entidades se describen en una base de datos mediante un conjunto de atributos. Una relación es una asociación entre varias entidades. [13]

El método entidad-relación parte del hecho de que uno de los resultados del análisis del sistema es la comprensión clara de cuales son las entidades incluidas.

Un modelo entidad-relación asiste al diseñador para definir y entender las cosas significativas acerca de cuál información necesita ser conocida o manejada y también las relaciones entre estas cosas.

Un grupo de entidades similares forman un conjunto de entidades. Las propiedades de las entidades son llamadas atributos, los valores para esos atributos están en un dominio de valores. Un atributo o conjunto de atributos cuyos valores identifican únicamente cada entidad en un conjunto de entidades es llamada una "llave" o "clave".

El modelo entidad-relación categoriza todos los elementos de un sistema como una entidad (una persona, lugar o cosa) o una relación entre entidades. Ambas construcciones son representadas por la misma estructura, una tabla.

La aplicación del modelo entidad-relación requiere los siguientes pasos:

1. Identificar las entidades del sistema y construir una tabla para representar cada entidad.
2. Identificar las relaciones entre las entidades y las tablas actuales extendidas o crear tablas nuevas para representar estas relaciones.

### RELACIONES.

Las relaciones representan las reglas y la información que el negocio necesita.

- Una relación es una asociación importante entre dos entidades.
- Una relación es "que tiene que ver una cosa con otra".
- Una relación es bi-direccional, asociación importante entre dos entidades.

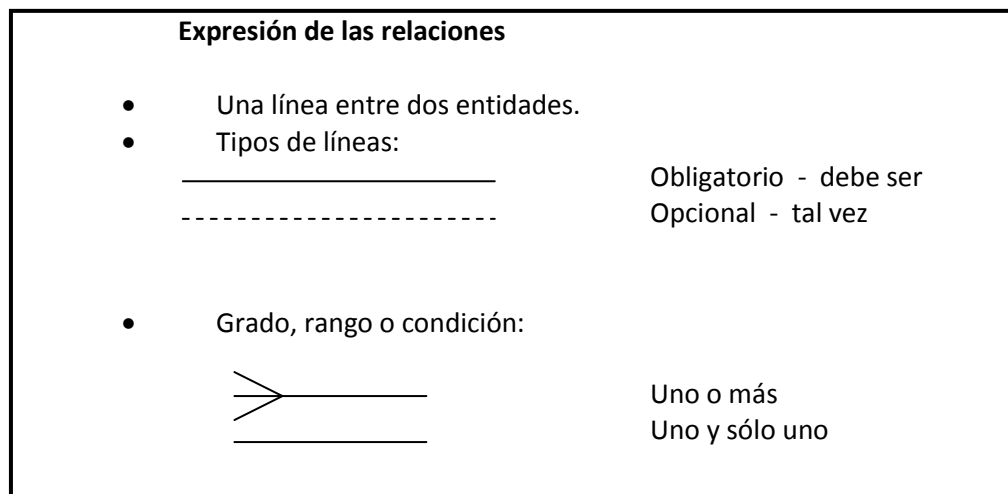


Tabla: Representación de las relaciones en el modelo E-R.



Los tipos de relaciones pueden ser:

**Relaciones Muchos-a-Uno:**

Estas son las más comunes y muestran que una relación tiene un grado de uno o más en una dirección y uno y solamente uno en la otra dirección.

**Relaciones Muchos-a-Muchos:**

Hay un grado de uno o más en ambas direcciones y también son muy comunes. Usualmente son opcionales en ambas direcciones.

**Relaciones Uno-a-Uno:**

Hay un grado de uno y sólo uno en ambas direcciones. Este tipo de relaciones es bastante raro, comúnmente indican que las dos entidades son realmente la misma entidad en términos de negocio. [8]

**ATRIBUTOS.**

Los atributos son información acerca de una entidad que necesita ser conocida o descrita. Los atributos describen una entidad por que la:

- Habilitan.
- Identifican.
- Clasifican.
- Numeran.
- Expresan su estado.

Los atributos representan un tipo de descripción o detalles para cada instancia.

**Encontrando atributos.**

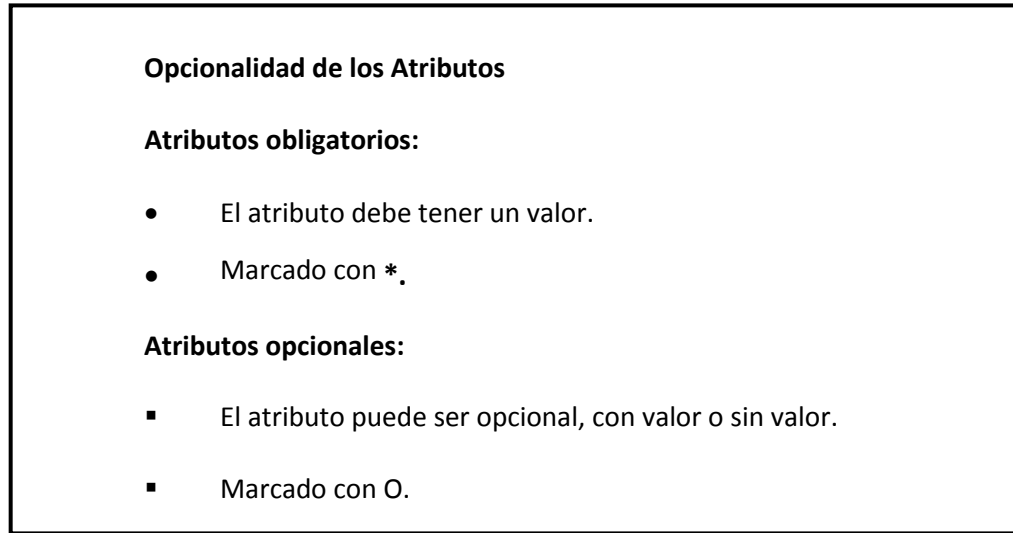
Los atributos quizá aparecen como:

- Palabras descriptivas y frases o nombres.
- Frases preposicionales (por ejemplo, cantidad de salario para cada empleado).
- Nombres posesivos y pronombres (por ejemplo, nombre del empleado).

La siguiente tabla nos muestra las características de una entidad y sus atributos.

<b>Características de la Entidad</b>	<b>Características de los atributos</b>
Una cosa acerca de cual información debe ser incluida.	Calificar una entidad.
Poseer uno o más atributos (si no tiene atributos, quizá no sea una entidad, tal vez sea un atributo).	No posee atributos propios (si tiene un atributo, entonces tal vez sea una entidad, o una combinación de los mismos atributos.
Quizá tiene múltiples ocurrencias y debe ser asociado con otra entidad por una relación.	Tiene un solo valor para cada instancia de la entidad.

Características de las Entidades y Atributos.



**Figura:** Muestra los detalles que acompañan un atributo.

Cuando tienes identificados los atributos necesitas adherir algunas marcas (ver figura ant.), considerando si el atributo siempre tiene un valor para cada instancia de la entidad. Si hay alguna posibilidad de que un atributo no tenga un valor, en todo caso, debes considerar un atributo opcional.

#### **Detalles de los Atributos.**

Cuando tienes identificados los atributos para una entidad, debes tener un valor para cada instancia de la entidad, puedes examinar los detalles de los atributos.

**Descripción de los Atributos:** Justamente una entidad necesita una descripción, entonces se hace un atributo.

**Formato de los atributos:** Cada atributo debe tener asignado un formato de información.

- TIPO. Caracter, Fecha, Imagen, Entero, Dinero, Fotografía, Sonido, Texto.
- MÁXIMA LONGITUD. Solamente es usado con los tipos como son: caracter o número.
- PROMEDIO LONGITUD. El tamaño normal del atributo.
- LUGAR DECIMAL. El número de dígitos después del lugar decimal.
- UNIDAD DE MEDIDA. Puedes asignar un peso, dimensión o capacidad.

**Valores admisibles.** Especificar valores válidos para los atributos:

- Rango de valores, por ejemplo 0-65 o A-M.
- Especificar la lista de valores, por ejemplo 4, 6, 8, 32 o Y, N.
- Una serie de rangos, por ejemplo 12-16, 20-15, 30-38.

#### **IDENTIFICADORES ÚNICOS.**

Cada entidad debe ser identificada, un identificador único (UID) es una combinación de atributos y relaciones que sirven para identificar una instancia de una entidad. Puedes especificar el identificador único durante el análisis, pero cada entidad debe tener un identificador único en orden para empezar a designar.



Debes encontrar que una instancia pueda ser fácilmente identificada por un solo atributo. Cuando estás seleccionando un camino para identificar una instancia de una entidad debes tener el modelo del negocio. Es decir, debes preguntarte que necesita el negocio, tiene o quiere usar una clave, entonces esa clave úsala como el identificador único. Si la clave no existe entonces realiza preguntas en el departamento para encontrar como diferencian entre las instancias. Cuando tengas identificado un atributo como un identificador único, puedes marcarlo con el símbolo #.

## **1.2 Ingeniería del Software**

### **1.2.1 Paradigma orientado a objetos**

El paradigma orientado a objetos utiliza objetos del mundo real, ligado por mensajes, para la solución de problemas y permite: [1]

- Proximidad de los conceptos de modelado respecto de las entidades del mundo real.
  - Mejora captura y validación de requisitos.
  - Acerca el “espacio del problema” y el “espacio de la solución”.
  - 
  - Modelado integrado de propiedades estáticas y dinámicas del ámbito del problema.
  - Facilita construcción, mantenimiento y reutilización.
  - 
  - Conceptos comunes de modelado durante el análisis, diseño e implementación
  - Facilita la transición entre distintas fases.
  - Favorece el desarrollo iterativo del sistema.
  - Disipa la barrera entre el “qué” y el “cómo”.
  -
- Sin embargo, existen problemas...

### **1.2.2 Metodología de desarrollo RUP**

Objetos

- Objeto = unidad atómica que encapsula estado y comportamiento.
- La encapsulación en un objeto permite una alta cohesión y un bajo acoplamiento.
- Un objeto puede caracterizar una entidad física (coche) o abstracta (ecuación matemática). [9]
  
- UML define una notación que se expresa como diagramas sirven para representar modelos/subsistemas o partes de ellos.**

### **Modelos y Diagramas**

Un **modelo** captura una vista de un sistema del mundo real. Es una abstracción de dicho sistema, considerando un cierto propósito. Así, el modelo describe completamente aquellos aspectos del sistema que son relevantes al propósito del modelo, y a un apropiado nivel de detalle. [9]



**Diagrama:** una representación gráfica de una colección de elementos de modelado, a menudo dibujada como un grafo con vértices conectados por arcos.

- Un proceso de desarrollo de software debe ofrecer un conjunto de modelos que permitan expresar el producto desde cada una de las perspectivas de interés
- El código fuente del sistema es el modelo más detallado del sistema (y además es ejecutable). Sin embargo, se requieren otros modelos...
- Cada modelo es completo desde su punto de vista del sistema, sin embargo, existen relaciones de trazabilidad entre los diferentes modelos

### **a) Metodologías:**

A través de la metodología RUP RationalObjectoryProcess, se lleva a cabo los pasos para el desarrollo del sistema: [9]

- Modelado de negocio, en base a los casos de uso del sistema.
- Recolección de requerimientos y necesidades de los usuarios.
- Análisis de requerimientos, una vez recabados los requerimientos se realizará un análisis de ellos preparándose para el diseño
- Diseño de datos, se realizará el diseño utilizando una metodología de la Ingeniería de Software
- Implementación, como resultado de los requerimientos, análisis y diseño.
- Pruebas del sistema.
- Despliegue del comportamiento del software con el hardware.

Comprende tres fases claves: Dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental.

La estructura dinámica de RUP es la que permite que éste sea un proceso de desarrollo fundamentalmente iterativo, y en esta parte se ven inmersas las 4 fases descritas anteriormente:

- Inicio (También llamado Incepción o Concepción)
- Elaboración
- Desarrollo (También llamado Implementación, Construcción)
- Cierre (También llamado Transición)

**Fase de Inicio:** Esta fase tiene como propósito definir y acordar el alcance del proyecto con los patrocinadores, identificar los riesgos asociados al proyecto, proponer una visión muy general de la arquitectura de software y producir el plan de las fases y el de iteraciones posteriores.

**Fase de elaboración:** En la fase de elaboración se seleccionan los casos de uso que permiten definir la arquitectura base del sistema, se realiza la especificación de los casos de uso seleccionados y el primer análisis del dominio del problema, diseñándose la solución preliminar.

**Fase de Desarrollo:** El propósito de esta fase es completar la funcionalidad del sistema, para ello se deben clarificar los requisitos pendientes, administrar los cambios de acuerdo a las evaluaciones realizados por los usuarios y se realizan las mejoras para el proyecto.



Fase de Cierre: El propósito de esta fase es asegurar que el software esté disponible para los usuarios finales, ajustar los errores y defectos encontrados en las pruebas de aceptación, capacitar a los usuarios y proveer el soporte técnico necesario. Se debe verificar que el producto cumpla con las especificaciones entregadas por las personas involucradas en el proyecto.

## ***b) UML***

\_ UML = Unified Modeling Language Un lenguaje de propósito general para el modelado orientado a objetos.

\_ Impulsado por el Object Management Group (OMG)

\_ Documento "OMG Unified Modeling Language Specification"

\_ UML combina notaciones provenientes desde:

o Modelado Orientado a Objetos

o Modelado de Datos Modelado de Componentes

o Modelado de Flujos de Trabajo (Workflows)

UML define una notación que se expresa como diagramas sirven para representar modelos/subsistemas o partes de ellos. [22]

### **Modelos y Diagramas**

Un **modelo** captura una vista de un sistema del mundo real. Es una abstracción de dicho sistema, considerando un cierto propósito. Así, el modelo describe completamente aquellos aspectos del sistema que son relevantes al propósito del modelo, y a un apropiado nivel de detalle.

**Diagrama:** una representación gráfica de una colección de elementos de modelado, a menudo dibujada como un grafo con vértices conectados por arcos.

Un proceso de desarrollo de software debe ofrecer un conjunto de modelos que permitan expresar el producto desde cada una de las perspectivas de interés

El código fuente del sistema es el modelo más detallado del sistema (y además es ejecutable). Sin embargo, se requieren otros modelos...

Cada modelo es completo desde su punto de vista del sistema, sin embargo, existen relaciones de trazabilidad entre los diferentes modelos.

## **1.2.3 EL LENGUAJE DE MODELADO UNIFICADO (UML)**

El Lenguaje de Modelado Unificado (UML - Unified Modeling Language) es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables. [22]



En función de las diferentes vistas del modelo, en UML se definen los siguientes diagramas gráficos:

- Diagrama de casos de uso
- Diagrama de clases
- Diagramas de comportamiento:
  - Diagrama de estados
  - Diagrama de actividades
- Diagramas de interacción:
  - Diagrama de secuencia
  - Diagrama de colaboración
- Diagramas de implementación
  - Diagrama de componentes
  - Diagrama de despliegue

Por otro lado podemos ver el modelo de una forma estática o de una forma dinámica. Estas perspectivas nos dan la siguiente clasificación:

- **Modelo estático (estructural):**
  - Diagrama de despliegue
  - Diagrama de componentes
  - Diagrama de clases
  - Diagrama de objetos
- **Modelo dinámico (comportamiento):**
  - Diagrama de estados
  - Diagrama de actividades
  - Diagrama de secuencia
  - Diagrama de colaboración
  - Diagrama de casos de uso

Los cuatro diagramas estructurales de UML permiten visualizar, especificar, construir y documentar los aspectos estáticos de un sistema:

Un **diagrama de clases** presenta un conjunto de clases, interfaces y colaboraciones, y las relaciones entre ellas.

Un **diagrama de objetos** representa un conjunto de objetos y sus relaciones. Se utilizan para describir estructuras de datos, instantáneas de los elementos encontrados en los diagramas de clases. Cubre los mismos aspectos que los diagramas de clases pero desde una perspectiva de casos reales o prototípicos.

Un **diagrama de componentes** muestra un conjunto de componentes y sus relaciones. Los diagramas de componentes se utilizan para describir la vista de implementación estática de un sistema. Los diagramas de componentes se relacionan con los diagramas de clases, ya que un componente normalmente se corresponde con una o más clases, interfaces o colaboraciones.

Un **diagrama de despliegue** muestra un conjunto de nodos y sus relaciones. Los diagramas de despliegue se utilizan para describir la vista de despliegue estática de un sistema. Los diagramas de



despliegue se relacionan con los diagramas de componentes, ya que un nodo normalmente incluye uno o más componentes.

Los cinco diagramas de comportamiento de UML se emplean para visualizar, especificar, construir y documentar los aspectos dinámicos de un sistema.

Los **diagramas de casos de uso** organizan los comportamientos del sistema. Un diagrama de caso de uso representa un conjunto de casos de uso y actores (un tipo especial de clases) y sus relaciones.

Los **diagramas de interacción**: Se da este nombre colectivo a los diagramas de secuencia y los diagramas de colaboración. Ambos diagramas son isomorfos, es decir, se puede convertir de uno a otro sin pérdida de información.

Un **diagrama de secuencia** es un diagrama de interacción que resalta la ordenación temporal de los mensajes. Un diagrama de secuencia presenta un conjunto de objetos y los mensajes enviados y recibidos por ellos., Los objetos suelen ser instancias con nombre o anónimas de clases, pero también pueden representarse instancias de otros elementos, tales como colaboraciones, componentes y nodos.

Un **diagrama de colaboración** es un diagrama de interacción que resalta la organización estructural de los objetos que envían y reciben mensajes. Un diagrama de colaboración muestra un conjunto de objetos, enlaces entre estos objetos y mensajes enviados y recibidos por estos objetos. Los objetos normalmente son instancias con nombre o anónimas de clases, pero también pueden representar instancias de otros elementos, como colaboraciones, componentes y nodos.

Un **diagrama de estado** representa una máquina de estados, constituida por estados, transiciones, eventos y actividades. Son especialmente importantes para modelar el comportamiento de una interfaz, una clase o una colaboración. Los diagramas de estados resaltan el comportamiento dirigido por eventos de un objeto, lo que es especialmente útil al modelar sistemas reactivos.

Un **diagrama de actividades** es un tipo especial de diagrama de estados que muestra el flujo secuencial o ramificado de actividades en un sistema. conjunto de actividades, Son especialmente importantes para modelar la función del sistema, así como para resaltar el flujo de control entre objetos.

## **Diagramas de Casos de Uso**

El diagrama de casos de uso representa la forma en como un Cliente (Actor) opera con el sistema en desarrollo, además de la forma, tipo y orden en como los elementos interactúan (operaciones o casos de uso).

Un diagrama de casos de uso muestra, por tanto, los distintos requisitos funcionales que se esperan de una aplicación o sistema y cómo se relaciona con su entorno (usuarios u otras aplicaciones). Un caso de uso, denotando un requisito funcional exigido al sistema, se representa en el diagrama por una elipse y un nombre significativo.

Un diagrama de casos de uso consta de los siguientes elementos:

- Actor.
- Casos de Uso.
- Relaciones de Uso, Herencia y Comunicación.



## **Elementos**

- **Actor:**

Una definición previa, es que un Actor es un rol que un usuario juega con respecto al sistema. Es importante destacar el uso de la palabra rol, pues con esto se especifica que un Actor no necesariamente representa a una persona en particular, sino más bien la labor que realiza frente al sistema.

Como ejemplo a la definición anterior, tenemos el caso de un sistema de ventas en que el rol de Vendedor con respecto al sistema puede ser realizado por un Vendedor o bien por el Jefe de Local.

- **Caso de Uso:**

Es una operación/tarea específica que se realiza tras una orden de algún agente externo, sea desde una petición de un actor o bien desde la invocación desde otro caso de uso.

- **Relaciones:**

- Asociación

Es el tipo de relación más básica que indica la invocación desde un actor o caso de uso a otra operación (caso de uso). Dicha relación se denota con una flecha simple.

- Dependencia o Instanciación

Es una forma muy particular de relación entre clases, en la cual una clase depende de otra, es decir, se instancia (se crea). Dicha relación se denota con una flecha punteada.

- Generalización

Este tipo de relación es uno de los más utilizados, cumple una doble función dependiendo de su estereotipo, que puede ser de Uso (<<uses>>) o de Herencia (<<extends>>). Este tipo de relación esta orientado exclusivamente para casos de uso (y no para actores).

extends: Se recomienda utilizar cuando un caso de uso es similar a otro (características).

uses: Se recomienda utilizar cuando se tiene un conjunto de características que son similares en más de un caso de uso y no se desea mantener copiada la descripción de la característica.

De lo anterior cabe mencionar que tiene el mismo paradigma en diseño y modelamiento de clases, en donde esta la duda clásica de usar o heredar.

## **Diagramas de Clases**

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

Un diagrama de clases esta compuesto por los siguientes elementos:

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

## Elementos

- **Clase**

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

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

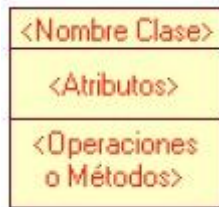


Figura 2.6 Esquema de una clase




En donde:

- **Superior:** Contiene el nombre de la Clase
- **Intermedio:** Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser private, protected o public).
- **Inferior:** Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: private, protected o public).

Atributos y Métodos:



- **Atributos:**


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

- **public (+, ):** Indica que el atributo será visible tanto dentro como fuera de la clase, es decir, es accesible desde todos lados.
- **private (-, ):** Indica que el atributo sólo será accesible desde dentro de la clase (sólo sus métodos lo pueden acceder).
- **protected (#, ):** Indica que el atributo no será accesible desde fuera de la clase, pero si podrá ser accedido por métodos de la clase además de las subclases que se deriven (ver herencia).

- **Métodos:**

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

- **public (+, ):** Indica que el método será visible tanto dentro como fuera de la clase, es decir, es accesible desde todos lados.
- **private (-, ):** Indica que el método sólo será accesible desde dentro de la clase (sólo otros métodos de la clase lo pueden acceder).

- `protected (#, )`: Indica que el método no será accesible desde fuera de la clase, pero si podrá ser accedido por métodos de la clase además de métodos de las subclases que se deriven (ver herencia).

- **Relaciones entre Clases:**

Ya definido el concepto de Clase, es necesario explicar como se pueden interrelacionar dos o más clases.

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

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

- **Herencia (Especialización/Generalización):** 

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

Agregación:

Para modelar objetos complejos, bastan los tipos de datos básicos que proveen los lenguajes: enteros, reales y secuencias de caracteres. Cuando se requiere componer objetos que son instancias de clases definidas por el desarrollador de la aplicación, tenemos dos posibilidades:

- **Por Valor:** Es un tipo de relación estática, en donde el tiempo de vida del objeto incluido esta condicionado por el tiempo de vida del que lo incluye. Este tipo de relación es comúnmente llamada Composición (el Objeto base se construye a partir del objeto incluido, es decir, es "parte/todo").

- **Por Referencia:** Es un tipo de relación dinámica, en donde el tiempo de vida del objeto incluido es independiente del que lo incluye. Este tipo de relación es comúnmente llamada Agregación (el objeto base utiliza al incluido para su funcionamiento).

- **Clase parametrizada:**

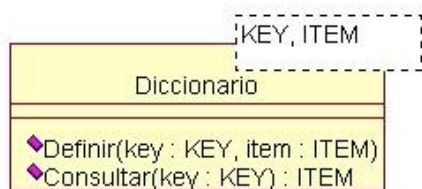


Figura 2.11 Clase Parametrizada

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



generalización puede venir dada de un Template (como en el caso de C++) o bien de alguna estructura predefinida (especialización a través de clases).

## **Diagramas de Estados**

Los diagramas de estado describen gráficamente los **eventos** y los **estados** de los objetos. Los diagramas de estado son útiles, entre otras cosas, para indicar los eventos del sistema en los casos de uso.

### **Elementos**

- Un **evento** es un acontecimiento importante a tomar en cuenta para el sistema.
- Un **estado** es la condición de un objeto en un momento determinado: el tiempo que transcurre entre eventos.
- Una **transición** es una relación entre dos estados, e indica que, cuando ocurre un evento, el objeto pasa del estado anterior al siguiente.

En UML, los estados se representan mediante óvalos. Las transiciones se representan mediante flechas con el nombre del evento respectivo. Se acostumbra poner un estado inicial (círculo negro).

Un diagrama de estado que describe los eventos globales del sistema y su secuencia en un caso de uso es un diagrama de estado para casos de uso.

## **Diagramas de Actividades**

Un diagrama de actividades puede considerarse como un caso especial de un diagrama de estados en el cual casi todos los estados son estados acción (identifican una acción que se ejecuta al estar en él) y casi todas las transiciones evolucionan al término de dicha acción (ejecutada en el estado anterior). Un diagrama de actividades puede dar detalle a un caso de uso, un objeto o un mensaje en un objeto. Permiten representar transiciones internas al margen de las transiciones o eventos externos.

La interpretación de un diagrama de actividades depende de la perspectiva considerada: en un diagrama conceptual, la actividad es alguna tarea que debe ser realizada; en un diagrama de especificación o de implementación, la actividad es un método de una clase. Generalmente se suelen utilizar para modelar los pasos de un algoritmo.

Los diagramas de actividad son similares a los diagramas de flujo procesales, con la diferencia de que todas las actividades están claramente unidas a objetos, siempre están asociados a una clase, a una operación o a un caso de uso. Soportan actividades tanto secuenciales como paralelas. La ejecución paralela se representa por medio de iconos de fork/espera, y en el caso de las actividades paralelas, no importa en qué orden sean invocadas (pueden ser ejecutadas simultáneamente o una detrás de otra).

Las actividades pueden formar jerarquías, lo que significa que una actividad puede estar formada de varias actividades «de detalle», en cuyo caso las transiciones entrantes y salientes deberían coincidir con las del diagrama de detalle.



Un diagrama de Actividad demuestra la serie de actividades que deben ser realizadas en un caso de uso, así como las distintas rutas que pueden irse desencadenando en el caso de uso. Es muy similar en definición a un diagrama de flujo (típicamente asociado en el diseño de Software), estos no son lo mismo.

## Elementos

- **Inicio:** El inicio de un diagrama de actividad es representado por un círculo de color negro sólido.
- **Actividad:** Una actividad representa la acción que será realizada por el sistema la cual es representada dentro de un ovalo.
- **Transición:** Una transición ocurre cuando se lleva acabo el cambio de una actividad a otra, la transición es representada simplemente por una línea con una flecha en su terminación para indicar dirección.
- **Ramificación (Branch):** Una ramificación ocurre cuando existe la posibilidad que ocurra más de una transición (resultado) al terminar determinada actividad. Este elemento es representado a través de un rombo.
- **Unión (Merge):** Una unión ocurre al fusionar dos o más transiciones en una sola transición o actividad. Este elemento también es representado a través de un rombo.
- **Expresiones Resguardadas (Guard Expressions):** Una expresión resguardada es utilizada para indicar una descripción explícita acerca de una transición. Este tipo de expresión es representada mediante corchetes [...] y es colocada sobre la línea de transición.
- **Fork :** Un fork representa una necesidad de ramificar una transición en más de una posibilidad. Aunque similar a una ramificación (Branch) la diferencia radica en que un fork representa más de una ramificación obligada, esto es, la actividad debe proceder por ambos o más caminos, mientras que una ramificación (Branch) representa una transición u otra para la actividad (como una condicional). Un fork es representado por una línea negra solida, perpendicular a las líneas de transición.
- **Join :** Una join ocurre al fusionar dos o más transiciones provenientes de un fork, y es empleado para dichas transiciones en una sola, tal y como ocurría antes de un fork .
- **Fin :** El fin de un diagrama de actividad es representado por un círculo, con otro círculo concéntrico de color negro sólido.
- **Canales (Swimlanes):** En determinadas ocasiones ocurre que un diagrama de actividad se expanda a lo largo de más de un entidad o actor, cuando esto ocurre el diagrama de actividad es particionada en canales (swimlines), donde cada canal representa la entidad o actor que esta llevando acabo la actividad.

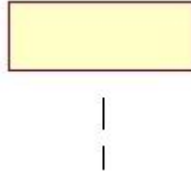
## Diagramas de Secuencia

En un diagrama de secuencia participan varios de los objetos o clases que forman parte del programa y se describen las llamadas que van haciendo unos a otros para realizar una tarea determinada.

Se realiza un diagrama de secuencia por cada caso de uso o para una parte de un caso de uso (lo que llamo subcaso de uso).

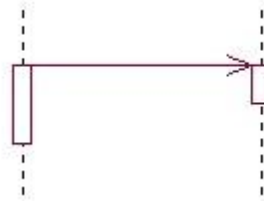
## Elementos

- **Objeto/Actor:**



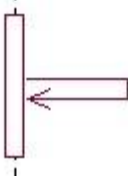
El rectángulo representa una instancia de un Objeto en particular, y la línea punteada representa las llamadas a métodos del objeto.

- **Mensaje a Otro Objeto:**



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

- **Mensaje al Mismo Objeto:**



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

En una primera fase de diseño se pueden definir clases grandes y ficticias, que representen un paquete/librería o, si el programa está compuesto por varios ejecutables corriendo a la vez, incluso clases que representen un ejecutable.

En cambio en una fase avanzada, se diseña el programa con los detalles necesarios para que los programadores tengan claro que métodos van a implementar, con sus parámetros respectivos. Incluso si es un diagrama para presentar al cliente, se puede realizar un diagrama de secuencia en el que sólo salga el actor "jugador" y una única clase "juego ajedrez" que represente el programa completo, de forma que el cliente vea qué datos y en qué orden los tiene que introducir en el programa y vea qué salidas y resultados le va a proporcionar el mismo.

## Diagramas de Colaboración

Un diagrama de colaboración es una forma alternativa al diagrama de secuencia de mostrar un escenario. Este tipo de diagrama muestra las interacciones entre objetos organizadas entorno a los objetos y los enlaces entre ellos.



## Elementos

- **Objeto:** Se representa con un rectángulo que contiene el nombre y la clase del objeto en un formato nombreObjeto: nombreClase.
- **Enlaces:** Un enlace es una instancia de una asociación en un diagrama de clases. Se representa como una línea continua que une a dos objetos, acompañada por un número que indica el orden dentro de la interacción. Pueden darse varios niveles de subíndices para indicar anidamiento de operaciones. Se pueden utilizar estereotipos para indicar si el objeto que recibe el mensaje es un atributo, un parámetro de un mensaje anterior, si es un objeto local o global.
- **Flujo de mensajes:** Expresa el envío de un mensaje. Se representa mediante una flecha dirigida cerca de un enlace.
- **Marcadores de creación y destrucción de objetos:** Puede mostrarse en la gráfica que objetos son creados y destruidos, agregando una restricción con la palabra new o delete respectivamente.
- **Objeto compuesto:** Es una representación alternativa de un objeto y sus atributos. En esta representación se muestran los objetos contenidos dentro del rectángulo que representa al objeto que los contiene. Un ejemplo es el objeto Window
- **Patrón de diseño:** Un diagrama de colaboración puede especificar un contrato entre objetos, parte esencial para la descripción de un patrón de diseño. Este diagrama contiene todos los elementos citados de un diagrama de colaboración, dejando libres posiblemente los tipos exactos de algunos objetos o con nombres genéricos para los mensajes. Una "instanciación" del patrón se representa como una elipse unida mediante flechas punteadas a los objetos o clases que participan realmente en el patrón. Estas flechas pueden tener roles, indicando cuál es el papel de cada elemento dentro del patrón. Por ejemplo, una instanciación del patrón Observer.
- **Contexto:** Un contexto es una vista de uno o más elementos dentro del modelo que colaboran en el desarrollo de una acción. Se usa para separar los demás elementos en el modelo de este problema en particular y darle énfasis. Puede mostrar sólo los detalles relevantes de las clases u objetos que contiene, para resaltar su utilidad.

Los diagramas de secuencia proporcionan una forma de ver el escenario en un orden temporal - qué pasa primero, qué pasa después -. Los clientes entienden fácilmente este tipo de diagramas, por lo que resultan útiles en las primeras fases de análisis.

Por el contrario los diagramas de colaboración proporcionan la representación principal de un escenario, ya que las colaboraciones se organizan entorno a los enlaces de unos objetos con otros. Este tipo de diagramas se utilizan más frecuentemente en la fase de diseño.

A diferencia de otras notaciones que muestran tanto el estado y el comportamiento de la clase en el diagrama de clases, UML separa el comportamiento de las clases en los diagramas de colaboración. Los diagramas de clase de UML no incluyen flujo de mensajes entre clases, es por esto que los diagramas de colaboración se deben crear en paralelo con los diagramas de clases. Aunque se puede indicar el orden del flujo de mensajes en un diagrama de colaboración numerando los mensajes, no se suele hacer, ya que para este propósito son mejores los diagramas de secuencia.



## Diagramas de Componentes

Lo que distingue a un diagrama de componentes de otros tipos de diagramas es su contenido. Normalmente contienen componentes, interfaces y relaciones entre ellos. Y como todos los diagramas, también puede contener paquetes utilizados para agrupar elementos del modelo.

Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes software, sean éstos componentes de código fuente, binarios o ejecutables. Desde el punto de vista del diagrama de componentes se tienen en consideración los requisitos relacionados con la facilidad de desarrollo, la gestión del software, la reutilización, y las restricciones impuestas por los lenguajes de programación y las herramientas utilizadas en el desarrollo.

### Elementos

- **Un componente:** es una parte modular de un sistema, cuyo comportamiento es definido por sus interfaces proporcionadas y requeridas; los funcionamientos internos del componente deben ser invisibles y su uso ambiente-independiente. Los archivos del código fuente, los Java Beans y otros artefactos que definen el sistema se pueden manifestar en componentes.
- **Un paquete:** es un espacio de nombres así como un elemento que puede estar contenido en otros espacios de nombre de paquetes. Un paquete puede poseer o combinarse con otros paquetes, y sus elementos se pueden importar dentro de un espacio de nombre de un paquete.

Un diagrama de componentes se representa como un grafo de componentes de software unidos por medio de relaciones de dependencia (generalmente de compilación). Puede mostrar también que un componente de software contiene una interfaz, es decir, la soporta.

## Diagramas de Despliegue

Es la etapa del desarrollo que describe la configuración del Sistema para su ejecución en un ambiente del mundo real.

Para el despliegue se deben tomar decisiones sobre los parámetros de la configuración, funcionamiento, asignación de recursos, distribución y concurrencia.

Un diagrama de despliegue muestra la configuración de nodos que participan en la ejecución y de los componentes que residen en ellos.

### Elementos

- **Nodos:** es un objeto físico en tiempo de ejecución que representa un recurso computacional generalmente tiene memoria y capacidad de procesamiento. Los nodos pueden contener objetos, instancias, instancias del componente.



Un nodo representa típicamente un procesador o un dispositivo sobre el que se pueden desplegar los componentes. Gráficamente un nodo se representa como un cubo 3D.

- **Componentes:** Representan el empaquetamiento físico de elementos lógicos tales como: clases, interfaces y colaboraciones, es decir representa una unidad de código (fuente, binario o ejecutable), es una parte física y reemplazable de un sistema.

Su representación es un rectángulo atravesado por dos rectángulos más pequeños a un lado.

## Diagramas de Objetos

Los diagramas de objetos modelan las instancias de elementos contenidos en los diagramas de clases. Un diagrama de objetos muestra un conjunto de objetos y sus relaciones en un momento concreto.

En UML, los diagramas de clase se utilizan para visualizar los aspectos estáticos del sistema y los diagramas de interacción se utilizan para ver los aspectos dinámicos del sistema, y constan de instancias de los elementos del diagrama de clases y mensajes enviados entre ellos. En un punto intermedio podemos situar los diagramas de objetos, que contienen un conjunto de instancias de los elementos encontrados en el diagrama de clases, representando sólo la parte estática de un interacción, consistiendo en los objetos que colaboran pero sin ninguno de los mensajes intercambiados entre ellos.

Los diagramas de objetos se emplean para modelar la vista de diseño estática o la vista de procesos estática de un sistema al igual que se hace con los diagramas de clases, pero desde la perspectiva de instancias reales o prototípicas. Esta vista sustenta principalmente los requisitos funcionales de un sistema. Los diagramas de objetos permiten modelar estructuras de datos estáticas.

En general los diagramas de objetos se utilizan para modelar estructuras de objetos, lo que implica tomar una instantánea de los objetos de un sistema en un cierto momento. Un diagrama de objetos representa una escena estática dentro de la historia representada por un diagrama de interacción. Los diagramas de objetos se utilizan para visualizar, especificar, construir y documentar la existencia de ciertas instancias en el sistema, junto a las relaciones entre ellas.

## Diagramas de Paquetes

Los Diagramas de Paquetes se usan para reflejar la organización de los paquetes y sus elementos, y para proveer una visualización de sus correspondientes nombres de espacio.

### **1.7 Seguridad en el Sistema**

Hoy en día la mayoría de las páginas web utilizan bases de datos para poder desarrollar portales dinámicos y así hacerlos más atractivos a la vez que útiles. Pero esta información que se guarda en la base de datos tiene que tener algún tipo de protección. Es por ello que algunos campos se



guardan encriptados en la base de datos, principalmente cuando una página requiere el nombre de usuario y contraseña, esta última se encripta y se guarda en la Base de datos. [18]

En PHP se utiliza la función MD5 (Message Digest 5), que es una función hash irreversible (de un sólo sentido), es decir, encripta el password tecleado por el usuario y es imposible que partiendo desde la cadena encriptada se vuelva a la contraseña origen. Por esto mismo no hay problema de que alguien pueda acceder al campo encriptado de la base de datos.

Como en la base de datos se guarda la contraseña encriptada, cuando un usuario quiere acceder, habrá que realizar una comparación entre el password que introduce encriptado en MD5, y lo que tenemos en la base de datos, (que es la contraseña encriptada en MD5), si coincide se le permite el acceso, si no, se rechaza.

MD5 se utiliza también para que cuando un usuario olvida su password, si quiere recuperar la contraseña se le pide que introduzca por ejemplo el correo, y se le envía un mail con una URL tal que si entra en ella genere una nueva contraseña que se le indica al usuario y se reescribe en md5 en la base de datos (borrando la anterior contraseña).

Hay que tener en cuenta que esto no es 100% seguro, puesto que la contraseña se encripta en el servidor, entonces al enviar la contraseña desde el cliente al servidor podría ser interceptada.

Para hacernos una idea, el algoritmo MD5 convierte el mensaje en un bloque múltiplo de 512 bits, (si hace falta añadirá bits por el final). Luego coge el primer bloque de 512 bits del mensaje y realiza diversas operaciones lógicas con los 128 bits de cuatro vectores iniciales ABCD de 32 bits cada uno. (Dichos vectores tendrán el valor inicial que nosotros queramos).

Información Es un conjunto de medidas para reducir los riesgos que tiene la información en cualquiera de sus formas.

Confidencialidad: Necesidad de que la información únicamente sea conocida por personas autorizadas.

Integridad: hace que el contenido permanezca inalterado a menos que sea modificado por personal autorizado, y esta modificación sea registrada, asegurando su precisión y confiabilidad. La integridad se obtiene agregando datos como es la huella digital.

Disponibilidad: Capacidad de la información de estar disponible para su procesamiento por las personas autorizadas.

Si alguna de estas fallas NO HAY SEGURIDAD

Las medidas: pueden ser preventivas, detectivas y correctivas.  
¿Cómo elegir esas medidas?

Criptografía: (del griego krypto, «oculto» y graphos, «escribir», literalmente «escritura oculta») es la técnica, bien sea aplicada al arte o la ciencia, que altera las representaciones lingüísticas de un mensaje.



**Criptoanálisis:** Se encarga de encontrar sistemas para descifrar la información que se transmiten a través de un medio. Considera un conjunto de técnicas que intentan encontrar la clave utilizada entre dos comunicaciones.

Las bases fundamentales de la criptografía, fueron establecidos por Claude Shannon: En un artículo titulado: A mathematical theory of communication, donde introduce las bases de la teoría de la información.

Whitfield Diffie: actualmente ingeniero de Sun Microsystem, fue el descubridor en 1975 del concepto de criptografía de clave pública.

**Mensaje:** Son los datos que actualmente con conciernen.  
Normalmente será un texto plano. Denotado por M

**Mensaje Cifrado:** Es el mensaje M encriptado. Llamamos al mensaje C

**Espacio de Mensajes:** Son todos los mensajes posibles, en la mayoría de los casos suele ser infinito, pero en otros la longitud puede estar limitada.

**Espacio de Mensajes cifrado:** Lo mismo que el punto anterior pero para mensajes cifrados.

**Espacio de claves:** Son el conjunto de todas las claves posibles. Se representa por la letra K (mayúscula, la k minúscula representa una clave en concreta)

La encriptación no es más que una aplicación desde el dominio de M al rango de C. Y la descryptación es la función inversa.

Encriptar se representa como  $C = E(M)$

Desencriptar se representa por  $M = D(C)$

Por lo que para recuperar el mensaje ,  $M=D(E(M))$

Si asignamos a cada letra un número (A =00, B =01, C=02,.....Z=25), y consideramos un alfabeto de 26 letras, la transformación criptográfica en términos matemáticos se puede explicar bajo la siguiente fórmula de congruencias:

$$C = (M + 3 \text{ mod } 26)$$

Obsérvese que este algoritmo ni siquiera posee clave, puesto que la transformación siempre es la misma. Obviamente, para descifrar basta con restar 3 al número de orden de las letras de criptograma.

Reglas de Kerckhoffs aceptadas mundialmente:

No debe existir ninguna forma de recuperar el texto claro a partir del criptograma (seguridad ante el primer ataque).

Todo sistema criptográfico debe estar compuesto por dos tipos de información:

a. Pública: se refiere a la familia de algoritmos que definen el sistema criptográfico. b. Privada: es conocida sólo por el usuario. La clave de cifrado de cada usuario en particular.

## CAPITULO II. Estado del Campo o del Arte.

### 2.1 Estado del Arte.

*Las organizaciones dependen de sus clientes, por lo que deben comprender las necesidades actuales y futuras de los clientes, satisfacer sus requerimientos y esforzarse en entender las expectativas de los clientes. Y mejorar su eficiencia y eficacia para no perder de vista su objetivo.*

**Dpointgroup Servicios profesionales** especializados en **congresos, ferias, espectáculos, deportes, eventos** y todas aquellas disciplinas en donde se requiera de personal capacitado así como también equipamientos e instrumental diseñados para instalaciones temporarias o definitivas. **Alquilamos y vendemos material audiovisual, pantallas de cine hinchables, pantallas de proyeccion, proyectores, etc.**

#### Servicio Proyecciones

La función de nuestro equipo son las proyecciones visuales en directo, sobre **pantallas u otros elementos creando una ambientación visual acorde, a tiempo y a ritmo, con la música.**



Mezclamos video a tiempo real creando una sesión adaptada a muy diferentes estilos de música, sesiones realizadas temáticamente a la medida del cliente o evento.



También nos encargamos de instalar **pantallas, proyectores de video y diapositivas** si es

requerido, creando así una decoración.

Salones y recepciones, residencias, hoteles para eventos, barcos, salones para conferencias, reuniones, lanzamientos de productos, exposiciones. Restaurantes, discotecas, pubs, bares.

Todas las necesidades para su evento, tenemos la solución para cada medida. Conozca algunos de nuestro catalogo de **Salas y Catering**.

## Comparación UNO

- <http://www.dpointgroup.com/servicios.html>
- **Dpointgroup Servicios profesionales especializados en congresos, ferias, espectáculos, deportes, eventos.**
- **Dirige sus servicios a clientes que desean realizar un evento social de grandes magnitudes, incluyendo los medios audiovisuales y catering**

Grupo de AudioVisuales - Salas Multimedia - Mozilla Firefox

ManualUsu.pdf (application/pdf Objeto) x Grupo de AudioVisuales - Salas Multimedia x París - La Oreja de Van Gogh. - YouTube x +

http://chaplín.urjc.es/z\_files/sj\_comu/aj04/SalasMultimedia.html

Universidad Rey Juan Carlos Comunidad Universitaria. Pas.

Salas Multimedia

**Introducción**  
En esta sección se describirán con detalle cada una de las salas multimedia con las que cuenta la Universidad Rey Juan Carlos.

**Presentación de las salas**  
Actualmente la Universidad Rey Juan Carlos dispone de las siguientes salas de audiovisuales:

- **Sala 1. Sala Audiovisual - Laboratorio II - ESCET**  
Sala destinada fundamentalmente a TeleDocencia, dentro del proyecto ADAMADRID.  
Localizada en el campus de Móstoles en el sótano del edificio de Laboratorios II. Tiene una capacidad aproximada de entre 30 y 40 personas.  
Está equipada con VideoProyector, Cámara de Documentos, PC para presentaciones, y cuenta con una cabina de producción con Reproductores VHS, CD y DVD, equipos de VideoConferencia, mesas de mezclas de audio y video, así como posibilidad de retransmitir los eventos realizados por streaming. En cuanto al sonido, la sala cuenta micrófonos inalámbricos y de sobremesa, que cubren las necesidades de la misma.
- **Sala 2. Sala de Gobierno - RECTORADO**  
Sala destinada fundamentalmente a reuniones y presentaciones para un número pequeño de personas.  
Localizada en en la 6ª Planta del Rectorado. Ver localización y planos.  
Tiene una capacidad aproximada de entre 40 y 45 personas.  
Está equipada con VideoProyector, Cámara de Documentos, Reproductores VHS, CD y DVD, Televisor, equipo de VideoConferencia. Cada puesto cuenta con un micrófono de sobremesa conectado al sistema de megafonía de la Sala.
- **Sala 3. Sala de Juntas - ESCET**  
Sala destinada fundamentalmente a reuniones y presentaciones para un número pequeño de personas.  
Localizada en en la 1ª Planta del Edificio de Gestión del Campus de Móstoles. Ver

Inicio 01:24:11 Grupo de AudioVisu... Descargas JUL&T@ (F) Documento1 - Micro... ES 06:48 p.m.

## Comparación Dos

- [http://www.chaplin.urjc.es/z\\_files/aj\\_comu/aj04/SalasMultimedia.html](http://www.chaplin.urjc.es/z_files/aj_comu/aj04/SalasMultimedia.html)
- UNIVERSIDAD Rey Juan Carlos
- Proporciona el reservado de sala para videoconferencia.
- Dirigido a los estudiantes y docentes dentro de la institución.



## Comparación Tres

- <http://www.cib.espol.edu.ec/sevicios/audioVisuales.aspx>
- Centro de Información bibliotecario
- Se pone a disposición de la comunidad politécnica las salas audiovisuales
- Permite el reservado a través de administración de las salas.

## SISTEMA DE MEDIOS AUDIOVISUALES

- El sistema permite el reservado de MEDIOS AUDIOVISUALES
- El objetivos es:  
*Facilitar a los usuarios el reservado, acceso y disposición de equipo audiovisual, infraestructura y servicios, haciendo uso de las Tecnologías de la información.*
- El sistema se puede ofrecer a toda Organización Empresarial o Institución, que cuente con medios audiovisuales y necesite, con el personal que labora o está adscrito, realizar conferencias, presentaciones o clases.
- El personal involucrado en el sistema; son: los empleados y administradores; o bien, estudiantes, docentes y administradores, según la organización a la que se refiera.
- El sistema permite:
  - Mostrar el catálogo de equipo servicios e infraestructura:
  - Realizar reservado de equipo, servicios e infraestructura, en la hora y fecha deseados.
- Permite confirmar el reservado por apartado, o indicar si se cancela.
- Emitir el préstamo de reservado.

### Por que recomiendo el sistema:

Porque es un software gratuito, agiliza el proceso de reservado, proporciona el reservado de coffee break previa autorización de él departamento o área, hace uso del servicio web.

The screenshot displays two views of the web application. The top view is the 'Consulta de Catálogo de Equipo' (Equipment Catalog Query) page. It features a navigation menu with options like 'Inicio', 'Consulta', 'Acceso', and 'Salir'. A sidebar on the left allows filtering by equipment type, with 'Proyector' selected. The main content area contains a table with the following data:

ID	EQUIPO	DESCRIPCION	NO. INVENTARIO	TIPO DE EQUIPO	ESTADO	CAPACIDAD	MODELO	UBICACION
10	proyector	acetatos	0	proyector	completo			
12	proyector	fijo	2345	projectores	completo			
22	proyector		4567		funcional	1200 lúme	Epson S1	0

The bottom view is the 'Acceso' (Access/Login) page. It includes a navigation menu and a login form with the following fields and options:

- Clave y contraseña (password field)
- Tipo de Usuario:  Usuario  Administrador
- Login: andres
- login Password: [masked]
- Ingresar button

At the bottom of the page, there is contact information: Telefonos: (222) 2 43 30 25, E-mail: secracad@hotmail.com.mx, and a navigation menu with links: home, salir, buap, hotmail, contacto.

## Capítulo III. Diseño del Sistema

### 3.1 Descripción de la información

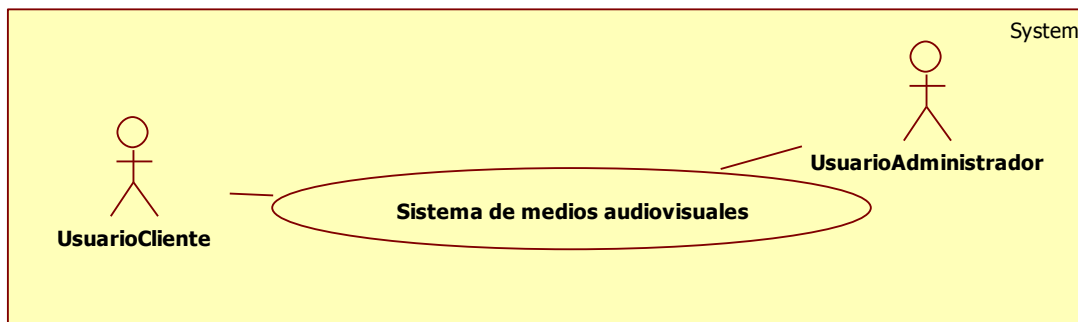
Se desarrolló un producto de software, implementando un sistema que permite el reservado de equipo audiovisual, infraestructura, servicios y administración de equipo.

#### ANÁLISIS DE CASOS DE USO

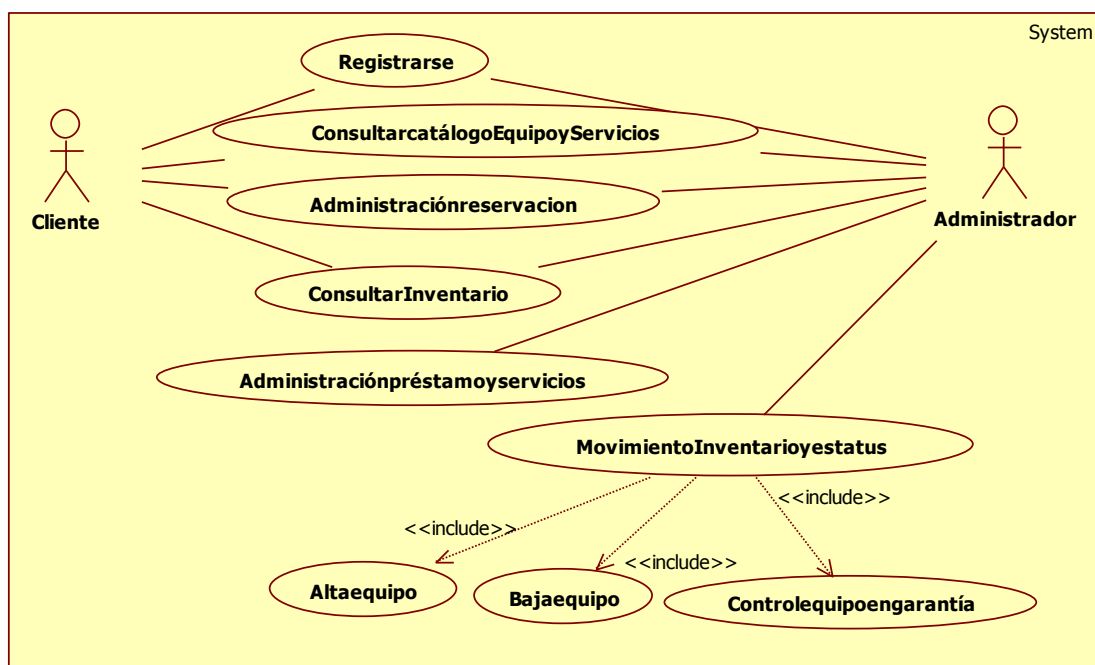
Permite identificar los límites, controles y entidades de los casos de uso del sistema.

El sistema se representó mediante los siguientes casos de uso:

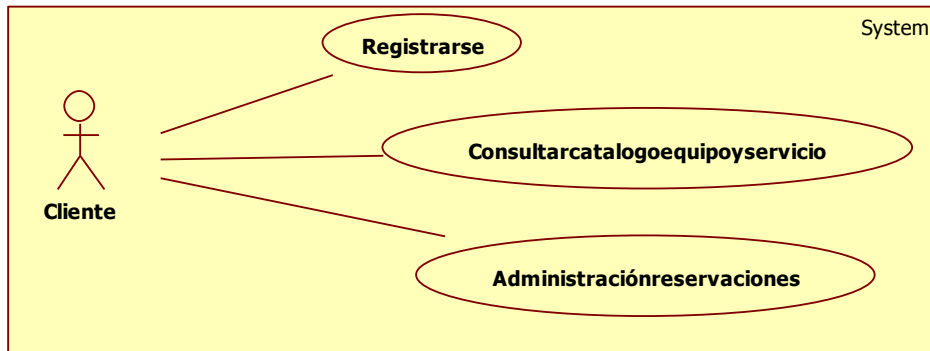
##### Caso de uso del sistema



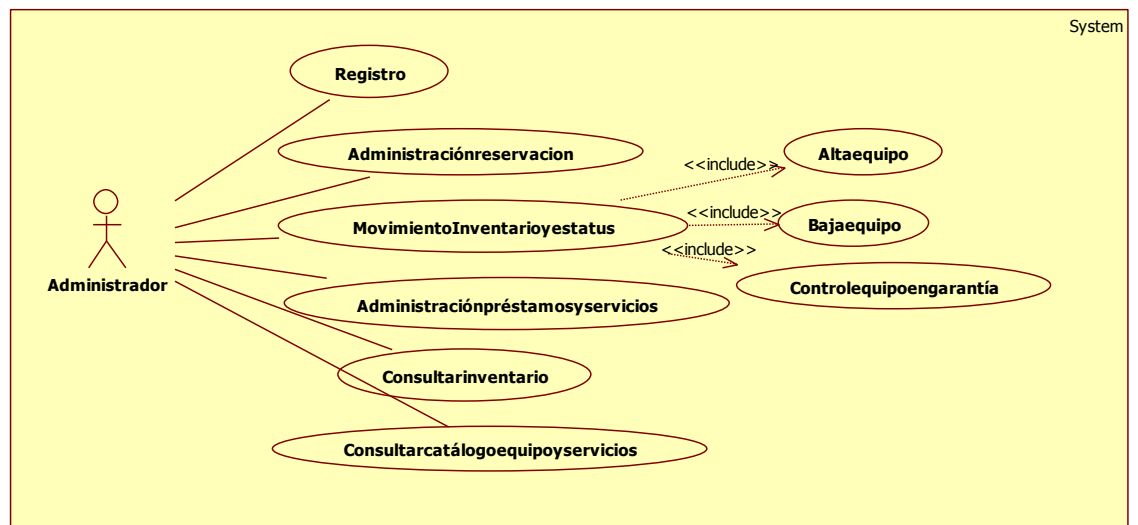
##### Casos de uso dentro del sistema



### Casos de uso del usuario cliente



### Casos de uso del usuario administrador



### DOCUMENTACION DE CASOS DE USO

#### Descripción de los casos de uso

<b>Caso de uso:</b>	Registrarse
<b>Participantes:</b>	Cliente, Administrador
<b>Tipo</b>	Primario
<b>Descripción:</b>	El usuario, para acceder al sistema, necesita ser personal de la empresa o institución, ya que al registrarse, el sistema le solicita su clave de personal y contraseña para validarlo y usar el sistema.



*Curso normal de eventos*

Acción del actor	Respuesta del sistema
1. Este caso de uso comienza cuando el cliente necesita del servicio de algún medio audiovisual y accede a el.	
	2. El sistema le muestra la pantalla, solicitándole que escriba sus datos para validar
3.El usuario ingresa los datos	
	4.El sistema los valida y le permite el uso de el
5.El usuario puede seleccionar varias opciones de medios audiovisuales, una vez que ha sido validado por el sistema. Y sale del sistema	

*Simplificaciones, metas y suposiciones.*

*Curso alterno:*

*Descripción de los casos de uso*

<b>Caso de uso:</b>	Consultar catálogo equipo y servicios
<b>Participantes:</b>	Cliente,Administrador
<b>Tipo</b>	Primario
<b>Descripción:</b>	Un usuario accede al sistema para tener conocimiento de el equipo que se puede utilizar y los servicios. El sistema acepta el registro del usuario y le permite acceder a la visualización del equipo y servicios.

*Curso normal de eventos*

Acción del actor	Respuesta del sistema
1.Este caso de uso comienza cuando el cliente necesita del servicio de algún medio audiovisual y se ha registrado en el sistema, elige la opción de consulta.	
	2. El sistema le muestra la lista de equipo y servicio que se proporciona.
3. El usuario observa la lista que se proporciona y sale de la opción	

*Curso alterno:*

Si no existe el equipo que requiere, deberá informarlo a el Usuario Administrador.

Si está el equipo que requiere debe revisar que se encuentre disponible.

*Descripción de los casos de uso*

<b>Caso de uso:</b>	Administración de reservaciones
<b>Participantes:</b>	Cliente, Administrador
<b>Tipo</b>	Primario
<b>Descripción:</b>	El usuario accede al sistema para realizar el reservado



	de equipo en la fecha y hora requerida, siempre y cuando exista y este disponible, así como servicios.
--	--

Curso normal de eventos

Acción del actor	Respuesta del sistema
1. El usuario ya validado por el sistema, ingresa a el para reservar el equipo o servicio necesario.	
	2. El sistema le pide la fecha y hora del reservado, para determinar que no este ocupado el equipo. En servicios se pide seleccionar el tipo y cantidad requerida.
3. El usuario confirma el reservado	
	4. El sistema hace el reservado conforme las necesidades
5. El usuario sale del sistema	

*Curso alterno:*

Si no hay equipo disponible, deberá regresar mas tarde u otro día.

*Descripción de los casos de uso*

<b>Caso de uso:</b>	Administración de préstamos de equipo y servicios
<b>Participantes:</b>	Cliente, Administrador
<b>Tipo</b>	Primario
<b>Descripción:</b>	El usuario accede al sistema para establecer los préstamos realizados.

Curso normal de eventos

Acción del actor	Respuesta del sistema
1. El usuario ya validado por el sistema, ingresa a el para activar o desactivar el préstamo.	
	2. El sistema le pide la fecha, hora, estado y a quién se realiza el préstamo.
3. El usuario inserta los datos.	
	4. El sistema almacena los datos.
5. El usuario sale del sistema	

*Curso alterno:*

Si el usuario no entrega su identificación, no podrá efectuarse el préstamo.

*Descripción de los casos de uso*



<b>Caso de uso:</b>	Consultar inventario
<b>Participantes:</b>	Administrador
<b>Tipo</b>	Primario
<b>Descripción:</b>	Permite que el usuario verifique la cantidad existente de equipo y servicios.

Curso normal de eventos

Acción del actor	Respuesta del sistema
1. El usuario ya validado por el sistema, ingresa a el para saber lo que se tienen en existencia.	
	2. El sistema le muestra los datos en existencia.
3. El usuario los visualiza y sale del sistema.	

*Curso alterno:*

**CASO DE USO:** Movimiento de inventario y estatus.

**ACTOR:** Administrador

**1.- DESCRIPCION:** A TRAVEZ DE ESTE CASO DE USO, EL SISTEMA LE PERMITE AL USUARIO AGREGAR, ELIMINAR y establecer el estado en que se encuentra EL EQUIPO E INFRAESTRUCTURA CON QUE SE CUENTA

**2.- FLUJO DE EVENTOS**

**2.1 PRE-CONDICION**

2.1.1 EL USUARIO YA DEBE HABERSE REGISTRADO EN EL SISTEMA PARA SER ACEPTADO

**2.2 FLUJO PRINCIPAL**

2.2.1 El sistema permite ACTUALIZAR EQUIPO y DEBE SELECCIONAR

AGREGAR 2.3.1

ELIMINAR 2.3.2

Estatus 2.3.3

2.2.1.2 EL USUARIO selecciona una opción

2.2.4 SALIR DE este caso de uso

**2.3 FLUJOS ALTERNO**

**2.3.1 AGREGAR EQUIPO**

2.3.1.1 EL USUARIO INGRESA LOS DATOS DEL NUEVO EQUIPO

2.3.1.2. CONTINUA EL FLUJO 2.2.3

**2.3.2 ELIMINAR EQUIPO**

2.3.2.1 EL USUARIO DA DE BAJA EL EQUIPO OBSOLETO O EN MAL ESTADO

2.3.2.2 CONTINUA EL FLUJO 2.2.3

**2.3.3 ESTATUS EQUIPO**

2.3.3.1 EL USUARIO indica el estado en que se encuentra el equipo

2.3.3.2 CONTINUA EL FLUJO 2.2.3



## **ESCENARIOS**

### **ESCENARIO 1: Reservar EQUIPO**

- CLAUDIA YA ELIGIO EL CASO DE USO Administración de reservaciones Y EL SISTEMA le pide que ingrese la fecha, hora y el equipo a utilizar.
- Claudia ingresa 25-October-11 de las 16:00-18:00 hrs con cañón, laptop, extensión y cortinas.
- El sistema verifica que haya disponible y le indica que si lo puede apartar.
- Claudia confirma el reservado y el sistema la acepta.
- Y SALE DEL CASO DE USO

### **ESCENARIO 2: Consultar catálogo equipo y servicios**

- . Alma entra a la página del sistema y se registra.
- . El sistema le muestra a Alma el menú para que consulte el equipo con el que se cuenta.
- . Alma entra a la opción, consultar catálogo de equipo.
- . El sistema muestra una lista de equipo.
- . Alma revisa el contenido y sale del caso de uso

### **ESCENARIO 3: Movimientos de inventario y estatus**

- Angel accede a el sistema como administrador y el sistema le solicita registrarse
- Angel escribe su registro y contraseña de empleado y el sistema le permite el acceso a administrar el equipo
- El sistema permite añadir, eliminar y establecer el estatus del equipo
- Angel desea agregar un nuevo equipo adquirido y elige la opción añadir
- El sistema muestra los campos de datos, para agregar los nuevos datos.
- Angel agrega los datos y visualiza la opción de consulta de equipo.

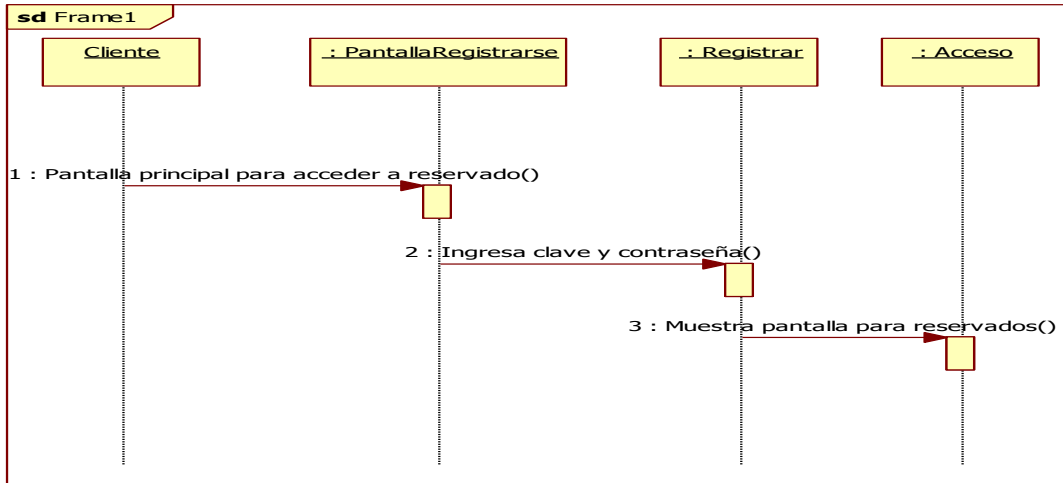
### **ESCENARIO 4: Registrarse**

- Angel accede al sistema
- El sistema le indica que se registre ingresando su clave y contraseña
- Angel escribe su clave 23456, y contraseña Angelito
- El sistema verifica que sean correctos y los acepta
- Angel ya entro a el sistema

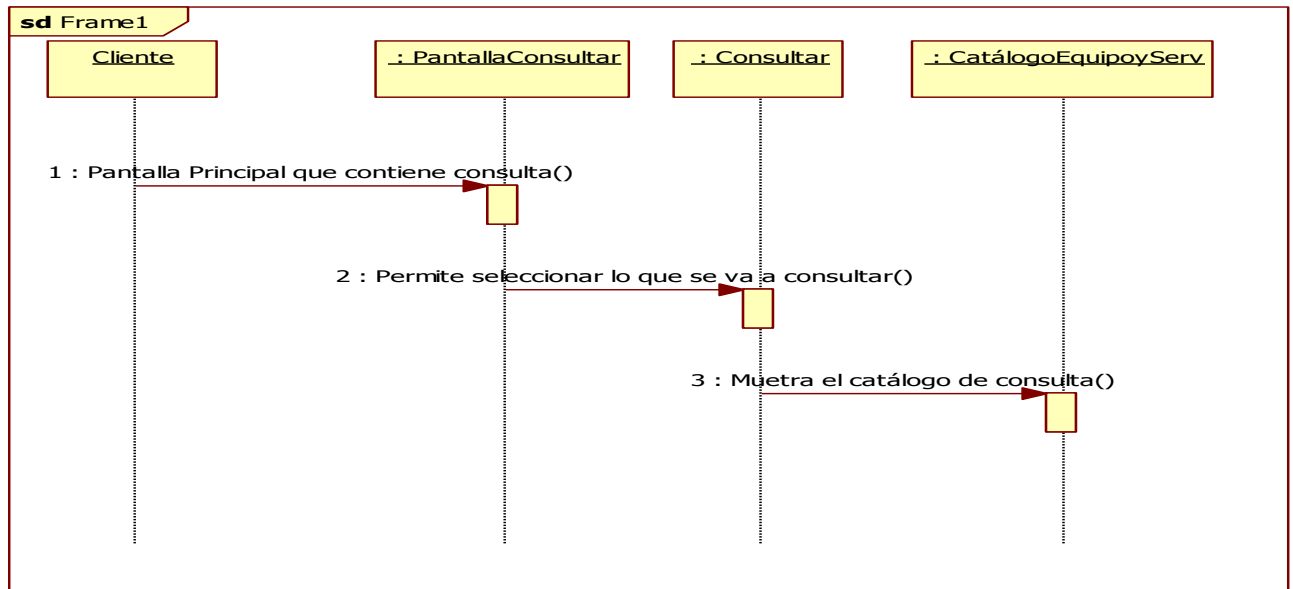
### 3.2 Representación del Flujo de Información

#### Diagramas de Secuencia:

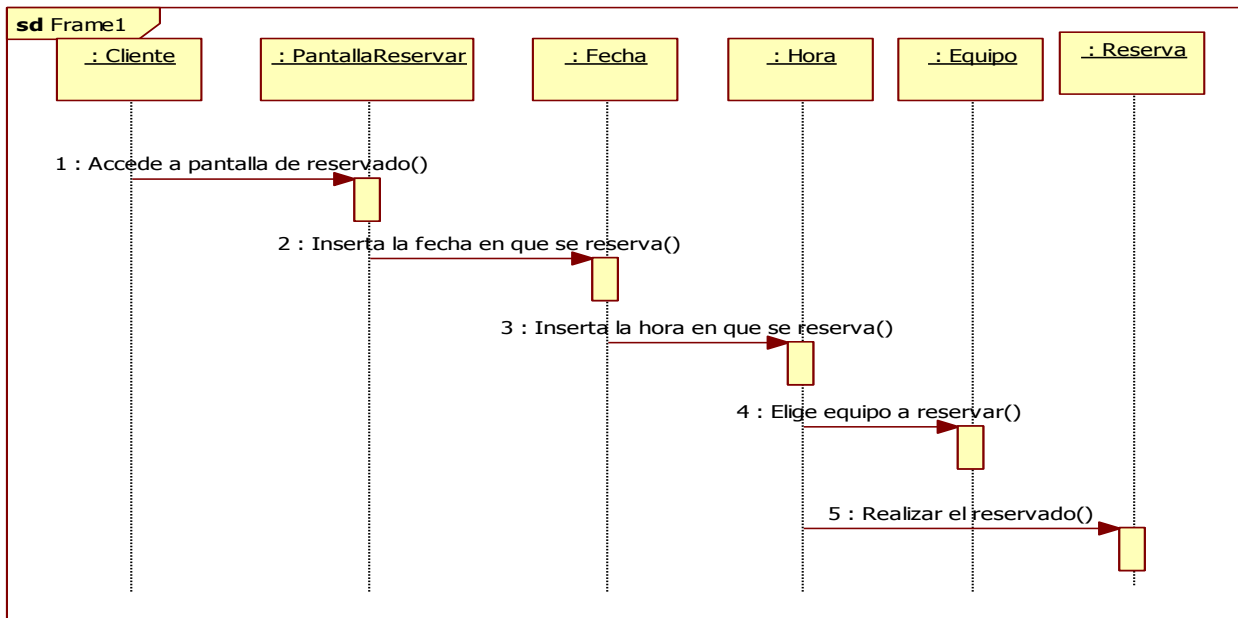
Secuencia de registro del cliente:



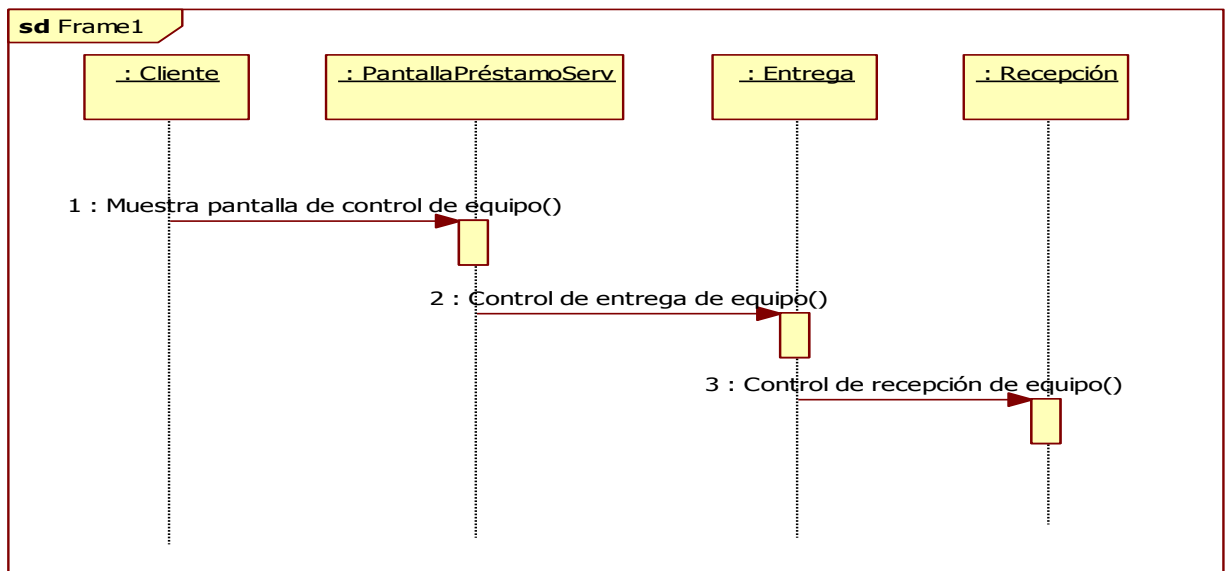
Secuencia de consulta de catálogo de equipo:



Secuencia de reservado de equipo:

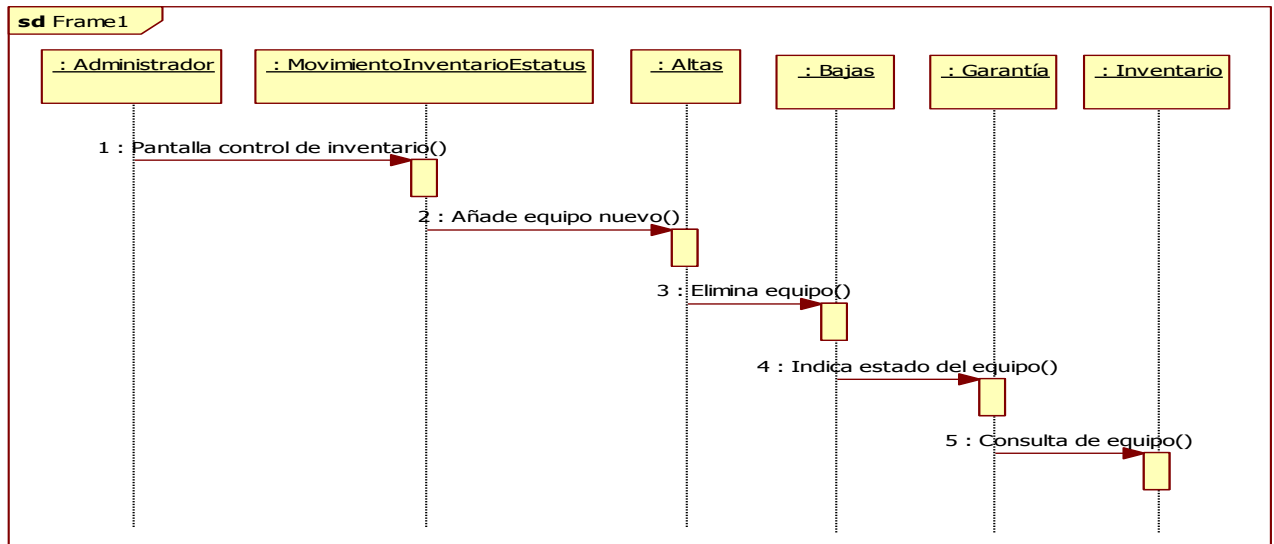


Secuencia de préstamo de equipo

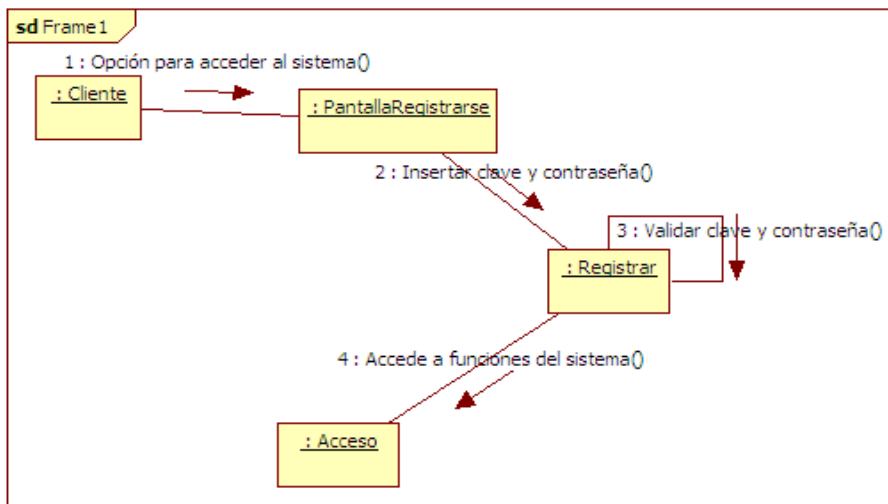




Secuencia de funciones del administrador.

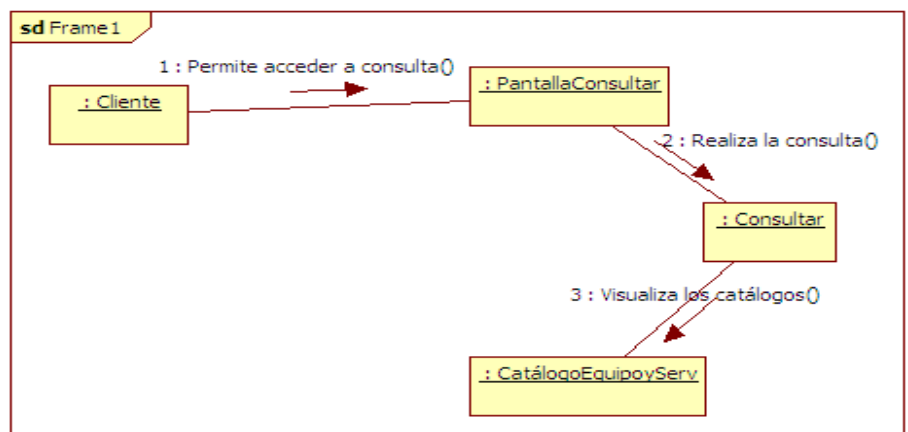


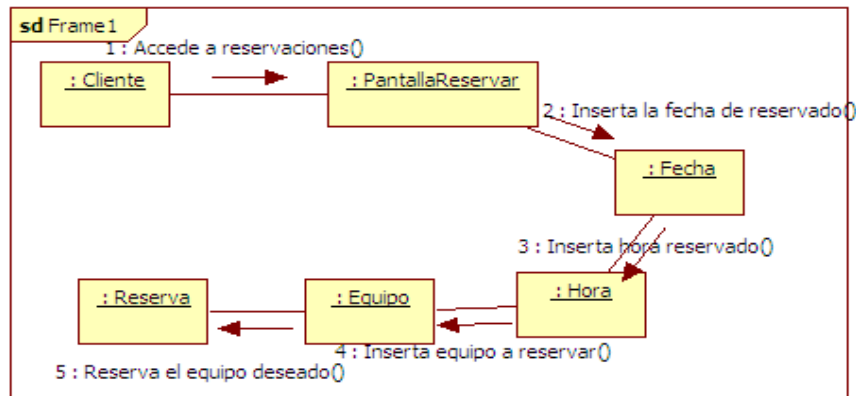
## Diagramas de colaboración



Muestra como es el flujo de datos para registrarse.

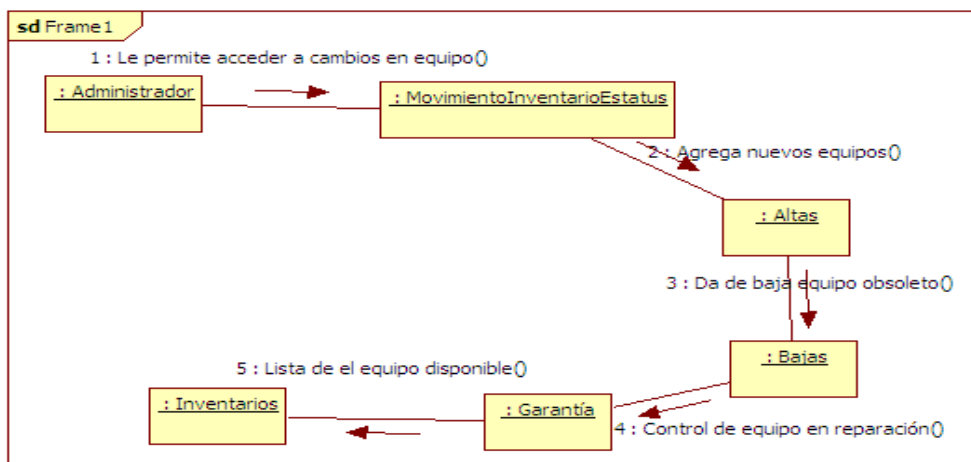
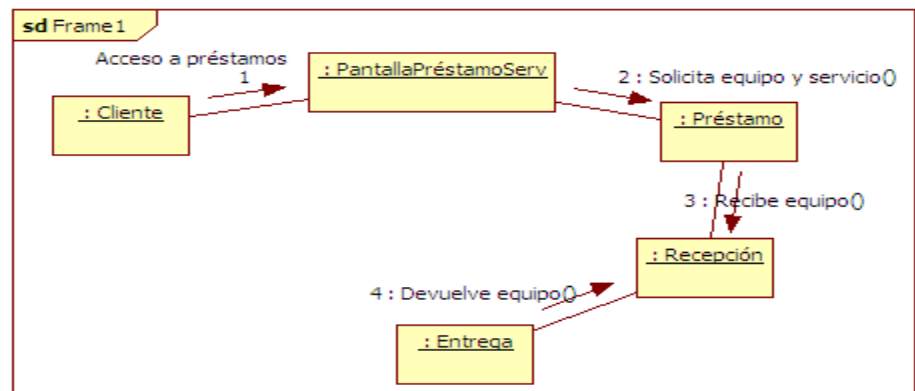
Muestra el flujo de datos para consultar el catálogo de equipo





Muestra el flujo de datos para realizar reservaciones de equipo.

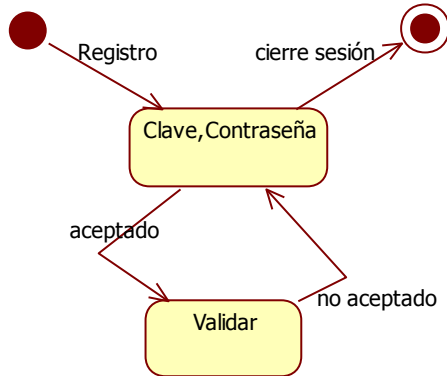
Muestra el flujo de datos para realizar préstamo de equipo.



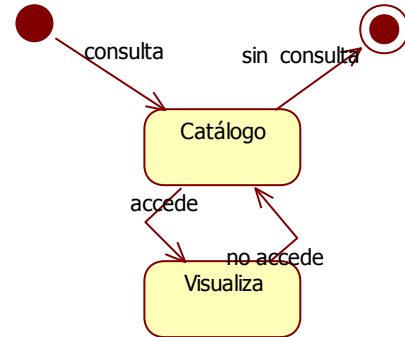
Muestra el flujo de datos para control de equipo y sistema.

## Diagramas de Estado

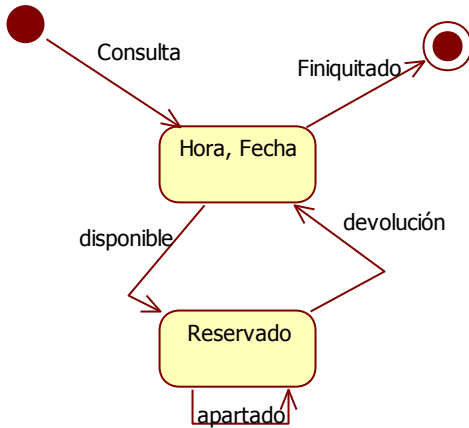
Permite registrarse si cuenta con clave y contraseña.



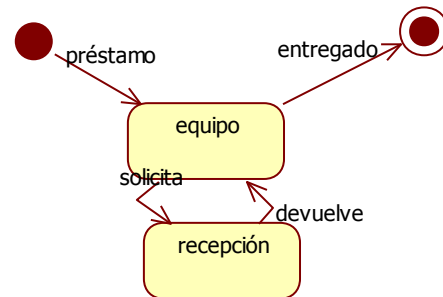
Muestra el catálogo de equipo por Equipo.



Reserva equipo por fecha y hora, siempre y Cuando se encuentre disponible.



Presta el equipo reservado con Previa identificación.

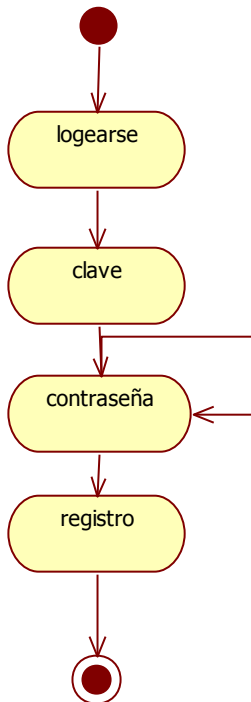


Realiza movimientos de equipo, usuario y reservados por parte del administrador.

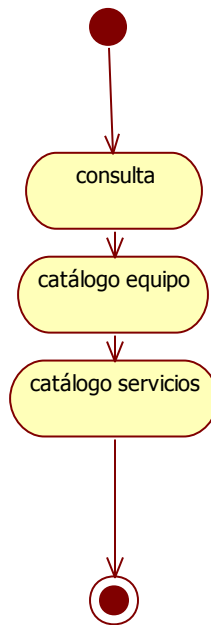


## Diagramas de actividad

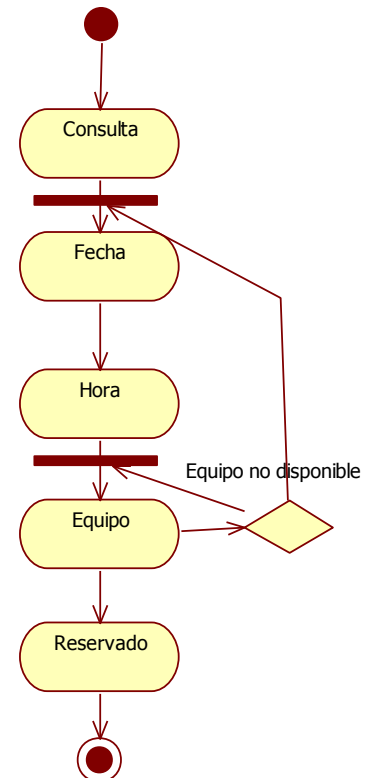
Permite registrarse si cuenta con clave y contraseña.



Muestra el catálogo de equipo  
Por equipo.

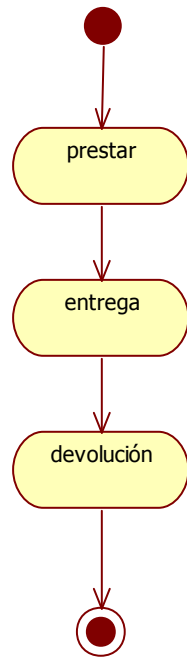


Reserva equipo por  
fecha y hora, si  
esta disponible.

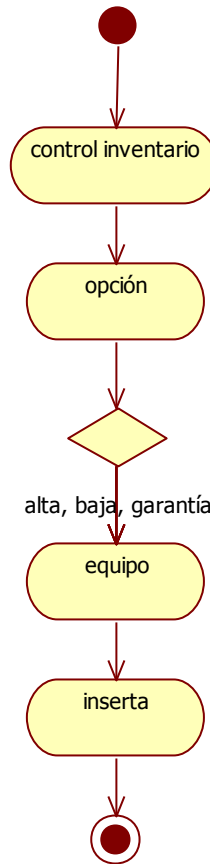




Se lleva el control de préstamo y devolución de equipo.

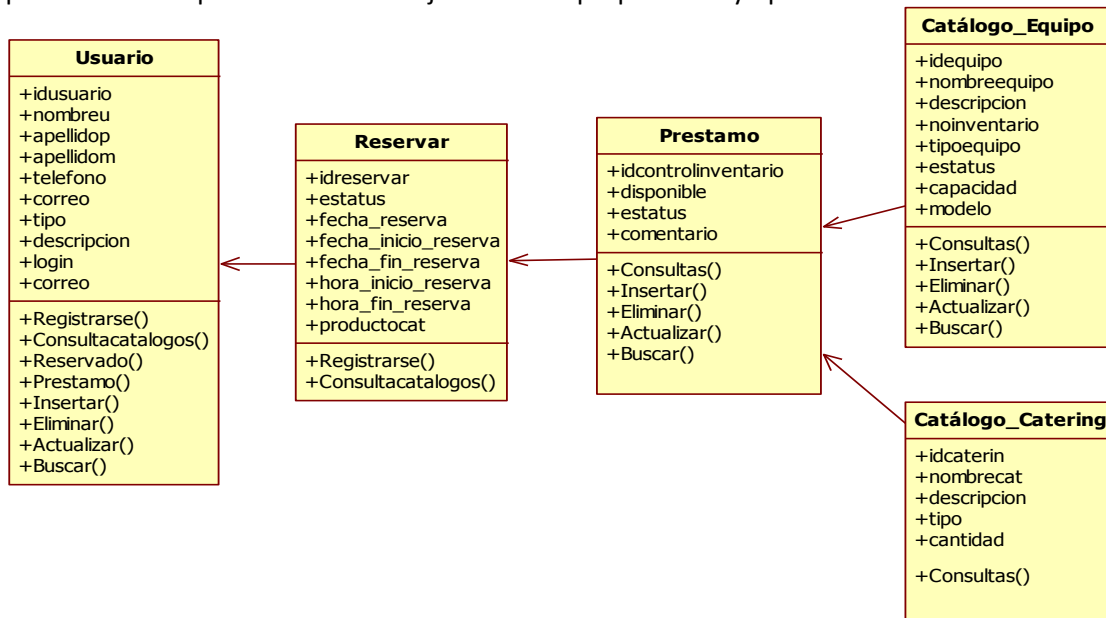


Realiza movimientos a el equipo, servicios y usuarios.



## DIAGRAMA DE CLASES

Representa el comportamiento del objeto con sus propiedades y operaciones.



## Diccionario del modelo conceptual

Define la función de cada objeto.

NOMBRE: Registrarse

Definición del Trabajo: Es el control de acceso a el sistema para poder utilizarlo

NOMBRE: UsuarioCliente

Definición del Trabajo: Información de una persona que interactuará con el sistema para reservar el equipo o servicio a utilizar

NOMBRE: UsuarioAdministrador

Definición del Trabajo: Información de la persona que administrará el sistema, controlando el equipo existente, los reservados y servicios.

NOMBRE: Administración de reservaciones.

Definición del Trabajo: Permite seleccionar el equipo a utilizar y apartarlo.

NOMBRE: ConsultarCatálogoEquipo y servicios

Definición del Trabajo: Lista del equipo en existencia en buenas condiciones para su uso y los servicios que se proporcionan.

NOMBRE: Movimiento de inventario y estatus.

Definición del Trabajo: Función ejercida por UsuarioAdmon, para controlar el estatus del equipo y su existencia.

NOMBRE: Consulta inventario.

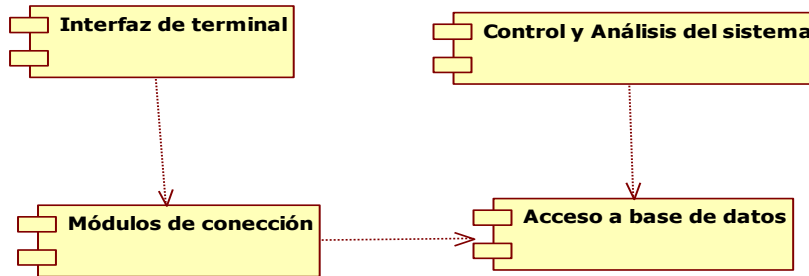
Definición del Trabajo: Función ejercida por UsuarioAdmon, para saber el equipo existente.

NOMBRE: Administración de préstamos

Definición del Trabajo: Función ejercida por UsuarioAdmon, al momento de entregar y recibir el equipo.

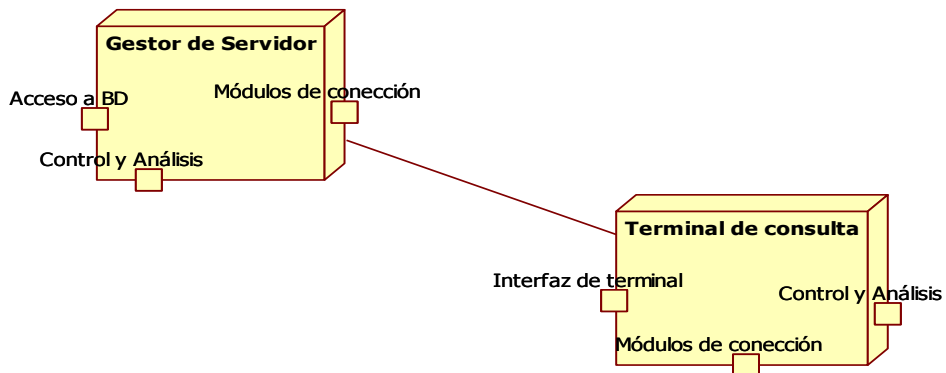
## Diagrama de Componentes

Representa lo que forma al sistema.



## Diagrama de Despliegue

Representa las conexiones entre servidor, base de datos y módulos del sistema.



## Diagrama Entidad-Relación

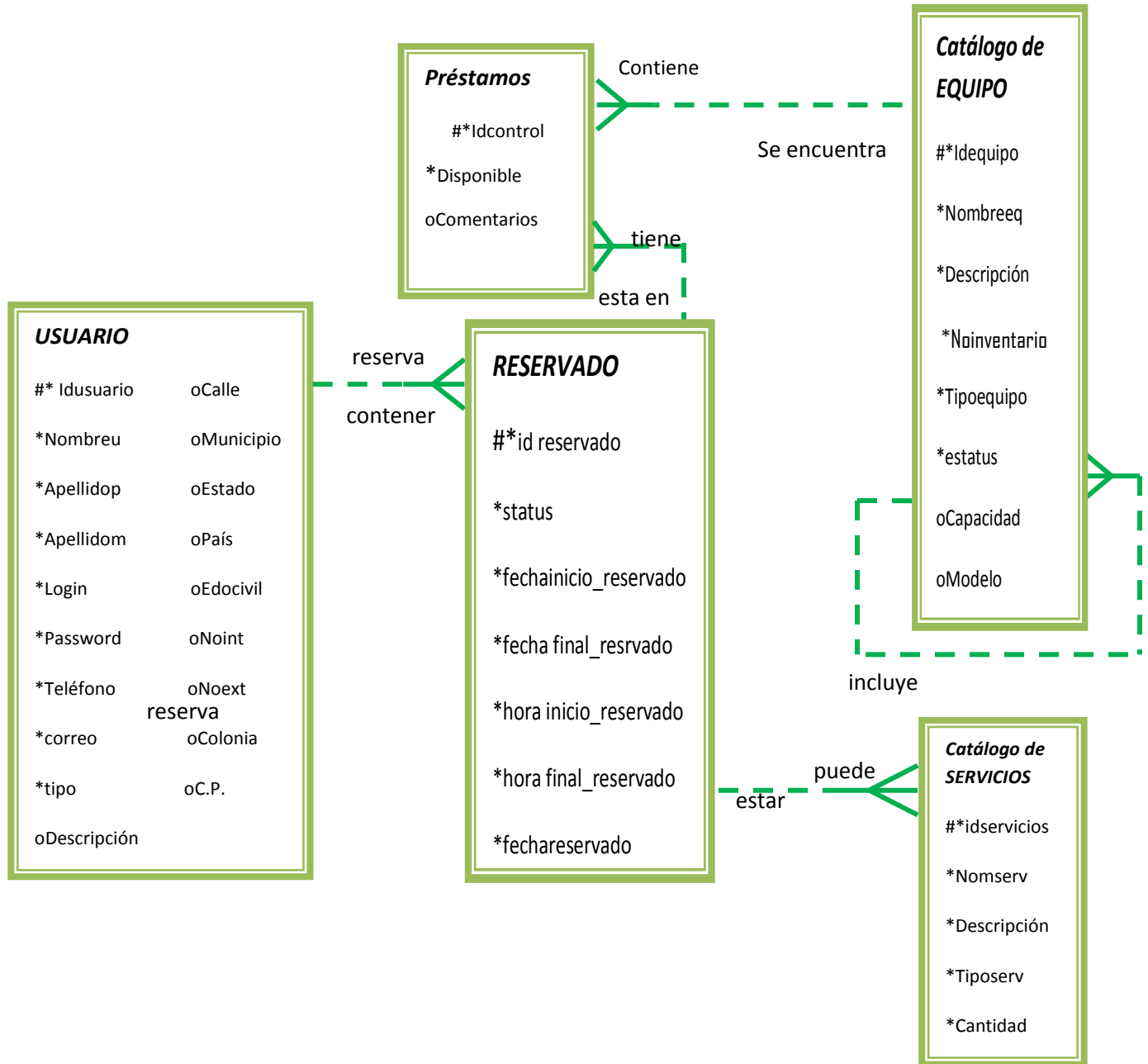


Diagrama E-R que representa el conjunto de entidades de un sistema y como se relacionan.

### 3.3 Partición Funcional

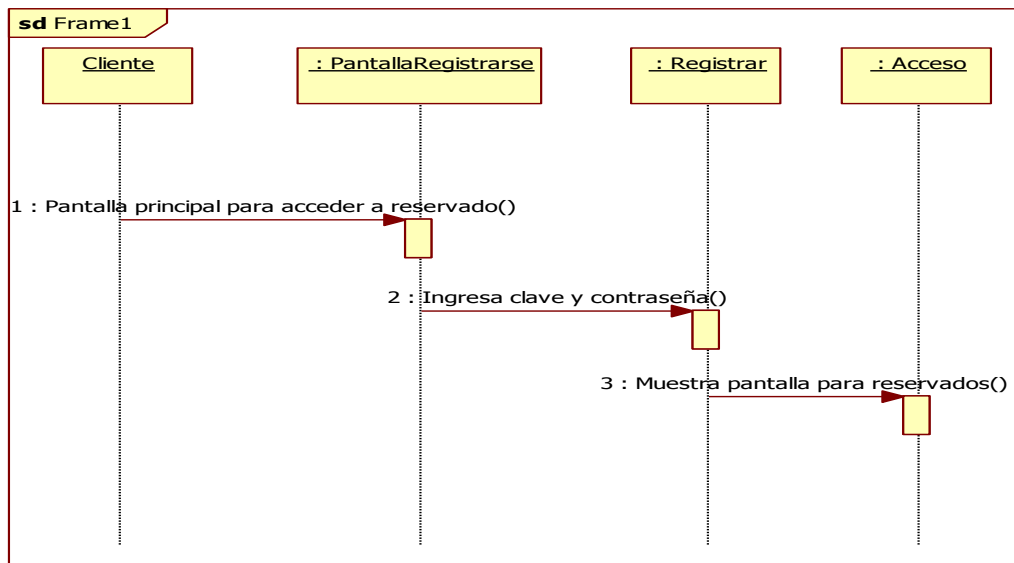
El sistema está conformado por los siguiente módulos:

- Registrarse
- Consultas
- Reservados
- Préstamo
- Control Inventario

#### Descripción funcional

##### Narración del procesamiento

Se muestra el flujo de trabajo del módulo REGISTRARSE.



#### Restricciones y limitaciones

El usuario no podrá pasar a otro módulo, hasta que después de registrarse, haya sido validado por un moderador.

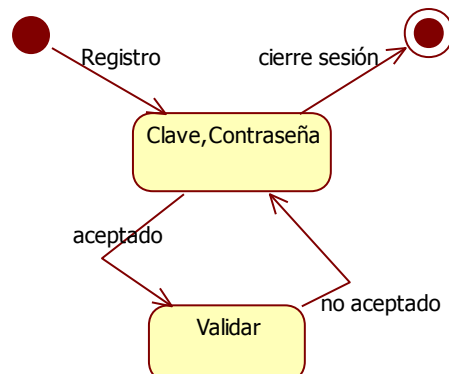
#### Requerimientos de Funcionamiento

Es necesario, que el usuario que hará las reservaciones se registre insertando su login y password, con su tipo de usuario.

#### Restricciones en Diseño

Solo si anexa su login y password hará uso de los demás módulos.

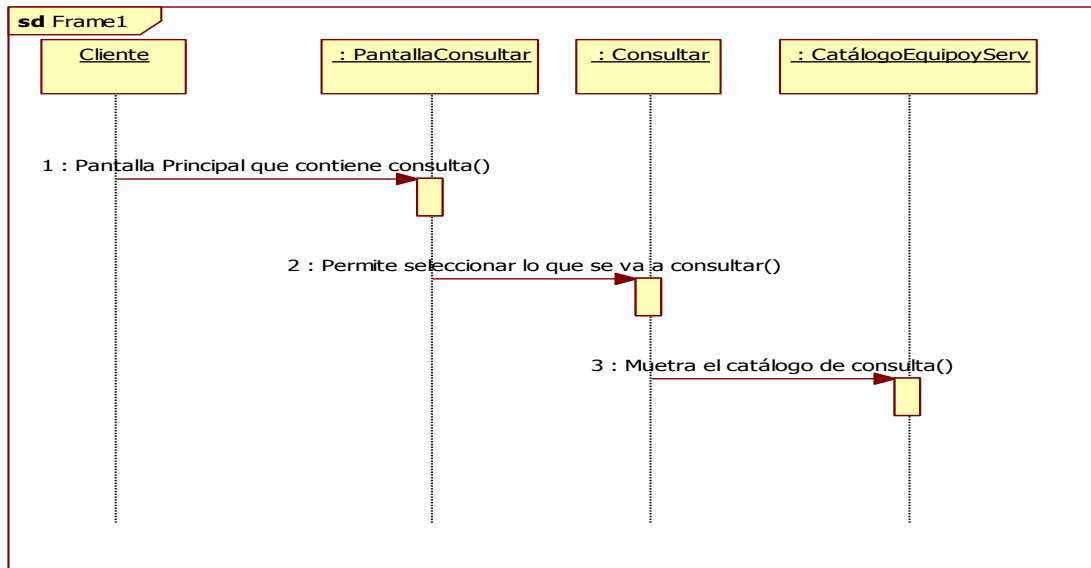
#### Diagrama



## Descripción funcional

### Narración del procesamiento

Se muestra el flujo de trabajo del módulo CONSULTAS.



### Restricciones y limitaciones

El usuario podrá visualizar el catálogo del equipo y servicios, únicamente para consulta, seleccionando el tipo de equipo que desea ver.

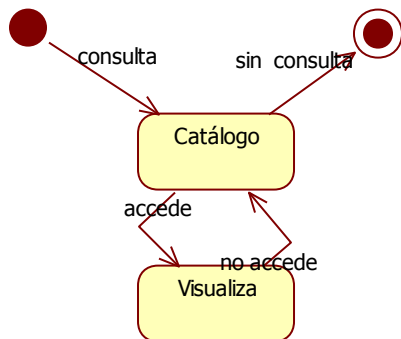
### Requerimientos de Funcionamiento

Los usuarios que hacen uso del sistema pueden ver la lista de los catálogos en cualquier momento, simplemente accediendo a la página.

### Restricciones en Diseño

El usuario puede realizar consultas del catálogo por:  
Nombre de Equipo, Area de equipo

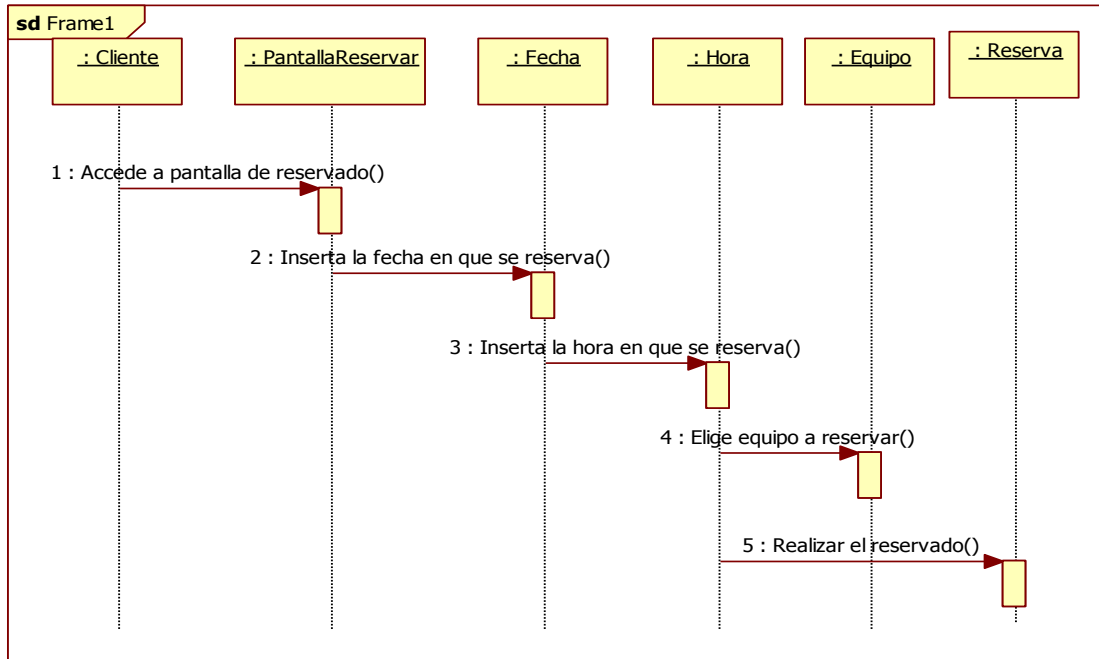
### Diagrama



## Descripción funcional

### Narración del procesamiento

Se muestra el flujo de trabajo del módulo RESERVACIONES.



### Restricciones y limitaciones

El usuario para hacer reservaciones de equipo, debe haberse registrado, insertando su login y password para ser validado por el sistema.

### Requerimientos de Funcionamiento

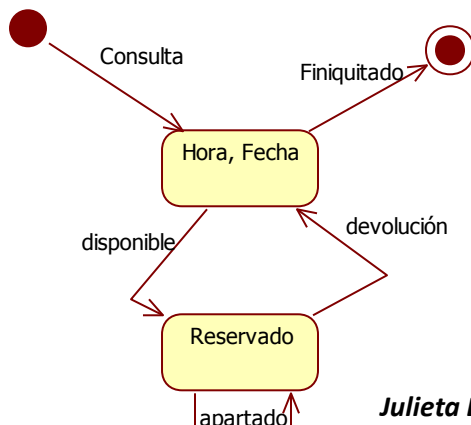
Es necesario que el usuario inserte la fecha, hora, numero de usuario y el equipo que desea reservar.

El sistema verifica que el equipo elegido se encuentre disponible en la fecha y hora para realizar el apartado.

### Restricciones en Diseño

El usuario no podrá hacer reservaciones si omitió alguno de los datos descritos.

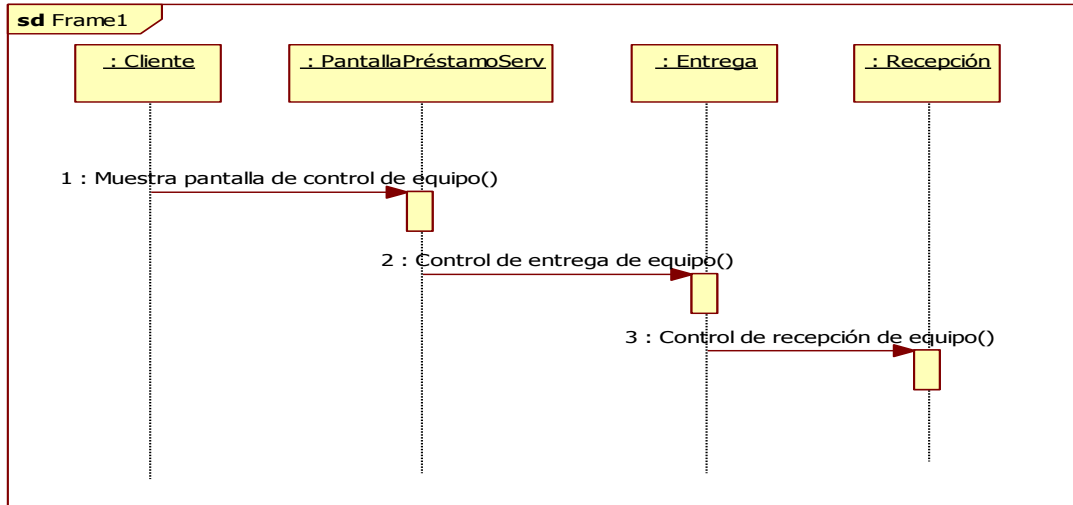
### Diagrama



## Descripción funcional

### Narración del procesamiento

Se muestra el flujo de trabajo del módulo PRESTAMOS.



### Restricciones y limitaciones

A el usuario no se le puede prestar ningún equipo si no ha reservado, y debe presentar una identificación oficial de la empresa o institución.

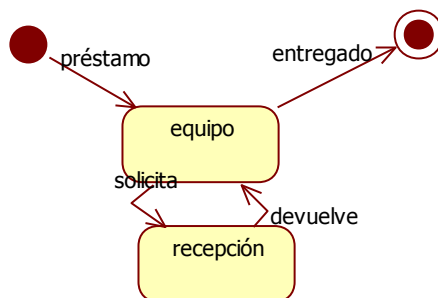
### Requerimientos de Funcionamiento

Se necesita hacer el reservado por parte del usuario con un mínimo de dos días de anticipación.

### Restricciones en Diseño

Se requiere el reservado de equipo y entrega a el usuario identificado.

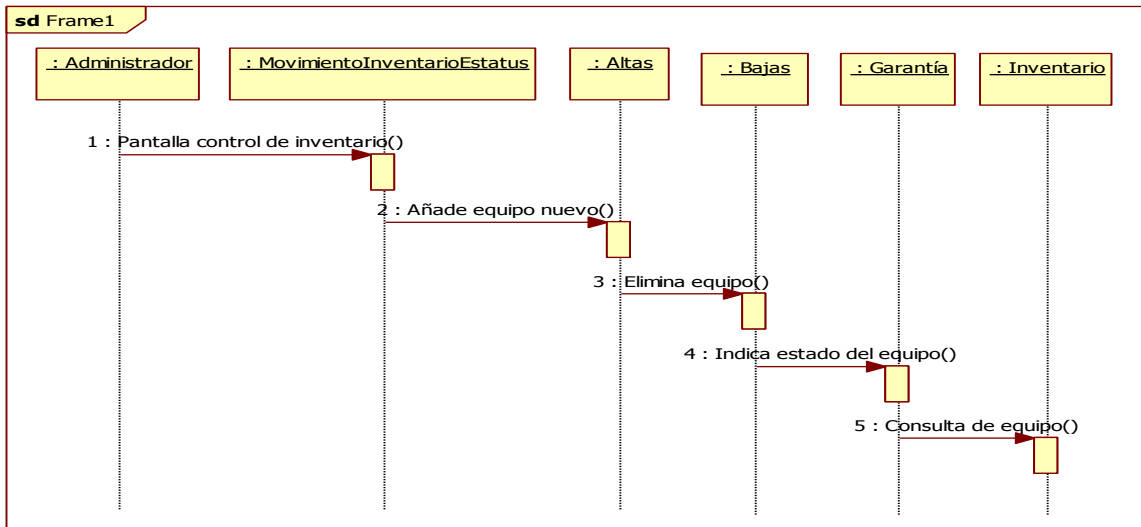
### Diagrama



## Descripción funcional

### Narración del procesamiento

Se muestra el flujo de trabajo del módulo CONTROL DE INVENTARIO.



### Restricciones y limitaciones

A este módulo solo se podrá acceder por parte del usuario administrador autorizado, donde al insertar su login y password , se muestra la pantalla de menús que le permite tener el control de los medios audiovisuales, como son: actualizaciones, eliminación, equipo en garantía, etc.

### Requerimientos de Funcionamiento

Se necesita haber ingresado con login y password validado como administrador.

### Restricciones en Diseño

Se muestran los menús de consulta, cambios, bajas, reservaciones, préstamos, altas.

### Diagrama



## Capítulo IV. Implementación y Pruebas del Sistema

### 4.1 IMPLEMENTACION Y PRUEBAS DEL SISTEMA

#### Criterios de Validación

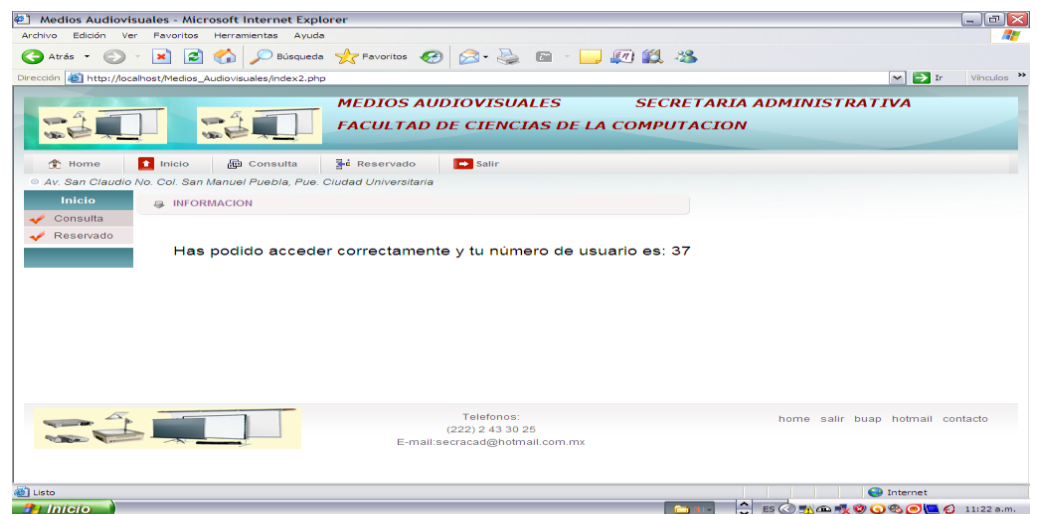
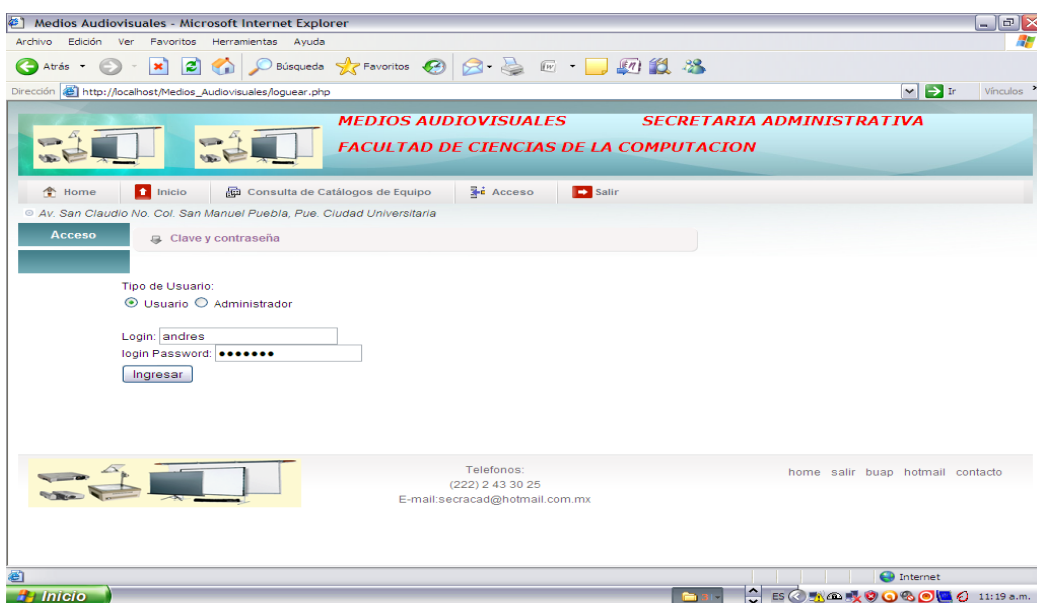
#### Límites de funcionamiento

Para registrarse no aplica

#### Clases de pruebas

Se llevan a cabo pruebas de validación, pruebas de módulos.

#### Respuesta esperada



**Restricciones especiales** Si el usuario no recuerda su clave y contraseña, puede acudir con el administrador del sistema para modificarla, previa identificación.

## Módulo de Consulta

Se llevan a cabo pruebas de validación y por módulos.

## Respuesta esperada

ID	EQUIPO	DESCRIPCION	NO.INVENTARIO	TIPO DE EQUIPO	ESTADO	CAPACIDAD	MODELO	UBICACION
10	proyector	acetatos	0	proyector	completo			
12	proyector	fijo	2345	proyectores	completo			
22	proyector		4567		funcional	1200 lúme	Epson S1	0

ID	PRODUCTO	MARCA	DESCRIPCION
1	cafe	internacional	descafeinado
2	cafe	internacional	con cafeina
3	galletas	garnesa	surtido
4	galletas	macma	surtido
5	refresco	coca cola	
6	refresco		sprite
7	refresco		manzanita

## Restricciones especiales

El usuario puede consultar el equipo por su tipo.

## Módulo de Reservados

Se llevan a cabo pruebas de validación y por módulos.

ID	Estatus	Equipo num.	Equipo	Descripción	Num. Inventario	Fecha_Reserva	Fecha_Inicial	Fecha_Final	Hora inicial	Hora final	Productos/Cantidad	Cancelar
2	Cancelado	17	pantalla	tripie	0	2012-06-02	2012-06-03	2012-06-03	14:00:00	16:00:00		<a href="#">Cancelar</a>
3	Apartado	25	grabadora	cd-dvd	2312	2012-06-02	2012-06-04	2012-06-04	12:00:00	14:00:00		<a href="#">Cancelar</a>
4	Cancelado	6	cañon	pentium	0	0000-00-00	2012-06-05	2012-06-05	16:00:00	17:00:00		<a href="#">Cancelar</a>
5	Cancelado	11	proyector	tripie	0	2012-06-05	2012-06-06	2012-06-06	14:00:00	16:00:00		<a href="#">Cancelar</a>
6	Apartado	19	cable	computadora a proyector	0	2012-06-05	2012-06-06	2012-06-06	17:00:00	19:00:00		<a href="#">Cancelar</a>
7	Apartado	6	cañon	pentium	0	2012-06-05	2012-06-07	2012-06-05	00:00:00	11:00:00	2 cajas de galletas	<a href="#">Cancelar</a>
8	Apartado	2	rotafolio	portatil	2134	2012-06-11	2012-06-11	2012-06-11	10:00:00	12:00:00		<a href="#">Cancelar</a>
9	Apartado	11			3456	2012-06-11	2012-06-11	2012-06-11	00:02:34	00:04:56		<a href="#">Cancelar</a>
10	Cancelado	11			3456	2012-06-11	2012-06-11	2012-06-11	56:34:00	00:00:00		<a href="#">Cancelar</a>
12	Apartado	11			3456	2012-06-11	2012-06-11	2012-06-11	00:00:00	00:00:00		<a href="#">Cancelar</a>
13	Apartado	11			3456	2012-06-11	2012-06-11	2012-06-11	00:00:00	00:00:05		<a href="#">Cancelar</a>
14	Apartado	4	sala		0	2012-06-12	2012-06-12	2012-06-12	13:00:00	14:30:00		<a href="#">Cancelar</a>
15	Apartado	18	pantalla	pared/techo enrollable	0	2012-06-13	2012-06-13	2012-06-13	16:00:00	19:00:00	café para 25 personas	<a href="#">Cancelar</a>

**Restricciones especiales.** Revisar el estatus de reservado.

## Módulo de Préstamos

Se llevan a cabo pruebas de validación y por módulos.

### Respuesta esperada

**MEDIOS AUDIOVISUALES SECRETARIA ADMINISTRATIVA FACULTAD DE CIENCIAS DE LA COMPUTACION**

Home Inicio Consulta Insertar Eliminar Actualizar Buscar Salir

Av. San Claudio No. Col. San Manuel Puebla, Pue. Ciudad Universitaria

**Insertar:** **INSERTAR PRESTAMO**

Usuario  
 Equipo  
 Préstamos

Administrador

Disponible(1-SI/0-NO): 1

Estatus: completo

Comentarios: se identifica con ife

Reservados: 2 -> 2012-06-02 -> 2012-06-03 -> 14:00:00 -> Angel -> Vargas -> 17 -> pantalla -> 0

Grabar

ID	DISPONIBLE	Estatus	COMENTARIO	EQUIPO No.	Equipo	Descripción	Inventario No.	RESERVADO No.	Fecha_Inicial	Hora_Inicial	Usuario No.	Usuario	Appat	Apmat
14	1	buenas condiciones		17	pantalla		0	2	0000-00-00		41	Angel		
15	1	completo	se identifica con ife	25	grabadora		0	3	0000-00-00		37	Andrés		
17	0	completo	se hizo la devoluci?	6	cañon		0	7	0000-00-00		41	Angel		
19	1	buenas condiciones	se identifica con ife	11			0	12	0000-00-00		37	Andrés		

**MEDIOS AUDIOVISUALES SECRETARIA ADMINISTRATIVA FACULTAD DE CIENCIAS DE LA COMPUTACION**

Home Inicio Consulta Insertar Eliminar Actualizar Buscar Salir

Av. San Claudio No. Col. San Manuel Puebla, Pue. Ciudad Universitaria

**Consultas:** **Consulta de Préstamos**

Usuario  
 Equipo  
 Préstamos  
 Reservados

Administrador

ID	DISPONIBLE	Estatus	COMENTARIO	Equi. No.	Equipo	Inv. No.	Descripción	No.res	Fecha_Inicial	Fecha_Final	Usu. No.	Usuario	Appat	Apmat
14	1	buenas condiciones		17	pantalla	0		2	0000-00-00	0000-00-00	41	Angel		
15	1	completo	se identifica con ife	25	grabadora	0		3	0000-00-00	0000-00-00	37	Andrés		
17	0	completo	se hizo la devoluci?	6	cañon	0		7	0000-00-00	0000-00-00	41	Angel		
19	1	buenas condiciones	se identifica con ife	11		0		12	0000-00-00	0000-00-00	37	Andrés		

Telefonos: (222) 2 43 30 25 home salir buap hotmail contacto  
E-mail:secracad@hotmail.com.mx

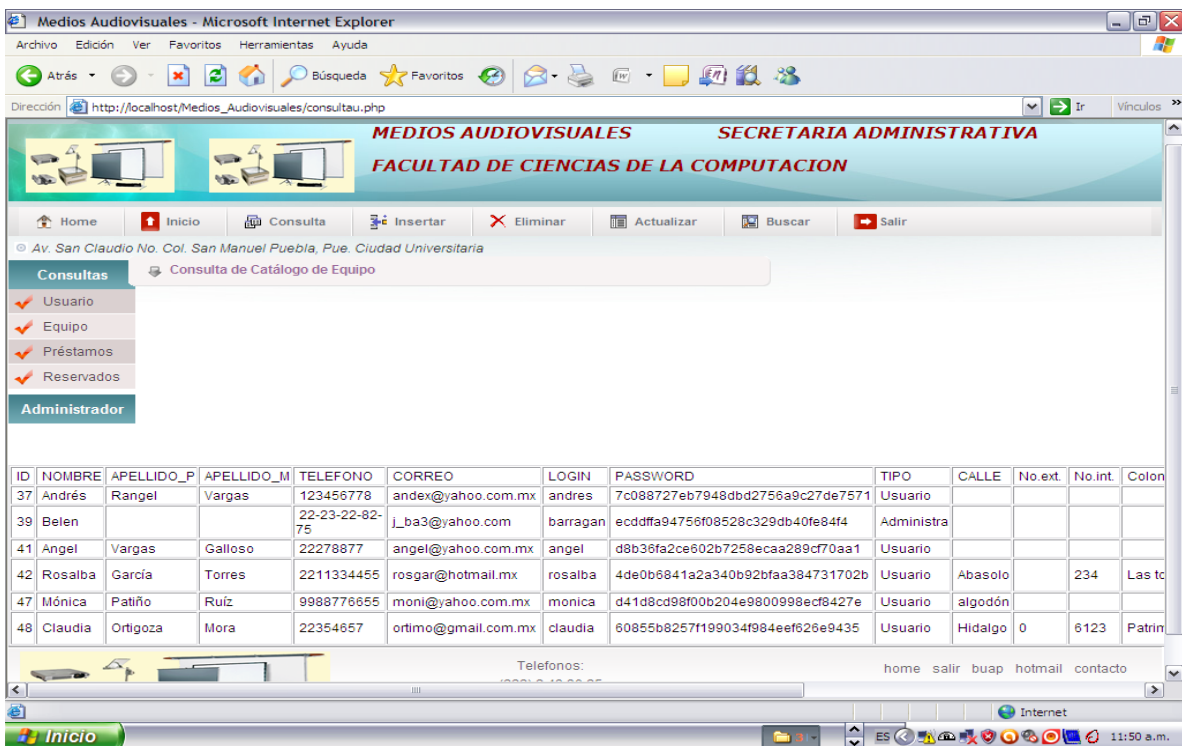
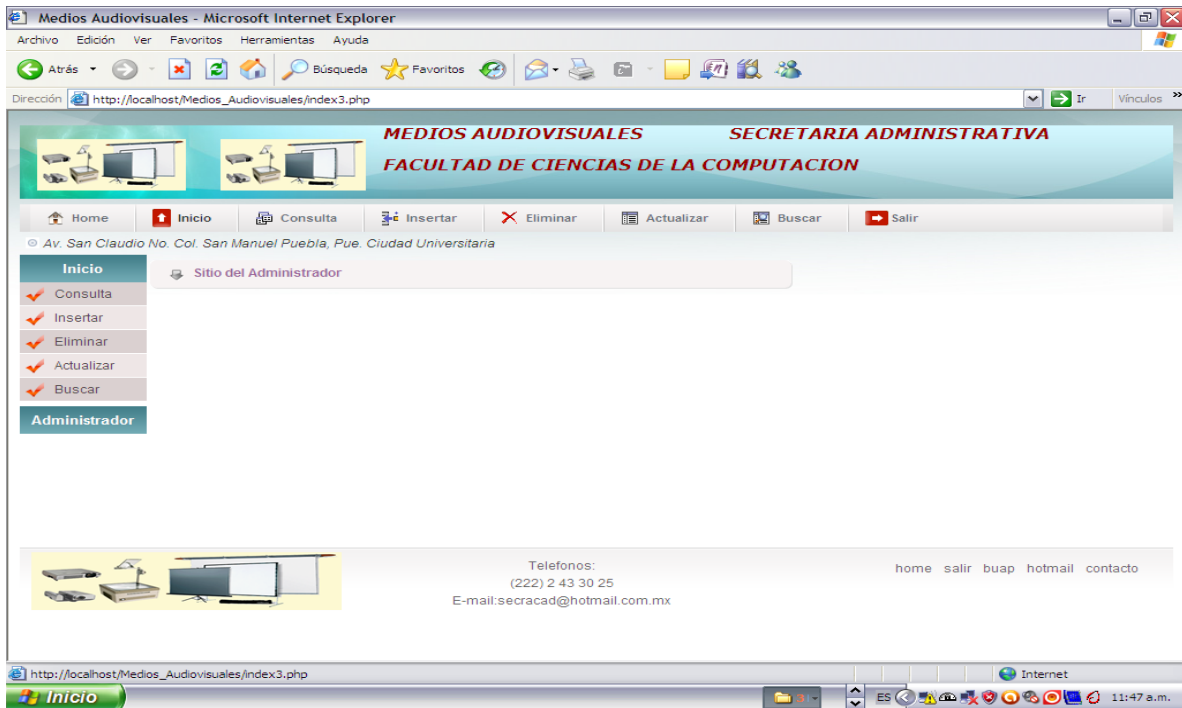
## Restricciones especiales

Seleccionar el usuario al que se le hizo el préstamo.

## Módulo de Control de inventario

Se llevan a cabo pruebas de validación y por módulos.

### Respuesta esperada



Pantalla Insertar Equipo:

**Medios Audiovisuales - Microsoft Internet Explorer**

Dirección: [http://localhost/Medios\\_Audiovisuales/insertare.php](http://localhost/Medios_Audiovisuales/insertare.php)

**FACULTAD DE CIENCIAS DE LA COMPUTACION**

Home Inicio Consulta Insertar Eliminar Actualizar Buscar Salir

Av. San Claudio No. Col. San Manuel Puebla, Pue. Ciudad Universitaria

**Insertar:** **INSERTAR EQUIPO**

Usuario  
 Equipo  
 Préstamos  
**Administrador**

No. Inventario:

Equipo:

Modelo:

Tipo Equipo:

Descripción:

Estatus:

Capacidad:

Ubicación:

ID	No. Inventario	Nombre Equipo	Modelo	Tipo Equipo	Estatus	Capacidad.....	Descripción	Ubicación
2	2134	rotafolio	aba	estandar			portatil	0
4	0	sala						0
6	0	cañon		existe	funciona		pentium	0
7	0	reproductor	samsung	buen estado	completo		cd-dvd	0
9	0	cañon			media vida		hp	4
10	0	proyector		proyector	completo		acetatos	
11	3456	television			completo			
12	2345	proyector		proyectores	completo		fijo	

Pantalla eliminar préstamos:

**Medios Audiovisuales - Microsoft Internet Explorer**

Dirección: [http://localhost/Medios\\_Audiovisuales/eliminarp.php](http://localhost/Medios_Audiovisuales/eliminarp.php)

**MEDIOS AUDIOVISUALES SECRETARIA ADMINISTRATIVA**  
**FACULTAD DE CIENCIAS DE LA COMPUTACION**

Home Inicio Consulta Insertar Eliminar Actualizar Buscar Salir

Av. San Claudio No. Col. San Manuel Puebla, Pue. Ciudad Universitaria

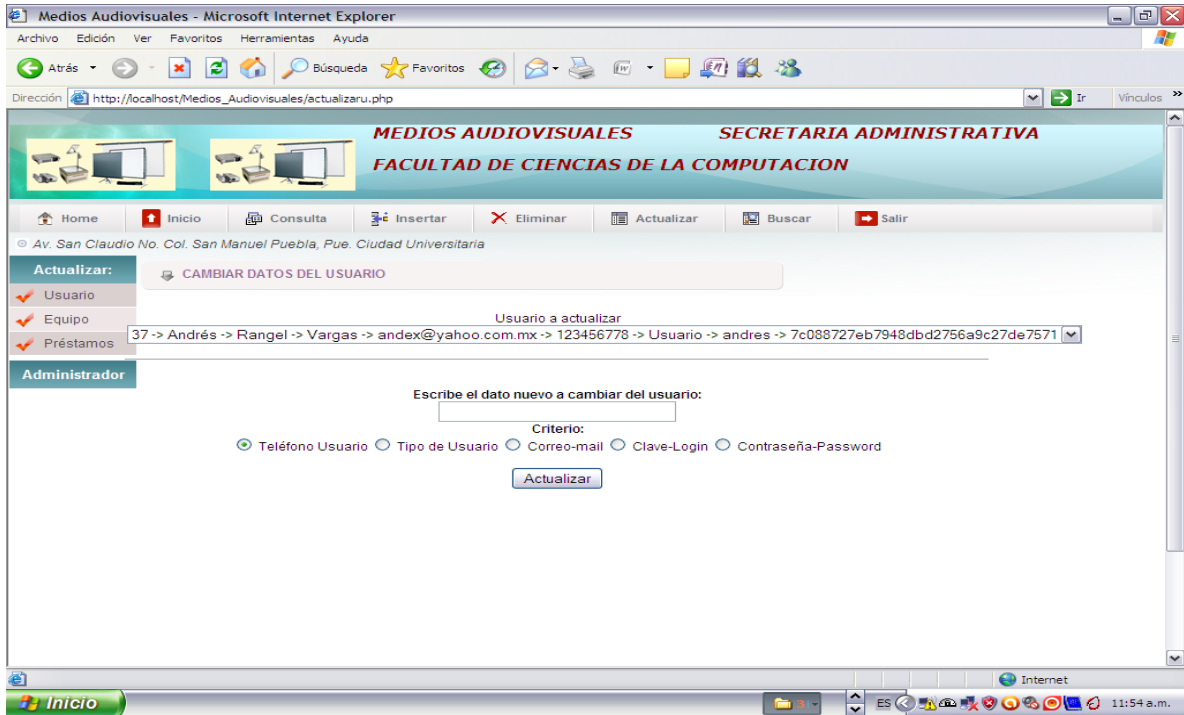
**Eliminar:** **BORRAR PRESTAMO**

Usuario  
 Equipo  
 Préstamos  
 Reservados  
**Administrador**

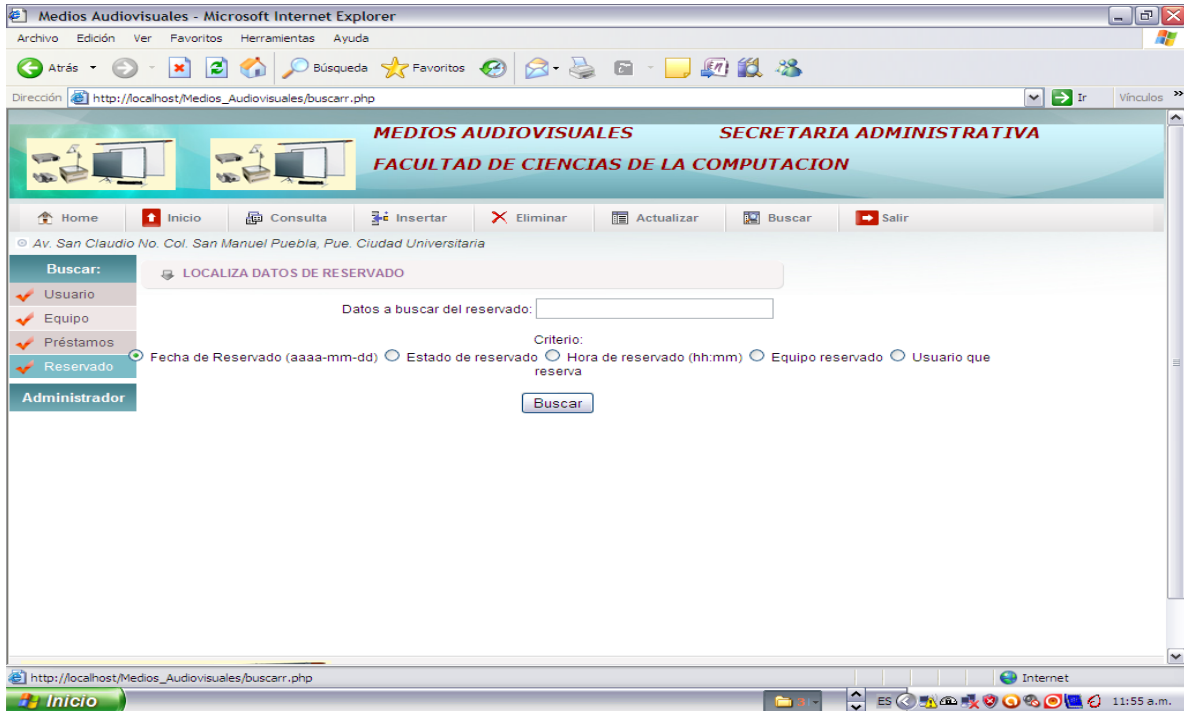
ID	DISPONIBLE	Estatus	COMENTARIO	Equipo No.	Equipo	Reserva No.	Fecha_Inicial	Fecha_Final	Usuario No.	Usuario	Apmat	Apmat	Eliminar	Actualizar
14	1	buenas condiciones		17	pantalla	2	0000-00-00	0000-00-00	41	Angel			<input type="button" value="Eliminar"/>	<input checked="" type="button" value="Actualizar"/>
15	1	completo	se identifica con ife	25	grabadora	3	0000-00-00	0000-00-00	37	Andrés			<input type="button" value="Eliminar"/>	<input checked="" type="button" value="Actualizar"/>
17	0	completo	se hizo la devoluci?	6	cañon	7	0000-00-00	0000-00-00	41	Angel			<input type="button" value="Eliminar"/>	<input checked="" type="button" value="Actualizar"/>
19	1	buenas condiciones	se identifica con ife	11		12	0000-00-00	0000-00-00	37	Andrés			<input type="button" value="Eliminar"/>	<input checked="" type="button" value="Actualizar"/>

Telefonos: (222) 2 43 30 25 home salir buap hotmail contacto

## Pantalla Actualizar Usuarios:



## Pantalla Buscar Reservados:



## Restricciones especiales

Insertar fecha y hora de reservado, y haberse logeado.



## CONCLUSIONES

*Se creó este sistema, para que agilice y facilite a los usuarios el trámite de reservado de equipos, infraestructura y servicios, en horarios y fechas solicitados quedando a entera satisfacción con lo que se reserva, haciendo uso de la web.*

*Con el lenguaje de modelado unificado UML y la metodología RUP se facilitó la especificación de requerimientos de cada módulo, para establecer el ciclo de vida del sistema, esquematizando los módulos que lo conforman, el análisis, diseño, implementación, pruebas e implantación.*

*Se usó el diagrama Entidad-Relación como una base necesaria para la construcción del diseño de la base de datos de este software; ya que se establecieron las bases de datos, atributos de las bases de datos y las relaciones entre ellas.*

*El sistema de medios audiovisuales proporciona calidad en el servicio agilizando los trámites de reservado, con la documentación adecuada para su desarrollo, operación y mantenimiento, haciendo uso de la Ingeniería de software.*

*Además de que se brinda la seguridad necesaria al sistema, para que no pueda ser intervenido por usuarios ajenos a la institución.*

*Por lo que es importante llevar y tener los conocimientos de la Ingeniería del software para crear y mantener aplicaciones de software.*

## PERSPECTIVAS

*Que en un futuro este sistema pueda ser aplicado en diferentes áreas de reservado y tener una actualización en el diseño.*



## Bibliografía.

- [1] Graharn, Jan: *Métodos orientados a objetos, Segunda edición, AddisonWe5ley IDíaz de Santos, 1996.*
- [2] Martin, James; Oclel}. James J.: *Métodos orientados a objetos: COI/ceptos ftllldalllell tales, Prentice-Hall Hispanoamericana, México, 1997.*
- [3] Rebecca Wirfs-Brock, Brian Wilkerson, and Lauren Wiener: *Designing Object-Oriented Software. Prentice Hall, 1990.*
- [4] UML gota a gota Martin Fowler Pearson Educacion Addison Wa fer longman de México. S.A. de C.V. México.1999
- [5] *Desarrollo de Sistemas. Roger S. Pressman; McGraw Hill*
- [6] *Desarrollo de sistemas de información. Una metodología basada en el modelado Fernández Alarcón, Vicenç, Ediciones UPC, S.L. 1ª ed., 1ª imp.(05/2006)*
- [7] Zavala R. 2000. *Diseño de un Sistema de Información Geográfica sobre internet. Tesis de Maestría en Ciencias de la Computación. Universidad Autónoma Metropolitana-Azcapotzalco. México, D.F. En prensa.*
- [8] Lara Martínez zuleyka. *Base de Datos Multimedia de Daños en inmuebles de la Zona Monumental de la Cd. De Puebla por le sismo del 15 de junio de 1999.*
- [9] Greiff W. R. *Paradigma vs Metodología; El Caso de la POO (Parte II). Soluciones Avanzadas. Ene-Feb 1994. pp. 31-39.*
- [10] Lewis G. 1994. "What is Software Engineering?" *DataPro (4015). Feb 1994. pp. 1-10.*
- [11] *Código de Etica del Ingeniero en Software y de la Práctica Profesional en el site de la Association for Computing Machinery <http://www.acm.org/serving/ethics.html>.*
- [12] Abraham Silbershatz, Henry F. Korth y S. Sudarshan. "Fundamentos de Bases de Datos". Tercera Edición, Editorial Mc Graw Hill, 1999.
- [13] Annette Scott. "Oracle Data Modeling and Relational Database Design", *Student Guide. Edición 1.0.*
- [14] Referencias: Date, C.J., *An Introduction to Database Systems, Vol. 1. Addison-Wesley, 1981.* Hawryszkiewicz, I.T., *Database Analysis and Design, SRA Inc., 1984.* Kroenke, D.M., *Database Processing: Fundamentals, Design, Implementation. SRA Inc., 1983.* "Diseño e implantación de Bases de Datos", *Capitulo 10. Diseño de bases de datos: formas normales de relaciones.*
- [15] *Applying UML and Patterns Introduction to Object-Oriented Analysis and Design*



- second edition – Craig Larman Editorial McGraw-Hill

[16] *El Lenguaje Unificado de Modelado* Grady Booch James Rumbaugh Ivar Jacobson  
Prentice Hall

[17] *Use Cases Requirements in Context* Daryl Kulak Eamonn Guiney  
Addison Wesley

[18] *Seguridad en la WEB* 0020 Mtro. Ivo Torres Pineda  
Curso Diplomado Tecnologías de la Información

[19] <http://users.dsic.upv.es/asignaturas/facultad/lsi/ejemplorup/>

[20] [http://es.wikipedia.org/wiki/Proceso\\_Unificado\\_de\\_Rational](http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational)

[21] <http://www.sysml.org>

[22] [http://www.objecteering.com/products\\_uml\\_profile\\_builder.php](http://www.objecteering.com/products_uml_profile_builder.php)

[23] <http://dsmforum.org>

## **APENDICE**

### **Glosario:**

Sistema de Bases de Datos: es un conjunto de registros, datos y relaciones que refleja las necesidades de información de una empresa.

Base de datos: consiste en una colección de datos relacionados entre sí, que contiene información acerca de una empresa determinada.

Base de datos relacional: consiste en un conjunto de tablas, a cada una de las cuales se le asigna un nombre exclusivo. Cada fila de la tabla representa una relación entre un conjunto de valores.

Criptografía: (del griego krypto, «oculto» y graphos, «escribir», literalmente «escritura oculta») es la técnica, bien sea aplicada al arte o la ciencia, que altera las representaciones lingüísticas de un mensaje.



Diagrama: una representación gráfica de una colección de elementos de modelado, a menudo dibujada como un grafo con vértices conectados por arcos.

□

Entidad: es una <<cosa>> u <<objeto>> en el mundo real que es distinguible de otros objetos.

Los atributos son información acerca de una entidad que necesita ser conocida o descrita.

Modelo entidad-relación: categoriza todos los elementos de un sistema como una entidad (una persona, lugar o cosa) o una relación entre entidades.

Primera forma normal: (o 1FN) si todos los campos en cada registro contienen un solo valor tomado de sus dominios respectivos.

3FN, si es 2FN y ningún atributo no-clave en la relación es funcionalmente dependiente de algún otro atributo no-clave.

4FN (cuarta forma normal) si es BFCN y no contiene dependencias multivalores.

Relación: es una asociación entre varias entidades.

Segunda forma normal: si es 1FN y cada atributo no-clave de la relación es total y funcionalmente dependiente de su clave principal.

Teoría de la normalización: ayuda a reconocer las cualidades no deseadas en un archivo y la forma de corregirlas.

UML = Unified Modeling Language Un lenguaje de propósito general para el modelado orientado a objetos.