

**SISTEMA MULTIMEDIA PARA LA ADMINISTRACIÓN DE CONTENIDOS DE LA
CARTELERA DIGITAL DE LA FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS**

MANUAL DE TÉCNICO



Autora: Sandra Enríquez

ÍNDICE

HERRAMIENTAS UTILIZADAS	4
FRAMEWORKS	4
DIAGRAMA ENTIDAD – RELACIÓN.....	5
ESTRUCTURA GENERAL DEL SISTEMA.....	5
DESCRIPCIÓN DE CLASES Y MÉTODOS	6
Package: multimedia.model.entities.....	6
Clase Configuracion	6
Clase ConfiguracionCartelera	7
Clase ListaReproduccion	8
Clase Lista.....	9
Clase Recurso	10
Clase Texto	11
Clase Usuario	12
APLICACIÓN WEB	13
Package multimedia.model.manager	13
Clase ManagerDAO	13
Clase ManagerMultimedia.....	15
Clase ManagerLista	18
Clase ManagerCartelera	21
Clase ManagerUsuario.....	24
Package multimedia.controller	26
Class ControllerMultimedia	26
Class ControllerListaReproduccion	30
Class ControllerCartelera	33
Class ControllerPresentacionCartelera	36
Class ControllerUsuario	37
Class ControllerLogin	38
Package utilitarios	40
Class JSFUtil.....	40
DICCIONARIO DE DATOS.....	40
Tabla recurso.....	40
Tabla lista_reproduccion	41

Tabla lista	42
Tabla configuracion.....	42
Tabla configuración_cartelera	43
Tabla texto	43
Tabla usuario.....	44

HERRAMIENTAS UTILIZADAS

- ✓ JDK 1.8

Java Development Kit es un conjunto de herramientas (programas y librerías) que permiten el desarrollo de aplicaciones en el lenguaje de programación Java. Está disponible para casi todos los sistemas operativos y es distribuido de forma gratuita para Windows, Solaris y Linux

- ✓ IDE de desarrollo Netbeans 8.0.1
- ✓ PostgreSQL 9.3

Es un sistema de gestión de base de datos relacional de código abierto, considerada una de las bases de datos más potentes y robustas del mercado. Utiliza el modelo cliente/servidor y usa multiprocesos para asegurar la estabilidad del sistema, permitiendo así que el sistema continúe funcionando si uno de los procesos llegara a fallar; además, funciona muy bien para grandes cantidades de datos y una alta concurrencia de usuarios accediendo al sistema.

- ✓ Servidor de aplicaciones web WildFly

WildFly es una aplicación gestionada flexible, ligera, que representa una nueva versión mejorada del servidor de aplicación JBoss. Está escrita en Java e implementa las especificaciones de la plataforma Java Enterprise Edition 7; es completamente gratis y de código abierto (Red Hat, 2013).

- ✓ Navegador Web: Mozilla Firefox 50.0.2 o versiones posteriores.

FRAMEWORKS

- ✓ JavaServer Faces 2.2

Es un entorno de trabajo (integrado en Netbeans) para desarrollar aplicaciones Web Java en el lado del servidor, que presentan una interfaz gráfica cuando se ejecutan. JSF provee a los desarrolladores un amplio conjunto de componentes de interfaz de usuario para el diseño Web.

- ✓ PrimeFaces 6.0

PrimeFaces es un componente de JavaServer Faces de código abierto que contiene un conjunto de componentes visuales para agilizar el desarrollo de aplicaciones web enriquecidas.

✓ Java Persistence API (JPA)

Proporciona a los desarrolladores de Java con un centro de mapeo objeto / relacional para la gestión de datos relacionales en aplicaciones Java

DIAGRAMA ENTIDAD – RELACIÓN

El diseño entidad – relación de la base de datos está desarrollado en Power Designer, mismo que se muestra en la Figura 1.

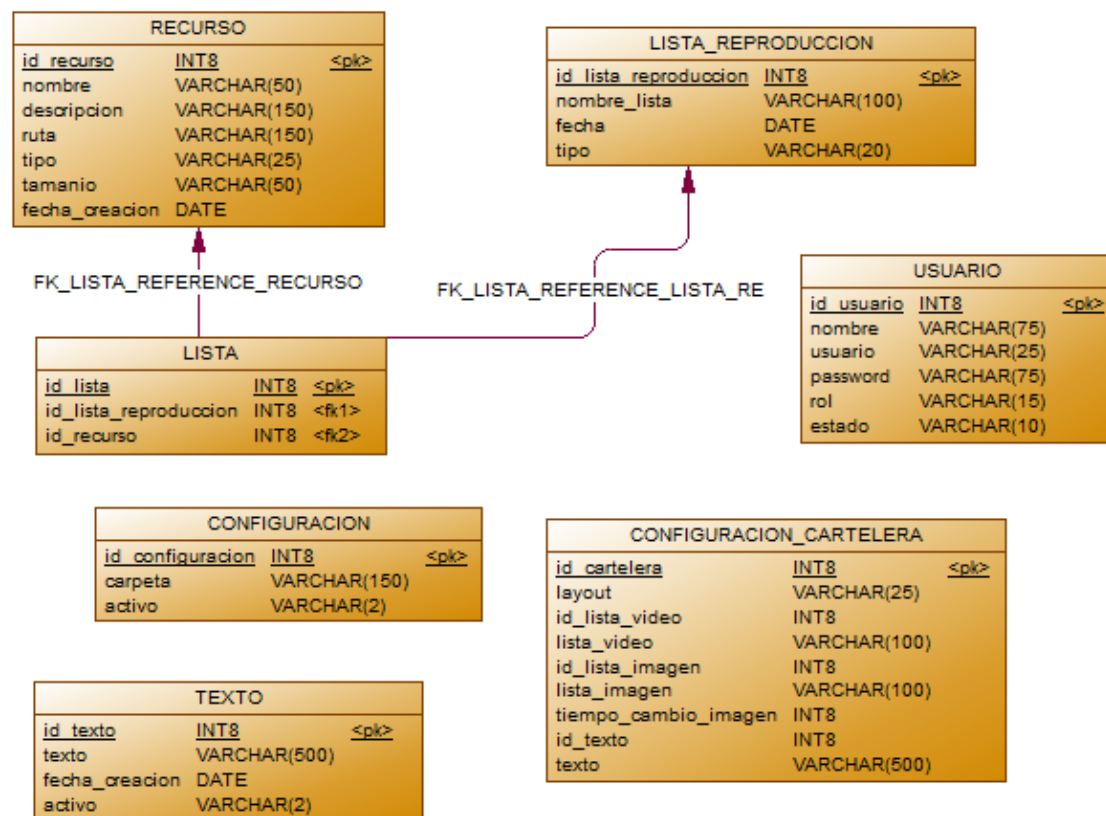


Figura 1

ESTRUCTURA GENERAL DEL SISTEMA

En la Figura 2 se muestra como está estructurado el sistema, en la carpeta Web Pages se encuentra las diferentes vistas, en la carpeta Source Packages están los diferentes

paquetes y en la carpeta Libraries se encuentran las diferentes librerías empleadas en el sistema.

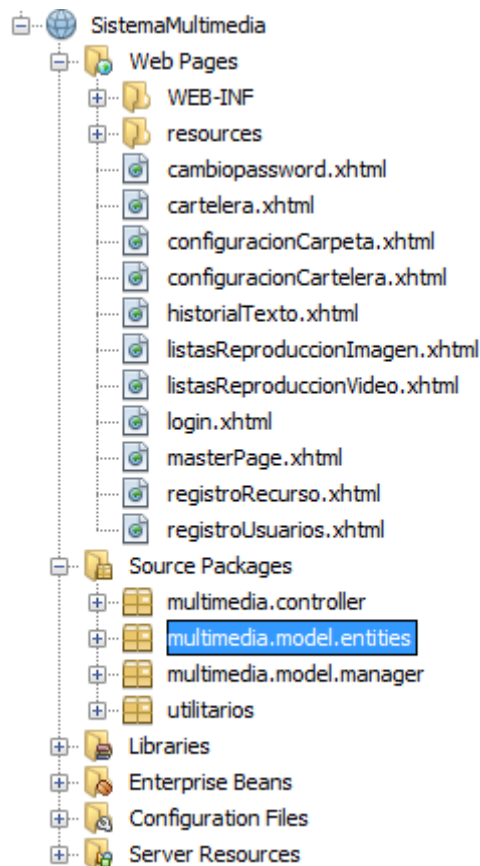


Figura 2

DESCRIPCIÓN DE CLASES Y MÉTODOS

Package: multimedia.model.entities

En este paquete se encuentran las diferentes entidades creadas con Java Persistence API, lo que permite la interacción con la base de datos mediante objetos POJO (clases simples java con métodos getter y setter básicos). A continuación se presenta la estructura del código de las diferentes clases.

Clase Configuracion

```
java.lang.Object
    multimedia.model.entities.Configuracion
```

All Implemented Interfaces:

```
java.io.Serializable
```

@Entity

public class **Configuracion**

extends java.lang.Object

implements java.io.Serializable

Constructor Summary

Constructor and Description
Configuracion()
Configuracion (java.lang.Integer idConfiguracion)

Method Summary

Modifier and Type	Method and Description
boolean	equals (java.lang.Object object)
java.lang.String	getActivo()
java.lang.String	getCarpeta()
java.lang.Integer	getIdConfiguracion()
int	hashCode()
void	setActivo (java.lang.String activo)
void	setCarpeta (java.lang.String carpeta)
void	setIdConfiguracion (java.lang.Integer idConfiguracion)
java.lang.String	toString()

Clase ConfiguracionCartelera

java.lang.Object

multimedia.model.entities.ConfiguracionCartelera

All Implemented Interfaces:

java.io.Serializable

@Entity

public class **ConfiguracionCartelera**

extends java.lang.Object

implements java.io.Serializable

Constructor Summary

Constructor and Description
ConfiguracionCartelera()
ConfiguracionCartelera (java.lang.Integer idCartelera)

Method Summary

Modifier and Type	Method and Description
boolean	equals (java.lang.Object object)
java.lang.String	getColorFondo ()
java.lang.Integer	getIdCartelera ()
java.lang.Integer	getIdListalmagen ()
java.lang.Integer	getIdListaVideo ()
java.lang.Integer	getIdTexto ()
java.lang.String	getLayout ()
java.lang.String	getListalmagen ()
java.lang.String	getListVideo ()
java.lang.String	getTexto ()
java.lang.Integer	getTiempoCambiolmagen ()
int	hashCode ()
void	setColorFondo (java.lang.String colorFondo)
void	setIdCartelera (java.lang.Integer idCartelera)
void	setIdListalmagen (java.lang.Integer idListalmagen)
void	setIdListaVideo (java.lang.Integer idListaVideo)
void	setIdTexto (java.lang.Integer idTexto)
void	setLayout (java.lang.String layout)
void	setListalmagen (java.lang.String listalmagen)
void	setListaVideo (java.lang.String listaVideo)
void	setTexto (java.lang.String texto)
void	setTiempoCambiolmagen (java.lang.Integer tiempoCambiolmagen)
java.lang.String	toString ()

Clase ListaReproduccion

java.lang.Object
multimedia.model.entities.ListaReproduccion

All Implemented Interfaces:

java.io.Serializable

@Entity

public class **ListReproduccion**

extends java.lang.Object

implements java.io.Serializable

Constructor Summary

Constructor and Description
ListaReproduccion()
ListaReproduccion(java.lang.Integer idListaReproduccion)

Method Summary

Modifier and Type	Method and Description
Lista	addLista(Lista lista)
boolean	equals(java.lang.Object object)
java.util.Date	getFecha()
java.lang.Integer	getIdListaReproduccion()
java.util.List<Lista>	getListList()
java.lang.String	getNombreLista()
java.lang.String	getTipo()
int	hashCode()
Lista	removeLista(Lista lista)
void	setFecha(java.util.Date fecha)
void	setIdListaReproduccion(java.lang.Integer idListaReproduccion)
void	setListList(java.util.List<Lista> listaList)
void	setNombreLista(java.lang.String nombreLista)
void	setTipo(java.lang.String tipo)
java.lang.String	toString()

Clase Lista

java.lang.Object
multimedia.model.entities.Lista

All Implemented Interfaces:

java.io.Serializable

@Entity

public class **Lista**

extends java.lang.Object

implements java.io.Serializable

Constructor Summary

Constructor and Description
Lista()
Lista (java.lang.Integer idLista)

Method Summary

Modifier and Type	Method and Description
boolean	equals (java.lang.Object object)
java.lang.Integer	getIdLista()
ListaReproduccion	getListReproduccion()
Recurso	getRecurso()
int	hashCode()
void	setIdLista (java.lang.Integer idLista)
void	setListaReproduccion (ListaReproduccion listaReproduccion)
void	setRecurso (Recurso recurso)
java.lang.String	toString()

Clase Recurso

java.lang.Object
multimedia.model.entities.Recurso

All Implemented Interfaces:

java.io.Serializable

@Entity
public class **Recurso**
extends java.lang.Object
implements java.io.Serializable

Constructor Summary

Constructor and Description
Recurso()
Recurso (java.lang.Integer idRecurso)

Method Summary

Modifier and Type	Method and Description
boolean	equals (java.lang.Object object)
java.lang.String	getDescripcion ()
java.util.Date	getFechaCreacion ()
java.lang.Integer	getIdRecurso ()
java.lang.String	getNombre ()
java.lang.String	getRuta ()
java.lang.String	getTamanio ()
java.lang.String	getTipo ()
int	hashCode ()
void	setDescripcion (java.lang.String descripcion)
void	setFechaCreacion (java.util.Date fechaCreacion)
void	setIdRecurso (java.lang.Integer idRecurso)
void	setNombre (java.lang.String nombre)
void	setRuta (java.lang.String ruta)
void	setTamanio (java.lang.String tamanio)
void	setTipo (java.lang.String tipo)
java.lang.String	toString ()

Clase Texto

java.lang.Object
multimedia.model.entities.Texto

All Implemented Interfaces:

java.io.Serializable

@Entity

public class **Texto**

extends java.lang.Object

implements java.io.Serializable

Constructor Summary

Constructor and Description
Texto ()
Texto (java.lang.Integer idTexto)

Method Summary

Modifier and Type	Method and Description
-------------------	------------------------

boolean	equals (java.lang.Object object)
java.lang.String	getActivo ()
java.util.Date	getFechaCreacion ()
java.lang.Integer	getIdTexto ()
java.lang.String	getTexto ()
int	hashCode ()
void	setActivo (java.lang.String activo)
void	setFechaCreacion (java.util.Date fechaCreacion)
void	setIdTexto (java.lang.Integer idTexto)
void	setTexto (java.lang.String texto)
java.lang.String	toString ()

Clase Usuario

java.lang.Object
multimedia.model.entities.Usuario

All Implemented Interfaces:

java.io.Serializable

@Entity

public class **Usuario**

extends java.lang.Object

implements java.io.Serializable

Constructor Summary

Constructor and Description
Usuario ()
Usuario (java.lang.Integer idUsuario)

Method Summary

Modifier and Type	Method and Description
boolean	equals (java.lang.Object object)
java.lang.String	getEstado ()
java.lang.Integer	getIdUsuario ()
java.lang.String	getNombre ()
java.lang.String	getPassword ()
java.lang.String	getRol ()
java.lang.String	getUsuario ()
int	hashCode ()
void	setEstado (java.lang.String estado)

void	setIdUsuario (java.lang.Integer idUsuario)
void	setNombre (java.lang.String nombre)
void	setPassword (java.lang.String password)
void	setRol (java.lang.String rol)
void	setUsuario (java.lang.String usuario)
java.lang.String	toString ()

APLICACIÓN WEB

Package **multimedia.model.manager**

Clase **ManagerDAO**

java.lang.Object
multimedia.model.manager.ManagerDAO

```
public class ManagerDAO
extends java.lang.Object
```

Objeto que encapsula la lógica básica de acceso a datos mediante JPA.

Autor:
Sandra Enríquez

Constructor Summary

Constructor and Description
ManagerDAO()

Method Detail

insertar

public void insertar(java.lang.Object object) throws java.lang.Exception

Throws:

java.lang.Exception

findAll

public java.util.List findAll(java.lang.Class clase, java.lang.String orderBy)

Finder Genérico que devuelve todas las entidades de una tabla.

Parámetros:

- ✓ clase: La clase que se desea consultar (ejemplo Recurso.class).

- ✓ orderBy: Expresión que indica la propiedad de la entidad por la que se desea ordenar la consulta. Para nombrar la(s) propiedad(es) por la cual se va a ordenar debe utilizar el alias "o" (ejemplo o.fecha). Se puede utilizar null o una cadena vacía y esto indicará que no se desea ordenar la consulta.

Valor de retorno:

Listado resultante.

actualizar

public void actualizar(java.lang.Object object) throws java.lang.Exception

Actualiza la información de un determinado objeto en la persistencia.

Parámetros:

object: Objeto que contiene la información que se va a actualizar.

Throws:

java.lang.Exception

findById

public java.lang.Object findById(java.lang.Class clase, java.lang.Object identificador)

throws java.lang.Exception

Finder genérico que permite encontrar un objeto específico.

Parámetros:

- ✓ clase: La clase que se desea consultar (ejemplo Recurso.class).
- ✓ identificador: Clave primaria que permite identificar el objeto consultado.

Throws:

java.lang.Exception

eliminar

public void eliminar(java.lang.Object object) throws java.lang.Exception

Método que permite eliminar un objeto cualquiera de la persistencia.

Parámetros:

- ✓ object: Objeto que contiene la información que se va a eliminar.

Throws:

java.lang.Exception

deleteWhere

public void deleteWhere(java.lang.Class clase, java.lang.Object object) throws

java.lang.Exception

Permite eliminar uno o más registros que coincidan con la condición que se está especificando.

Parámetros:

- ✓ clase: La clase que se desea consultar (ejemplo Recurso.class).

- ✓ object: Objeto que contiene la condición para eliminar (ejemplo idRecurso = 1).

Throws:

java.lang.Exception

findWhere

public java.util.List findWhere(java.lang.Class clase, java.lang.Object obj)

Finder genérico que permite utilizar la cláusula where para realizar consultas.

Parámetros:

- ✓ clase: La clase que se desea consultar (ejemplo Recurso.class).
- ✓ obj: Objeto que contiene la condición para eliminar (ejemplo tipo = 'IMAGEN').

Clase ManagerMultimedia

java.lang.Object

multimedia.model.manager.ManagerMultimedia

public class **ManagerMultimedia**

extends java.lang.Object

Constructor Summary

Constructor and Description
ManagerMultimedia()

Method Detail

findAllRecurso

public java.util.List<Recurso> findAllRecurso()

Retorno:

Listado de Recursos.

insertarRecurso

public void insertarRecurso(java.lang.String nombre, java.lang.String descripcion, java.lang.String tipo, java.lang.String ruta, java.lang.String tamaño) throws java.lang.Exception

Almacena la información de un recurso en la tabla Recurso.

Parámetros:

- ✓ nombre: Nombre del recurso que se está registrando.
- ✓ descripcion: Breve descripción que hace referencia al recurso.

- ✓ tipo: Tipo de recurso puede ser IMAGEN, VIDEO o ANIMACIÓN.
- ✓ ruta: Ruta relativa del recurso (ejemplo /LogoUTN.png).
- ✓ tamaño: Tamaño en megas del recurso.

Throws:

java.lang.Exception

encontrarRecursoPorNombre

public Recurso encontrarRecursoPorNombre(java.lang.String nombre)

Permite encontrar un recurso mediante el atributo nombre.

Parámetros:

- ✓ nombre: Indica el nombre del recurso que se está buscando.

Retorno:

Objeto de tipo Recurso.

actualizarRecurso

public void actualizarRecurso(java.lang.Integer id, java.lang.String nombre, java.lang.String descripcion) throws java.lang.Exception

Permite actualizar un registro en la tabla Recurso.

Parámetros:

- ✓ id: Clave primaria del objeto que se está modificando.
- ✓ nombre: Nombre del recurso.
- ✓ descripcion: Descripción que hace referencia al contenido del recurso.

Throws:

java.lang.Exception

eliminarRecursoMultimedia

public void eliminarRecursoMultimedia(java.lang.Integer id) throws java.lang.Exception

Permite eliminar un registro de la tabla Recurso.

Parámetro:

- ✓ id: Clave primaria del objeto que se va a eliminar.

Throws:

java.lang.Exception

eliminarRecursoDeLista

public void eliminarRecursoDeLista(java.lang.Integer id) throws java.lang.Exception

Permite eliminar recursos que forman parte de una lista que coincidan con la clave primaria del recurso que se indica en el parámetro.

Parámetro:

- ✓ id: Clave primaria que identifica el recurso que se va a eliminar de la tabla Lista.

Throws:

java.lang.Exception

listaReproduccionPorIdRecurso

public java.util.List<Lista> listaReproduccionPorIdRecurso(java.lang.Integer id)

Devuelve una listado con las diferentes listas de reproducción a las que pertenece un recurso.

Parámetro:

- ✓ id: Clave primaria que identifica el objeto por el que se está realizando la consulta.

Retorno:

Listado de objetos tipo Lista.

buscarRecurso

public Recurso buscarRecurso(java.lang.Integer id) throws java.lang.Exception

Permite encontrar un recurso mediante la clave primaria.

Parámetro:

- ✓ id: Clave primaria que identifica el objeto que se está consultando.

Retorno:

Objeto de tipo recurso.

Throws:

java.lang.Exception

guardarRuta

public void guardarRuta(java.lang.String ruta) throws java.lang.Exception

Almacena la ruta donde se almacenarán todos los recursos multimedia en la tabla Configuracion.

Parámetro:

- ✓ ruta: Indica la ruta para el almacenamiento de recursos (ejemplo F:\Cartelera\RecursosMultimedia).

Throws:

java.lang.Exception

findAllRutas

public java.util.List<Configuracion> findAllRutas()

Retorno:

Listado de objetos tipo Configuracion.

findRutaActiva

```
public Configuracion findRutaActiva()
```

Retorno:

Objeto de tipo Configuracion.

Clase ManagerLista

java.lang.Object

multimedia.model.manager.ManagerLista

```
public class ManagerLista
```

```
extends java.lang.Object
```

Constructor Summary

Constructor and Description

ManagerLista()

Method Detail

listaRecursoPorTipo

```
public java.util.List<Recurso> listaRecursoPorTipo(java.lang.String tipo)
```

Parámetro:

✓ tipo: Puede ser IMAGEN o VIDEO.

Retorno:

Lista de objetos de tipo Recurso.

findListaPorNombre

```
public ListaReproduccion findListaPorNombre(java.lang.String nombre)
```

Parámetro:

✓ nombre: Atributo nombre del objeto ListaReproducción.

Retorno:

Objeto de tipo ListaReproduccion.

crearListaReproduccionTemp

```
public ListaReproduccion crearListaReproduccionTemp(java.lang.String nombre)
throws java.lang.Exception
```

Crear en memoria un objeto de tipo ListaReproduccion.

Parámetros:

- ✓ nombre: Valor del atributo nombre de la lista temporal que se está creando.

Retorno:

Objeto de tipo ListaReproduccion.

Throws:

java.lang.Exception

agregarRecurso

```
public void agregarRecurso(ListaReproduccion listaReproduccionTemp,
java.lang.Integer idRecurso) throws java.lang.Exception
```

Permite agregar un recurso a la lista de reproducción temporal creada.

Parámetros:

- ✓ listaReproduccionTemp: Objeto temporal de tipo ListaReproduccion.
- ✓ idRecurso: Clave primaria del objeto de tipo Recurso que se va a agregar a la lista de reproducción.

Throws:

java.lang.Exception

añadirRecurso

```
public void añadirRecurso(ListaReproduccion lr, Recurso r) throws
java.lang.Exception
```

Agrega un recurso a una lista de reproducción existente (Guardar un registro en la tabla Lista).

Parámetros:

- ✓ lr: Lista de reproducción a la que se le está agregando el recurso.
- ✓ r: Objeto de tipo Recurso que se está agregando a la lista.

Throws:

java.lang.Exception

quitarRecurso

```
public void quitarRecurso(ListaReproduccion listaReproduccionTemp, Lista l) throws
java.lang.Exception
```

Permite eliminar un recurso de la lista temporal creada.

Parámetros:

- ✓ listaReproduccionTemp: Objeto de tipo ListaReproduccion.
- ✓ l: Objeto de tipo Lista que contiene la información del recurso que se está eliminando de la lista temporal.

Throws:

java.lang.Exception

guardarListaReproduccion

public void guardarListaReproduccion(ListaReproduccion listaReproduccionTemp, java.lang.String tipo) throws java.lang.Exception

Almacena la información de una lista de reproducción en las tablas ListaReproduccion y Lista.

Parámetros:

- ✓ listaReproduccionTemp: Objeto de tipo ListaReproduccion que contiene toda la información de la lista que se está creando.
- ✓ tipo: Atributo que indica si es una lista de IMAGEN o VIDEO.

Throws:

java.lang.Exception

findAllListasReproduccion

public java.util.List<ListaReproduccion> findAllListasReproduccion(java.lang.String tipo)

Retorna todas las listas de reproducción ya sea de video o de imagen dependiendo del valor del parámetro.

Parámetro:

- ✓ tipo: Puede ser uno de los siguientes valores: IMAGEN o VIDEO.

Retorno:

Lista de objetos tipo ListaReproduccion

listaReproduccionPorId

public java.util.List<Lista> listaReproduccionPorId(java.lang.Integer id)

Parámetro:

- ✓ id: Valor que indica la lista de reproducción por la cual se realiza la consulta.

Retorno:

Lista de objetos tipo Lista.

eliminarListaReproduccion

public void eliminarListaReproduccion(java.lang.Integer id) throws java.lang.Exception

Borra un registro de la tabla ListaReproduccion.

Parámetro:

- ✓ id: Clave primaria que identifica el objeto que se está eliminando.

Throws:

java.lang.Exception

eliminarRecursoListaReproduccion

```
public void eliminarRecursoListaReproduccion(java.lang.Integer id) throws  
java.lang.Exception
```

Elimina un registro de la tabla Lista.

Parámetro:

- ✓ id: Clave primaria que identifica el objeto de tipo Lista que se está eliminando.

Throws:

```
java.lang.Exception
```

eliminarDetallesListaReproduccion

```
public void eliminarDetallesListaReproduccion(java.lang.Integer id) throws  
java.lang.Exception
```

Elimina uno o varios registros de la tabla Lista.

Parámetro:

- ✓ id: Valor que identifica los recursos de la tabla Lista que se desea eliminar.

Throws:

```
java.lang.Exception
```

Clase ManagerCartelera

```
java.lang.Object
```

```
multimedia.model.manager.ManagerCartelera
```

```
public class ManagerCartelera
```

```
extends java.lang.Object
```

Constructor Summary

Constructor and Description
ManagerCartelera()

Method Detail

guardarConfiguracion

```
public void guardarConfiguracion(java.lang.String layout, java.lang.Integer  
idListaVideo, java.lang.String listaVideo, java.lang.Integer idListamagen,  
java.lang.String listamagen, java.lang.String texto, java.lang.Integer idT,  
java.lang.Integer tiempo, java.lang.String color)
```

Almacena la configuración de la cartelera en la tabla ConfiguracionCartelera.

Parámetros:

- ✓ layout: Valor que representa el diseño de la cartelera.
- ✓ idListaVideo: Valor que representa la clave primaria de la lista de video.
- ✓ listaVideo: Valor que representa el nombre de la lista de video.
- ✓ idListamagen: Valor que representa la clave primaria de la lista de imagen.
- ✓ listamagen: Valor que representa el nombre de la lista de imagen.
- ✓ texto: Valor que representa el texto que se mostrará en la cartelera.
- ✓ idT: Valor que representa la clave primaria del texto que se está utilizando.
- ✓ tiempo: Valor que representa el tiempo que se tarda en cambiar las imágenes.
- ✓ color: Valor que representa el color de fondo de la cartelera (ejemplo ffffff).

actualizarConfiguracion

public void actualizarConfiguracion(java.lang.Integer id, java.lang.String layout, java.lang.Integer idV, java.lang.String lv, java.lang.Integer idI, java.lang.String li, java.lang.String texto, java.lang.Integer idT, java.lang.Integer tiempo, java.lang.String color) throws java.lang.Exception

Actualiza un registro de la tabla ConfiguracionCartelera.

Parámetros:

- ✓ id: Valor que representa la clave primaria del objeto que se está actualizando.
- ✓ layout: Valor que representa el diseño de la cartelera.
- ✓ idV: Valor que representa la clave primaria de la lista de video.
- ✓ lv: Valor que representa el nombre de la lista de video.
- ✓ idI: Valor que representa la clave primaria de la lista de imagen.
- ✓ li: Valor que representa el nombre de la lista de imagen.
- ✓ texto: Valor que representa el texto que se mostrará en la cartelera.
- ✓ idT: Valor que representa la clave primaria del texto que se está utilizando.
- ✓ tiempo: Valor que representa el tiempo que se tarda en cambiar las imágenes.
- ✓ color: Valor que representa el color de fondo de la cartelera (ejemplo #ffffff).

Throws:

java.lang.Exception

findConfiguracionAnterior

public ConfiguracionCartelera findConfiguracionAnterior()

Retorno:

Objeto de tipo ConfiguracionAnterior

guardarTexto

public void guardarTexto(java.lang.String texto) throws java.lang.Exception

Almacena el texto que se va a mostrar en la cartelera en la tabla Texto.

Parametro:

✓ texto: Valor que representa el atributo texto de la tabla Texto.
Throws:
java.lang.Exception

textoActivoInactivo

public void textoActivoInactivo(java.lang.Integer idT) throws java.lang.Exception
Actualiza el estado de un registro en la tabla Texto.
Parámetro:
✓ idT: Representa la clave primaria del objeto de tipo Texto que se va a modificar.
Throws:
java.lang.Exception

findListaReproduccionPorId

public ListaReproduccion findListaResproduccionPorId(java.lang.Integer id) throws java.lang.Exception
Parámetro:
✓ id: Valor que representa la clave primaria del objeto tipo ListaReproduccion que se está consultando.
Retorno:
Objeto de tipo ListaReproduccion.
Throws:
java.lang.Exception

findTextoPorId

public Texto findTextoPorId(java.lang.Integer idT) throws java.lang.Exception
Parámetro:
✓ idT: Valor que representa la clave primaria del objeto tipo Texto que se está consultando.
Retorno:
Objeto de tipo Texto.
Throws:
java.lang.Exception

findTextoActivo

public Texto findTextoActivo() throws java.lang.Exception
Retorno:
Objeto de tipo Texto.
Throws:
java.lang.Exception

findAllTexto

```
public java.util.List<Texto> findAllTexto()
```

Retorno:

Lista de objetos de tipo Texto.

listaReproduccionPorId

```
public java.util.List<Lista> listaReproduccionPorId(java.lang.Integer id)
```

Parámetro:

✓ id: Valor por el cual se está realizando la búsqueda.

Retorno:

Lista de objetos de tipo Lista.

Clase ManagerUsuario

```
java.lang.Object  
    multimedia.model.manager.ManagerUsuario
```

```
public class ManagerUsuario  
    extends java.lang.Object
```

Constructor Summary

Constructor and Description
ManagerUsuario()

Method Detail**findAllUsuarios**

```
public java.util.List<Usuario> findAllUsuarios()
```

Retorno:

Lista de objetos de tipo Usuario.

registrarUsuario

```
public void registrarUsuario(java.lang.String nombre, java.lang.String usuario,  
    java.lang.String pass, java.lang.String rol) throws java.lang.Exception
```

Almacena los datos de un usuario en la tabla Usuario.

Parámetros:

✓ nombre: Nombre completo del usuario que se está creando.

- ✓ usuario: Nombre de usuario.
- ✓ pass: Contraseña para el usuario.
- ✓ rol: Rol que tendrá el usuario en el sistema: ADMINISTRADOR u OPERADOR.

Throws:

java.lang.Exception

findByUsuario

public Usuario findByUsuario(java.lang.String usuario)

Parámetro:

- ✓ usuario: Nombre de usuario mediante el cual se realiza la consulta.

Retorno:

Objeto de tipo Usuario.

actualizarUsuario

public void actualizarUsuario(java.lang.Integer id, java.lang.String nombre, java.lang.String usuario, java.lang.String rol) throws java.lang.Exception

Actualiza un registro en la tabla Usuario.

Parámetros:

- ✓ id: Clave primaria del objeto que se está actualizando.
- ✓ nombre: Nombre completo del usuario que se está creando.
- ✓ usuario: Nombre de usuario.
- ✓ rol: Rol que tendrá el usuario en el sistema: ADMINISTRADOR u OPERADOR.

Throws:

java.lang.Exception

cambiarPass

public void cambiarPass(java.lang.Integer id, java.lang.String pass) throws java.lang.Exception

Actualiza el valor del atributo password de un registro de la tabla Usuario.

Parámetros:

- ✓ id: Clave primaria del objeto tipo Usuario que se está actualizando.
- ✓ pass: Contraseña nueva para el usuario que se está actualizando.

Throws:

java.lang.Exception

cambiarEstado

public void cambiarEstado(java.lang.Integer id, java.lang.String estado) throws java.lang.Exception

Actualiza el atributo estado de un registro de la tabla Usuario.

Parámetros:

- ✓ id: Clave primaria del objeto que se está actualizando.
- ✓ estado: Puede ser ACTIVO o INACTIVO.

Throws:

java.lang.Exception

iniciarSesion

public Usuario iniciarSesion(Usuario u) throws java.lang.Exception

Verifica si los datos del usuario son correcto o no para poder iniciar sesión.

Parámetros:

- ✓ u: Objeto de tipo usuario.

Retorno:

Objeto de tipo usuario.

Throws:

java.lang.Exception

findAllTexto

public java.util.List<Texto> findAllTexto()

Retorno:

Lista de objetos de tipo Texto.

getMD5

public java.lang.String getMD5(java.lang.String cadena) throws java.lang.Exception

Método que encripta una cadena utilizando el algoritmo MD5.

Parámetro:

- ✓ cadena: Cadena que se va a encriptar.

Retorno:

Cadena encriptada.

Throws:

java.lang.Exception

Package multimedia.controller

Class ControllerMultimedia

java.lang.Object

multimedia.controller.ControllerMultimedia

public class **ControllerMultimedia**

extends java.lang.Object

Constructor Summary

Constructor and Description
ControllerMultimedia()

Method Summary

Modifier and Type	Method and Description
void	actualizarRecurso()
void	buscarRecurso(Recurso recurso)
void	cargaRecursoAuxiliar(Recurso recurso)
void	eliminarRecurso(Recurso recurso)
org.primefaces.model.Uploa dedFile	getArchivo()
java.lang.String	getDescripcion()
java.util.List<Recurso>	getListaRecurso()
java.util.List<Configuracion>	getListaRutas()
java.lang.String	getNombre()
java.lang.String	getNombreRuta()
Recurso	getRecursoAux()
java.lang.String	getRuta()
org.primefaces.model.Strea medContent	getVistaPreviaImagen(java.lang.String rutaRelativa)
void	guardarArchivo()
void	ingresarRutaRecursos()
void	iniciar()
void	prepararImagen(java.lang.String rutaCompleta)
static double	redondearDecimales(double valorInicial, int numeroDecimales)
java.lang.String	rutaVideo(java.lang.String rutaRelativa)
void	setArchivo(org.primefaces.model.UploadedFile archivo)
void	setDescripcion(java.lang.String descripcion)
void	setListaRecurso(java.util.List<Recurso> listaRecurso)
void	setListaRutas(java.util.List<Configuracion> listaRutas)
void	setNombre(java.lang.String nombre)
void	setNombreRuta(java.lang.String nombreRuta)
void	setRecursoAux(Recurso recursoAux)
void	setRuta(java.lang.String ruta)
void	setVistaPreviaImagen(org.primefaces.model.StreamedCont ent vistaPreviaImagen)
void	subirArchivo(java.lang.String nombreArchivo, java.io.InputStream entrada)
java.lang.String	tipoArchivo(java.lang.String tipo)
boolean	verificarTipoRecurso(java.lang.String tipo)

Method Detail

iniciar

@PostConstruct

```
public void iniciar()
```

ingresarRuta

```
public void ingresarRutaRecursos()
```

Método que permite ingresar la ruta en donde se van a almacenar los diferentes recursos multimedia que se registren en el sistema.

prepararImagen

```
public void prepararImagen(java.lang.String rutaCompleta)
```

Método que recupera un recurso multimedia a partir de la ruta de la carpeta.

- ✓ rutaCompleta: Ruta en donde se encuentra el recurso que se está leyendo.

getVistaPrevialImagen

```
public org.primefaces.model.StreamedContent
```

```
getVistaPrevialImagen(java.lang.String rutaRelativa)
```

Método que devuelve un objeto de tipo StreamedContent mediante el cual es posible visualizar una imagen o animación usando la etiqueta p:graphicImage.

- ✓ rutaRelativa: Ruta relativa de la imagen o animación que se desea mostrar.

rutaVideo

```
public java.lang.String rutaVideo(java.lang.String rutaRelativa)
```

Método que a partir de la ruta relativa de un determinado video devuelve la ruta completa para poder visualizar el video.

Retorno:

Ruta completa de un determinado video.

subirArchivo

```
public void subirArchivo(java.lang.String nombreArchivo, java.io.InputStream  
entrada)
```

Método que permite guardar un recurso multimedia en la carpeta que el usuario haya especificado en la tabla Configuración.

- ✓ nombreArchivo: Nombre del archivo que se está guardando.

tipoArhivo

```
public java.lang.String tipoArchivo(java.lang.String tipo)
```

Método que devuelve el tipo de archivo que se está guardando a partir de la extensión.

- ✓ tipo: Extensión del archivo (ejemplo png, jpeg, mp4, etc.).

Retorno:

Tipo de archivo que se está subiendo: IMAGEN, VIDEO o ANIMACION.

verificarTipoRecurso

```
public boolean verificarTipoRecurso(java.lang.String tipo)
```

Método que verifica si el archivo que se está guardando tiene la extensión permitida por el sistema.

- ✓ tipo: Extensión del archivo que se está guardando (ejemplo png).

Retorno:

Boolean true o false.

guardarArchivo

```
public void guardarArchivo()
```

Método que permite al usuario guardar los datos del recurso que se está guardando en la tabla Recurso y el archivo en la carpeta que el usuario haya especificado en la tabla Configuración.

redondearDecimales

```
public static double redondearDecimales(double valorInicial, int numeroDecimales)
```

Método que permite redondear un número con varios decimales, dependiendo de la cantidad de decimales que desee obtener.

- ✓ valorInicial: Cantidad con todos los decimales.
- ✓ numeroDecimales: Cantidad de decimales que se desea obtener.

Retorno:

Valor con los decimales requeridos.

eliminarRecurso

```
public void eliminarRecurso(Recurso recurso)
```

Método que elimina un determinado recurso tanto de la base de datos como de la carpeta en donde estaba almacenado. El recurso que se está eliminando también se borra de las listas de reproducción a las que haya pertenecido.

actualizarRecurso

```
public void actualizarRecurso()
```

Método que permite actualizar un recurso.

cargaRecursoAuxiliar

```
public void cargaRecursoAuxiliar(Recurso recurso)
```

Permite almacenar los datos de un recurso en una variable auxiliar para poder manipular dichos datos.

Class ControllerListaReproduccion

```
java.lang.Object
```

```
multimedia.controller.ControllerListaReproduccion
```

```
public class ControllerListaReproduccion
```

```
extends java.lang.Object
```

Constructor Summary

Constructor and Description
ControllerListaReproduccion()

Method Detail**Iniciar**

```
@PostConstruct
```

```
public void iniciar()
```

cargarListaImágenes

```
public void cargarListaImágenes()
```

Método que recupera en una lista todos los recursos de tipo IMAGEN o ANIMACION.

cargarListaVideos

```
public void cargarListaVideos()
```

Método que recupera en una lista todos los recursos de tipo VIDEO.

cargarListasReproduccionVideos

```
public void cargarListasReproduccionVideos()
```

Método que recupera en una lista todas las listas de reproducción de videos.

cargarListasReproduccionImágenes

```
public void cargarListasReproduccionImágenes()
```

Método que recupera en una lista todas las listas de reproducción de imágenes.

cargarListaPorIdLista

```
public void cargarListaPorIdLista(ListaReproduccion lr)
```

Método que recupera en una lista todos los recursos que pertenecen a una determinada lista de reproducción.

Parámetro:

- ✓ lr: Lista de reproducción de la cual se desea obtener los recursos.

cargarRutaVideo

```
public void cargarRutaVideo(java.lang.String rutaRelativa)
```

Método que genera la ruta completa de un video.

Parámetro:

- ✓ rutaRelativa: Ruta relativa del video que se desea obtener la ruta completa.

lipiarRuta

```
public void lipiarRuta()
```

Borra los valores de la variable que contiene el valor de la ruta, carga las listas de los recursos de tipo video e imagen.

cargarRecursoVistaPrevia

```
public void cargarRecursoVistaPrevia(Recurso r)
```

Método que acumula los valores de un recurso en una variable auxiliar para poder manipular los datos.

Parámetro:

- ✓ r: Objeto de tipo Recurso.

crearListaReproduccionTemp

```
public void crearListaReproduccionTemp()
```

Crea una lista de reproducción en memoria, un objeto de tipo ListaReproduccion que almacena temporalmente el nombre de la lista.

agregarRecurso

public void agregarRecurso(Recurso r)

Permite agregar un video a la lista de reproducción que se encuentra creada en memoria.

Parámetro:

- ✓ r: Objeto de tipo Recurso que se va a agregar a la lista.

quitarVideo

public void quitarVideo(Lista l)

Permite quitar un video de la lista que se encuentra creada temporalmente en memoria.

Parámetro:

- ✓ l: Objeto de tipo Lista.

guardarListaReproduccionVideo

public void guardarListaReproduccionVideo()

Almacena en la base de datos la lista que se encuentra creada en memoria.

crearListaReproduccionTemplmg

public void crearListaReproduccionTemplmg()

Crea una lista de reproducción en memoria, un objeto de tipo ListaReproduccion que almacena temporalmente el nombre de la lista.

agregarRecursoImg

public void agregarRecursoImg(Recurso r)

Permite agregar una imagen a la lista de reproducción que se encuentra creada en memoria.

Parámetro:

- ✓ r: Objeto de tipo Recurso que se va a agregar a la lista.

quitarImagen

public void quitarImagen(Lista l)

Permite quitar una imagen de la lista que se encuentra creada temporalmente en memoria.

Parámetro:

- ✓ l: Objeto de tipo Lista.

guardarListaReproduccionImg

```
public void guardarListaReproduccionImg()
```

Almacena en la base de datos la lista que se encuentra creada en memoria.

añadirNuevoRecurso

```
public void añadirNuevoRecurso(Recurso r)
```

Agrega un nuevo recurso a una lista de reproducción existente ya sea de video o de imagen.

Parámetro:

- ✓ r: Objeto de tipo Recurso.

eliminarListaDeReproduccion

```
public void eliminarListaDeReproduccion(ListaReproduccion lr)
```

Elimina una lista de reproducción existente ya sea de imagen o video.

Parámetro:

- ✓ lr: Objeto de tipo ListaReproduccion que se va a eliminar.

eliminarRecursoDeLista

```
public void eliminarRecursoDeLista(Lista l)
```

Elimina una imagen o video de una lista de reproducción.

Parámetro:

- ✓ l: Objeto de tipo Lista que contiene la información requerida para eliminar el recurso de la lista de reproducción.

Class ControllerCartelera

java.lang.Object

multimedia.controller.ControllerCartelera

All Implemented Interfaces:

java.io.Serializable

```
public class ControllerCartelera
extends java.lang.Object
implements java.io.Serializable
```

Constructor Summary

Constructor and Description
ControllerCartelera()

Method Detail

iniciar

@PostConstruct
public void iniciar()

cargarValores

public void cargarValores(java.lang.Integer idV, java.lang.Integer idI,
java.lang.Integer idT)

Recupera los valores de la configuración realizada sobre la cartelera (Lista de video, lista de imagen, texto que se utilizó en la cartelera la última vez). Verifica que los valores que contienen los parámetros sean diferentes de null.

Parámetros:

- ✓ idV: Clave primaria de la lista de video que se mostró en la cartelera la última vez.
- ✓ idI: Clave primaria de la lista de imagen que se mostró en la cartelera la última vez.
- ✓ idT: Clave primaria del texto que se mostró en la cartelera la última vez.

seleccionarListaVideo

public void seleccionarListaVideo(ListaReproduccion lv)

Captura los valores de la lista de video que se está seleccionando en una variable para su posterior manipulación.

Parámetro:

- ✓ lv: Objeto de tipo ListaReproduccion que contiene la información de la lista seleccionada.

seleccionarListaImagen

public void seleccionarListaImagen(ListaReproduccion li)

Captura los valores de la lista de imagen que se está seleccionando en una variable para su posterior manipulación.

Parámetro:

- ✓ lv: Objeto de tipo ListaReproduccion que contiene la información de la lista seleccionada.

definirTexto

```
public void definirTexto()
```

Permite asignar un nuevo texto a la cartelera.

seleccionarTextoAnterior

```
public void seleccionarTextoAnterior(Texto t)
```

Permite seleccionar un texto que se ha utilizado anteriormente para mostrar nuevamente en la cartelera.

guardar

```
public java.lang.String guardar(java.lang.String layout)
```

Ejecuta el método guardarConfiguracionActual() después de validar que todos los campos de la configuración se hayan completado (de acuerdo al diseño de cartelera seleccionado).

Parámetro:

- ✓ latouy: Diseño de cartelera que se ha seleccionado.

Retorno:

Valor (String) que retorna el método guardarConfiguracionActual().

guardarConfiguracionActual

```
public java.lang.String guardarConfiguracionActual()
```

Almacena en la base de datos la configuración que se ha ingresado; verifica si se está ingresando un texto nuevo o se está reutilizando un anterior, dependiendo del resultado guarda el nuevo texto. Además se verifica si es la primera vez que se ingresa una configuración, de ser así se crea un nuevo registro en la tabla ConfiguracionCartelera caso contrario se actualiza la configuración existente.

Retorno:

Nombre de la página a donde se tiene que redirigir.

seleccionarLayout

```
public void seleccionarLayout(java.lang.String layout)
```

Verifica el diseño de cartelera que se ha seleccionado para de acuerdo a ese valor mostrar los campos que se debe completar para realizar la configuración de la cartelera.

Parámetro:

- ✓ layout: Diseño de cartelera seleccionado.

Class ControllerPresentacionCartelera

java.lang.Object

multimedia.controller.ControllerPresentacionCartelera

public class **ControllerPresentacionCartelera**

extends java.lang.Object

Constructor Summary

Constructor and Description
ControllerPresentacionCartelera()

Method Detail

iniciar

@PostConstruct

public void iniciar()

cargarValores

public void cargarValores(java.lang.Integer idV, java.lang.Integer idI,
java.lang.Integer idT)

Recupera los valores de la configuración de la cartelera (lista de video, lista de imagen, tiempo de cambio de las imágenes, texto). Verifica que los valores de los parámetros sean diferentes de null; además, ejecuta los métodos cargarRutasVideos() y cargarRutasImágenes().

Parámetros:

- ✓ idV: Clave primaria de la lista de video.
- ✓ idI: Clave primaria de la lista de imagen.
- ✓ idT: Clave primaria del texto.

cargarRutasImágenes

public void cargarRutasImágenes()

Recupera en una sola variable todas las rutas de las imágenes que pertenecen a la lista de reproducción de imágenes que se está utilizando.

cargarRutasVideos

```
public void cargarRutasVideos()
```

Recupera en una sola variable todas las rutas de los videos que pertenecen a la lista de reproducción de video que se está utilizando.

Class ControllerUsuario

```
java.lang.Object
```

```
multimedia.controller.ControllerUsuario
```

```
public class ControllerUsuario
```

```
extends java.lang.Object
```

Constructor Summary

Constructor and Description
ControllerUsuario()

Method Detail

iniciar

```
@PostConstruct
```

```
public void iniciar()
```

guardarUsuario

```
public void guardarUsuario()
```

Verifica que los valores que los valores sean diferentes de null para comprobar que el usuario haya completado todos los campos y guardar los datos del usuario en la base de datos y recupera la lista de todos los usuarios en una lista.

actualizarUsuario

```
public void actualizarUsuario()
```

Actualiza los datos de un usuario y se controla que no se ingrese valores nulos.

cambiarEstado

```
public void cambiarEstado(Usuario u)
```

Cambia de estado ACTIVO a INACTIVO o viceversa dependiendo del estado actual que tenga el usuario.

Parámetro:

- ✓ u: Objeto de tipo Usuario que contiene la información del usuario que se está actualizando.

cargarListaTexto

```
public void cargarListaTexto()
```

Recupera en una lista todos los textos que se han mostrado en la cartelera.

cargarUsuarioAux

```
public void cargarUsuarioAux(Usuario u)
```

Acumula en una variable auxiliar los datos de un usuario.

Parámetro:

- ✓ u: Objeto de tipo Usuario que contiene los datos de un usuario.

Class ControllerLogin

```
java.lang.Object
```

```
multimedia.controller.ControllerLogin
```

All Implemented Interfaces:

```
java.io.Serializable
```

```
public class ControllerLogin
```

```
extends java.lang.Object
```

```
implements java.io.Serializable
```

Constructor Summary

Constructor and Description
ControllerLogin()

Method Detail

iniciarSesion

```
public java.lang.String iniciarSesion()
```

Método que verifica los datos que proporciona el usuario para ingresar al sistema, además si los datos son correctos se crea la variable de sesión con los datos del usuario.

Retorno:

Nombre de la página a donde se tiene que redirigir si los datos del usuario son correctos.

cerrarSesion

```
public void cerrarSesion()
```

Método que borra los datos de la variable de sesión del usuario que ingresó al sistema e invalida la sesión.

verificarSesion

```
public void verificarSesion()
```

Recupera los datos de la variable de sesión para verificar si la sesión esta activa o no.

cambiarPass

```
public void cambiarPass()
```

Permite al usuario cambiar la contraseña, verifica que la contraseña no sea null luego guarda los cambios.

verificarPasswords

```
public void verificarPasswords()
```

Verifica que la contraseña que ingresa el usuario sea igual a la contraseña que actualmente está utilizando, para comprobar que la persona que está cambiando la contraseña sea la dueña de la cuenta de usuario.

activarMenu

```
public void activarMenu(java.lang.String rol)
```

Activa las diferentes opciones del menú dependiendo del rol que tenga el usuario que está ingresando al sistema.

Parámetro:

- ✓ rol: Rol correspondiente al usuario que ingresa al sistema

desactivarMenu

```
public void desactivarMenu()
```

Inactiva todas las opciones del menú para que cuando el usuario finalice la sesión no exista ningún menú activo.

Package utilitarios**Class JSFUtil**

```
java.lang.Object  
    utilitarios.JSFUtil
```

```
public class JSFUtil  
    extends java.lang.Object
```

Constructor Summary

Constructor and Description
JSFUtil()

Method Summary

Modifier and Type	Method and Description
static void	crearMensaje (javax.faces.application.FacesMessage.Severity severity, java.lang.String mensaje, java.lang.String detalle) Crea un mensaje JSF
static void	crearMensajeERROR (java.lang.String mensaje, java.lang.String detalle) Crea un mensaje JSF de tipo de severidad ERROR.
static void	crearMensajeINFO (java.lang.String mensaje, java.lang.String detalle) Crea un mensaje JSF de tipo de severidad INFORMATIVE.
static void	crearMensajeWARN (java.lang.String mensaje, java.lang.String detalle) Crea un mensaje JSF de tipo de severidad WARNING.

DICCIONARIO DE DATOS**Tabla recurso**

NOMBRE:	recurso
----------------	---------

DESCRIPCIÓN:	Tabla en la que se almacena la descripción de todos los recursos multimedia.	
NOMBRE	TIPO DATO	DESCRIPCIÓN
id_recurso	serial	Clave primaria de la tabla. Generada automáticamente por la secuencia recurso_id_recurso_seq.
nombre	character varying(50)	Nombre del recurso.
descripcion	character varying(150)	Descripción que hace referencia al contenido del recurso.
ruta	character varying(150)	Ruta relativa del recurso.
tipo	character varying(25)	Tipo de archivo multimedia IMAGEN, VIDEO o ANIMACION.
tamano	character varying(50)	Tamaño del archivo multimedia en megas.
fecha_creacion	date	Fecha actual en la que se está creando el registro.

Restricciones

NOMBRE	TIPO	DEFINICION
pk_recurso	primary key	(id_recurso)

Tabla lista_reproduccion

NOMBRE:	Lista_reproduccion	
DESCRIPCIÓN:	Tabla donde se almacena la información general de la lista de reproducción.	
NOMBRE	TIPO DATO	DESCRIPCIÓN
id_lista_reproduccion	serial	Clave primaria de la tabla. Generada automáticamente por la secuencia lista_reproduccion_id_lista_reproduccion_seq.
nombre_lista	character varying(100)	Nombre correspondiente a la lista de reproducción.
fecha	date	Fecha actual en la que se está creando el registro.

Restricciones

NOMBRE	TIPO	DEFINICION
pk_lista_reproduccion	primary key	(pk_lista_reproduccion)

Tabla lista

NOMBRE:	lista	
DESCRIPCIÓN:	Tabla que almacena el detalle de las listas de reproducción.	
NOMBRE	TIPO DATO	DESCRIPCIÓN
id_lista	serial	Clave primaria de la tabla. Generada automáticamente por la secuencia lista_id_lista_seq.
id_lista_reproduccion	integer	Clave foránea que hace referencia a la lista de reproducción (tabla lista_reproduccion) a la que pertenece el detalle.
id_recurso	integer	Clave foránea que hace referencia a la tabla recurso.

Restricciones

NOMBRE	TIPO	DEFINICION
pk_lista	primary key	(id_lista)
fk_lista_reference_lista_re	foreign key	(id_lista_reproduccion) REFERENCES lista_reproduccion (id_lista_reproduccion) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT
fk_lista_reference_recurso	foreign key	(id_recurso) REFERENCES recurso (id_recurso) MATCH SIMPLE ON UPDATE RESTRICT ON DELETE RESTRICT

Tabla configuracion

NOMBRE:	configuracion	
DESCRIPCIÓN:	Tabla que almacena la ruta de la carpeta donde se van a almacenar todos los recursos multimedia.	
NOMBRE	TIPO DATO	DESCRIPCIÓN
id_configuracion	serial	Clave primaria de la tabla. Generada automáticamente por la secuencia configuracion_id_configuracion_seq.
carpeta	character varying(150)	Campo que guarda la ruta de la carpeta.
activo	character varying(2)	Campo que almacena el estado de la ruta (si o no).

Restricciones

NOMBRE	TIPO	DEFINICION
pk_configuracion	primary key	(id_configuracion)

Tabla configuración_cartelera

NOMBRE:	configuración_cartelera	
DESCRIPCIÓN:	Tabla que almacena la configuración de la cartelera.	
NOMBRE	TIPO DATO	DESCRIPCIÓN
id_cartelera	serial	Clave primaria de la tabla. Generada automáticamente por la secuencia configuracion_cartelera_id_cartelera_seq.
layout	character varying(25)	Campo que almacena el diseño que tendrá la cartelera.
id_lista_video	integer	Campo que almacena el valor de la clave primaria de la lista de video que se va a mostrar en la cartelera.
lista_video	character varying(100)	Campo que guarda el nombre de la lista de video.
id_lista_imagen	integer	Campo que almacena el valor de la clave primaria de la lista de video que se va a mostrar en la cartelera.
lista_imagen	character varying(100)	Campo que guarda el nombre de la lista de imagen.
tiempo_cambio_imagen	integer	Campo que guarda el tiempo que se tarda en cambiar cada imagen de la lista.
id_texto	integer	Campo que almacena el valor de la clave primaria del texto que se va a mostrar en la cartelera.
texto	character varying(500)	Campo que almacena el texto que se muestra en la cartelera.
color_fondo	character varying(25)	Campo que almacena el código hexadecimal del color de fondo de la cartelera.

Restricciones

NOMBRE	TIPO	DEFINICION
pk_configuracion_cartelera	primary key	(id_cartelera)

Tabla texto

NOMBRE:	texto	
DESCRIPCIÓN:	Tabla que almacena el texto que se muestra en la cartelera.	
NOMBRE	TIPO DATO	DESCRIPCIÓN
id_texto	serial	Clave primaria de la tabla. Generada automáticamente por la secuencia texto_id_texto_seq.
texto	character varying(500)	Campo que almacena el texto que se va a mostrar en la cartelera.

fecha_creacion	date	Campo que almacena la fecha actual en la que se realiza el registro.
activo	character varying(2)	Campo que almacena el estado del texto (si o no).

Restricciones

NOMBRE	TIPO	DEFINICION
pk_texto	primary key	(id_texto)

Tabla usuario

NOMBRE:	usuario	
DESCRIPCIÓN:	Tabla en la que se almacenan los usuarios del sistema.	
NOMBRE	TIPO DATO	DESCRIPCIÓN
id_recurso	serial	Clave primaria de la tabla. Generada automáticamente por la secuencia usuario_id_usuario_seq.
nombre	character varying(75)	Campo que almacena el nombre completo del usuario.
usuario	character varying(25)	Campo que almacena el alias que se usará como nombre de usuario.
password	character varying(75)	Campo que almacena la contraseña (encriptada con md5) que el usuario utilizará para ingresar al sistema.
rol	character varying(15)	Campo que almacena el rol (administrador u operador) que tendrá el usuario.
estado	character varying(10)	Campo que almacena el estado del usuario (activo o inactivo).

Restricciones

NOMBRE	TIPO	DEFINICION
pk_usuario	primary key	(id_usuario)