

UNIVERSIDAD AUTÓNOMA
METROPOLITANA

UNIDAD AZCAPOTZALCO

División de Ciencias Básicas e Ingeniería

Licenciatura en Ingeniería en Computación

REPORTE DE PROYECTO TERMINAL

*“Sistema Tutor Web para el Aprendizaje de
Programación Orientada a Objetos”*

Alumno: Juan José Perea Sánchez

Matricula: 2042040030

Asesor: Irma Fernanda Ardón Pulido

Tabla de Contenido

Introducción.....	5
Capitulo 1. Diseño e Implementación de la Base de Daros.....	7
Capitulo 2. Navegación de las Páginas Web.....	11
Capitulo 3. Funcionamiento de los Módulos.....	16
Conclusiones.....	26
Anexo A.....	28
Anexo B.....	36

Índice de Figuras.

Figura 1. Diagrama Entidad-Relación.....	10
Figura 2. Diagrama de Navegación Alumnos.....	11
Figura 3. Diagrama de Navegación Administración.....	11
Figura 4. Diagrama de Navegación General.....	12
Figura 5. Página Principal del Sitio Web.....	12
Figura 6. Página Principal para Alumnos.....	13
Figura 7. Preguntas de Evaluación.....	13
Figura 8. Chat en Línea.....	14
Figura 9. Página Principal para Administración.....	14
Figura 10. Página Principal de Contenido de Estudio.....	15
Figura 11. Módulos del Sistema Tutor.....	16
Figura 12. Lista de Alumnos de la unidad uno.....	17
Figura 13. Lista de Alumnos que terminaron el curso.....	17
Figura 14. Resultado de Buscar Alumno.....	18
Figura 15. Actualizar Alumno.....	18
Figura 16. Administración de Contenidos de Estudio.....	19
Figura 17. Subir Contenido PDF.....	19
Figura 18. Resultado de Agregar PDF.....	20
Figura 19. Resultado de la Búsqueda de un Contenido.....	20
Figura 20. Asociar a una Actividad.....	21
Figura 21. Preguntas de Evaluación.....	21
Figura 22. Botón para Evaluación.....	22
Figura 23. Chat en Línea del Sistema.....	22
Figura 24. Preguntas de la Unidad.....	23

Figura 25. Contenidos de Estudio.....	24
Figura 26. Inscripción al Tutorial.....	25
Figura 27. Diagrama de Clases.....	37

INTRODUCCIÓN

En el presente documento se muestra el proceso de creación del Proyecto Terminal Sistema Tutor Web para el Aprendizaje de Programación Orientada a Objetos, se describe el desarrollo de la Base de Datos, el diseño, tanto de las páginas Web como de su navegación y finalmente, el funcionamiento de los módulos implementados.

El objetivo del Proyecto Terminal es, proporcionar un medio en el cual los alumnos de la UAM-Azcapotzalco, de la carrera de Ingeniería en Computación, puedan estudiar los mecanismos y características principales de la Programación Orientada a Objetos (POO) en lenguaje Java.

Para implementar la Base de Datos de manera física, se seleccionó como manejador de bases de datos a MySQL para ello se estudió, la forma de realizar un script en SQL y cómo cargar éste al manejador desde la línea de comandos.

Se utilizó el Entorno de Desarrollo Integrado (IDE) Netbeans 6.8, en el cual, haciendo uso de los frameworks Struts e Hibernate se programó el funcionamiento de los módulos que componen el sistema, los cuales fueron cargados en el servidor de aplicaciones Tomcat 6.

Se seleccionó el software antes mencionado debido a que facilita el desarrollo del sistema propuesto, además de ser herramientas vigentes en el ámbito laboral.

En el primer Capítulo, se describen los procesos llevados a cabo para la realización de la Base de Datos, se identificaron las Entidades principales, seguido de la relaciones existentes entre ellas, utilizando tanto una Matriz de Entidades como un Diagrama Entidad-Relación, para depurar las relaciones previamente definidas y finalmente, mediante un Script SQL se implementó la Base de Datos utilizada por el Sistema Tutor.

En el segundo Capítulo, se muestra la manera en que el usuario navega por los diferentes servicios que ofrece el Sistema Tutor, mostrando las interfaces correspondientes tanto para los Alumnos como para el Administrador.

En el tercer Capítulo, se describe el funcionamiento de los módulos que componen el Sistema Tutor, señalando los pasos necesarios para la ejecución de las actividades tanto de Alumnos como del Administrador.

Finalmente, se presentan las conclusiones en términos de utilidad del proyecto, herramientas utilizadas, mejoras propuestas y líneas de investigación derivadas del proyecto.

CAPÍTULO 1. DISEÑO E IMPLEMENTACIÓN DE LA BASE DE DATOS

El Diseño consiste en identificar, Entidades, Atributos y Relaciones, para elaborar el Diagrama Entidad-Relación.

I. Identificar las Entidades

Se hizo una lectura de la propuesta de Proyecto Terminal para poder obtener las entidades u objetos principales, personas, lugares, cosas o eventos de interés necesarios para la elaboración de la Base de Datos del Sistema Tutor.

Después de analizar diversos candidatos a Entidades, se determinaron como principales las siguientes:

- Alumnos
- Actividades_realizadas
- Evaluaciones_alumno
- Reglas_de_producción
- Actividades
- Contenidos_de_estudio
- Preguntas
- Respuestas

II. Entidades y Atributos

Una vez obtenidas las Entidades principales, se definieron sus atributos y Llave Primaria (PK Primary Key) tal y como se muestra en las Tablas uno y dos.

Actividades	Actividades_Realizadas	Alumnos	Contenido_de_estudio
idActividad (PK)	idActividad_realizada (PK)	login (PK)	idContenido (PK)
descripcion	login_usuario	Password	idActividad
Regla	idActividad	Mail	url
Unidad	numero_unidad	Dirección	Nombre
	realizada	numero_unidad	Unidad
		ultimo_acceso	Tipo
		Tipo	nombreO
		Teléfono	

Tabla 1. Atributos de las Entidades: Actividades, Actividades_Realizadas, Alumnos y Contenidos_de_estudios

Evaluaciones_alumnos	Reglas_de_produccion	Preguntas	Respuestas
idEvaluacion (PK)	idRegla (PK)	idPregunta (PK)	idRespuesta (PK)
login_usuario	unidad	numero_unidad	idPregunta
numero_unidad	sig_reglaS	Pregunta	Respuesta
Calificación	pregunta	respuesta_correcta	
	sig_reglaN		

Tabla 2. Atributos de las Entidades: Evaluaciones_alumnos, Reglas_de_Produccion, Unidad, Preguntas y Respuestas

III. Identificar Relaciones entre Entidades

En la Tabla tres se representan las asociaciones entre Entidades.

Entidad	Relación	Entidad Relacionada
Alumnos	Realizan	Acciones
Reglas_de_produccion	Reaccionan a las	Acciones
Acciones_realizadas	Realizadas por	Alumnos
Alumnos	Reciben	Evaluaciones_alumnos
Preguntas	Contiene	Respuestas

Tabla 3. Relaciones entre Entidades

Matriz de Entidades

En el diseño de una base de datos, la Matriz de Entidades permite detectar interrelaciones que no están explícitamente representadas en los enunciados del problema, pero que son localizables por sentido común, además de depurar las relaciones previamente definidas.

En la Matriz de Entidades se utilizarán las siguientes abreviaturas:

AL = Alumnos

AC = Actividades

CE = Contenidos_de_estudio

PR = Preguntas

AR = Actividades_realizadas

PR = Preguntas

RP = Reglas_de_producción

EA = Evaluaciones_por_alumno

	AL	AC	AR	RP	CE	PR	RE	EA
AL	X	--	Realiza	--	--	--	--	recibe
AC	X	X	es hecha	son programadas por	Utilizan	--	--	--
AR	X	X	X	--	--	--	--	--
RP	X	X	X	X	--	--	--	--
CE	X	X	X	X	X	--	--	--
PR	X	X	X	X	X	X	contiene	--
RE	X	X	X	X	X	X	X	--
EA	X	X	X	X	X	X	X	X

Tabla 4. Matriz de Entidades

Creación del Diagrama Entidad-Relación

Utilizando la información proporcionada por la Matriz de Entidades, se construye el Diagrama Entidad-Relación (DER) cuya versión final, sin ciclos, se muestra en la Figura uno.

Una vez identificadas las Entidades con sus Atributos y Relaciones, se implementó la Base de Datos con un script SQL, el cual se presenta en el Anexo A.

Dicho script crea, tanto la base de datos, como sus respectivas tablas, y también las tareas automáticas de eliminación de alumnos que tengan inactividad en el Sistema Tutor, es decir aquellos que no han ingresado a él en los últimos 15 días.

El Sistema Tutor, hace uso del conector Java para el manejador MySQL, permitiendo el acceso a la Base de Datos y la manipulación de sus elementos.

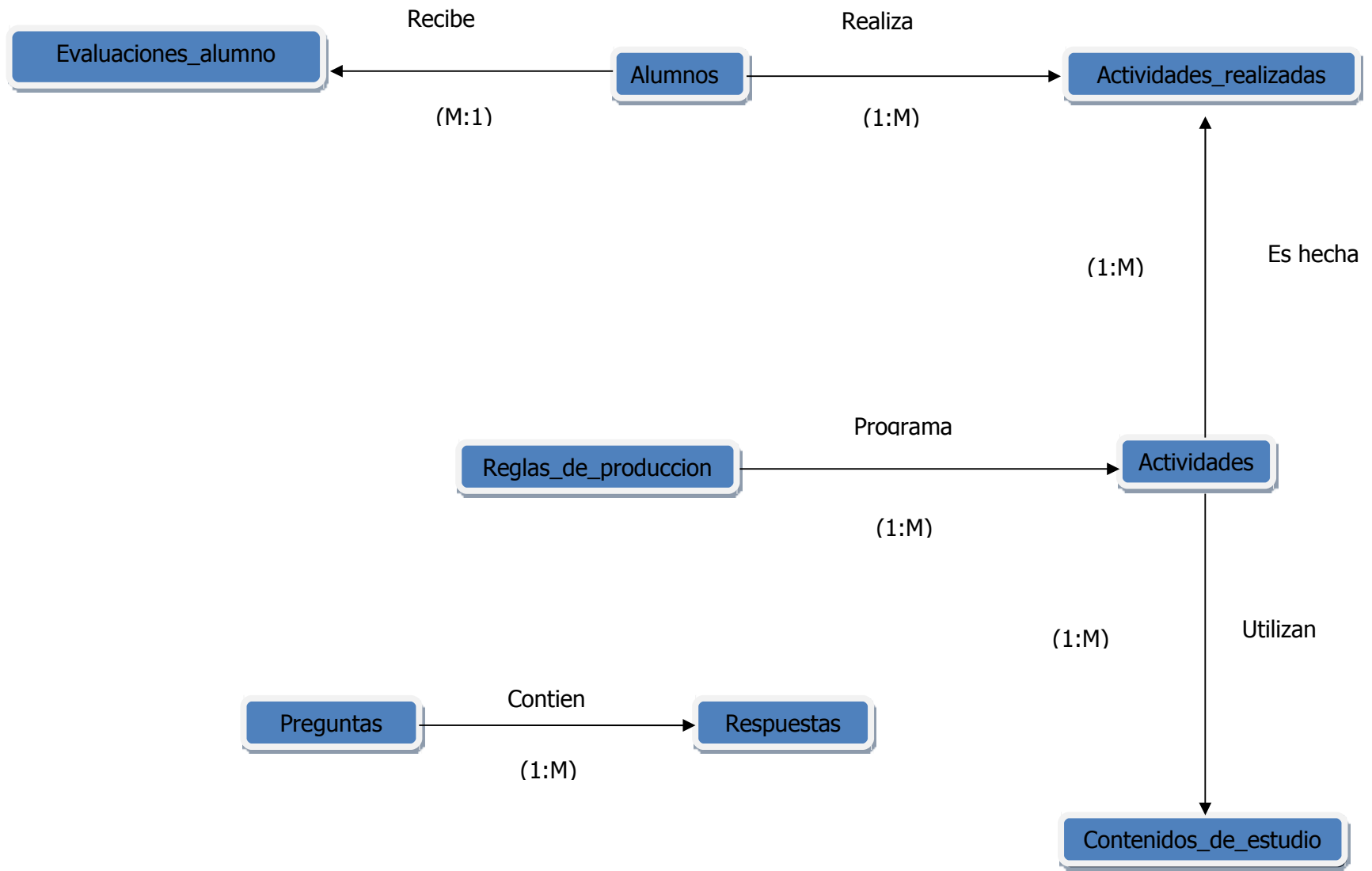


Figura 1. Diagrama Entidad-Relación

CAPÍTULO 2. NAVEGACIÓN DE LAS PÁGINAS WEB

El sitio Web del Sistema Tutor se compone de dos secciones: Administración y Alumnos.

A continuación se presentan los Diagramas de Navegación donde se ilustra la manera en que los usuarios utilizarán dichas secciones.

Figura 2.

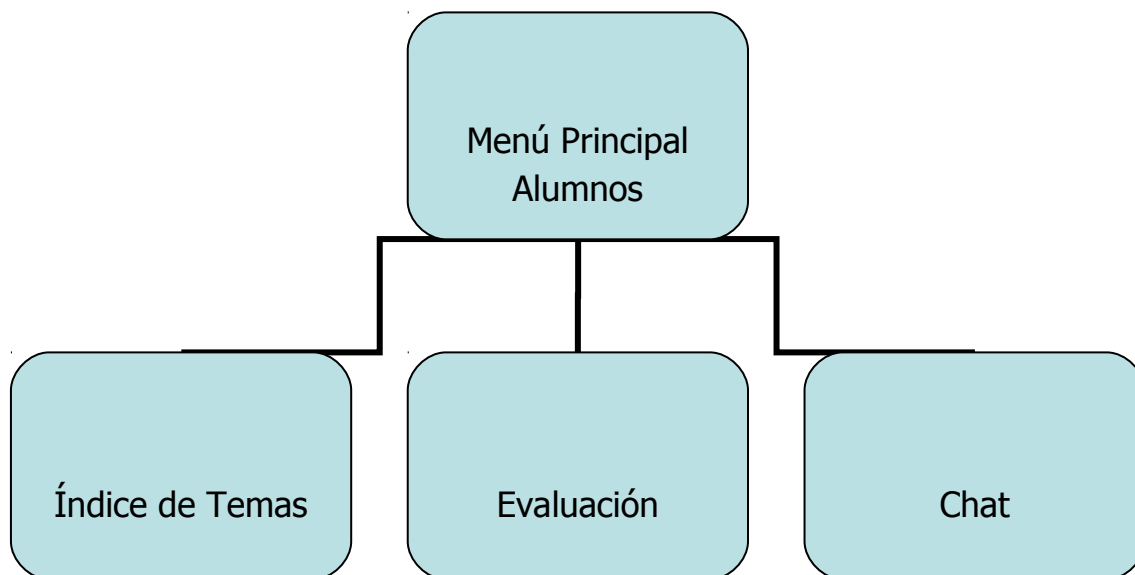


Figura 2. Diagrama de Navegación Alumnos

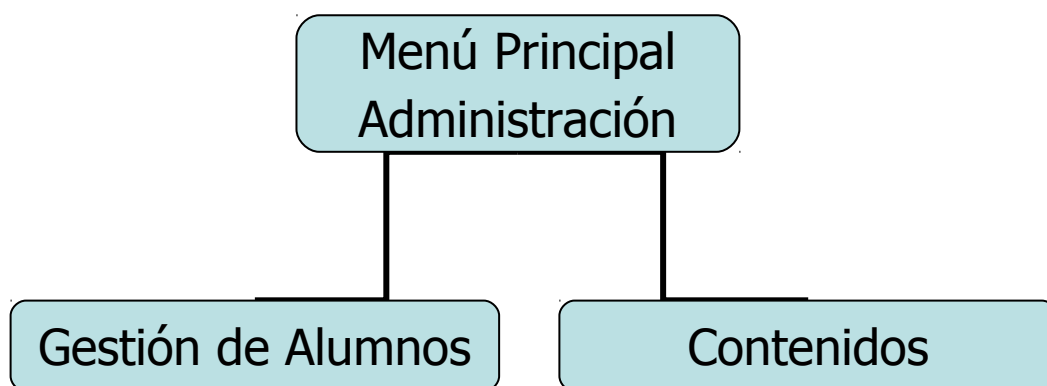


Figura 3. Diagrama de Navegación Administración

El Diagrama general de Navegación es el siguiente:

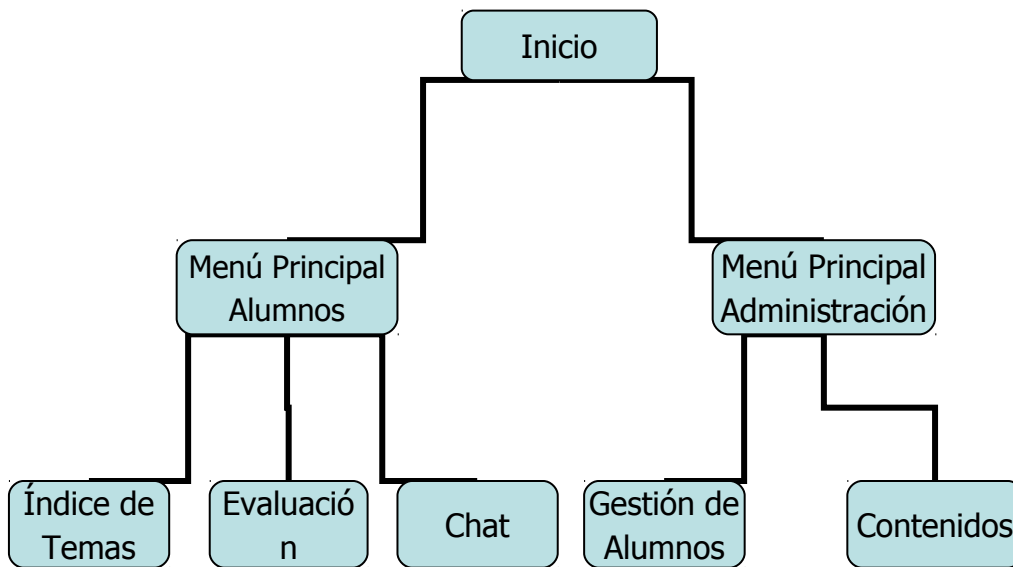


Figura 4. Diagrama de Navegación General

Una vez elaborados los Mapas de Navegación, se diseñaron las interfaces correspondientes, mismas que a continuación se presentan, empezando con la página Principal del Sitio Web (Figura cinco).

Tutor Virtual.

Programacion Orientada a Objetos en JAVA.

Para entrar al curso, escribe tu login y password

Log in:
Password:

¿Deseas inscribirte haz click [aquí](#)



Figura 5. Página Principal del Sitio Web

Una vez que el usuario ingresa al Sistema, si es alumno, verá una interfaz con acceso a: Índice de Unidades, Evaluación y Chat.

Dicho Índice de Unidades, se muestra en la siguiente Figura seis.

Índice de Unidades.



- [1.- Introduccion a la POO](#)
- [2.- Caracteristicas de la POO](#)
- [3.- Iniciacion y Destruccion de Objetos](#)
- [4.- Control de Acceso](#)
- [5.- Herencia y Polimorfismo](#)
- [6.- Clases Abstractas e Interfaces](#)

Figura 6. Página Principal para Alumnos

En caso de que el alumno acceda a la Evaluación, el Sistema le mostrará la siguiente interfaz:

Preguntas de Evaluacion.



¿Que es la programacion orientada a objetos?

- Una forma de programar sin problemas
- Un modelo para simular los objetos cotidianos
- Es un modelo de programacion que utiliza objetos, ligados mediante mensajes, para la solucion de problemas

¿Que es un mensaje?

- Es una forma de comunicacion entre los objetos para ejecutar una accion
- Es una llamada de atencion
- Es cuando un objeto reporta un error

¿Que es un objeto?

- Es una abstraccion de las cosas vivas
- Implementacion de los datos
- Es una encapsulacion generica de datos y de los metodos para manipularlos

Figura 7.Preguntas de Evaluación

Si el alumno desea entrar al Chat, deberá pulsar en la opción correspondiente y verá una página como la mostrada en la Figura ocho.

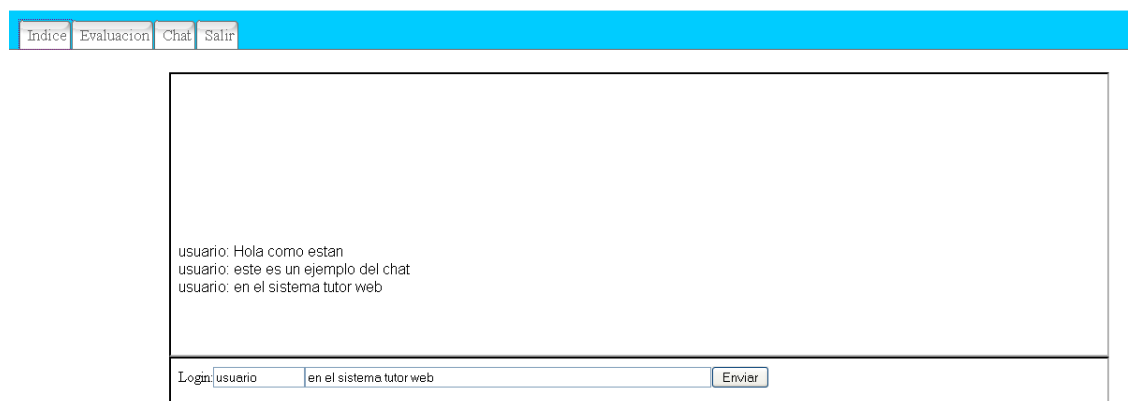


Figura 8. Chat en Línea

En caso de que el usuario sea el Administrador, se muestra una interfaz con acceso a: Administración de Alumnos y Contenidos de Estudio.

Para Administración de Alumnos, se presenta la página mostrada en la siguiente figura.



Figura 9. Página Principal para Administración *

Si el Administrador selecciona el enlace Contenidos de Estudio se mostrará la página de la Figura 10.

*El sistema se encarga de validar que el campo Login no quede vacío.

Administracion.

Contenido de Estudios.

Alumnos	Contenido de Estudios	Salir
---------	-----------------------	-------

- [Agregar Contenido PDF.](#)
- [Agregar Contenido HTML.](#)
- [Agregar Contenido VIDEO FLA.](#)
- [Agregar Contenido FLASH.](#)

Buscar Contenido

Nombre del Contenido:

Buscar Borrar Actualizar

Unidad:

Figura 10. Página Principal de Contenidos de Estudio *

*El sistema se encarga de validar que el campo Nombre del Contenido no quede vacío.

CAPÍTULO 3. FUNCIONAMIENTO DE LOS MÓDULOS

A continuación se presenta el funcionamiento de los módulos del Sistema Tutor, mismos que se muestran en la siguiente figura:

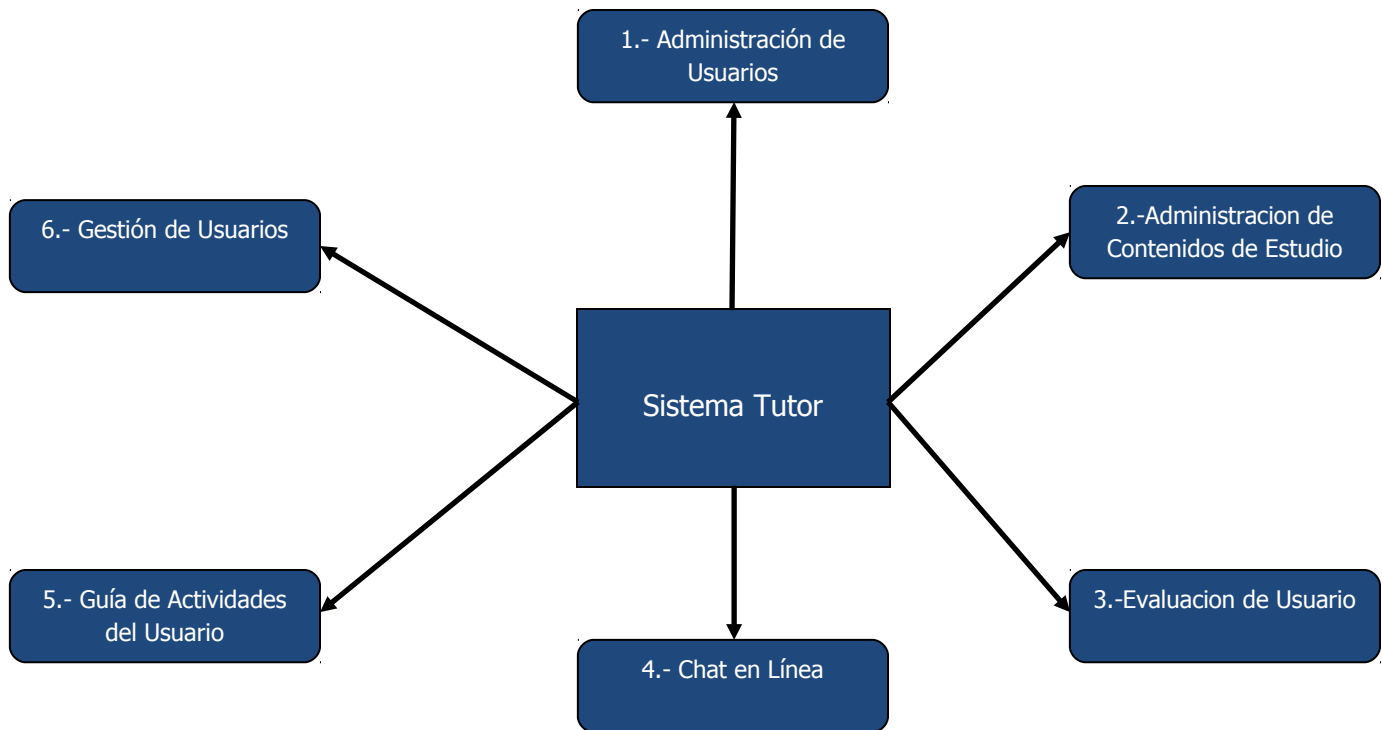


Figura 11. Módulos del Sistema Tutor

Descripción del manejo de módulos.

1.- Administración de Usuarios

En una interfaz como la mostrada en la Figura nueve, el Administrador del sistema podrá llevar a cabo las siguientes tareas:

1.1 Consultas

Listado de alumnos que:

- i. Cursan cada Unidad.
- ii. Concluyeron el Tutorial.

Y de un alumno en particular:

- iii. Expediente (Calificaciones de las evaluaciones realizadas).

1.2 Bajas

- i. Por alumno.
- ii. Alumnos que concluyeron el Tutorial.

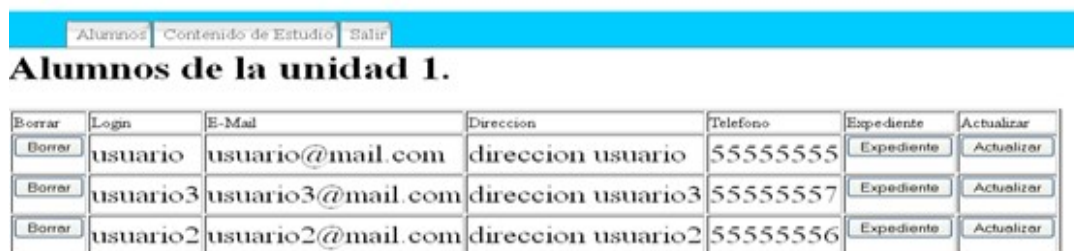
1.3 Cambios

De manera individual:

- i. Datos de los alumnos.

A continuación, se muestran las interfaces de las consultas de alumnos.

En la Figura 12 se muestra de ejemplo a los alumnos que están cursando la unidad uno.



Borrarr	Login	E-Mail	Direccion	Telefono	Expediente	Actualizar
<input type="button" value="Borrarr"/>	usuario	usuario@mail.com	direccion usuario	55555555	<input type="button" value="Expediente"/>	<input type="button" value="Actualizar"/>
<input type="button" value="Borrarr"/>	usuario3	usuario3@mail.com	direccion usuario3	55555557	<input type="button" value="Expediente"/>	<input type="button" value="Actualizar"/>
<input type="button" value="Borrarr"/>	usuario2	usuario2@mail.com	direccion usuario2	55555556	<input type="button" value="Expediente"/>	<input type="button" value="Actualizar"/>

Figura 12. Lista de Alumnos que cursan la unidad uno.

Como se aprecia en la figura anterior, además del listado, en esta interfaz, el usuario podrá consultar el expediente del alumno, actualizar sus datos o bien borrarlo.

En la Figura 13 se muestra un ejemplo de listado de alumnos que terminaron el curso.



Borrarr	Login	E-Mail	Direccion	Telefono	Expediente
<input type="button" value="Borrarr"/>	usuarioF	usuarioF@mail.com	direccion usuarioF	55555550	<input type="button" value="Expediente"/>

Figura 13. Lista de Alumnos que terminaron el curso.

En la Figura 14 se muestra el resultado de buscar a un alumno.



Login	E-Mail	Direccion	Telefono
usuario	usuario@mail.com	direccion usuario	55555555

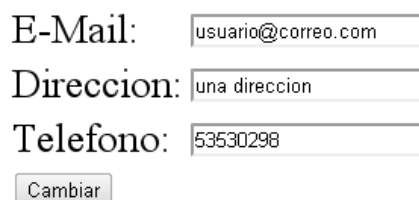
Expediente

Figura 14. Resultado de buscar a un alumno.

En una interfaz como la mostrada en la Figura nueve, al activar la casilla Actualizar, el administrador del Sistema podrá ingresar el login de un estudiante y si éste existe, modificar datos por medio de un formulario como el que se presenta en la figura 15.

Actualizar Alumno

Cambiar datos del alumno: "usuario"



E-Mail:

Direccion:

Telefono:

Figura 15. Actualizar Alumno

2.- Administración del Contenido de Estudios.

En este modulo se utiliza la página de contenido de estudio mostrada en la Figura 10 (página 12) en ella, el administrador podrá realizar las siguientes acciones.

2.1 Altas

2.2 Bajas

2.3 Cambios

En la siguiente Figura 16 se detallan dichas acciones.

Administracion.

Contenido de Estudios.

Alumnos Contenido de Estudios Salir

- [Agregar Contenido PDF.](#)
- [Agregar Contenido HTML.](#)
- [Agregar Contenido VIDEO FLA.](#)
- [Agregar Contenido FLASH.](#)

Buscar Contenido

Nombre del Contenido:

Buscar Borrar Actualizar

Unidad:

Figura 16. Administración de Contenidos de Estudio

Como se puede apreciar, la página muestra enlaces para dar de alta cada tipo de contenido de estudio.

Una vez que el usuario en la interfaz anterior, activa el enlace para agregar un cierto tipo de contenido, deberá pulsar el botón Examinar (Figura 17).

Alumnos Contenido de Estudio Salir

Archivo PDF:

Unidad:

Figura 17. Subir un contenido PDF.

En la Figura 18 se muestra el resultado de subir un archivo PDF.

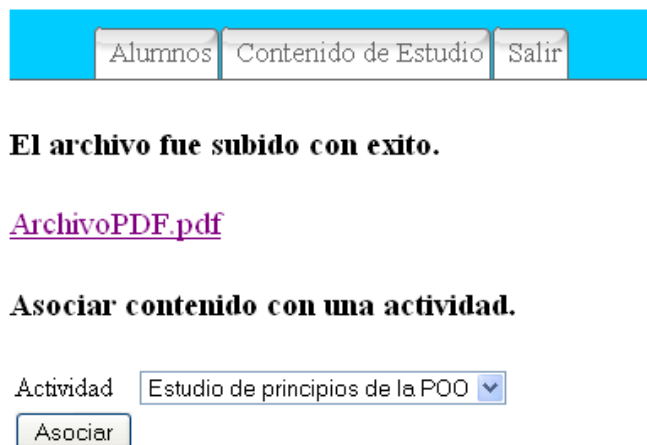


Figura 18. Resultado de Agregar PDF

Para los videos, a diferencia de los demás tipos de contenido, como resultado, aparece la reproducción del material dado de alta en el sistema.

Para buscar, eliminar o actualizar un contenido de estudio, el Administrador, deberá seleccionar la opción correspondiente en la interfaz de la Figura 16.

En la Figura 19 se muestra el resultado de la búsqueda.

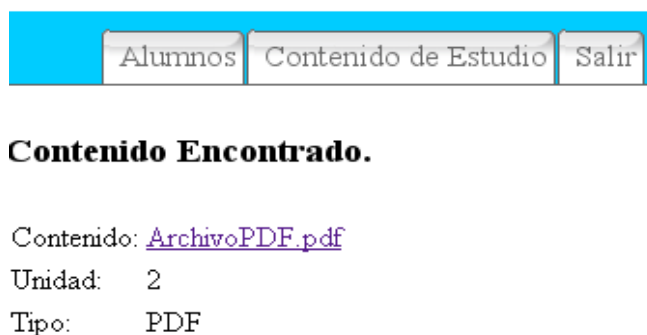


Figura 19. Resultado de la búsqueda de un contenido

Si se cambia el número de unidad, el Sistema pide que el Administrador asocie el contenido con una de las actividades de la nueva unidad tal y como se muestra en la Figura 20.



Contenido ACTualizado.

Asociar contenido con una actividad de la nueva unidad.

Actividad

Figura 20. Asociar a una actividad

3.-Evaluación del Usuario

En este módulo, el Sistema Tutor proporciona las preguntas para la evaluación de los alumnos, cada pregunta viene con tres posibles respuestas, de las cuales, solo una es correcta; tanto las preguntas como las respuestas son adquiridas de la Base de Datos implementada con el script SQL presentado en el Anexo A; un ejemplo de evaluación se muestra en la Figura 21.

Preguntas de Evaluación



¿Que es la herencia?

- Que los atributos de un objeto son heredados a otros
- Acceso a la informacion contenida en otras clases
- Cuando las clases disponende de todos los metodos y propiedades de la clase de la que heredan

¿Cuales son las características de la POO?

- Abstraccion,Encapsulamiento,Herencia y Polimorfismo
- Abstraccion,Declaracion de variables y Metodos
- Ejecucion,Compilacion y Abstraccion

¿Como se implementa el polimorfismo?

- Con solo cambiar el tipo de dato devuelto y cambiar el nombre del metodo
- Se nombran los metodos de la misma forma pero el tipo de dato o los parametros son diferentes
- Con cambiarle el nombre al metodo

¿Que es el polimorfismo?

- Permite que las clases se comporten de formas diferentes
- Permite implementar de diferentes formas un mismo metodo
- Permite que los metodos regresen varios tipos de resultados

Figura 21. Preguntas de Evaluación

Una vez que conteste todas las preguntas, el alumno debe pulsar el botón Evaluar (Figura 22) para saber su calificación

¿Como se crean las clases de objetos?

- Se crean asi: `new class nombreClase{}`
- Se crean asi: `clase nombreClase{}`
- Se crean asi: `class nombreClase{}`

Figura 22. Botón para Evaluación

El resultado de la evaluación depende de la cantidad de respuestas correctas, esto se determina comparando la respuesta seleccionada por el alumno con la respuesta asignada a cada pregunta; si el número de respuestas correctas es mayor a cinco, la evaluación se considera aprobatoria, en caso contrario, el alumno tendrá que repetir la evaluación.

4.- Chat en Línea

En este módulo se utiliza la página de chat en la cual se proporcionan pequeños campos de texto donde el alumno escribirá tanto su login como el texto que compartirá con los demás alumnos.

Cabe resaltar que todo el que se conecte verá la conversación y podrá participar en ella.

En la Figura 23 se muestra dicha página del chat.

Indice Evaluacion Chat Salir

usuario: Hola como estan
usuario: este es un ejemplo del chat
usuario: en el sistema tutor web

Login [usuario] en el sistema tutor web

Figura 23. Chat en línea del Sistema

5.- Guía de Actividades del Usuario.

Este módulo se encarga, por medio de las reglas de producción (almacenadas en la Base de Datos) de planear las actividades que los alumnos deberán seguir; esto es, de acuerdo a sus respuestas, el sistema proporcionará los contenidos de estudio adecuados para un cierto tema.

Preguntas de la unidad.



The screenshot shows a user interface with a blue header bar containing four buttons: 'Indice', 'Evaluacion', 'Chat', and 'Salir'. Below the header, a question is displayed: '¿Conoces los principios basicos de la POO?'. There are two radio button options: 'SI' and 'NO'. Below the 'SI' option is a button labeled 'Aceptar'.

Figura 24. Preguntas de la unidad

En la Figura 24 se puede observar un ejemplo de una regla de producción con sus respuestas, dependiendo de la pregunta y de la respuesta, el Sistema muestra, o bien la siguiente regla o los Contenidos de Estudio correspondientes.

Si se muestran los Contenidos de Estudio, se crea una nueva Actividad con contenidos asociados, los cuales serán mostrados al Alumno cada vez que ingrese a la unidad que estudia actualmente. Una vez estudiados y comprendidos dichos contenidos, el alumno deberá pulsar el botón Actividad Realizada para seguir con los demás temas de la Unidad.

En la siguiente figura, se muestra un ejemplo de los contenidos de estudio desplegados para la regla de producción mostrada en la Figura 24.

Te recomiendo estudies lo siguiente:

¿QUE ES LA POO?

- ❖ Se puede definir **POO** como una técnica o estilo de programación que utiliza objetos como bloque esencial de construcción.
- ❖ La programación orientada a objetos es un modelo de programación que le da más importancia a los datos que a los algoritmos con la finalidad de imitar la forma en la que se percibe el mundo real: Un conjunto de objetos en constante interacción.

[ArchivoPDF.pdf](#)
[ArchivoHTML.html](#)

Si ya estudiaste todo el contenido de estudio marca esta actividad como realizada

Actividad Realizada

Figura 25. Contenidos de Estudio.

6.- Gestión de Usuarios

Además de la validación de usuarios en la página principal del Sistema, descrita anteriormente, este módulo permite la inscripción al Tutorial para dar de alta un nuevo Alumno en el Sistema (validando que no queden campos vacíos).

En la Figura 26 se muestra la página de inscripción al tutorial.

Inscripcion al tutorial.

Login:

Password:

Repetir Password:

E-mail:

Direccion:

Telefono:

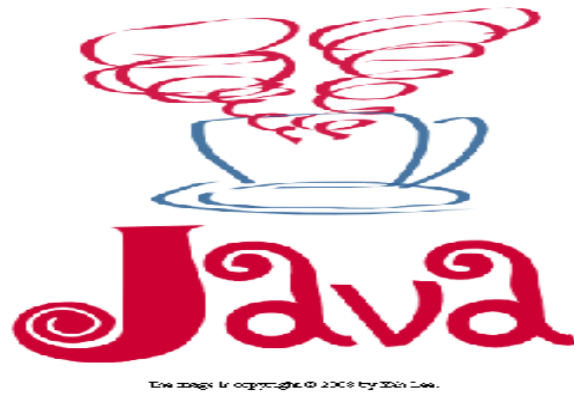


Figura 26. Inscripción al Tutorial

Una vez descrito el Sistema Tutor, se presentan las conclusiones.

CONCLUSIONES.

Razón y Utilidad del Proyecto

El objetivo de Diseñar e Implementar el Sistema Tutor es, proporcionar un medio de aprendizaje, en donde los Alumnos de la UAM-Azcapotzalco puedan estudiar Programación Orientada a Objetos en Java, aprovechando Tecnologías de la Información tales como videos, documentos PDF y HTML, y con un Chat donde los Alumnos pueden intercambiar opiniones y dudas.

Herramientas Utilizadas

En cuanto a las herramientas de desarrollo que se utilizaron, fueron de gran ayuda y facilitaron la elaboración del Sistema, los frameworks Struts e Hibernate, son herramientas vigentes de trabajo, muy utilizados en la práctica profesional; por su parte el manejador MySQL y el servidor Tomcat al ser de libre distribución y fáciles, tanto de instalar como de utilizar, optimizaron el tiempo de desarrollo.

Mejoras Propuestas

Algunas de las mejoras propuestas para el Proyecto Terminal son:

Implementar una interacción que se adecue al perfil de cada estudiante.

Cambio de colores, distribución y dinamismo de las páginas Web.

Líneas de Investigación Derivadas del Proyecto

Algunas líneas de investigación derivadas del presente Proyecto Terminal son:

Diseñar de manera multidisciplinaria, con expertos en el tema, los módulos de “Guía de Actividades del Usuario” y “Evaluación del Usuario”.

Incluir diversos Contenidos Temáticos (ciencias exactas, redes, estructura de datos, etc.)

Conclusión Final

Una de las cosas que aprendí en la elaboración de este Proyecto Terminal es saber integrar los frameworks utilizados, esto es de gran utilidad en mi desarrollo profesional, al igual que las tareas automáticas en MySQL.

El Entorno de Desarrollo Integrado (IDE) Netbeans 6.9, en el cual se desarrolló el Sistema Tutor, proporciona una manera fácil y rápida de programar.

Para la elaboración del Proyecto Terminal integré la información de diversas materias, Base de Datos, Interoperabilidad e Integración de Aplicaciones y Metodologías de Análisis y Diseño de Sistemas de Información.

Este trabajo me motiva a seguir desarrollando proyectos de este tipo, ya que utilicé herramientas de las cuales quiero seguir aprendiendo.

En vista de que el Análisis fue previo a la propuesta de PT y los casos de uso fueron parte de ella, no se realizó el manual del programador que se tenía planeado, la documentación de las clases así como el diagrama respectivo se añadieron a este documento en el Anexo B.

Script utilizado para crear la base de datos

```
CREATE DATABASE IF NOT EXISTS tutorial;
USE tutorial;
--Crear la tabla actividades
DROP TABLE IF EXISTS `tutorial`.`actividades`;
CREATE TABLE `tutorial`.`actividades` (
  `idActividad` int(11) NOT NULL AUTO_INCREMENT,
  `descripcion` varchar(60) NOT NULL,
  `regla` int(11) NOT NULL,
  `unidad` int(11) NOT NULL,
  PRIMARY KEY (`idActividad`)
) ENGINE=MyISAM AUTO_INCREMENT=37 DEFAULT CHARSET=latin1;
INSERT INTO `tutorial`.`actividades` VALUES (1,'Introduccion a la POO',3,1),
(2,'Las clases',5,1),
(3,'Los objetos',7,1),
(4,'Los metodos',9,1),
(5,'Los mensajes',11,1),
(6,'Caracteristicas de la POO',14,2),
(7,'La herencia',15,2),
(8,'La abstraccion',18,2),
(9,'El encapsulamiento',20,2),
(10,'El polimorfismo',21,2),
(11,'Constructores',25,3),
(12,'Sobrecarga de constructor',27,3),
(13,'Llamar al constructor',28,3),
(14,'Asignar un objeto',30,3),
(15,'Constructor copia',33,3),
(16,'Destruccion',35,3),
(17,'Recolector de basura',36,3),
(18,'Alcance de las clases',40,4),
(19,'Acceso a los miembros',41,4),
(20,'Referencia this',44,4),
(21,'Miembros Static',45,4),
(22,'Miembros final',48,4),
(23,'Metodos Sobrecargados',49,4),
(24,'Metodos get y set',52,4),
(25,'Sub-Clases y Super-Clases',54,5),
(26,'Definicion Sub-Clase',57,5),
(27,'Redefinicion de metodos',59,5),
(28,'Constructores y destructores de la sub-clase',61,5),
(29,'Referencias de objetos',63,5),
(30,'Sobreescribir metodos',65,5),
(31,'Clases Abstractas',68,6),
(32,'Implementacion de Abstractas',69,6),
(33,'Definir interfaces',72,6),
(34,'Utilizar Interfaces',74,6),
(35,'Interfaces como herencia multiple',75,6),
(36,'Multiples interfaces',78,6);

--Crear la tabla actividades_realizadas
DROP TABLE IF EXISTS `tutorial`.`actividades_realizadas`;

CREATE TABLE `tutorial`.`actividades_realizadas` (
  `idActividad_realizada` int(11) NOT NULL AUTO_INCREMENT,
  `login_usuario` varchar(10) NOT NULL,
  `idActividad` int(11) NOT NULL,
  `numero_unidad` int(11) NOT NULL,
  `realizada` int(11) NOT NULL,
  PRIMARY KEY (`idActividad_realizada`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```

--Crear la tabla alumnos
DROP TABLE IF EXISTS `tutorial`.`alumnos`;
CREATE TABLE `tutorial`.`alumnos` (
  `login` varchar(10) NOT NULL,
  `password` varchar(20) NOT NULL,
  `mail` varchar(25) NOT NULL,
  `direccion` varchar(20) NOT NULL,
  `numero_unidad` int(11) NOT NULL,
  `ultimo_acceso` date NOT NULL,
  `tipo` int(11) NOT NULL,
  `telefono` varchar(20) NOT NULL,
  PRIMARY KEY (`login`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--Crear al admin
INSERT INTO `tutorial`.`alumnos` VALUES ('admin','admin','admin@admin.com','direccion del admin',1,'2011-02-15',1,'53530292');
DROP TABLE IF EXISTS `tutorial`.`contenidos`;

--Crear la tabla contenidos
CREATE TABLE `tutorial`.`contenidos` (
  `idContenido` int(11) NOT NULL AUTO_INCREMENT,
  `idActividad` int(11) NOT NULL,
  `url` varchar(100) NOT NULL,
  `nombre` varchar(100) NOT NULL,
  `unidad` int(11) NOT NULL,
  `tipo` varchar(50) NOT NULL,
  `nombreO` varchar(100) NOT NULL,
  PRIMARY KEY (`idContenido`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--Crear la tabla evaluaciones_alumnos
DROP TABLE IF EXISTS `tutorial`.`evaluaciones_alumnos`;
CREATE TABLE `tutorial`.`evaluaciones_alumnos` (
  `idEvaluacion` int(11) NOT NULL AUTO_INCREMENT,
  `login_usuario` varchar(10) NOT NULL,
  `numero_unidad` int(11) NOT NULL,
  `calificacion` int(11) NOT NULL,
  PRIMARY KEY (`idEvaluacion`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--Crear la tabla preguntas
DROP TABLE IF EXISTS `tutorial`.`preguntas`;
CREATE TABLE `tutorial`.`preguntas` (
  `idPregunta` int(11) NOT NULL AUTO_INCREMENT,
  `numero_unidad` int(11) NOT NULL,
  `pregunta` varchar(100) NOT NULL,
  `respuesta_correcta` varchar(150) NOT NULL,
  PRIMARY KEY (`idPregunta`)
) ENGINE=MyISAM AUTO_INCREMENT=61 DEFAULT CHARSET=latin1;
INSERT INTO `tutorial`.`preguntas` VALUES (1,1,'¿Que es la programacion orientada a objetos?','Es un modelo de programacion que utiliza objetos, ligados mediante mensajes, para la solucion de problemas'),
(2,1,'¿Que es un mensaje?','Es una forma de comunicacion entre los objetos para ejecutar una accion'),
(3,1,'¿Que es un objeto?','Es una encapsulacion generica de datos y de los metodos para manipularlos'),
(4,1,'¿Que es una clase?','Es un tipo de objeto definido por el usuario'),
(6,1,'¿Como es la forma en que un metodo devuelve un valor?','Se debe declarar el tipo de dato en la declaracion del metodo y se regresa con return'),
(5,1,'¿Que son los metodos?','Es la forma que determina como tiene que actuar el objeto cuando recibe un mensaje'),
(10,1,'¿Cuales son los mecanismos basicos de la POO?','Objetos,Mensajes,Metodos y Clases'),
(9,1,'¿Que es lo que define el comportamiento de una clase?','Los metodos'),
(8,1,'¿Cual es el metodo principal en una clase en Java?','Es el metodo main'),
(7,1,'¿Como se crean las clases de objetos?','Se crean asi: class nombreClase {}),

```

(12,2,'¿Que es la herencia?','Cuando las clases disponen de todos los metodos y propiedades de la clase de la que heredan'),

(11,2,'¿Cuales son las características de la POO?','Abstraccion,Encapsulamiento,Herencia y Polimorfismo'),

(17,2,'¿Como se implementa el polimorfismo?','Se nombran los metodos de la misma forma pero el tipo de dato devuelto o los parametros son diferentes'),

(16,2,'¿Que es el polimorfismo?','Permite implementar de diferentes formas un mismo metodo'),

(15,2,'El encapsulamiento nos ayuda a:','Ver un objeto como una caja negra, permitiendo ocultar su estructura'),

(14,2,'¿Para que sirve la abstraccion?','Para generalizar y centrarse en los aspectos que permitan tener una vision global del problema'),

(13,2,'¿De que forma podemos acceder a un metodo de la super clase?','Con la palabra clave super'),

(30,3,'¿Como se implementa el destructor de un objeto?','protected void finalize() throws throwable{}'),

(29,3,'¿Que es la asignacion de un objeto?','Es cuando dos objetos hacen referencia a un solo objeto'),

(28,3,'¿Que es un constructor copia?','Es cuando a un objeto se le asigna otro objeto de su misma clase al crearlo'),

(27,3,'¿Que es el constructor predeterminado?','Es el constructor que se ejecuta en caso de no escribir uno propio'),

(26,3,'¿Como invocar al recolector de basura?','System.gc();'),

(25,3,'¿Como funciona el recolector de basura?','Es un subproceso que se ejecuta paralelamente a una aplicacion y limpia los objetos de la memoria'),

(24,3,'¿Para que sirve el destructor?','Para destruir un objeto liberando recursos'),

(23,3,'¿Como se llama al constructor de un objeto?','nombreClase nombreObjeto = new NombreClase();'),

(22,3,'¿Para que sirva la sobrecarga del constructor?','Para iniciar un objeto de diferentes maneras'),

(21,3,'¿Que es un constructor?','Es un procedimiento especial de una clase que es llamado automaticamente cuando se crea un objeto'),

(20,2,'La herencia se implementa de la siguiente forma:','public class nombreClase extends nombreSuperClase{}'),

(19,2,'¿Como se llama la clase de la que se hereda?','Superclase'),

(18,2,'¿En java se puede implementar la herencia multiple?','No es posible implementarla'),

(36,4,'¿Cual es la caracteristicade los miembros static?','Un metodo static carece de la referencia this por lo que no puede ser invocado por un objeto de su clase'),

(35,4,'¿Para que sirve la referencia this?','Un mecanismo para que un metodo conozca el objeto para el que ha sido invocado'),

(34,4,'¿Que es el acceso a los miembros?','Es la forma en que se puede acceder a los atributos de una clase'),

(33,4,'¿Cual es la diferencia entre el acceso privado y el protegido?','El acceso protegido se comporta como publico para metodos de las clases del mismo paquete'),

(32,4,'¿Cuantos tipos de acceso existen?','Publico,Privado y Protegido'),

(31,4,'¿Que es el alcance de las clases?','Es la forma de determinar la relacion que tiene una clase con otras del mismo paquete'),

(37,4,'¿Que es un miembro final?','Es un miembro cuyo valor no puede ser modificado'),

(50,5,'¿Que nos permiten los metodos protected?','Nos ofrece un nivel intermedio de acceso entre publico y privado'),

(49,5,'¿Se puede acceder a los miembros privados de la superclase?','Solo atravez de los metodos de la superClase'),

(48,5,'¿Cual de las siguientes relaciones representa a la herencia?','Es un'),

(47,5,'Si se tiene una clase llamada figura cual de las siguientes declaraciones es incorrecta','class rectangulo extends Figura{} class Cuadrado extends Figura,Rectangulo{}'),

(46,5,'¿Cuales son los tipos de conversion de objetos?','Implicitas y explicitas'),

(45,5,'¿Como se define una subclase?','class subClase extends superClase{}'),

(44,5,'¿Como se implementa el destructor de una subclase?','protected void finalize() throws throwable{ super.finalize(); }'),

(43,5,'¿Como se implementa el constructor de una subclase?','subclase(parametros){ super(parametros); }'),

(42,5,'¿Como se redefinen los metodos?','Se escribe @override y se declara el metodo como en la superclase y se implementa su funcionalidad'),

(41,5,'¿Que miembros hereda una clase?','Hereda todos los miembros de su superclase '),

(40,4,'¿Cual de los siguientes ejemplos de sobrecarga es correcto?','bool metodo(){ bool metodo(parametro1){ bool metodo(parametro1,parametro2){ }')'),

(39,4,'¿Para que sirven los metodos get y set?','Para asignar y obtener los valores de los atributos de los objetos'),

(38,4,'¿Que son los metodos sobrecargados?','Son los metodos con el mismo nombre, pero con distinto numero de parametros'),

(60,6,'¿Donde se implementan los metodos de las interfaces?','Se implementan en la clase que utiliza la interfaz'),

(59,6,'¿Como deben ser los metodos de las interfaces?','Todos los metodos deben ser abstractos'),

(58,6,'¿Como se implementa la herencia multiple con interfaces?','class nombreClase implements interfaz1, interfaz2,...,interfazn{}'),

(57,6,'¿Como se utiliza una interfaz?','class nombreClase implements nombreInterfaz{}'),

(56,6,'¿Como se define una interfaz?','interface nombreInterfaz{}'),

(55,6,'¿Que es una interfaz?','Una forma de darle a una clase diferente comportamiento'),

(54,6,'¿Cuando se implementan los metodos abstractos de la clase abstracta?','Dentro de una clase que hereda de la clase abstracta'),

(53,6,'¿Como deben ser los metodos de una clase abstracta?','Pueden contener metodos normales y abstractos'),

(52,6,'¿Como se define una clase abstracta?','public abstract class nombreClase{}'),

(51,6,'¿Que es una clase abstracta?','Son clases que se utilizan para ser superclases pero que no instancian objetos');

--Crear la tabla reglas_produccion

DROP TABLE IF EXISTS `tutorial`.`reglas_produccion`;

CREATE TABLE `tutorial`.`reglas_produccion` (
 `idRegla` int(11) NOT NULL AUTO_INCREMENT,
 `unidad` int(11) NOT NULL,
 `sig_reglaS` int(11) NOT NULL,
 `pregunta` varchar(100) NOT NULL,
 `sig_reglaN` int(11) NOT NULL,
 PRIMARY KEY (`idRegla`)

) ENGINE=MyISAM AUTO_INCREMENT=79 DEFAULT CHARSET=latin1;

INSERT INTO `tutorial`.`reglas_produccion` VALUES (1,1,2,'¿Conoces los principios basicos de la POO?',3),

(2,1,4,'¿Sabes lo que es una clase?',5),

(3,1,2,'Entonces te recomiendo estudios lo siguiente:',0),

(4,1,6,'¿Sabes lo que es un objeto?',7),

(5,1,4,'Entonces te recomiendo estudios lo siguiente:',0),

(6,1,8,'¿Sabes lo que son los metodos?',9),

(7,1,6,'Entonces te recomiendo estudios lo siguiente:',0),

(8,1,10,'¿Sabes que son los mensajes?',11),

(9,1,8,'Entonces te recomiendo estudios lo siguiente:',0),

(10,1,0,'Entonces te recomiendo realizar la evaluacion',0),

(11,1,10,'Entonces te recomiendo estudios lo siguiente:',0),

(12,2,13,'¿Conoces las caracteristicas de la POO?',14),

(13,2,15,'¿Quieres estudiar el concepto de herencia?',16),

(14,2,13,'Entonces te recomiendo estudios lo siguiente:',0),

(15,2,16,'Entonces te recomiendo estudios lo siguiente:',0),

(16,2,17,'¿Sabes lo que es la abstraccion?',18),

(17,2,19,'¿Sabes como funciona el encapsulamiento?',20),

(18,2,17,'Entonces te recomiendo estudios lo siguiente:',0),

(19,2,21,'¿Quieres saber como funciona el polimorfismo?',22),

(20,2,19,'Entonces te recomiendo estudios lo siguiente:',0),

(21,2,22,'Entonces te recomiendo estudios lo siguiente:',0),

(22,2,0,'Entonces te recomiendo realizar la evaluacion',0),

(23,3,24,'¿Sabes para que sirve un constructor?',25),

(24,3,26,'¿Sabes como funciona la sobre carga del constructor?',27),

(25,3,24,'Entonces te recomiendo estudios lo siguiente:',0),

(26,3,28,'¿Quieres saber como se llama a un constructor?',29),

(27,3,26,'Entonces te recomiendo estudios lo siguiente:',0),

(28,3,29,'Entonces te recomiendo estudios lo siguiente:',0),

(29,3,30,'¿Quieres saber como asignar un objeto?',31),

(30,3,31,'Entonces te recomiendo estudios lo siguiente:',0),

(31,3,32,'¿Saber lo que es un constructor copia?',33),

(32,3,34,'¿Sabes lo que es un destructor?',35),

(33,3,32,'Entonces te recomiendo estudios lo siguiente:',0),

(34,3,36,'¿Quieres saber como llamar al recolector de basura?',37),

(35,3,34,'Entonces te recomiendo estudios lo siguiente:',0),

(36,3,37,'Entonces te recomiendo estudios lo siguiente:',0),

(37,3,0,'Entonces te recomiendo realizar la evaluacion',0),

(38,4,39,'¿Sabes lo que es el alcance de las clases?',40),

(39,4,41,'¿Quieres saber que es el control de acceso y sus tipos?',42),

(40,4,39,'Entonces te recomiendo estudios lo siguiente:',0),

(41,4,42,'Entonces te recomiendo estudios lo siguiente:',0),

(42,4,43,'¿Sabes como funciona la referencia this?',44),

(43,4,45,'¿Quieres saber como funcionan los miembros static?',46),

(44,4,43,'Entonces te recomiendo estudios lo siguiente:',0),

(45,4,46,'Entonces te recomiendo estudios lo siguiente:',0),

(46,4,47,'¿Sabes como funcionan los miembros de instancia finales?',48),

(47,4,49,'¿Quieres saber como funcionan los metodos sobrecargados?',50),

(48,4,47,'Entonces te recomiendo estudios lo siguiente:',0),
 (49,4,50,'Entonces te recomiendo estudios lo siguiente:',0),
 (50,4,51,'¿Sabes como funcionan los metodos get y set?',52),
 (51,4,0,'Entonces te recomiendo realizar la evaluacion',0),
 (52,4,51,'Entonces te recomiendo estudios lo siguiente:',0),
 (53,5,54,'¿Quieres saber lo que son las sub-clases y super-clases?',55),
 (54,5,55,'Entonces te recomiendo estudios lo siguiente:',0),
 (55,5,56,'¿Sabes como se define una sub-clase y como funciona?',57),
 (56,5,58,'¿Conoces la redefinicion de metodos de la super-clase?',59),
 (57,5,56,'Entonces te recomiendo estudios lo siguiente:',0),
 (58,5,60,'¿Sabes como funcionan los constructores y destructores de las sub-clases?',61),
 (59,5,58,'Entonces te recomiendo estudios lo siguiente:',0),
 (60,5,62,'¿Sabes como se hacen las referencias a objetos de una sub-clase o viceversa?',63),
 (61,5,60,'Entonces te recomiendo estudios lo siguiente:',0),
 (62,5,64,'¿Sabes como sobre escribir metodos para usar polimorfismo?',65),
 (63,5,62,'Entonces te recomiendo estudios lo siguiente:',0),
 (64,5,0,'Entonces te recomiendo realizar la evaluacion',0),
 (65,5,64,'Entonces te recomiendo estudios lo siguiente:',0),
 (66,6,67,'¿Sabes como se definen las clases abstractas?',68),
 (67,6,69,'¿Quieres saber como se implementan las clases abstractas?',70),
 (68,6,67,'Entonces te recomiendo estudios lo siguiente:',0),
 (69,6,70,'Entonces te recomiendo estudios lo siguiente:',0),
 (70,6,71,'¿Sabes como se definen las interfaces?',72),
 (71,6,73,'¿Sabes como utilizar las interfaces?',74),
 (72,6,71,'Entonces te recomiendo estudios lo siguiente:',0),
 (73,6,75,'¿Quieres saber como utilizar interfaces como herencia multiple?',76),
 (74,6,73,'Entonces te recomiendo estudios lo siguiente:',0),
 (75,6,76,'Entonces te recomiendo estudios lo siguiente:',0),
 (76,6,77,'¿Sabes implementar multiples interfaces?',78),
 (77,6,0,'Entonces te recomiendo realizar la evaluacion',0),
 (78,6,77,'Entonces te recomiendo estudios lo siguiente:',0);

--Crear la tabla respuestas

```
DROP TABLE IF EXISTS `tutorial`.`respuestas`;
```

```
CREATE TABLE `tutorial`.`respuestas` (  
  `idRespuesta` int(11) NOT NULL AUTO_INCREMENT,  
  `idPregunta` int(11) NOT NULL,  
  `respuesta` varchar(150) NOT NULL,  
  PRIMARY KEY (`idRespuesta`)
```

```
) ENGINE=MyISAM AUTO_INCREMENT=182 DEFAULT CHARSET=latin1;
```

```
INSERT INTO `tutorial`.`respuestas` VALUES (1,1,'Es un modelo de programacion que utiliza objetos, ligados mediante mensajes, para la solucion de problemas'),
```

```
(2,1,'Un modelo para simular los objetos cotidianos'),
```

```
(6,2,'Es cuando un objeto reporta un error'),
```

```
(5,2,'Es una llamada de atencion'),
```

```
(4,2,'Es una forma de comunicacion entre los objetos para ejecutar una accion'),
```

```
(3,1,'Una forma de programar sin problemas'),
```

```
(31,11,'Abstraccion,Encapsulamiento,Herencia y Polimorfismo'),
```

```
(30,10,'Datos,Estructuras,Clases y Metodos'),
```

```
(29,10,'Objetos,Interfaces y Operadores'),
```

```
(28,10,'Objetos,Mensajes,Metodos y Clases'),
```

```
(27,9,'Los parametros'),
```

```
(26,9,'Los atributos'),
```

```
(25,9,'Los metodos'),
```

```
(24,8,'No hay metodo principal'),
```

```
(23,8,'El elmetodo que el programador indique'),
```

```
(22,8,'Es el metodo main'),
```

```
(21,7,'Se crean asi: new class nombreClase {}'),
```

```
(20,7,'Se crean asi: clase nombreClase {}'),
```

```
(19,7,'Se crean asi: class nombreClase {}'),
```

```
(18,6,'Los metodos no regresan valores'),
```

```
(17,6,'Solo se regresa el valor con return'),
```


(16,6,'Se debe declarar el tipo de dato en la declaracion del metodo y se regresa con return'),
(15,5,'El modelo en que esta construido el objeto'),
(14,5,'La forma de interpretar un mensaje'),
(13,5,'Es la forma que determina como tiene que actuar el objeto cuando recibe un mensaje'),
(12,4,'Encapsulacion de datos'),
(11,4,'Es una forma de clasificar los objetos'),
(10,4,'Es un tipo de objeto definido por el usuario'),
(9,3,'Implementacion de los datos'),
(8,3,'Es una abstraccion de las cosas vivas '),
(7,3,'Es una encapsulacion generica de datos y de los metodos para manipularlos'),
(43,15,'Ver un objeto como una caja negra, permitiendo ocultar su estructura'),
(42,14,'Para implementar los metodos'),
(41,14,'Para hacer que la construccion de las clases sean mas facil'),
(40,14,'Para generalizar y centrarse en los aspectos que permitan tener una vision global del problema'),
(39,13,'No hay forma de acceder a un metodo de la superclase'),
(38,13,'Con la palabra clave Arriba'),
(37,13,'Con la palabra clave super'),
(36,12,'Acceso a la informacion contenida en otras clases'),
(35,12,'Que los atributos de un objeto son heredados a otros '),
(34,12,'Cuando las clases disponende de todos los metodos y propiedades de la clase de la que heredan'),
(33,11,'Ejecucion,Compilacion y Abstraccion'),
(32,11,'Abstraccion,Declaracion de variables y Metodos'),
(90,30,'private void finalize() {}'),
(89,30,'private int finalize() throws throwable {}'),
(88,30,'protected void finalize() throws throwable {}'),
(87,29,'Es usar el constructor de otro objeto para iniciar otro'),
(86,29,'Es copiar los datos de un objeto a otro'),
(85,29,'Es cuando dos objetos hacen referencia a un solo objeto'),
(84,28,'Es el constructor sin argumentos asignado a un objeto'),
(83,28,'Es la copia del constructor predeterminado de un objeto'),
(82,28,'Es cuando a un objeto se le asigna otro objeto de su misma clase al crearlo'),
(81,27,'No existe un constructor predeterminado'),
(80,27,'Es el unico constructor que debe ser invocado'),
(79,27,'Es el constructor que se ejecuta en caso de no escribir uno propio'),
(78,26,'finalize.gc();'),
(77,26,'garbageColector.run();'),
(76,26,'System.gc();'),
(75,25,'Es un proceso que se ejecuta paralelamente y borra todos los datos de un objeto'),
(74,25,'Borra todos los objetos de una aplicacion '),
(73,25,' Es un subproceso que se ejecuta paralelamente a una aplicacion y limpia los objetos de la memoria'),
(72,24,'Un procedimiento utilizado para destruir una clase'),
(71,24,'Un procedimiento que borra los datos inicializados de un objeto '),
(70,24,'Para destruir un objeto liberando recurzos'),
(68,23,'new nombreObjeto::nombreClase();'),
(67,23,'nombreClase nombreObjeto = new nombreClase();'),
(69,23,'nombreObjeto = new nombreClase();'),
(66,22,'Para iniciar los metodos mas importantes de un objeto'),
(65,22,'Para iniciar un objeto de acuerdo a la situacion'),
(64,22,'Para iniciar un objeto de diferentes maneras'),
(63,21,'Es un procedimiento que le da valores a los metodos'),
(62,21,'Es un procedimiento que ensambla y da valores a los elementos de un objeto'),
(61,21,'Es un procedimiento especial de una clase que es llamado automaticamente cuando se crea un objeto'),
(60,20,'public class extends nombreSuperClass::nombreClase {}'),
(59,20,'public class nombreClase::nombreSuperClase {}'),
(58,20,'public class nombreClase extends nombreSuperClase {}'),
(57,19,'Over clase'),
(56,19,'Clase de arriba'),
(55,19,'Superclase'),
(54,18,'Solo en casos especiales'),
(53,18,'No es posible implementarla'),
(52,18,'Si es posible implementarla'),
(51,17,'Con cambiarle el nombre al metodo'),

(50,17,'Con solo cambiar el tipo de dato devuelto y cambiar el nombre del metodo'),
(49,17,'Se nombran los metodos de la misma forma pero el tipo de dato o los parametros son diferentes'),
(48,16,'Permite que las clases se comporten de formas diferentes'),
(47,16,'Permite que los metodos regresen varios tipos de resultados'),
(46,16,'Permite implementar de diferentes formas un mismo metodo'),
(45,15,'Permite organizar las aplicaciones en forma de paquetes'),
(44,15,'Permite agrupar los objetos en paquetes'),
(111,37,'No existen los miembros final'),
(110,37,'Es un miembro cuyo valor siempre cambia'),
(109,37,'Es un miembro cuyo valor no puede ser modificado'),
(108,36,'Un metodo static solo es declarado en clases abstractas'),
(107,36,'Un metodo static no puede hacer uso de los atributos del objeto'),
(106,36,'Un metodo static carece de la referencia this por lo que no puede ser invocado por un objeto de su clase'),
(105,35,'Es la forma de acceder a los metodos y atributos de un objeto diferente al que lo invoca'),
(104,35,'Para invocar varios metodos de las clases del mismo paquete'),
(103,35,'Un mecanismo para que un metodo conozca el objeto para el que ha sido invocado'),
(102,34,'Hacer que el programador no tenga acceso a los atributos'),
(101,34,'Es la forma en que se puede acceder a los atributos de una clase'),
(100,34,'Los atributos de una clase no pueden ser accedidos directamente'),
(99,33,'El acceso protegido se comporta como privado para metodos de las clases del mismo paquete'),
(98,33,'No existe diferencia entre ellos'),
(97,33,'El acceso protegido se comporta como publico para metodos de las clases del mismo paquete'),
(96,32,'Privado,Protegido y REferenciado'),
(95,32,'Publico,de Paquete y Expuesto'),
(94,32,'Publico,Privado y Protegido'),
(93,31,'Es la forma de determinar en donde puede ser invocada una clase'),
(92,31,'Es la forma de determinar en que se deben declarar las clases'),
(91,31,'Es la forma de determinar la relacion que tiene una clase con otras del mismo paquete'),
(133,45,'class subClase extends superClase {}'),
(132,44,'protected void finalize() throws throwable{ return finalize(); }'),
(131,44,'protected void finalize(super.finalize()) throws throwable {}'),
(130,44,'protected void finalize() throws throwable{ super.finalize(); }'),
(129,43,'subClase(super(parametros)) {}'),
(128,43,'superClase(parametros){ subclase(parametros); }'),
(127,43,'subclase(parametros){ super(parametros); }'),
(126,42,'Se escribe el nuevo metodo y dentro de este se declara el metodo de la superclase'),
(125,42,'No se pueden redefinir los metodos de la superclase'),
(124,42,'Se escribe @override y se declara el metodo como en la superclase y se implementa su funcionalidad'),
(123,41,'Hereda solo a los miembros que son private'),
(122,41,'Hereda solamente a los constructores'),
(121,41,'Hereda todos los miembros de su superclase '),
(120,40,'bool metodo(){} int metodo(parametro1,parametro2){} object metodo(parametro2){}'),
(119,40,'bool metodo(){} bool metodo2(parametro1){} int metodo(){}'),
(118,40,'bool metodo(){} bool metodo(parametro1){} bool metodo(parametro1,parametro2){}'),
(117,39,'Para acceder a otros metodos de los objetos'),
(116,39,'Para borrar los valores de los atributos de los objetos'),
(115,39,'Para asignar y obtener los valores de los atributos de los objetos'),
(114,38,'Son los metodos con el mismo nombre y con el mismo numero de parametros'),
(113,38,'Son los metodos con diferente nombre, pero con el mismo numero de parametros'),
(112,38,'Son los metodos con el mismo nombre, pero con distinto numero de parametros'),
(181,60,'Se implementan en una clase que herede de la que implementa la interfaz'),
(180,60,'Se implementan en la clase que hereda de la interfaz'),
(179,60,'Se implementan en la clase que utiliza la interfaz'),
(178,59,'Puede tener metodos normales y abstractos'),
(177,59,'Todos los metodos deben ser implementados'),
(176,59,'Todos los metodos deben ser abstractos'),
(175,58,'class nombreClase implements(interfaz1,interfaz2,...,interfazn) {}'),
(174,58,'class nombreClase implements interfaz1 implements interfaz2 ... interfazn {}'),
(173,58,'class nombreClase implements interfaz1, interfaz2,...,interfazn {}'),
(172,57,'implements extends nombreClase::nombreInterfaz {}'),
(171,57,'implements nombreInterface in class nombreClase {}'),
(170,57,'class nombreClase implements nombreInterfaz {}'),

(169,56,'nombreInterfaz extends interface{}'),
 (168,56,'class interface nombreInterfaz{}'),
 (167,56,'interface nombreInterfaz{}'),
 (165,55,'Una forma de ejecutar codigo de manera dinamica'),
 (164,55,'Una forma de comunicar diferentes clases con el mismo comportamiento'),
 (163,55,'Una forma de darle a una clase diferente comportamiento'),
 (162,54,'Nunca se implementan'),
 (161,54,'En la implementacion de la clase abstracta'),
 (160,54,'Dentro de una clase que hereda de la clase abstracta'),
 (159,53,'Pueden contener metodos normales y abstractos'),
 (158,53,'No puede haber metodos abstractos'),
 (157,53,'Todos los metodos deber ser abstractos'),
 (156,52,'abstract public nombreClase{}'),
 (155,52,'public class nombreClase abstract{}'),
 (154,52,'public abstract class nombreClase{}'),
 (153,51,'Son clases cuyos metodos son implementados'),
 (152,51,'Son clases que se utilizan para instanciar objetos abstractos'),
 (151,51,'Son clases que se utilizan para ser superClases pero que no instancian objetos'),
 (150,50,'Nos ofrece acceso a los atributos de las clases abstractas'),
 (149,50,'Nos ofrece un nivel alto de encapsulamiento'),
 (148,50,'Nos ofrece un nivel intermedio de acceso entre publico y privado'),
 (147,49,'Solo atravez de los metodos de la superClase'),
 (146,49,'No es posible'),
 (145,49,'Si es posible'),
 (144,48,'Contiene un'),
 (143,48,'Tiene un'),
 (142,48,'Es un'),
 (141,47,'class Rectangulo extends Figura{} class Cuadrado extends Figura,Rectangulo{}'),
 (140,47,'class Cuadrado extends Figura{} class Rectangulo extends Cuadrado{}'),
 (139,47,'class Cuadrado extends Figura{}'),
 (138,46,'Internas y Externas'),
 (137,46,'Publicas,Privadas y Protegidas'),
 (136,46,'Implicitas y explicitas'),
 (135,45,'superClase::extends subClase{}'),
 (134,45,'class superClase extends subClase{}');

--Tarea programada borrar alumnos inactivos

CREATE EVENT 'borrar_alumnos'

ON SHEDUKLE

EVERY 1 DAY

DO

delete alumnos,actividades_realizadas,evaluaciones_alumnos from alumnos,actividades_realizadas,evaluaciones_alumnos

where current_date() - ultimo_acceso > 15 and alumnos.login = actividades_realizadas.login_usuario and alumnos-login = evaluaciones_alumnos.login_usuario

Anexo B.

1.- SELECCIÓN DE LAS CLASES

1.1.- Clases elegidas

Las Clases que conforman el Sistema Tutor son las siguientes:

- a) Alumnos
- b) ActividadesRealizadas
- c) EvaluacionesAlumnos
- d) Actividades
- e) ReglasDeProduccion
- f) ContenidosDeEstudio
- g) Preguntas
- h) Respuestas

1.2.-Diccionario de Clases

- a) **Alumnos.-** Contiene la información de los alumnos, como login, e-mail, unidad, etc.
- b) **ActividadesRealizadas.-** Guarda la información de actividades que ha realizado el alumno durante el tutorial.
- c) **EvaluacionesAlumnos.-** Guarda las calificaciones de los alumnos.
- d) **Actividades.-** Se refiere a todas las posibles acciones que un usuario puede realizar durante el estudio de los Contenidos de Estudio.
- e) **ReglasDeProduccion.-** Es aquella que contiene la pregunta y actividades asociadas a la Actividad actual del Alumno.
- f) **ContenidosDeEstudio.-** Contiene información de los Contenidos de Estudio, como nombre, unidad, etc.
- g) **Preguntas.-** Contiene la pregunta y respuesta para evaluar una unidad.
- h) **Respuestas.-** Contiene aquellas respuestas tanto incorrectas como a la respuesta correcta para alguna pregunta de evaluación.

1.3.- Diagrama de Clases.

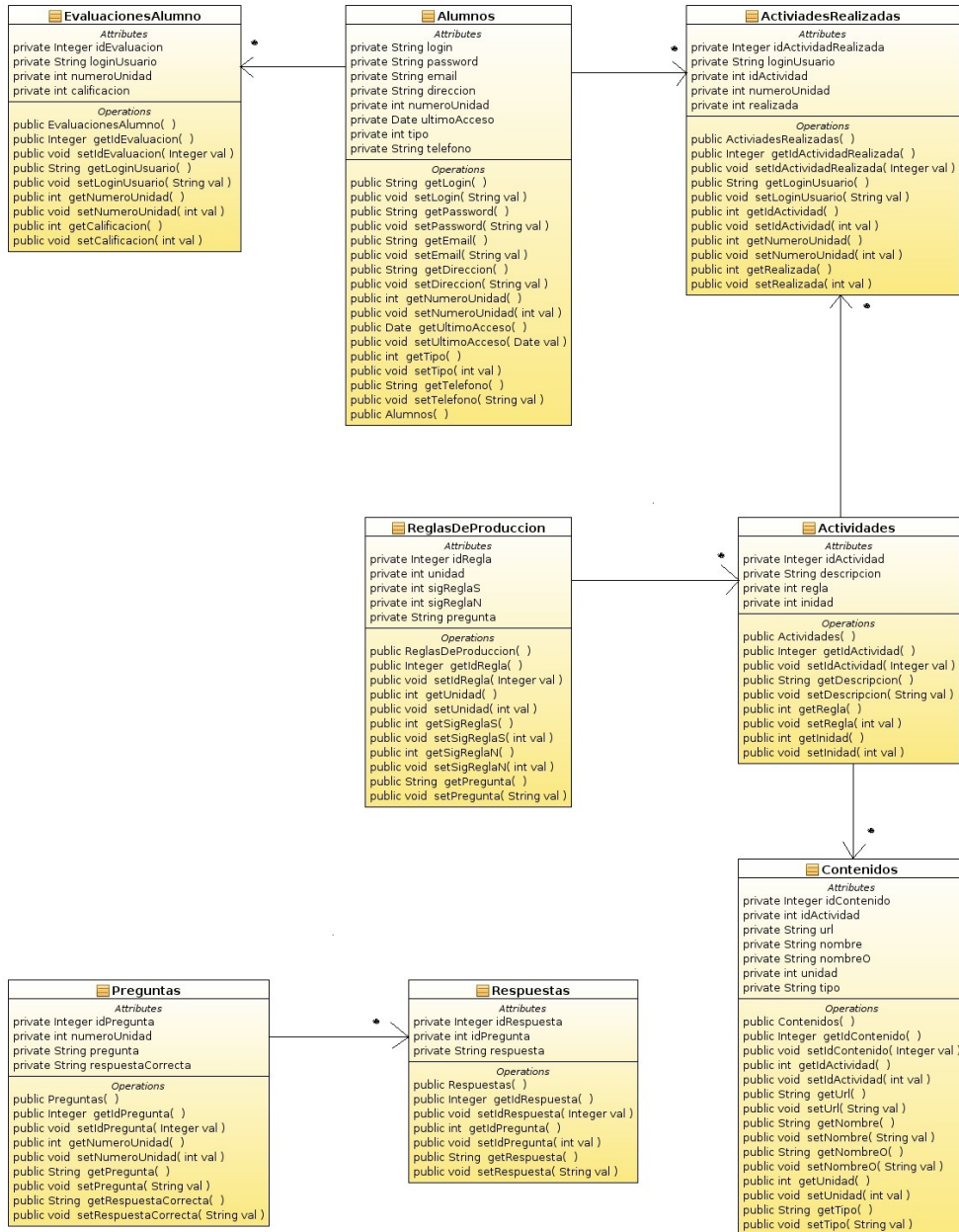


Figura 27. Diagrama de Clases

UNIVERSIDAD AUTÓNOMA
METROPOLITANA

UNIDAD AZCAPOTZALCO

División de Ciencias Básicas e Ingeniería
Licenciatura en Ingeniería en Computación

Guía del Administrador

*“Sistema Tutor Web para el Aprendizaje de
Programación Orientada a Objetos”*

Alumno: Juan José Perea Sánchez

Matricula: 2042040030

Asesor: Irma Fernanda Ardón Pulido

Tabla de Contenido

Introducción.....	3
1.- Ingreso al Sistema.....	3
2.- Operaciones Administrativas con Alumnos.....	4
3.- Operaciones Administrativas con Contenidos de Estudio.....	7
4.- Salir del Sistema.....	11

INTRODUCCIÓN

El presente documento muestra la forma de usar el Sistema Tutor Web para las operaciones administrativas, las cuales están divididas tanto para Alumnos como para Contenidos de Estudio, detallando todos los procesos que pueden ser de utilidad.

1.- INGRESO AL SISTEMA

Para que el Administrador ingrese, debe ir a la Página Principal del Sistema Tutor (Figura uno) e ingresar login y password (ambos como "admin") si alguno o los dos datos son incorrectos el Sistema le indicara un error (Figura dos).

Tutor Virtual Programacion Orientada a Objetos en JAVA

Para entrar al curso, escribe tu login y password

Login:
Password:

Deseas inscribirte haz click [aquí](#)



Figura1. Página Principal

Login:
Password:

"El password es incorrecto"

Figura 2.Password incorrecto

Si los datos son correctos, el sistema mostrará la Página Principal de Administración la cual contiene el Menú Principal (siguiente figura).



Figura 3. Menú Principal

2.- OPERACIONES ADMINISTRATIVAS CON ALUMNOS

En la Página Principal, además del menú, se muestran las tareas administrativas para los alumnos, en donde el Administrador por medio de un formulario (Figura cuatro) tiene las opciones de Buscar, Actualizar o Borrar.

Operaciones con alumnos

Login:

Buscar Actualizar Borrar

Figura 4. Formulario para operaciones con alumnos

Para Buscar un alumno se debe ingresar su login en el campo correspondiente y seleccionar la opción Buscar, posteriormente pulsar el botón Ejecutar, al hacerlo, el Sistema buscará al Alumno (en caso de no existir, se muestra el siguiente mensaje de error).

- "El alumno no existe"

Figura 5. No existe el alumno

Si no ocurre ningún error, el Sistema muestra los datos del alumno (Figura seis).

Listar Alumno

Login	E-Mail	Direccion	Telefono
usuario	usuario@correo.com	una direccion	53530298

Figura 6. Listar alumno

Como se puede ver en la figura anterior, se proporciona la opción Expediente, con la cual el Administrador puede consultar las evaluaciones realizadas por el Alumno como se muestra en la Figura siete.

Expediente

Unidad	Calificacion
1	9
1	5

Figura 7. Expediente del alumno

En caso de actualizar los datos de un Alumno, se utiliza el mismo formulario de la Figura cuatro, donde se ingresa el login correspondiente, se selecciona la opción Actualizar y se pulsa el botón Ejecutar, al hacerlo, el Sistema mostrara el siguiente formulario.

Actualizar Alumno

Cambiar datos del alumno: "usuario"

E-Mail:

Direccion:

Telefono:

Figura 8. Formulario de actualización

Una vez actualizados los datos del alumno, el Sistema muestra que los cambios fueron realizados (Figura nueve).

Datos Actualizados

Figura 9. Datos actualizados

Para borrar un alumno se ingresa el login, se selecciona la opción Borrar y se pulsa el botón Ejecutar, al hacerlo, el Sistema muestra el siguiente mensaje (Figura 10).

Alumno Borrado

Figura 10. Alumno Borrado

Además, en esta página se puede listar a todos los alumnos que estén cursando alguna de las unidades del Sistema Tutor:

Listar alumnos por unidad

1 ▾
Listar

Figura 11. Formulario para listar por unidad

Y por último, se cuenta con la posibilidad de listar a los alumnos que recién terminaron el curso, como se muestra en las Figuras 12 y 13.

Listar alumnos que terminaron el curso

Listar

Figura 12. Listar alumnos que terminaron el tutorial

Listado de Alumnos

Borrar	Login	E-Mail	Direccion	Telefono	Expediente
Borrar	juanito	juanito@correo.com	una direccion	53530292	Expediente

Figura 13. Alumnos que finalizaron el tutorial

Cuando se listan los alumnos, el sistema proporciona las opciones de: Borrar, Expediente y Actualizar, mismas que se describieron anteriormente.

3.- OPERACIONES ADMINISTRATIVAS CON LOS CONTENIDOS

Para realizar las operaciones administrativas con los Contenidos de Estudio se debe seleccionar la opción correspondiente del Menú Principal.

Si se desea Agregar un contenido de estudio, el Sistema proporciona cuatro enlaces, cada uno de ellos para los diferentes tipos de contenido (Figura 14).

- [Agregar Contenido PDF](#)
- [Agregar Contenido HTML](#)
- [Agregar Contenido VIDEO FLA](#)
- [Agregar Contenido FLASH](#)

Figura 14. Agregar contenido

Por ejemplo, para agregar un contenido PDF, lo primero que se debe hacer es, pulsar el enlace correspondiente (Figura 14), el cual muestra el formulario para subir un contenido de estudio (Figura 15), en segundo lugar, pulsar en el botón Seleccionar archivo y buscar en el Sistema por medio de un cuadro de dialogo, el archivo que deseamos agregar (Figura 16).

Archivo PDF: No se el... archivo

Unidad:

Figura 15. Formulario para subir contenido

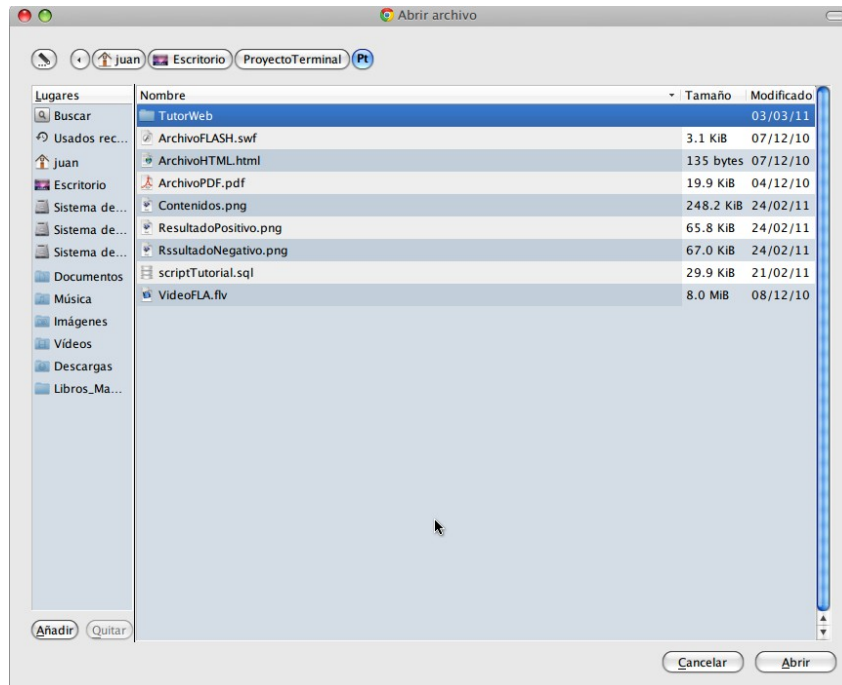


Figura 16. Buscar contenido en el sistema

Una vez seleccionado el archivo, se debe elegir la unidad para la cual servirá el Contenido de Estudio y pulsar el botón Subir, hecho esto, el Sistema confirma proporcionando un enlace para desplegarlo en el navegador (Figura 17).

Y finalmente, se debe asociar el Contenido de Estudio con una de las Actividades de la unidad (Figura 18).

El archivo fue subido con éxito.

[ArchivoPDF.pdf](#)

Figura 17. Archivo subido con éxito

Asociar contenido con una actividad

Actividad

Figura 18. Asociar contenido

Este procedimiento es el mismo para todos los tipos de contenido a excepción de los videos pues una vez agregado se reproduce, tal como se ve en la Figura 19.

El archivo fue subido con éxito

¿QUE ES LA POO?

- ❖ Se puede definir **POO** como una técnica o estilo de programación que utiliza objetos como bloque esencial de construcción.
- ❖ La programación orientada a objetos es un modelo de programación que le da más importancia a los datos que a los algoritmos con la finalidad de imitar la forma en la que se percibe el mundo real: Un conjunto de objetos en constante interacción.

Figura 19. Video subido.

En la Página Principal de Contenidos de Estudio, además de los enlaces para Agregar Contenidos, esta la posibilidad de Buscar, Borrar y Actualizar, por medio de un formulario (Figura 20).

Nombre del Contenido:

Buscar Borrar Actualizar

Unidad:

Figura 20. Formulario para operaciones con contenidos

Para buscar un contenido se ingresa el nombre del archivo correspondiente, se selecciona la opción Buscar y se pulsa el botón Ejecutar, esto mostrara el resultado de la búsqueda (Figura 21).

Contenido Encontrado

Contenido: [ArchivoPDF.pdf](#)
Unidad: 1
Tipo: PDF

Figura 21. Resultado de buscar contenido

Además, en la Figura 21 se muestran los datos del contenido encontrado, si se pulsa en el enlace el material se desplegará en el navegador (Figura 22).

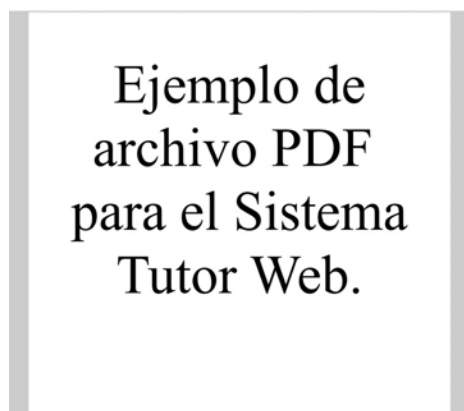


Figura 22. Ejemplo de contenido PDF

Si se desea borrar un Contenido de Estudio se procede de la misma manera, solo que hay que seleccionar la opción Borrar en el formulario, al hacerlo, el Sistema indicará que el contenido fue borrado (Figura 23).

El contenido fue borrado

Figura 23. Contenido borrado

También existe la posibilidad de Actualizar un Contenido de Estudio para ello, se procede de la misma manera que en las opciones anteriores y como resultado se mostrará un formulario como el de la Figura 24.

Datos del contenido

Nombre:	<input type="text" value="ArchivoPDF.pdf"/>
Unidad:	<input type="text" value="1"/>
Actividad:	<input type="text" value="Introduccion a la POO ▼"/>
<input type="button" value="Actualizar"/>	

Figura 24. Formulario actualizar contenido

Por medio de este formulario se puede modificar el archivo, la unidad a la que pertenece o la Actividad con la cual el contenido está asociado.

Cabe aclarar que, si se cambia la unidad entonces también se tendrá que asociar nuevamente el contenido de estudio con una de las actividades de la nueva unidad; en la Figura 25 se muestra este proceso.

Contenido ACTualizado

Asociar contenido con una actividad de la nueva unidad

Actividad

Figura 25. Asociar contenido con nueva actividad

4. SALIR DEL SISTEMA

Para terminar la sesión actual, se debe seleccionar la opción Salir del Menú Principal, el Sistema indicará que ha terminado (Figura 26) y si se desea ingresar de nuevo, se puede pulsar el enlace Iniciar sesión.



The image is copyright © 2008 by Koh Lee.

Haz terminado tu sesión

[Iniciar sesión](#)

Figura 26. Terminar sesión

UNIVERSIDAD AUTÓNOMA
METROPOLITANA

UNIDAD AZCAPOTZALCO

División de Ciencias Básicas e Ingeniería
Licenciatura en Ingeniería en Computación

Guía de Usuario

*“Sistema Tutor Web para el Aprendizaje de
Programación Orientada a Objetos”*

Alumno: Juan José Perea Sánchez

Matricula: 2042040030

Asesor: Irma Fernanda Ardón Pulido

Tabla de Contenido

Introducción.....	3
1.- Darse de alta en el Sistema.....	3
2.- Ingreso al Sistema.....	4
3.- Acceso a la Unidad.....	4
4.- Realizar la Evaluación.....	6
5.- Uso del Chat.....	7
6.- Salir del Sistema.....	7

INTRODUCCIÓN.

En este documento se presenta la Guía de Usuario, donde se explica la forma de utilizar el Sistema Tutor, en específico para los alumnos.

1.- DARSE DE ALTA EN EL SISTEMA

Para que un alumno utilice los servicios que el Sistema Tutor ofrece, debe darse de alta, para ello existe un enlace en la Página Principal (Deseas inscribirte haz clic aquí) como se muestra en la Figura uno.

Tutor Virtual **Programacion Orientada a Objetos en JAVA**

Para entrar al curso, escribe tu login y password

Login:
Password:

Deseas inscribirte haz click [aquí](#)



Figura 1. Página Principal

A continuación, se desplegará una página con un formulario (Figura dos) en donde el alumno deberá ingresar sus datos, los cuales serán validados y en caso de que el e-mail y login ya existan, el alumno deberá ingresar datos diferentes.

Inscripcion al tutorial

Login:
Password:
Repetir Password:
E-mail:
Direccion:
Telefono:



Figura 2. Página Alta en el Sistema

2.- INGRESO AL SISTEMA

El usuario registrado debe introducir su login y password, de ingresar datos erróneos el sistema indicará que son incorrectos (Figura tres) en caso contrario, se muestra el Menú de Opciones que ofrece el Sistema Tutor, mostrados en la Figura cuatro.

Login:
Password:

"El alumno no existe"

Figura 3. Datos incorrectos.



Figura4. Menú de Opciones

3.- ACCESO A LA UNIDAD Y ESTUDIO DE LOS TEMAS

Para estudiar los temas del Sistema se muestra la lista de las unidades en donde el alumno selecciona la deseada, sin embargo no puede entrar a una unidad mas allá de la que cursa actualmente, por ejemplo si el alumno está en la unidad dos, debe aprobarla, para poder cursar la tres, esto se muestra en la Figura cinco.

- "No puedes acceder a esta unidad"
- [1.- Introduccion a la POO](#)
- [2.- Caracteristicas de la POO](#)
- [3.- Iniciacion y Destrucion de Objetos](#)
- [4.- Control de Acceso](#)
- [5.- Herencia y Polimorfismo](#)
- [6.- Clases Abstractas e Interfaces](#)

Figura 5. Acceso a la unidad

Una vez dentro de una unidad, el Sistema desplegara las Reglas de Producción correspondientes (Figura seis) en donde se debe seleccionar la respuesta que se considere adecuada y dependiendo de esta, el Sistema planeará la siguiente Actividad y desplegará una de dos cosas:

1. La siguiente Regla de Producción.
2. Los Contenidos de Estudio.

¿Sabes lo que son los metodos?

SI NO

Figura 6. Regla de Producción

Para el segundo caso, el alumno una vez que estudió y comprendió los temas deberá marcar esta Actividad como realizada con el botón correspondiente (Figura siete), con lo que el Sistema desplegará la siguiente Regla de Producción y el procedimiento continua.

[ArchivoPDF.pdf](#)
[ArchivoHTML.html](#)
[ArchivoPDF.pdf](#)

Si ya estudiaste todo el contenido de estudio marca esta actividad como realizada

Figura 7. Marcar Actividad Realizada

Cuando se contesten todas las Reglas de Producción, el Sistema mostrará un mensaje indicando que es tiempo de realizar la evaluación de la unidad como lo muestra la Figura ocho.

Te recomiendo realices la evaluacion de esta unidad

Figura 8. Realizar evaluación

4.- REALIZAR LA EVALUACIÓN

Para realizar la evaluación se debe pulsar el enlace correspondiente del menú, el cual desplegara diez preguntas de la unidad que se está cursando (Figura nueve), las cuales tienen tres posibles respuestas, el alumno debe seleccionar la que considere correcta, para obtener su resultado, debe pulsar el botón Evaluar (Figura 10) y el Sistema desplegará, una calificación aprobatoria o no aprobatoria.

¿Que es la herencia?

- Acceso a la informacion contenida en otras clases
- Cuando las clases disponende de todos los metodos y propiedades de la clase de la que heredan
- Que los atributos de un objeto son heredados a otros

¿Cuales son las características de la POO?

- Abstraccion,Encapsulamiento,Herencia y Polimorfismo
- Ejecucion,Compilacion y Abstraccion
- Abstraccion,Declaracion de variables y Metodos

Figura 9. Preguntas de evaluación

¿En java se puede implementar la herencia multiple?

- No es posible implementarla
- Si es posible implementarla
- Solo en casos especiales

Evaluar

Figura 10. Botón Evaluar

De obtener una calificación aprobatoria el alumno podrá ingresar a la siguiente unidad.

En la Figura 11 se muestra el resultado de obtener una calificación aprobatoria.

Felicidades aprobaste la unidad

Tu calificacion es:8

Figura 11. Calificación aprobatoria

5.- USO DEL CHAT

Para entrar al chat se debe pulsar la opción correspondiente del menú, al hacerlo se desplegará una página que contiene dos marcos, en el inferior, el alumno deberá escribir su login y el mensaje que desea mostrar, en el marco superior se desplegarán los mensajes de todos los usuarios del Sistema como se muestra en la Figura 12.



```
martin: hola
martin: como estan todos
usuario2: como estan
usuario2: que hay martin
```

Login:

Figura 12. Chat en línea.

6.-SALIR DEL SISTEMA

Para salir del Sistema solo hay que pulsar la opción correspondiente del menú y se desplegará la página que se muestra en la Figura 13.

Haz terminado tu sesion

[Iniciar sesion](#)

Figura 13. Salir del sistema

UNIVERSIDAD AUTÓNOMA
METROPOLITANA

UNIDAD AZCAPOTZALCO

División de Ciencias Básicas e Ingeniería
Licenciatura en Ingeniería en Computación

Manual de Instalación

*“Sistema Tutor Web para el Aprendizaje de
Programación Orientada a Objetos”*

Alumno: Juan José Perea Sánchez

Matricula: 2042040030

Asesor: Irma Fernanda Ardón Pulido

Tabla de Contenidos.

Introducción.....	3
1.- Instalación en Windows.....	3
1.1. Instalación de MySQL.....	3
1.2.- Instalación de Tomcat.....	7
1.3.- Instalación del Sistema Tutor.....	10
2.- Instalación en Linux.....	12
2.1. Instalación de MySQL.....	12
2.2.- Instalación de Tomcat.....	13
2.3.- Instalación del Sistema Tutor.....	14

INTRODUCCIÓN

El presente documento describe los pasos necesarios para la instalación de los servidores Tomcat y MySQL como contenedor web y manejador de base de datos respectivamente, además de la ejecución del Script SQL utilizado para la creación de la Base de Datos del Sistema Tutor y de la aplicación Web (TutorWeb.war).

Este manual, contiene los pasos de instalación tanto para Windows como para Linux.

Como único requisito previo para seguir correctamente este manual, se debe contar con una maquina virtual de Java instalada en el equipo.

1.- INSTALACIÓN EN WINDOWS

1.1.-Instalacion de MySQL

Como primer paso, se debe instalar el manejador de base de datos MySQL, para ello es necesario descargar la versión más reciente de la siguiente dirección web: <http://dev.mysql.com/downloads/mysql/>.

Una vez terminada la descarga de MySQL, pulsar dos veces en el ícono de instalación, para iniciar el asistente mostrado en la Figura uno.



Figura 1.Inicio del asistente de de instalación

Posteriormente, para el tipo de instalación, seleccionar Typical (Figura dos).

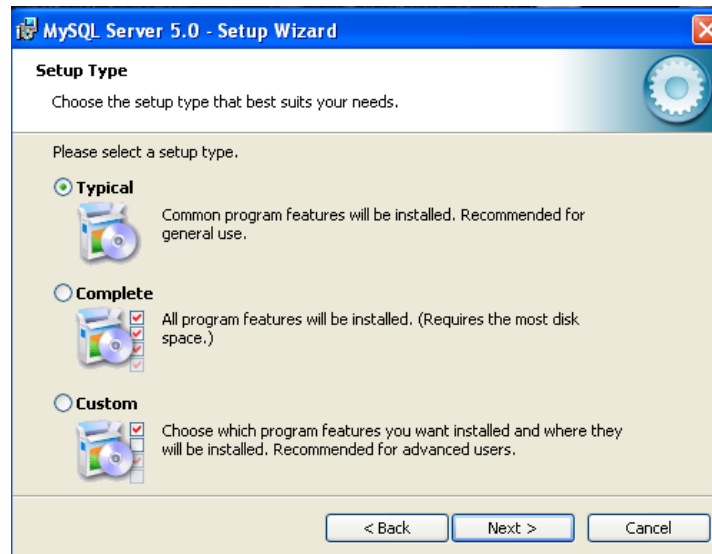


Figura 2. Escoger tipo de configuración

Por último, pulsar Next seguido de Install para iniciar el proceso de instalación del manejador, como se muestra en la Figura tres.

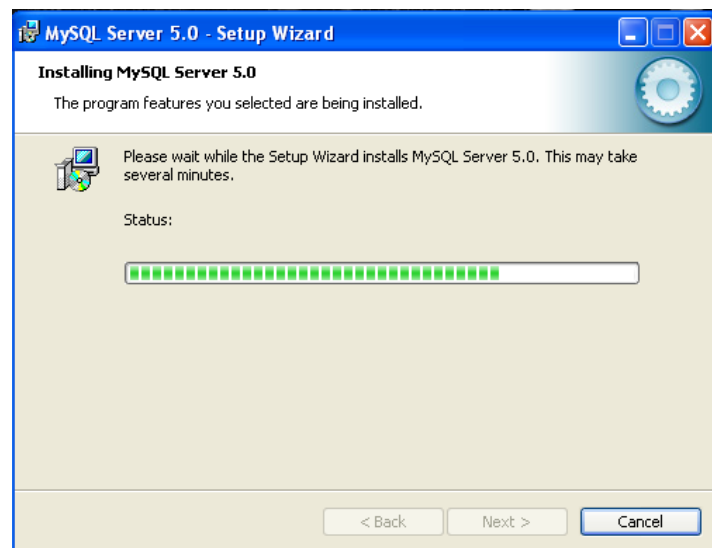


Figura 3. Proceso de instalación del manejador

El paso siguiente es, configurar el manejador, con un asistente como el mostrado en la Figura cuatro.

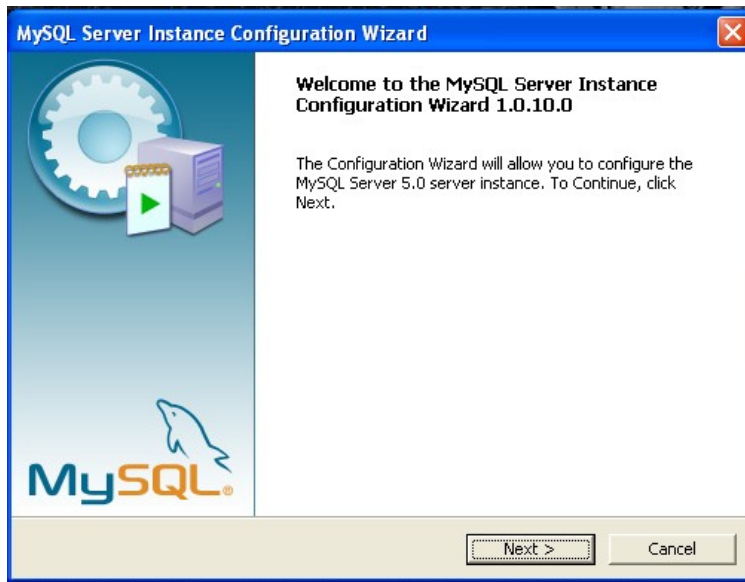


Figura 4. Asistente para la configuración de MySQL

La configuración de MySQL puede ser la que se desee, sin embargo, el único punto requerido por la aplicación, es que el password de root sea: *jj11p09s83*

La Figura cinco muestra este proceso.



Figura 5. Asignación del password de root

A continuación, el asistente empezará a configurar al manejador MySQL como se muestra en la Figura seis.

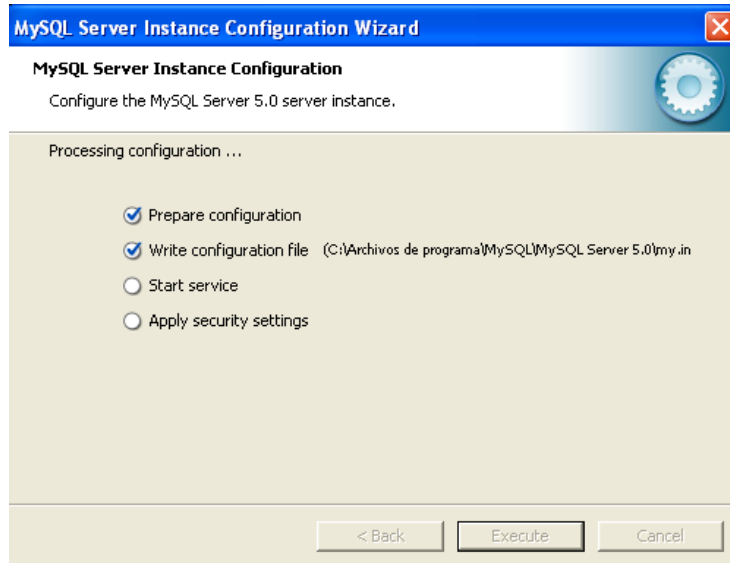


Figura 6. Configuración del manejador

Para asegurar el correcto funcionamiento de la instalación, basta con abrir una sesión en MySQL, primero se debe abrir una consola de MSDOS e ingresar el comando: `mysql -u root -p`, posteriormente escribir el password de root.

Si se muestra en pantalla el prompt de MySQL significa que todo funciona correctamente.

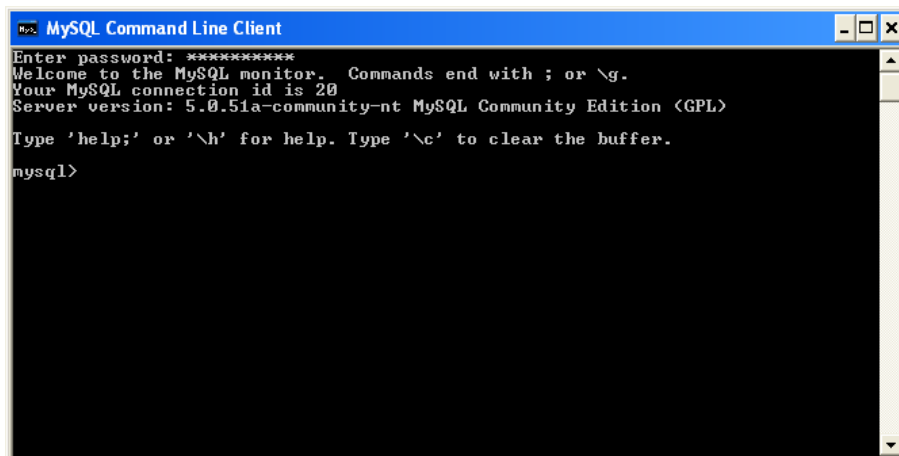


Figura 7. Prueba de MySQL

1.2.-Instalacion de Tomcat

Para instalar Tomcat es necesario, descargarlo desde la siguiente dirección web:
<http://tomcat.apache.org/download-60.cgi>

Una vez que finalizó la descarga de Tomcat, hay que pulsar dos veces sobre el ícono de instalación y seguir el asistente (Figura ocho).



Figura 8. Inicia asistente para instalar Tomcat

Aceptar las condiciones de uso, seleccionar el tipo de instalación que se utilizará y activar la opción Normal como se muestra en la Figura nueve.

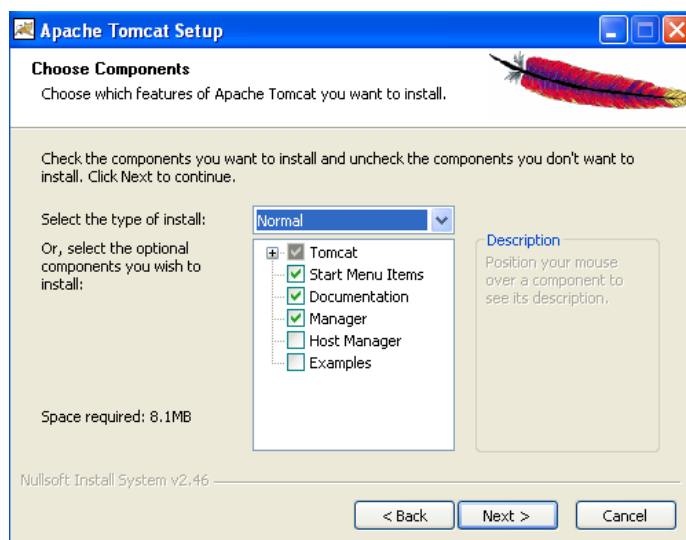


Figura 9. Seleccionar tipo de instalación

El siguiente paso es seleccionar el directorio en el cual se instalará Tomcat como se muestra en la Figura 10.

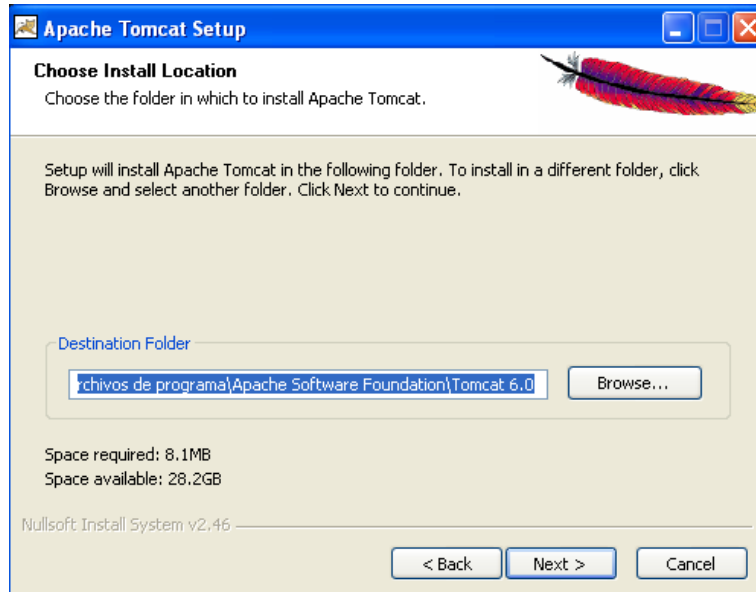


Figura 10. Directorio de instalación

A continuación, se debe asignar un puerto (que de manera predefinida es 8080), además de un login y un password para el Administrador (definidos por el usuario) este proceso se muestra en la siguiente figura.

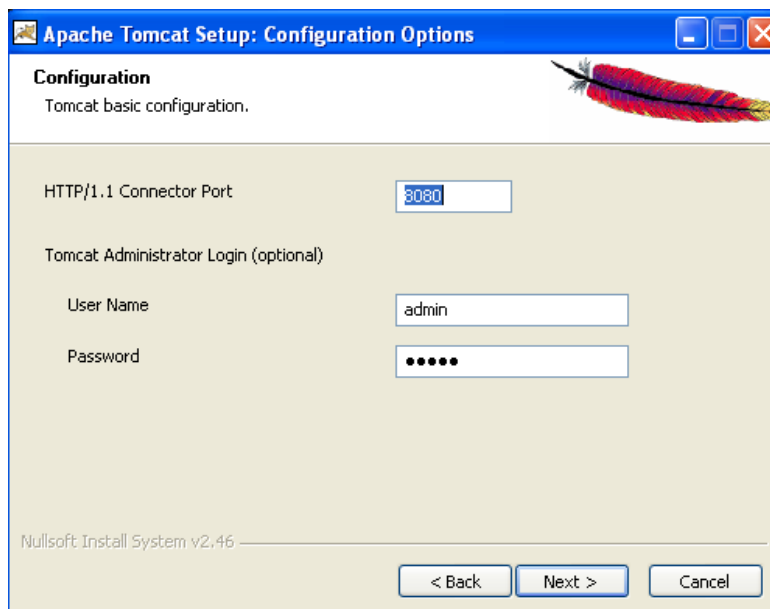


Figura 11. Configuración esencial

Con esto, el asistente comenzará a instalar Tomcat, la Figura 12 muestra este paso.

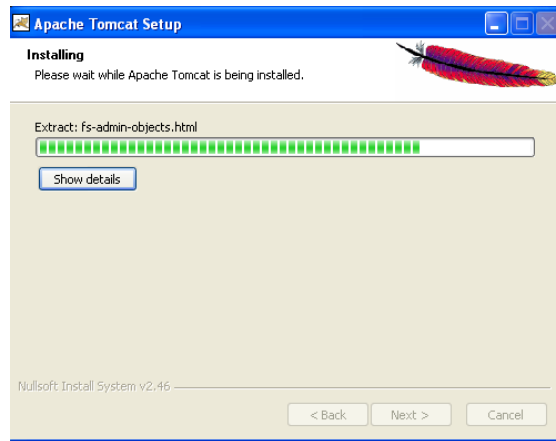


Figura 12. Instalación de Tomcat

Una vez finalizada la instalación, se podrá apreciar un ícono en la parte derecha de la barra de tareas de Windows como se muestra en la Figura 13.

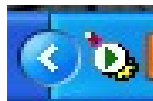


Figura 13. Icono de Tomcat

Para comprobar la instalación de Tomcat, en un navegador web escribir la dirección: <http://localhost:8080> y se desplegará una página parecida a la mostrada en la Figura 14.

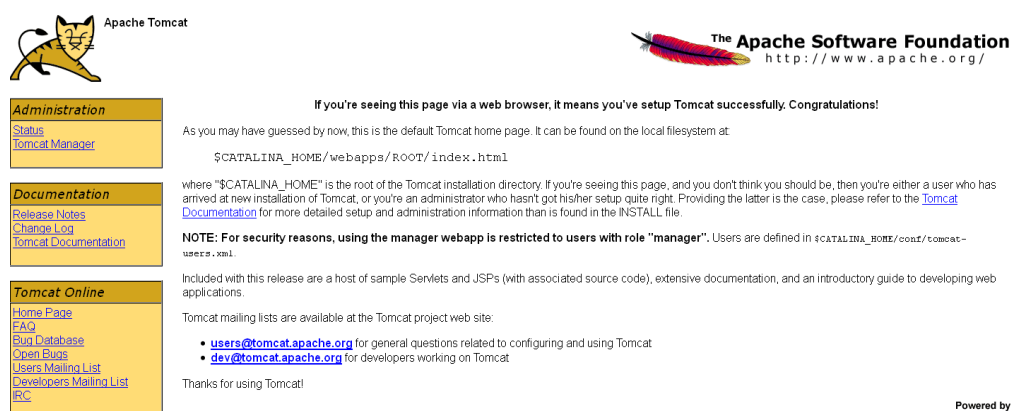
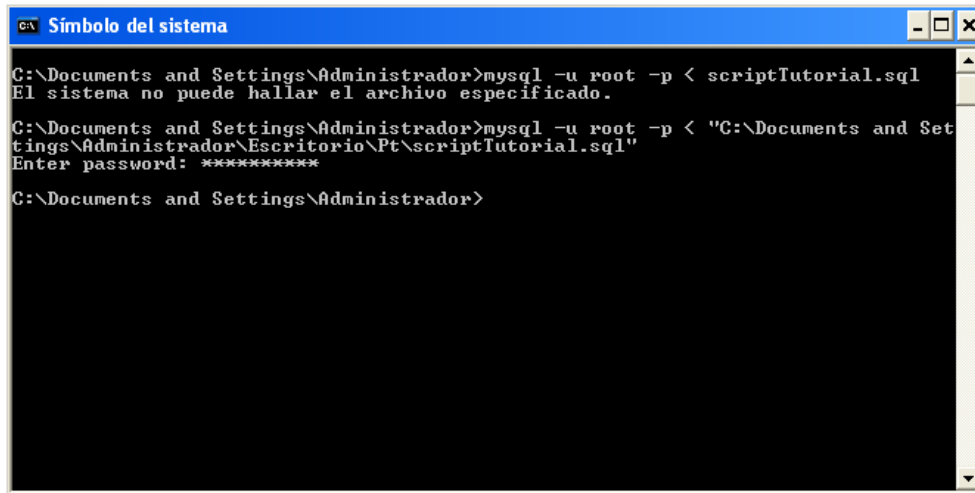


Figura 14. Página de Tomcat

1.3.-Instalacion del Sistema Tutor.

El primer paso para la instalación de la aplicación del Sistema Tutor es cargar el Script SQL que contiene la construcción de la Base de Datos, abrir una consola de MSDOS e introducir el siguiente comando: `mysql -u root -p < scripTutor.sql` y después, ingresar el password de root. En la Figura 15 se muestra este proceso.



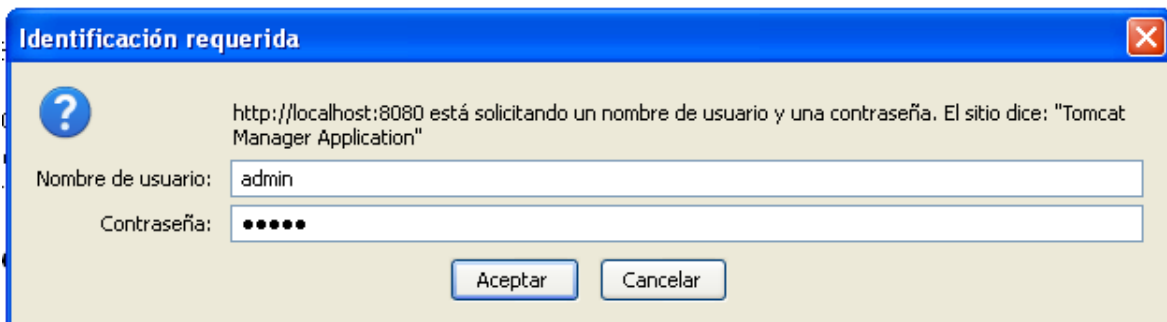
```
C:\Documents and Settings\Administrador>mysql -u root -p < scripTutor.sql
El sistema no puede hallar el archivo especificado.

C:\Documents and Settings\Administrador>mysql -u root -p < "C:\Documents and Settings\Administrador\Escritorio\Pt\scripTutor.sql"
Enter password: *****

C:\Documents and Settings\Administrador>
```

Figura 15. Creación de la Base de Datos Tutorial.

El segundo paso es, desplegar la aplicación en el servidor Tomcat, para ello, abrir un navegador web e introducir la siguiente dirección: <http://localhost:8080>, en una página como la mostrada en la Figura 14, pulsar el enlace Tomcat Manager, a continuación, se debe ingresar, el nombre de usuario y contraseña (admin en ambos casos) como en la Figura 16.



Identificación requerida

?

http://localhost:8080 está solicitando un nombre de usuario y una contraseña. El sitio dice: "Tomcat Manager Application"

Nombre de usuario:

Contraseña:

Aceptar Cancelar

Figura 16. Petición de nombre y contraseña

Si se introducen los datos correctos, se mostrará la siguiente página.



Gestor de Aplicaciones Web de Tomcat

Mensaje: OK

Gestor

[Listar Aplicaciones](#)
 [Ayuda HTML de Gestor](#)
 [Ayuda de Gestor](#)
 [Estado de Servidor](#)

Aplicaciones

Trayectoria	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Welcome to Tomcat	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/docs	Tomcat Documentation	true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
/manager	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos

Figura 17. Página de administración de Tomcat

En la parte inferior de esta página, esta el formulario para cargar un archivo WAR que contiene la aplicación; pulsando el botón Examinar se buscará y seleccionará el archivo TutorWeb.war, finalmente, pulsar el botón Desplegar (Figura 18)

Seleccione archivo WAR a cargar

Figura 18. Formulario de despliegue de archivo WAR

Una vez desplegado el archivo WAR, se muestra un enlace del Tutor Web tal y como se muestra en la Figura 19.

/TutorWeb		true	0	Arrancar Parar Recargar Replegar Expirar sesiones sin trabajar ≥ 30 minutos
---------------------------	--	------	---	--

Figura 19. Sistema Tutor Web desplegado

Al pulsar el enlace se mostrará la Página Principal del Sistema Tutor (Figura 20).

Tutor Virtual.

Programacion Orientada a Objetos en JAVA.

Para entrar al curso, escribe tu login y password

Login:
Password:

Deseas inscribirte haz click [aqui](#)



Figura 20. Página principal del Sistema Tutor

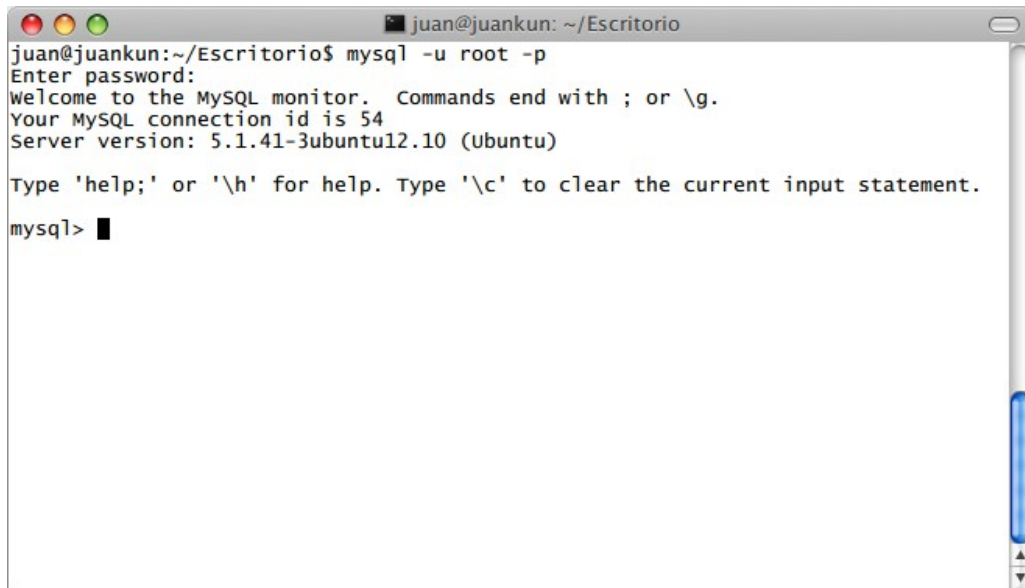
2.- INSTALACIÓN EN LINUX

2.1.-Instalación de MySQL

Para la instalación de MySQL en Linux en primer lugar hay que abrir una terminal y como root ingresar el siguiente comando: `apt-get install mysql-server mysql-client`.

Una vez instalado el manejador MySQL el segundo paso es modificar el password de Administrador de la siguiente forma: `root: /usr/bin/mysqladmin -u root password jj11p09s83`, de esta manera se establece el super usuario (root) y su password (jj11p09s83).

Hecho esto hay que conectarse a MySQL de la siguiente forma: `mysql -u root -p`, posteriormente ingresar el password establecido anteriormente, la Figura 21 muestra el resultado.



```
juan@juankun: ~/Escritorio
juan@juankun:~/Escritorio$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 54
Server version: 5.1.41-3ubuntu12.10 (Ubuntu)

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> █
```

Figura 21. MySQL funcionando en Linux

2.2.-Instalación de Tomcat

Para la instalación de Tomcat, el primer paso es abrir una terminal e introducir como root el comando: `apt-get install tomcat6`, para verificar que la instalación sea correcta, abrir un navegador web e introducir la dirección: <http://localhost:8080> esto mostrara una página como la siguiente.

It works !

If you're seeing this page via a web browser, it means you've setup Tomcat successfully. Congratulations!

This is the default Tomcat home page. It can be found on the local filesystem at: `/var/lib/tomcat6/webapps/ROOT/index.html`

Tomcat6 veterans might be pleased to learn that this system instance of Tomcat is installed with `CATALINA_HOME` in `/usr/share/tomcat6` and `CATALINA_BASE` in `/var/lib/tomcat6`, following the rules from `/usr/share/doc/tomcat6-common/RUNNING.txt.gz`.

You might consider installing the following packages, if you haven't already done so:

tomcat6-docs: This package installs a web application that allows to browse the Tomcat 6 documentation locally. Once installed, you can access it by clicking [here](#).

tomcat6-examples: This package installs a web application that allows to access the Tomcat 6 Servlet and JSP examples. Once installed, you can access it by clicking [here](#).

tomcat6-admin: This package installs two web applications that can help managing this Tomcat instance. Once installed, you can access the [manager webapp](#) and the [host-manager webapp](#).

NOTE: For security reasons, using the manager webapp is restricted to users with role "manager". The host-manager webapp is restricted to users with role "admin". Users are defined in `/etc/tomcat6/tomcat-users.xml`.

Figura 22. Página principal de Tomcat

El segundo paso es configurar el nombre del Administrador así como su contraseña, por lo que es necesario editar el archivo `tomcat-users.xml`, el cual se localiza en el directorio `/etc/tomcat6/`, para esto, con permisos de root agregarle, las siguientes líneas.

```
<tomcat-users>
<role rolename="admin"/>
<role rolename="manager"/>
<user username="tomcat" password="contraseña" roles="admin,manager"/>
```

Donde: username es el nombre del admin y password es la contraseña, estos pueden ser definidos por el usuario.

Por último es necesario reiniciar Tomcat con el siguiente comando: `service tomcat6 restart`.

1.3.-Instalacion del Sistema Tutor

El primer paso para la instalación del Sistema Tutor es cargar el Script SQL que contiene la construcción de la base de datos, para esto, abrir una terminal e introducir el comando: `mysql -u root -p < scripTutor.sql` y a continuación ingresar el password de root. En la Figura 23 se muestra este proceso.



Figura 23. Creación de la base de datos Tutorial

El segundo paso es desplegar la aplicación en el servidor Tomcat, para esto, abrir un navegador web e introducir la siguiente dirección: <http://localhost:8080> y se mostrará una página como la de la Figura 22, pulsar el enlace manager-webapp a continuación se debe ingresar el nombre de usuario y contraseña, admin en ambos casos, (Figura 24).



El servidor localhost:8080 en Tomcat Manager Application requiere un nombre de usuario y una contraseña.

Nombre de usuario:

Contraseña:

Figura 24. Petición de nombre de usuario y contraseña

Si los datos ingresados son correctos, se muestra en pantalla la siguiente página.

Gestor de Aplicaciones Web de Tomcat

Mensaje: OK

Gestor

[Listar Aplicaciones](#) [Ayuda HTML de Gestor](#) [Ayuda de Gestor](#) [Estado de Servidor](#)

Aplicaciones				
Trayectoria	Nombre a Mostrar	Ejecutándose	Sesiones	Comandos
/	Welcome to Tomcat	true	0	Arrancar Parar Recargar Replegar <input type="button" value="Expirar sesiones"/> sin trabajar > <input type="text" value="30"/> minutos
/docs	Tomcat Documentation	true	0	Arrancar Parar Recargar Replegar <input type="button" value="Expirar sesiones"/> sin trabajar > <input type="text" value="30"/> minutos
/manager	Tomcat Manager Application	true	1	Arrancar Parar Recargar Replegar <input type="button" value="Expirar sesiones"/> sin trabajar > <input type="text" value="30"/> minutos

Figura 25. Página de administración de Tomcat

En la parte inferior de la página mostrada en la figura anterior, está el formulario para cargar un archivo WAR el cual contiene la aplicación; buscar y seleccionar el archivo TutorWeb.war y pulsar el botón Desplegar (Figura 26).

Seleccione archivo WAR a cargar

Figura 26. Formulario de despliegue de archivo WAR

Una vez desplegado el archivo WAR, se muestra un enlace del Tutor Web tal y como se muestra en la Figura 27.

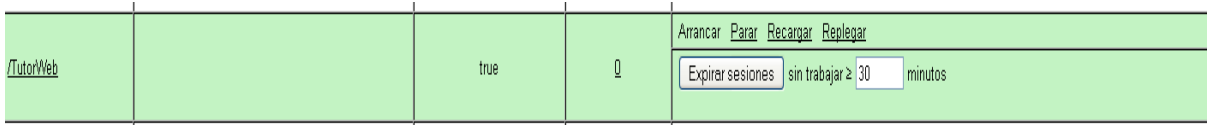


Figura 27. Sistema Tutor Web desplegado

Al pulsar el enlace se mostrará la Página Principal del Sistema Tutor (Figura 28).

Tutor Virtual.

Programacion Orientada a Objetos en JAVA.

Para entrar al curso, escribe tu login y password

Login:
Password:

Deseas inscribirte haz click [aquí](#)



The image is copyright © 2008 by Xuh Lee.

Figura 28. Página Principal del Sistema Tutor