Lando

Programación II

30/05/2019

Aritz Garitano

Rubén Domínguez

# Índice

Contenido

[Índice 1](#_Toc10153737)

[Contexto y motivación 2](#_Toc10153738)

[Estructura de aplicación desarrollada 2](#_Toc10153739)

[Paquetes 2](#_Toc10153740)

[Común 2](#_Toc10153741)

[Excepciones 2](#_Toc10153742)

[LD 2](#_Toc10153743)

[LN 3](#_Toc10153744)

[LP 3](#_Toc10153745)

[Desarrollos de terceros 3](#_Toc10153746)

[Detalles Técnicos 4](#_Toc10153747)

[Planificación 5](#_Toc10153748)

[Desvíos 5](#_Toc10153749)

[Objetivos fijados 5](#_Toc10153750)

[Desarrollos futuros y opciones de ampliación 5](#_Toc10153751)

[Bibliografía 6](#_Toc10153752)

# Contexto y motivación

La idea de nuestra aplicación se nos ocurrió debido a que los días anteriores al comienzo del proyecto de estuvimos conversando acerca de comics, CDs, DVDs, libros… que teníamos por casa. En particular de los problemas de uno de los integrantes de tener en varios sitios diferentes objetos y luego echarlos en falta.

Ya había aplicaciones de solo libros, de solo películas… pero nosotros pensamos en una que abarcara todo lo que tuviéramos y tenerlo en la misma aplicación. Por ello originalmente nuestra base de datos estaba formada por libros, canciones, películas, videojuegos y objetos de colección. Pero al crearla nos dimos cuenta de que quizá estábamos abarcando demasiado, por lo que la recortamos a libros, canciones y películas.

Nuestros requisitos para el proyecto era que pudieras agrupar canciones, películas y libros en “Librerías Multimedia”. Estos grupos podrían representar por ejemplo una colección en particular y una descripción ej: Marvel (Nombre), colección de comics y películas de Marvel que está en casa de los abuelos (Descripción).

# Estructura de aplicación desarrollada

## Paquetes

### Común

Consta de una clase formada por las constantes de inserción y consulta y la interfaz itfpropertyV2 (Mejora de la interfaz itfProperty presente en otro paquete) usada para mantener la independencia entre los paquetes LN y LP, devuelve la propiedad solicitada como un objeto.

### Excepciones

Alberga las excepciones de la aplicación, en este caso 4. Una para los campos con contenidos vacíos, otra para cuando programamos mal la petición de un case, una tercera para las repeticiones de títulos y una ultima para en caso de pedir una propiedad que no existe.

### LD

Alberga la clase clsConexiónBD esta clase es la que establece conexión con la BD, se le dan los parámetros de conexión y con ellos forma una expresión de conexión que luego es leída por el driver de MySQL.

También alberga antiguas clases de inserción, consulta y datos de cuando comenzamos a hacer pruebas, ahora están en desuso.

El resto del paquete esta formado por las clases encargadas de la intercomunicación tabla a tabla entre la aplicación y la base BD.

### LN

Esta formada por una interfaz itfproperty (sustituida por una versión superior) que devolvía en diferentes tipos de datos en lugar de un objeto único como la versión nueva.

También cuenta con un gestor que se encarga de la comunicación entre LP y LN, una clase de comparación con Comparator y un motón de clases de tratamiento de datos.

### LP

Este paquete principalmente es para las pantallas y la interacción con el usuario.

## Desarrollos de terceros

La clase de UtilidadesLP por cortesía de Javier Cerro Fernandez.

Conector de MySQL de ALUD.

# Detalles Técnicos

# Planificación

## Desvíos

Nada más comenzar con el proyecto ya tuvimos problemas para coincidir y perdíamos mucho tiempo en desplazamientos, esto se solucionó cuando comenzamos a usar git y discord, que nos permitieron tener más flexibilidad.

Uno de los primeros problemas de programación que nos encontramos fue el de conexión entre aplicación y la BD, fue por inexperiencia principalmente se solucionó rápidamente. Aunque luego nos volvimos a encontrar problemas similares que solucionamos dedicándole tiempo.

Luego tuvimos un problema con los autoincrementales, que no incrementaban básicamente. Javi becario nos enseñó como arreglar ese problema.

Luego tuvimos que reestructurar todo LD y la BD para adecuarla a los requisitos que tendríamos más adelante perdimos una cantidad muy grande de tiempo y esto se repitió bastantes veces.

Por último un problema que se ido repitiendo bastante es, debido a que usamos intellij teníamos problemas para usar el Windows Builder por lo que al final tuvimos que hacer las ventanas a pelo por lo que perdimos mucho tiempo, para la siguiente usaremos eclipse que debe de ser más fácil.

## Objetivos fijados

Inicialmente los objetivos estaban un poco diluidos, teníamos una idea aproximada pero no sabíamos cómo íbamos a ir avanzando, queríamos tener unas librerías con la información de películas, libros, canciones, videojuegos… pero no sabíamos cómo quedaría. Por lo general nuestros objetivos han sido los de hacer lo que era necesario en el proyecto.

## Desarrollos futuros y opciones de ampliación

El proyecto no está al 100% de su potencial, se podrían añadir más tablas en la BD, con sus respectivos inserts, consultas, deletes… también se podrían añadir diferentes tipos de ordenación a la hora de visualizar los datos de pantalla. Otra cosa que podría sumar seria la de hacer infalibles las inserts haciendo que no se pudieran repetir los datos.

Aparte de eso tampoco estaría de mas generar mas javadocs, limpiar el código y hacer la interfaz de usuario mas decorada y con más imágenes.

# Bibliografía

stackoverfLow:

<https://stackoverflow.com/questions/1990817/how-to-make-a-jtable-non-editable>

Otras páginas:

<https://www.codejava.net/java-se/swing/editable-jtable-example>

You tube:

Canales:

<https://www.youtube.com/channel/UCdulIs-x_xrRd1ezwJZR9ww>

<https://www.youtube.com/channel/UCS3W5vFugqi6QcsoAIHcMpw>

Videos sueltos:

<https://www.youtube.com/watch?v=oBQjLgiBruM&t=8s>

<https://www.youtube.com/watch?v=yH_g6QGFqes&t=83s>

<https://www.youtube.com/watch?v=GAl1FSKvoFY>

<https://www.youtube.com/watch?v=nJBNMN4Dwss&frags=wn&ab_channel=pildorasinformaticas>

<https://www.youtube.com/watch?v=ZlyTn8PZ3Fc>