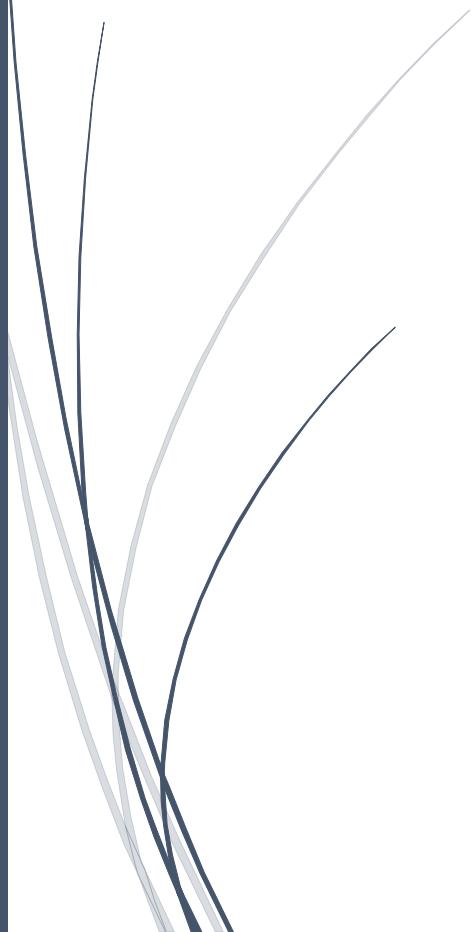


14 de marzo del 2018

MusicSelector

Proyecto Final Ciclo DAW



Autor: David Ruiz Requejo
Curso: 2º DAW (2017/2018)
Tutor: Teresa Martínez Suñer
Centro: C.P.I.F.P. Los Enlaces

Índice

1.	<i>Propuesta de Proyecto: MusicSelector</i>	3
1.1.	Título del Proyecto	3
1.2.	Título Corto	3
1.3.	Contexto del Proyecto	3
1.4.	Objetivo breve	3
2.	<i>Documento Descripción del Proyecto</i>	4
2.1.	Contexto del Proyecto	4
2.1.1.	Ámbito y entorno	4
2.1.2.	Análisis de la realidad	4
2.1.3.	Solución y Justificación de la solución propuesta	4
2.1.4.	Destinatarios	4
2.2.	Objetivos	5
2.2.1.	Objetivos (Versión en español)	5
2.2.2.	Objectives (English version)	5
3.	<i>Documento de Acuerdo en el Proyecto</i>	6
3.1.	Requisitos Funcionales y No Funcionales	6
3.2.	Tareas	7
3.3.	Metodología	7
3.3.1.	Metodologías Ágiles	7
3.3.2.	Modelo en Espiral	8
3.4.	Planificación de las tareas	8
3.5.	Presupuesto	9
3.5.1.	Beneficios	9
3.6.	Contrato y Pliego de condiciones	9
4.	<i>Documento y Análisis de Diseño</i>	10
4.1.	Modelado de Datos. Análisis y Diseño de la BBDD	10
4.1.1.	Patrones	10
4.1.2.	Ánalisis y Diseño de la BBDD	10
4.1.3.	Packages y Procedures	12
4.2.	Ánalisis y Diseño del Sistema Funcional	14
4.2.1.	Diagramas de Clases	15
4.2.2.	Diagramas de Actividad	15
4.2.3.	Diagramas de Casos de Uso	17
4.2.4.	Diagramas de Secuencia	24
4.3.	Ánalisis y Diseño de Interfaz de Usuario	44
4.3.1.	Intranet	44
4.3.2.	WebApp	45
4.4.	Diseño de la Arquitectura de la Aplicación	46
4.4.1.	Tecnologías y Herramientas usadas y descripción de las mismas	46
4.4.2.	Arquitectura y Componentes de la Aplicación	47
4.5.	Estructura de las Aplicaciones	49
5.	<i>Documento de Implementación e Instalación del Sistema</i>	50

5.1.	Pruebas	50
5.1.1.	Documento de “Planificación y registro de casos de prueba”.....	50
5.1.2.	Pruebas Funcionales.....	53
5.2.	Instalación y Configuración	54
5.3.	Manual de Usuario	54
5.3.1.	Intranet.....	54
5.3.2.	WebApp.....	62
6.	Documento de Cierre.....	66
6.1.	Resultados Obtenidos y Conclusiones.....	66
6.1.1.	Resultados Obtenidos.....	66
6.1.2.	Conclusiones.....	66
6.2.	Diario de Bitácora	66
7.	Bibliografía	67
8.	Anexos	68
8.1.	Contrato y Pliego de Condiciones (Apartado 3.6).....	68
8.2.	Scripts Base de Datos	73
8.2.1.	Creates.....	73
8.2.2.	Drops	77
8.2.3.	Sequences.....	78
8.2.4.	Triggers	79
8.2.5.	Inserts	81
8.2.6.	Packages and Procedures	87
8.3.	Requisitos Funcionales y No Funcionales	117
8.3.1.	Requisitos Funcionales	117
8.3.2.	Requisitos No Funcionales.....	131
8.4.	Diagramas de Secuencia (Apartado 4.2.2.).....	133
8.5.	Pruebas	149
8.5.1.	Pruebas Funcionales.....	149
8.5.2.	Tablas de Datos.....	158

1. Propuesta de Proyecto: MusicSelector

1.1. Título del Proyecto

Programa de gestión del catálogo musical para emisora de radio.

1.2. Título Corto

MusicSelector

1.3. Contexto del Proyecto

La famosa emisora de radio Los 50 primordiales ha solicitado a la empresa Strixes S.L. una aplicación para la gestión de su catálogo musical. Entre los diferentes puntos pactados con el cliente destacan:

- Una aplicación de escritorio o una aplicación web de uso interno (intranet) que optimice la búsqueda de canciones para emitir por antena.
- Una aplicación web en la que se puedan registrar los usuarios para poder disfrutar de contenidos musicales vía web: escuchar música, ver vídeos, petición de canciones que no estén en el catálogo...

1.4. Objetivo breve

Planificar y desarrollar las diferentes aplicaciones solicitadas por el cliente para la gestión de su catálogo.

La aplicación de escritorio o intranet será de uso interno para la emisora y en ella se podrá añadir, modificar, eliminar y listar contenido del catálogo además de seleccionar las canciones que se van a emitir por antena.

Desde la aplicación web los usuarios administradores podrán realizar las mismas tareas que desde la aplicación de escritorio. El resto de usuarios podrán, entre otras cosas, escuchar o visualizar contenidos seleccionando por título de la canción, autor, década, tipo de música.

2. Documento Descripción del Proyecto

2.1. Contexto del Proyecto

2.1.1. Ámbito y entorno

El proyecto va a consistir en la realización de un producto solicitado por la empresa “Los 50 Primordiales”. Esta empresa se dedica a la gestión de una emisora musical en la radio. Tras un año desde su puesta en escena han decidido intentar captar un mayor número de oyentes. Para ello han solicitado a la empresa Strixe S.L. la digitalización de sus datos y contenidos para tener un mayor control sobre los mismos, además de la creación de dos aplicaciones que permitirá la inserción de estos contenidos musicales en el mundo web donde se encuentra la mayor parte de los usuarios potenciales hoy en día.

Las aplicaciones van a ir dirigidas a los departamentos de gestión (intranet) y marketing (web). El departamento de gestión se encarga de la retransmisión y selección de los contenidos que se reproducen desde la emisora. La aplicación web (intranet) les permitirá la gestión, selección y programación del contenido musical. El apartado de marketing se encarga de la relación con el cliente. La web servirá, no solo para dar a conocer a la emisora, sino que también permitirá el acceso a una oferta mucho más versátil con opciones hasta ahora fuera del alcance de la emisión de radio tradicional.

2.1.2. Análisis de la realidad

Las aplicaciones parten desde cero ya que la emisora no cuenta con antecedentes en este campo de desarrollo. Aunque existen en el mercado productos desarrollados por otras emisoras en la misma línea que los propuestos por esta empresa, intentaremos crear un producto novedoso y que destaque del resto tanto por sus características, como por su diseño y contenido. Emisoras como “M80” o “Cadena Ser” cuentan con productos similares, si bien, la interacción con los usuarios se ve dificultada en gran medida por la cantidad de banners publicitarios.

2.1.3. Solución y Justificación de la solución propuesta

Antes de comenzar el desarrollo se han mantenido varias reuniones con los representantes de la emisora para decidir la estructura y requisitos mínimos que deberán cumplir las aplicaciones. Partiendo de esos requisitos mínimos indispensables propuestos por el cliente nos hemos reunido los representantes de los distintos departamentos de Strixe S.L. que van a interactuar y colaborar en el desarrollo de las aplicaciones para, por medio del “Brainstorming”, alcanzar un producto original y novedoso.

La primera aplicación consistirá en una aplicación web o de escritorio (se decidirá en próximas reuniones) para uso interno de la emisora. Básicamente constará de una base de datos del catálogo musical de la emisora y de un entorno amigable para poder gestionar su contenido.

La segunda aplicación será una aplicación web por medio de la cual los oyentes podrán acceder a contenidos musicales “a la carta”.

2.1.4. Destinatarios

Como ya he dicho anteriormente, el destinatario de este proyecto es la famosa emisora de radio “Los 40 Primordiales”, más concretamente sus departamentos de Gestión y Marketing si bien, todos los visitantes de la página web de la emisora serán destinatarios potenciales de la aplicación.

2.2. Objetivos

2.2.1. Objetivos (Versión en español)

El objetivo del proyecto, como ya hemos avanzado anteriormente, es la realización de dos aplicaciones (una de escritorio y una aplicación web, o dos aplicaciones web) para mejorar la gestión de los archivos musicales pertenecientes a la emisora, y la comunicación entre ésta y los usuarios.

Desde la primera aplicación usuarios registrados como “administradores”, de entre los miembros del personal de la emisora, podrán seleccionar las canciones que se vayan a emitir en cada momento por diversos criterios: Título, Autor, Época, Genero, etc.

Esto facilitará en gran medida el trabajo diario de los responsables de realizar la selección de canciones ya que, en la actualidad, se vienen gestionando estas tareas de forma manual a través de listados impresos en papel sin un orden lógico establecido. Con esto cumplimos uno de los objetivos fundamentales de este desarrollo, que es facilitar y agilizar la gestión del material de emisora y mejorar las condiciones laborales de los trabajadores de la misma.

A través de la segunda aplicación los usuarios que visiten la web podrán acceder a una serie filtros similares a los mencionados anteriormente para poder seleccionar y reproducir las canciones que deseen si están disponibles en el catálogo de la emisora. Además, podrán también tener a un catálogo de videos musicales que se podrá visualizar desde cualquier ordenador.

2.2.2. Objectives (English version)

The aim of the project, as I said before, is to create two applications (a desktop one and a web application, or two web applications) to improve the way the radio station staff handle with the company musical catalogue, and the communication between this one and the users.

With the first application users registered as admin will be able to select any song from this catalogue to listen to them in every moment by different criteria: Title, Author, Period, Genre, etc.

This development will make easier the hard daily work of the persons in charge of selecting the songs to broadcast because at present they are managing these tasks in a manual way, with printed lists without a logical established order. Thanks to this we get one of the main goals of this development, which is to facilitate and to improve the management of the radio station musical material and to improve the working conditions of the company staff.

With the second application the users who visit the web will be able to use different filters, similar to the ones I mentioned previously to be able to select and reproduce the songs that they wish, if they are available in the catalogue. In addition, they will be able to have also to a catalogue of musical videos to be enjoyed from any computer.

3. Documento de Acuerdo en el Proyecto

3.1. Requisitos Funcionales y No Funcionales

A lo largo de este apartado vamos a definir los requisitos funcionales y no funcionales de las aplicaciones a desarrollar.

Gracias a las diferentes reuniones realizadas con el cliente hemos podido definir de forma específica los requisitos base de las aplicaciones. Utilizar este método de análisis de requisitos va a permitir, no solo una mayor claridad a la hora de describir la estructura de los proyectos, sino que también habrá una mayor eficiencia a la hora de implementar las diferentes funcionalidades.

Hemos clasificado dichos requisitos en funcionales y no funcionales:

- Los requerimientos funcionales de una aplicación son aquellos que describen cualquier tarea y actividad que esta deba realizar.
- Los requerimientos no funcionales representan las características generales y restricciones de la aplicación que se esté desarrollando.

El formato que se ha decidido utilizar para la descripción de los requerimientos es el siguiente:

Requisitos Funcionales

Requisito: RF-NN ~ "Nombre del requisito"			
Tipo	Funcional	Prioridad	"Alta, Media, Baja"
Código	RF-NN	Aplicación	"Intranet, WebApp"
Etapa Implementación	"Etapa de Implementación"		
Nombre Requisito	"Nombre Requisito"		
Descripción Requisito	"Descripción Requisito"		

Requisitos No Funcionales

Requisito: RNF-NN ~ "Nombre del requisito"			
Tipo	No Funcional	Prioridad	"Alta, Media, Baja"
Código	RNF-NN	Aplicación	"Intranet, WebApp"
Fecha Impl.	"Fecha"	Fecha Límite Impl.	"Fecha"
Nombre Requisito	"Nombre Requisito"		
Descripción Requisito	"Descripción Requisito"		

Las tablas específicas de cada requisito funcional y no funcional pueden verse en el Anexo 8.2.

3.2. Tareas

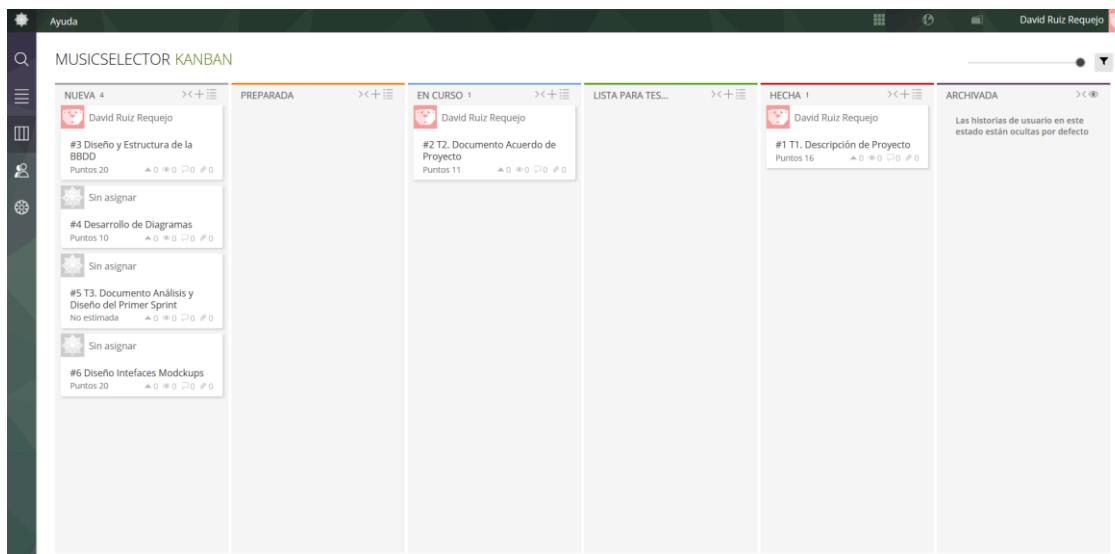
A continuación, adjunto el listado de las diferentes tareas que se van a realizar para esto proyecto:

TAREA	FCH INICIO	FCH FIN
T1. Descripción de Proyecto	-	25/03/2018
T2. Documento Acuerdo de Proyecto	26/03/2018	08/04/2018
T3. Documento Análisis y Diseño del Primer Sprint	09/04/2018	22/04/2018
T3.1. Diseño Estructura BBDD	09/04/2018	13/04/2018
T3.2. Diseño Interfaz Mockup	14/04/2018	15/04/2018
T3.3. Aprender Spring	15/04/2018	22/04/2018
T4. Primer Sprint (Aplicación Intranet)	23/04/2018	08/05/2018
T4.1. Sección Usuarios (Login, Registro, Editar)	23/04/2018	26/04/2018
T4.2. Sección Catálogo (Añadir, Eliminar, Editar, Lista Reproducción)	26/04/2018	06/04/2018
T5. Documento análisis y diseño del Segundo Sprint	09/05/2018	20/05/2018
T5.1. Diseño Interfaz Mockup	09/05/2018	12/05/2018
T6. Segundo Sprint	21/05/2018	01/06/2018
T6.1. Sección Usuarios (Login, Registro, Editar)	21/05/2018	22/05/2018
T6.2. Sección Catálogo (Listado, Filtro, Información, Lista Reproducción, Exportar)	22/05/2018	24/05/2018
T6.3. Sección Información (Contacto)	24/05/2018	01/06/2018
T7. Documento de instalación, documento de manual de usuario y documento de cierre	01/06/2018	03/06/2018

3.3. Metodología

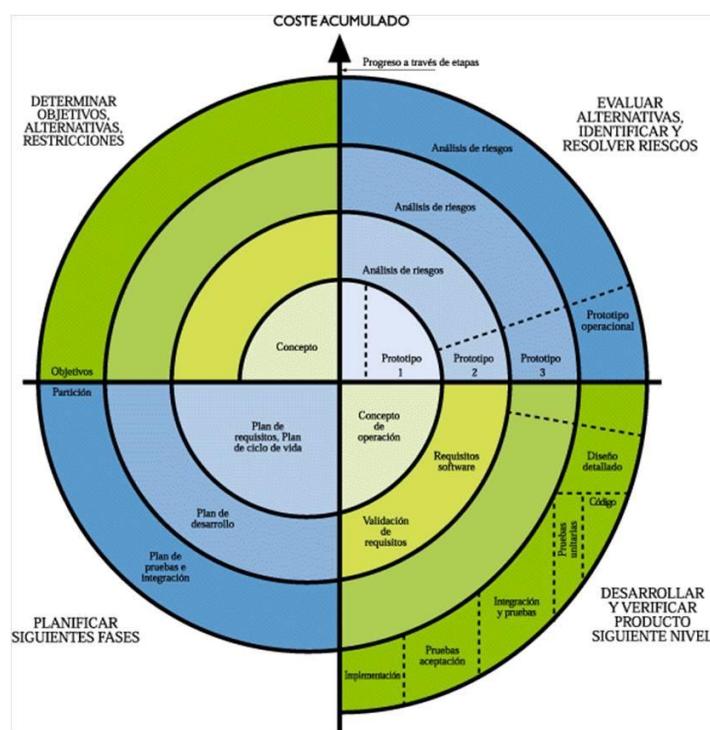
3.3.1. Metodologías Ágiles

Para el desarrollo de este proyecto se ha decidido utilizar la metodología ágil, Kanban, que consiste en dividir en bloques las diferentes tareas que hay que realizar a lo largo del proyecto, de manera que el trabajo en curso se limita, y sólo se empieza con un nuevo bloque cuando el anterior ha sido entregado o ha pasado a una función posterior de la cadena. Para poder realizar esto se ha seleccionado la herramienta Taiga.



3.3.2. Modelo en Espiral

Para el desarrollo de las aplicaciones se ha optado por la utilización de un Modelo en Espiral. Para seleccionar este modelo se han tenido en cuenta diferentes pautas. Un ejemplo de éstas es que, gracias a su desarrollo continuado, ayuda a la gestión de riesgo. Además, me va a permitir diferenciar los procesos de alta prioridad pudiendo, de esta forma, desarrollarlos de una forma más óptima. Otro dato importante es que, como el desarrollo del proyecto se realiza en pequeños fragmentos, la estimación de costos se simplifica. Finalmente, cabe destacar que el hecho de utilizar este modelo me va a permitir mantener informado al cliente de todos los cambios mientras nuestro sistema crece, lo que permite mejorar el desarrollo e irlo adaptando a las necesidades de éste.



3.4. Planificación de las tareas

3.5. Presupuesto

El Presupuesto disponible por la emisora para el desarrollo de las aplicaciones es de 8.000 €. Debido a las continuas modificaciones solicitadas por el cliente a lo largo de la elaboración de las mismas, el número de horas invertidas ha sido mayor del que originalmente preveíamos, por lo que el beneficio neto es menor del esperado.

En esta tabla podemos observar las horas empleadas en la elaboración del proyecto, agrupadas en función de las distintas tareas. Los precios por hora han sido calculados aplicando un importe inferior al habitual para dichas tareas, debido al interés que tenemos en establecer una buena relación con un cliente potencialmente importante en el devenir de nuestra empresa:

Apartado	Precio/Hora	Importe
Análisis	10,00 €	450,00 €
Diseño	10,00 €	1.170,00 €
Programación	20,00 €	2.810,00 €
Pruebas	10,00 €	800,00 €
Documentación	10,00 €	600,00 €
TOTAL		5.830,00 €

El total de gastos de distribuye entre los costes fijos, que serán cargados en su totalidad al proyecto (luz, teléfono, Internet, horas, etc.), y costes variables (herramientas, software, etc.). Estos últimos, por corresponderse a elementos que engrosarán el patrimonio de la empresa, y que podrán ser utilizados para futuros proyectos, repercutirán en un porcentaje en el cálculo total del presupuesto.

Costes Variables			Costes Fijos	
Concepto	Cantidad	Precio	Concepto	Precio
Ordenador	1	150,00 €	Luz y Teléfono	60,00 €
Programas	-	100,00 €	Internet	70,00 €
TOTAL		250,00 €	TOTAL	5.960,00 €

3.5.1. Beneficios

El precio técnico (Pt), que en este caso coincide con el coste total, por encontrarnos ante un único proyecto, asciende a 6.210,00 € (suma entre costes fijos y variables). Para realizar el cálculo hemos aplicado la fórmula $Pt = \frac{CF}{Q} + CVu$, siendo "Q" el número de proyectos, en este caso 1, "CF" los costes fijos, y "CVu" los costes variables por unidad.

Ya podemos calcular el beneficio neto, pues el precio de venta (P), como ya hemos avanzado anteriormente es de 8.000 €. Aplicando la fórmula $B = Q \times P - (Q \times CVu + CF)$, obtenemos que el beneficio neto será el resultado de la operación $1 \times 8.000 - (1 \times 250 + 5.960)$, es decir 1.790,00 euros.

3.6. Contrato y Pliego de condiciones

La copia del contrato firmado con la Emisora, así como el Pliego de Condiciones en el que se exponen las condiciones a las que deben sujetarse las dos partes que formalizan este contrato, se adjuntará en el Anexo de esta documentación.

4. Documento y Análisis de Diseño

4.1. Modelado de Datos. Análisis y Diseño de la BBDD

4.1.1. Patrones

Otro dato importante es la utilización del patrón de Modelo Vista Controlador. Este patrón me va a permitir realizar el proyecto de una forma modular, lo que implica que, a la hora de realizar mejoras o correcciones de fallos, únicamente tendremos que modificar el bloque correspondiente y no todo el mecanismo de la aplicación. Tras la experiencia obtenida durante mis años de formación, he decidido utilizar el patrón de DTO y el patrón DAO, ya que he llegado a la conclusión de que este patrón me permite un diseño bien elaborado pues las aplicaciones que lo presentan tienen una extensibilidad y una mantenibilidad únicas, comparadas con otras aplicaciones basadas en otros patrones.

La capa DTO contiene una clase java conocida como Value Object que únicamente contiene sus atributos, constructor, getters y setters. Esta clase no tiene comportamiento.

La capa DAO contiene todos los métodos CRUD (Create, Read, Update, Delete). Por lo general se tiene un DAO para cada tabla en la base de datos.

```
package com.musicselector.webapp.dao.user;

import ...

public class UserDAO extends DAOBase {

    LogUtil logger = new LogUtil();
    private final String CLASSNAME = "SecQuestionDAO";
    private String METHODNAME = "";

    public boolean checkUser(UserDTO loginUser) {...}

    public boolean checkNick(UserDTO dtoUser) {...}

    public boolean checkEmail(UserDTO dtoUser) {...}

    public UserDTO selectUserByNick(UserDTO dtoUser) {...}

    public ArrayList<UserDTO> listUsers() {...}

    public UserDTO selectUser(Integer idUser) {...}

    public boolean insertUser(UserDTO dtoUser) {...}

    public boolean updateUser(UserDTO dtoUser) {...}

    public boolean deleteUser(UserDTO dtoUser) {...}
```

DAO

```
package com.musicselector.webapp.dto.user;

import ...

/**
 * Clase UserDTO
 *
 * @version 1.0
 * @since 23/04/2018
 */
public class UserDTO {

    private IntegeridUser;
    @Size(min = 2, max = 20, message = "Tiene que tener entre (mi
    @NotEmpty(message = "No puede estar vacio.")
    private String nick;
    @Size(min = 4, max = 15, message = "Tiene que tener entre (mi
    @NotEmpty(message = "No puede estar vacio")
    private String password;
    @NotEmpty(message = "No puede estar vacio")
    private String name;
    @NotEmpty(message = "No puede estar vacio")
    private String surname;
    private String age;
    @Email
    @NotEmpty(message = "No puede estar vacio")
    private String email;
    @Size(min = 1, max = 60, message = "Tiene que tener entre (mi
    private String secAnswer;
    private Integer novelMusic;
    private String profileImage;
```

DTO

4.1.2. Análisis y Diseño de la BBDD

Se ha llevado a cabo la utilización de una BBDD desarrollada en SQL Developer (Oracle SQL), que permitirá tener almacenada la información acerca de los usuarios, catálogo musical y demás datos introducidos por ellos o generados por el propio programa.

Para el desarrollo de la base de datos se ha tenido en cuenta que en las distintas aplicaciones habrá que diferenciar entre los usuarios. Para ello, los usuarios tendrán asignados un rol que limitará las funcionalidades de cada uno dentro de la misma. A la aplicación de uso interno solo se podrá acceder con usuario y contraseña. Este usuario tendrá que tener como rol “administrador” para poder entrar, ya que

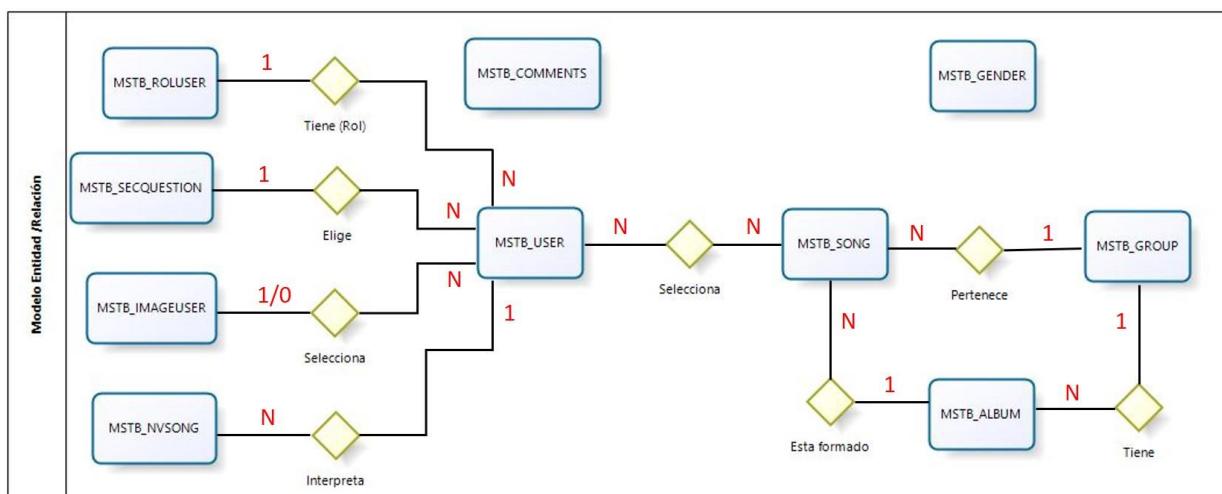
una vez dentro, se le dará control a la gestión musical, de los contenidos y de los datos de los usuarios de la emisora. En la aplicación, los usuarios “no admin” podrán suscribirse a la plataforma. A la hora de registrarse, los usuarios deberán elegir un Nick único con el que podrán loggearse en la aplicación. A los usuarios se les permitirá la personalización de sus perfiles desde la aplicación web.

También se tendrá una pantalla de comentarios donde los usuarios podrán abrir temas de conversación para hablar con otros suscriptores. Esta opción solo estará disponible si el usuario se encuentra loggeado en la aplicación web.

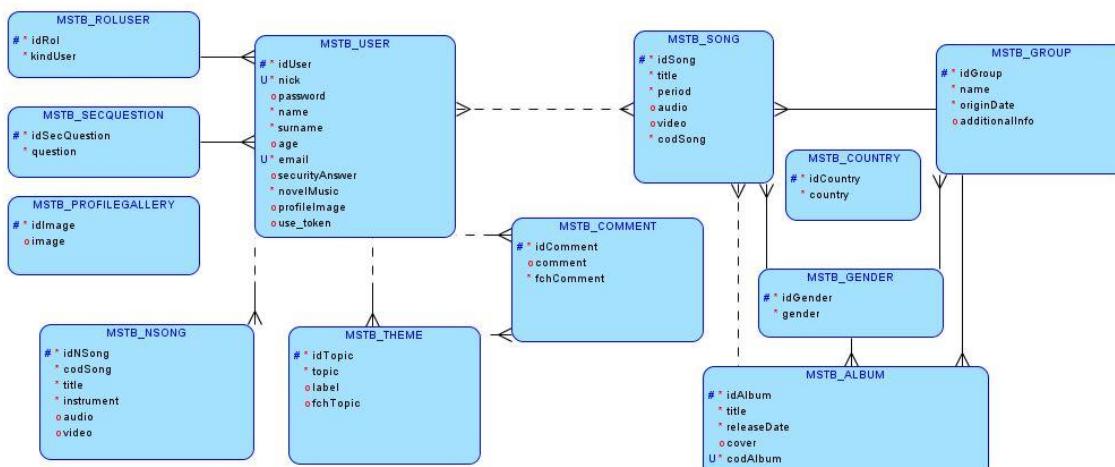
En el apartado de músicos nóveles, los usuarios loggeados podrán subir su propio contenido musical. A este contenido se le asignará un código para identificarlo que deberá ser único (Ej. 3 Primeras letras del Nick del usuario + “-“ + identificador del usuario + “-“ + número de archivos subidos).

Los usuarios podrán gestionar sus propias listas de reproducción seleccionando diferentes canciones del catálogo. A continuación, podemos observar el modelo entidad/relación, el modelo lógico y el modelo relacional de la BBDD.

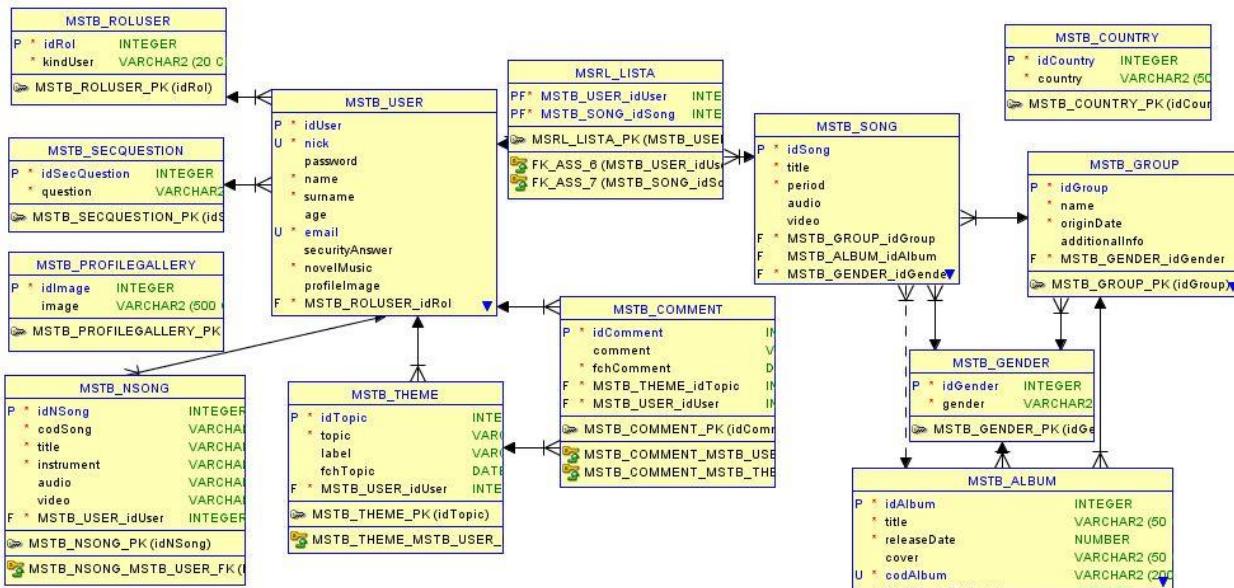
4.1.2.1. Diagrama E/R



4.1.2.2. Modelo Lógico



4.1.2.3. Modelo Relacional



4.1.3. Packages y Procedures

Además de la programación realizada en Java, se ha decidido implementar el uso de procedimientos programados en PL/SQL. Estos procedimientos se encargarán de realizar las diferentes acciones que se efectúan sobre las tablas de la BBDD. También se ha optado por realizar el proyecto de esta forma con el fin de obtener una organización que permite una mejor comprensión del código. Los procedimientos han sido almacenados en Packages según la parte del proyecto a la que afectan. Los packages que se han diseñado en el proyecto son los siguientes:

- **MSPKG_ALBUM**: En este paquete se encuentran todos los procedimientos que afectan a la tabla MSTB_ALBUM.
 - *LIST_ALBUMS*: Devuelve un listado de los Álbumes.
 - *ALBUMFILTER*: Devuelve un listado de los Álbumes filtrado por los parámetros que se le pasan al procedimiento.
 - *ALBUM_I*: Inserta un Álbum.
 - *ALBUM_U*: Actualiza un Álbum.
 - *ALBUM_D*: Elimina un Álbum.
 - *SELECT_ALBUM_CODALBUM*: Selecciona un Álbum por Código de Álbum.
 - *SELECT_ALBUM_ID*: Selecciona un Álbum por id.
 - *CHECK_CODALBUM*: Comprueba que el código del álbum no esté ya registrado en la aplicación.
 - *LIST_ALBUM_BY_GROUP*: Devuelve un listado con los Álbumes de un grupo.
 - *CHECK_ALBUM_OF_GROUP*: Comprueba que un Álbum pertenezca a un grupo.
- **MSPKG_COUNTRY**: En este paquete se encuentran todos los procedimientos que afectan a la tabla MSTB_COUNTRY.
 - *LIST_COUNTRIES*: Procedimiento que devuelve el Listado de los países.
- **MSPKG_GENDER**: En este paquete se encuentran todos los procedimientos que afectan a la tabla MSTB_GENDER.

- *LIST_GENDERS*: Procedimiento que devuelve el Listado de los géneros.
- **MSPKG_GROUP**: En este paquete se encuentran todos los procedimientos que afectan a la tabla MSTB_GROUP.
 - *LIST_GROUPS*: Devuelve un listado de los Artistas registrados en la aplicación.
 - *GROUPFILTER*: Devuelve un listado de los Artistas filtrado por los parámetros que se le pasan al procedimiento.
 - *GROUP_I*: Inserta un Artista.
 - *GROUP_U*: Actualiza un Artista.
 - *GROUP_D*: Elimina un Artista.
 - *SELECT_GROUP_ID*: Selecciona un Artista por id.
- **MSPKG_ROL**: En este paquete se encuentran todos los procedimientos que afectan a la tabla MSTB_ROL.
 - *LIST_ROLS*: Procedimiento que devuelve el Listado de los roles.
- **MSPKG_SECQUESTION**: En este paquete se encuentran todos los procedimientos que afectan a la tabla MSTB_SECQUESTION.
 - *LIST_QUESTION*: Procedimiento que devuelve el Listado de las preguntas de seguridad.
- **MSPKG_SONG**: En este paquete se encuentran todos los procedimientos que afectan a la tabla MSTB_SONG.
 - *LIST_SONGS*: Devuelve un listado de las Canciones catalogadas en la aplicación. (Estructura SongDTO)
 - *SONGFILTER*: Devuelve un listado de las Canciones filtrado por los parámetros que se le pasan al procedimiento. (Estructura SongDTO)
 - *SONG_I*: Añade una canción al Catálogo.
 - *SONG_U*: Actualiza una canción.
 - *SONG_D*: Elimina una canción.
 - *SELECT_SONG_ID*: Selecciona una canción por id.
 - *SELECT_SONG_COD*: Selecciona una canción por código.
 - *CHECK_SONG_COD*: Comprueba que el código de la canción no esté ya registrado en la aplicación.
 - *CHECK_SONG*: Comprueba si una canción por título y artista.
 - *SELECT_LAST_SONGS*: Devuelve un listado de las 3 últimas canciones introducidas en la Base de Datos.
 - *SELECT_SONGS_CATALOG*: Devuelve un listado de todas las canciones del catálogo. (Estructura CatalogoDTO)
 - *CATALOG_FILTER*: Devuelve un listado de las Canciones filtrado por los parámetros que se le pasan al procedimiento. (Estructura CatalogoDTO)
- **MSPKG_USER**: En este paquete se encuentran todos los procedimientos que afectan a la tabla MSTB_USER.
 - *CHECK_USER_ADMIN*: Comprueba que los datos introducidas se correspondan con un usuario con rol administrador dado de alta en la aplicación.
 - *CHECK_USER*: Comprueba que los datos introducidas se correspondan con un usuario dado de alta en la aplicación.
 - *SELECT_USER_DATA*: Selecciona un usuario por su Nick o Email.

- *LIST_USERS*: Procedimiento que devuelve el Listado de usuarios registrados en la aplicación.
- *SELECT_USER_ID*: Procedimiento que devuelve el usuario que tiene el id que se le pasa como parámetro.
- *USER_I*: Procedimiento que inserta un usuario.
- *USER_U*: Procedimiento que actualiza un usuario.
- *USER_D*: Procedimiento que elimina un usuario.
- *USERFILTER*: Devuelve un listado de los Usuarios filtrado por los parámetros que se le pasan al procedimiento.
- *CHECK_NICK*: Comprueba que un Nick no se esté usando ya en la aplicación.
- *CHECK_EMAIL*: Comprueba que un Email no se esté usando ya en la aplicación.
- *SELECT_USER_BY_EMAIL*: Selecciona un usuario por Email.
- *SELECT_USER_BY_TOKEN*: Selecciona un usuario por Token (Clave única).
- *UPDATE_TOKEN*: Se encarga de actualizar el campo Token de un Usuario.
- *REGENERATE_PASS*: Actualiza la contraseña de un Usuario.

4.2. Análisis y Diseño del Sistema Funcional

En este apartado veremos cómo se ha diseñado las aplicaciones. Para ello se han realizado diversos diagramas que abarcan las principales acciones de la aplicación. En estos diagramas se muestran tanto los procesos o acciones que se llevan a cabo en la aplicación como la autoría de las mismas.

Para realizar los diferentes diagramas se han establecido tablas que contendrán la información acerca de los mismos. Por ejemplo, los diagramas de Casos de Uso y de Secuencia van a ser representados de la siguiente forma:

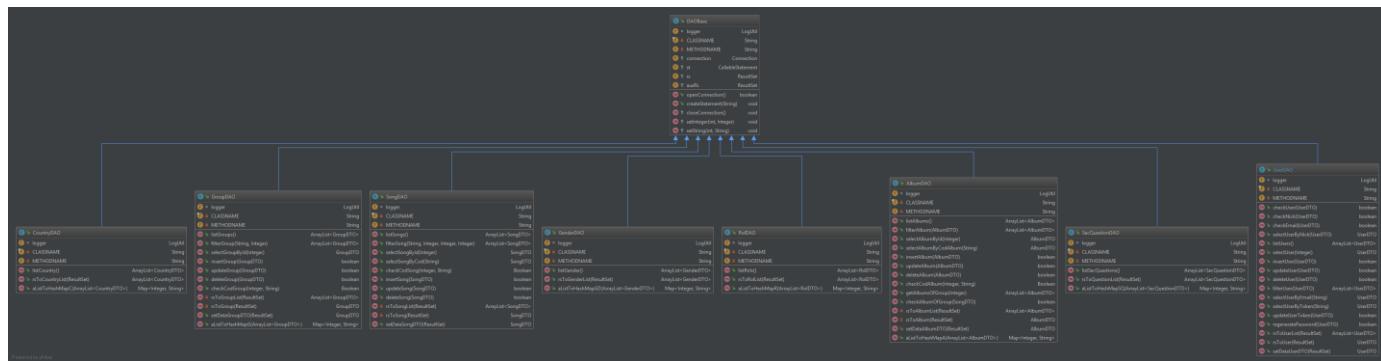
"DCU-XX" ~ "*****"			
D. Caso de Uso:	"*****"	Código:	"DCU-XX"
Implementación:	"Sprint 1/2"	Aplicación:	"Intranet/WebApp"
Descripción Corta:	"*****"		
Actores:	"*****"		
Flujo:	"*****"		

"DS-XX" ~ "*****"			
D. Secuencia:	"*****"	Código:	"DS-XX"
Implementación:	"Sprint 1/2"	Aplicación:	"Intranet/WebApp"
Descripción Corta:	"*****"		
Listado de Objetos:	• "*****"		
Proceso:	"*****"		

Junto a cada una de las tablas de los diagramas habrá una imagen del diagrama en sí.

4.2.1. Diagramas de Clases

Debido al gran tamaño resultante de los Diagramas de Clase no me es posible adjuntarlo en la documentación escrita. Por ello, adjunto los archivos en formato .uml y .png en una carpeta que será entregada en el CD al cliente. Véase un ejemplo a escala del resultado final.

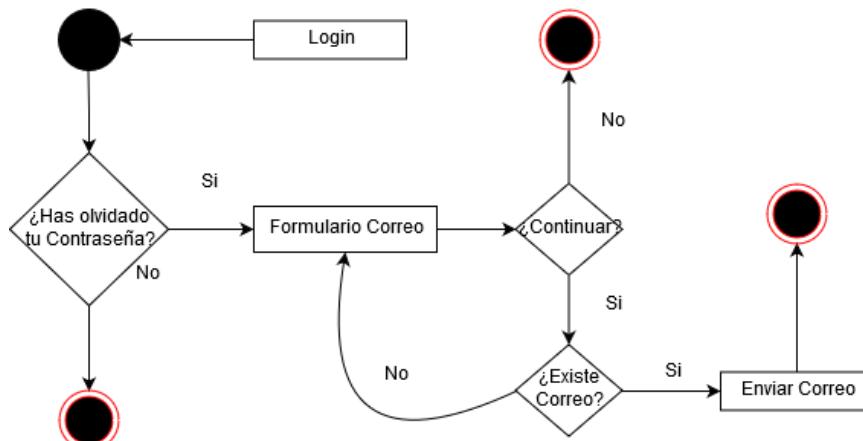


4.2.2. Diagramas de Actividad

Tras analizar los diferentes diagramas de Actividad que puede haber en la aplicación se ha decidido combinar algunos procesos similares en un único diagrama

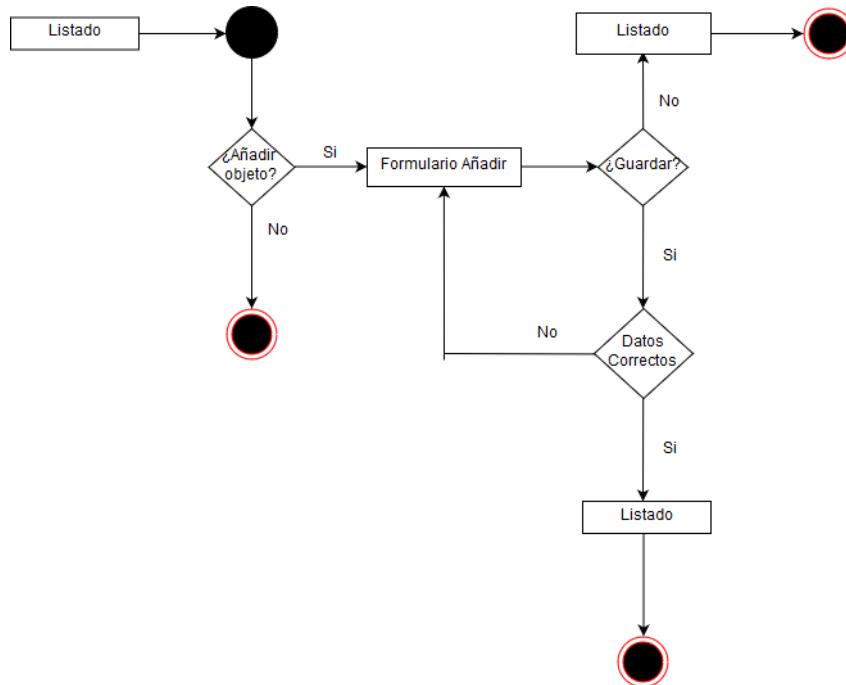
4.2.2.1. Recuperar Contraseña

DA-01 ~ Recuperar Contraseña			
D. Actividad:	Recuperar Contraseña	Código:	DA-01
Implementación:	Sprint 1/2	Aplicación:	Intranet/WebApp
Descripción Corta:	En este diagrama de Actividad se muestra las diferentes elecciones que hay que tomar a la hora de Recuperar nuestra contraseña y los caminos a donde nos llevan cada una de las elecciones.		



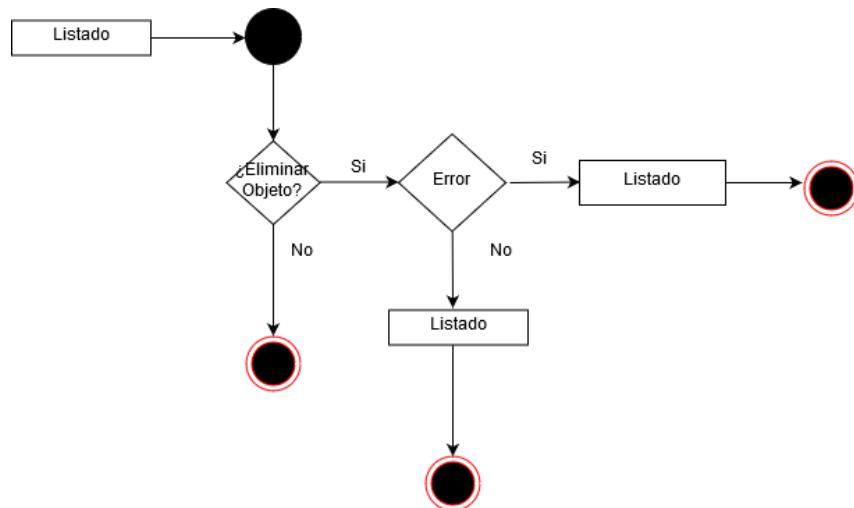
4.2.2.2. Añadir Usuario, Canción, Álbum, Artista

DA-02 ~ Añadir/Editar Usuario, Canción, Álbum, Artista			
D. Actividad:	Añadir/Editar Usuario, Canción, Álbum, Artista	Código:	DA-02
Implementación:	Sprint 1/2	Aplicación:	Intranet/WebApp
Descripción Corta:	En este diagrama de Actividad se muestra las diferentes elecciones que hay que tomar a la hora de Añadir/Editar Usuarios, Canciones, Álbumes y Artistas y los caminos a donde nos llevan cada una de las elecciones.		



4.2.2.3. Eliminar Usuario, Canción, Álbum, Artista

DA-03 ~ Eliminar Usuario, Canción, Álbum, Artista			
D. Actividad:	Eliminar Usuario, Canción, Álbum, Artista	Código:	DA-03
Implementación:	Sprint 1/2	Aplicación:	Intranet/WebApp
Descripción Corta:	En este diagrama de Actividad se muestra las diferentes elecciones que hay que tomar a la hora de Eliminar Usuarios, Canciones, Álbumes y Artistas y los caminos a donde nos llevan cada una de las elecciones.		

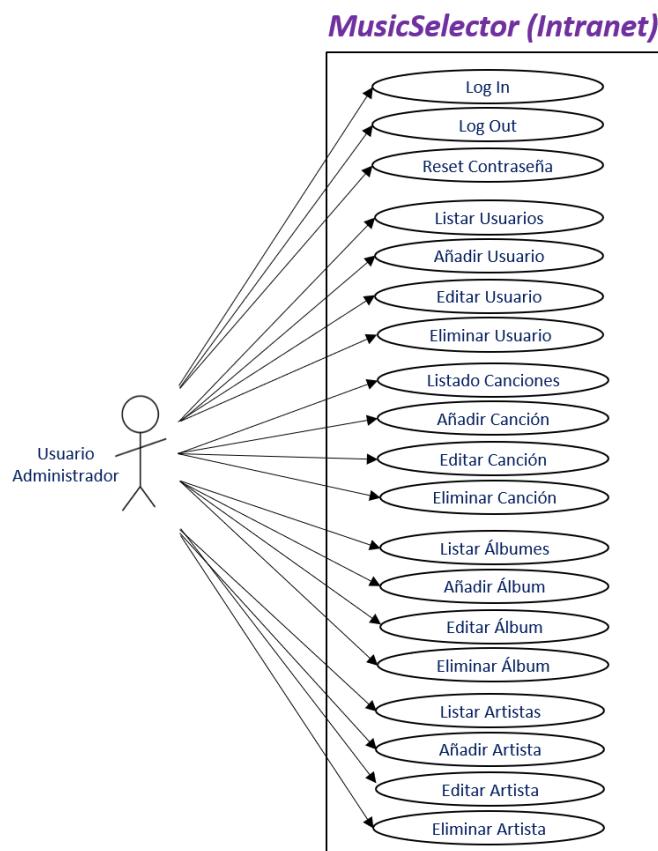


4.2.3. Diagramas de Casos de Uso

4.2.3.1. Intranet

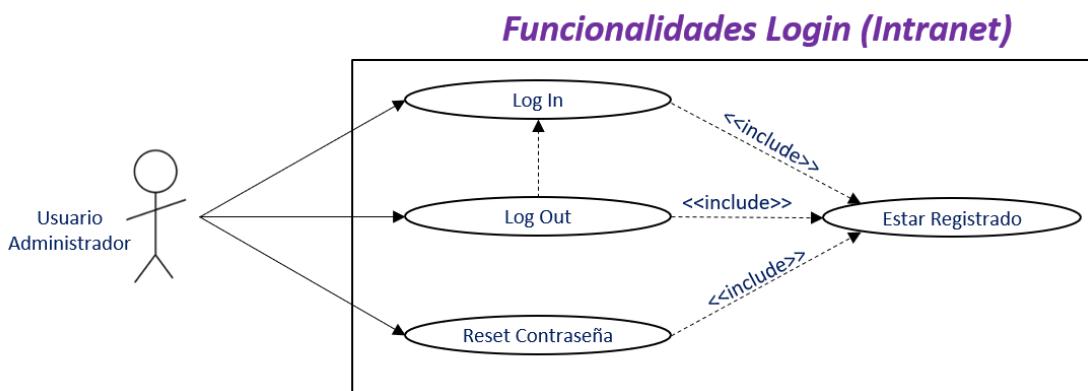
4.2.3.1.1. Diagrama Funcionalidades Intranet

DCU-01 ~ Funcionalidades Intranet			
D. Caso de Uso:	Funcionalidades Intranet	Código:	DCU-01
Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este caso de uso se explican las diferentes acciones que puede ejecutar el Usuario en la aplicación web de la Intranet del Proyecto de MusicSelector.		
Actores:	Usuario Administrador		
Flujo:	<ul style="list-style-type: none"> • Login • Logout • Recuperar Contraseña • Listar Usuarios • Dar de Alta Usuarios • Editar Usuarios • Eliminar Usuarios • Listar Canciones • Añadir Canciones • Editar Canciones • Eliminar Canciones • Listar Álbumes • Añadir Álbumes • Editar Álbumes • Eliminar Álbumes • Listar Artistas • Añadir Artistas • Editar Artistas • Eliminar Artistas 		



4.2.3.1.2. Diagrama Funcionalidades Login (Intranet)

DCU-02 ~ Funcionalidades Login (Intranet)			
D. Caso de Uso:	Funcionalidades Login (Intranet)	Código:	DCU-02
Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este caso de uso se explican las diferentes acciones que puede ejecutar el Usuario en la aplicación web de la Intranet del Proyecto de MusicSelector.		
Actores:	Usuario Administrador		
Procesos:	Login, Logout, Recuperar Contraseña		

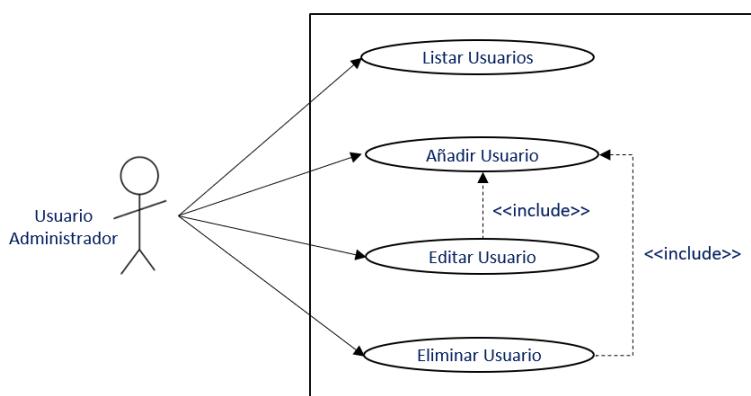


4.2.3.1.3. Diagrama Sección Usuarios

DCU-03 ~ Sección Usuarios

D. Caso de Uso:	Sección Usuarios	Código:	DCU-03
Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este caso de uso se explican las diferentes acciones que puede se pueden realizar desde la Sección de Usuarios de la Intranet del Proyecto de MusicSelector.		
Actores:	Usuario Administrador		
Procesos:	Listar Usuarios, Dar de Alta Usuarios, Editar Usuarios, Eliminar Usuarios		

Sección Usuarios (Intranet)

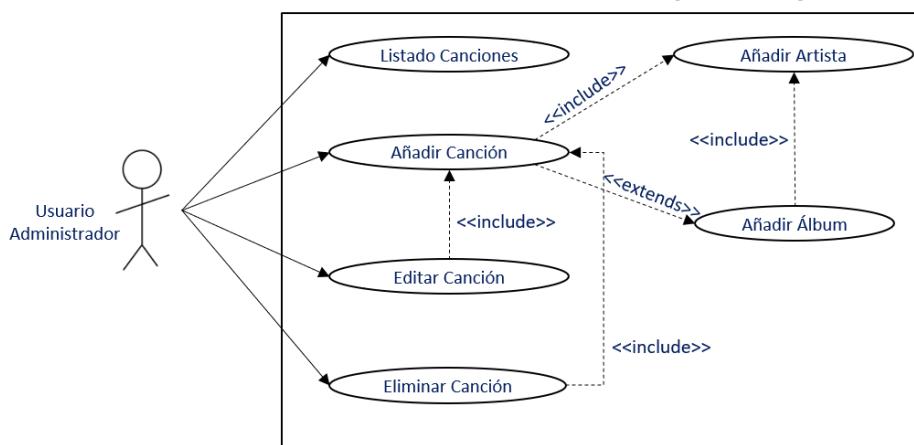


4.2.3.1.4. Diagrama Sección Canciones

DCU-04 ~ Funcionalidades Intranet

D. Caso de Uso:	Sección de Canciones	Código:	DCU-04
Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este caso de uso se explican las diferentes acciones que puede se pueden realizar desde la Sección de Canciones de la Intranet del Proyecto de MusicSelector.		
Actores:	Usuario Administrador		
Procesos:	Listar Canciones, Añadir Canciones, Editar Canciones, Eliminar Canciones.		

Sección Canciones (Intranet)

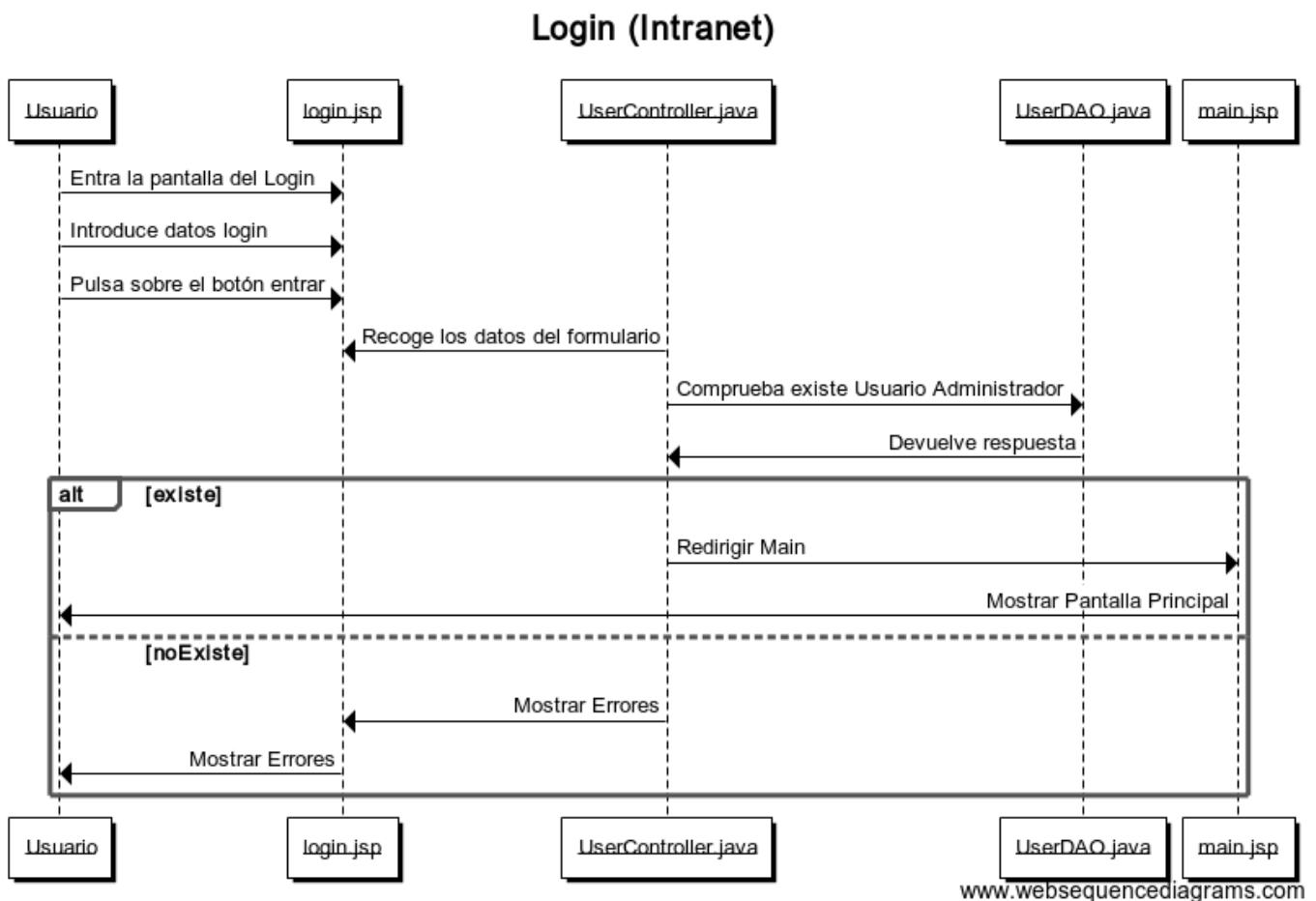


4.2.4. Diagramas de Secuencia

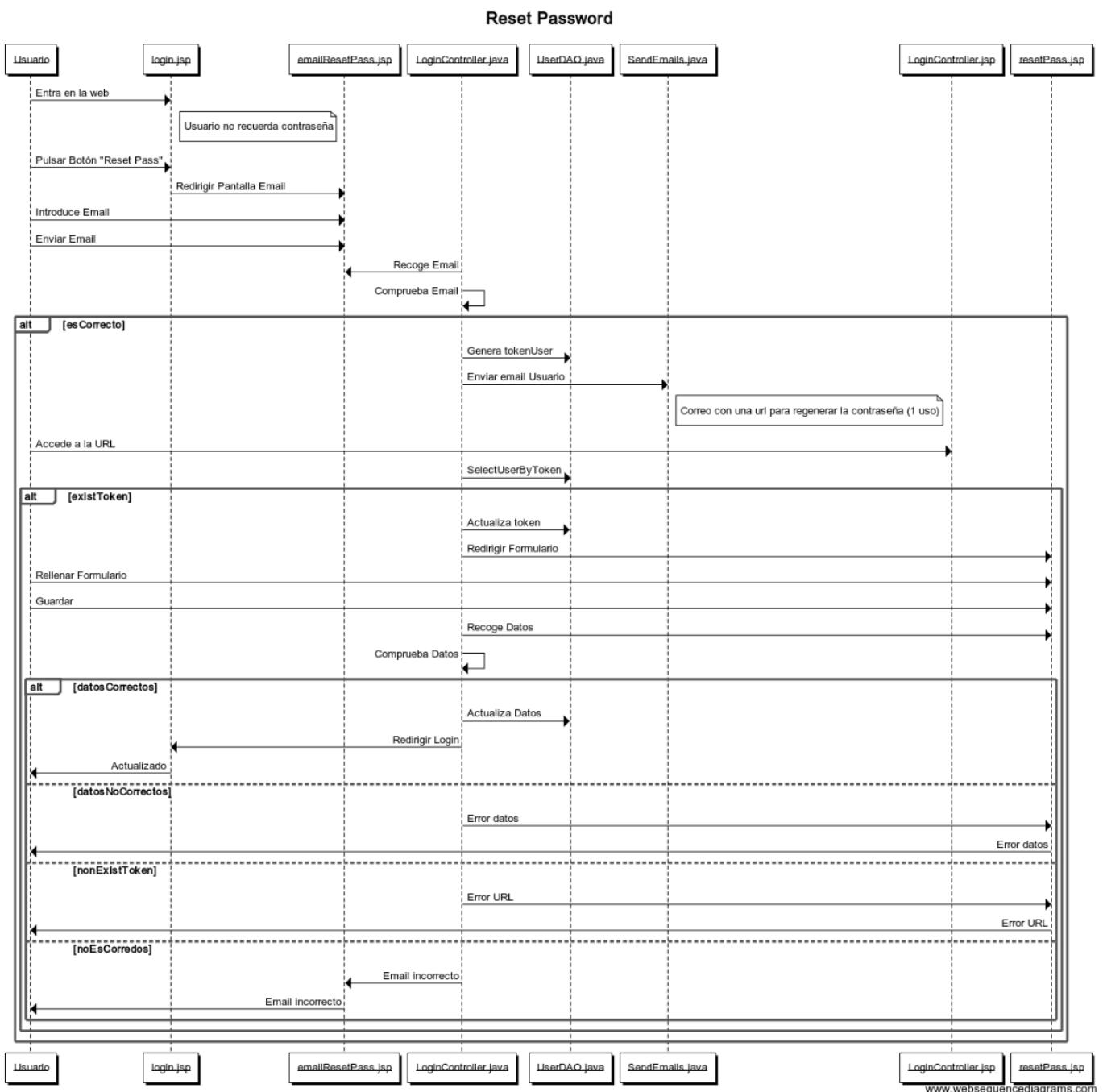
Las imágenes de los diagramas de secuencia se adjuntarán en el apartado “Anexo” al final del Documento.

4.2.4.1. Intranet

4.2.4.1.1. Login (Intranet) (Apartado 4.2.2.1.1.)

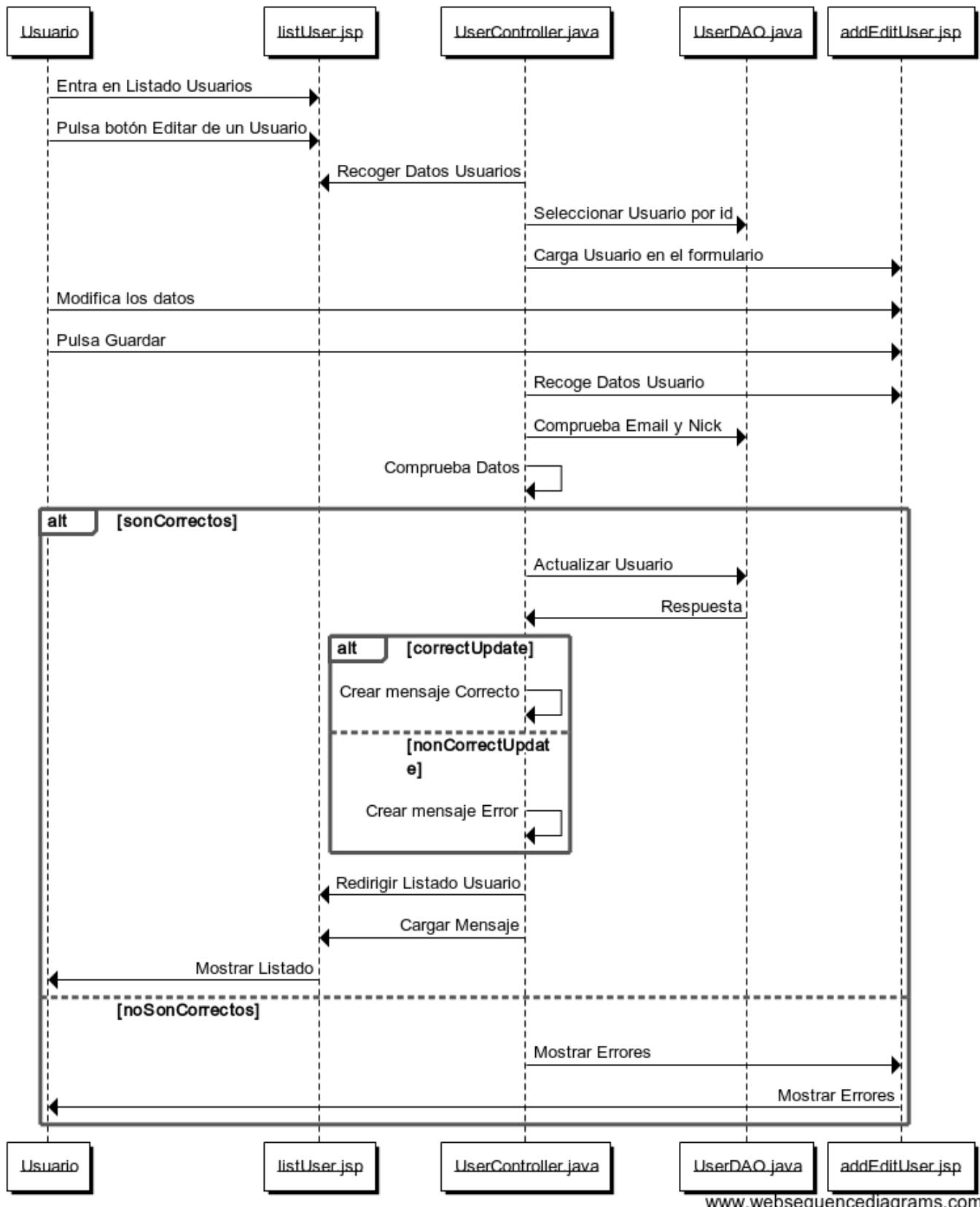


4.2.4.1.2. Recuperar Password (Intranet) (Apartado 4.2.2.1.2.)

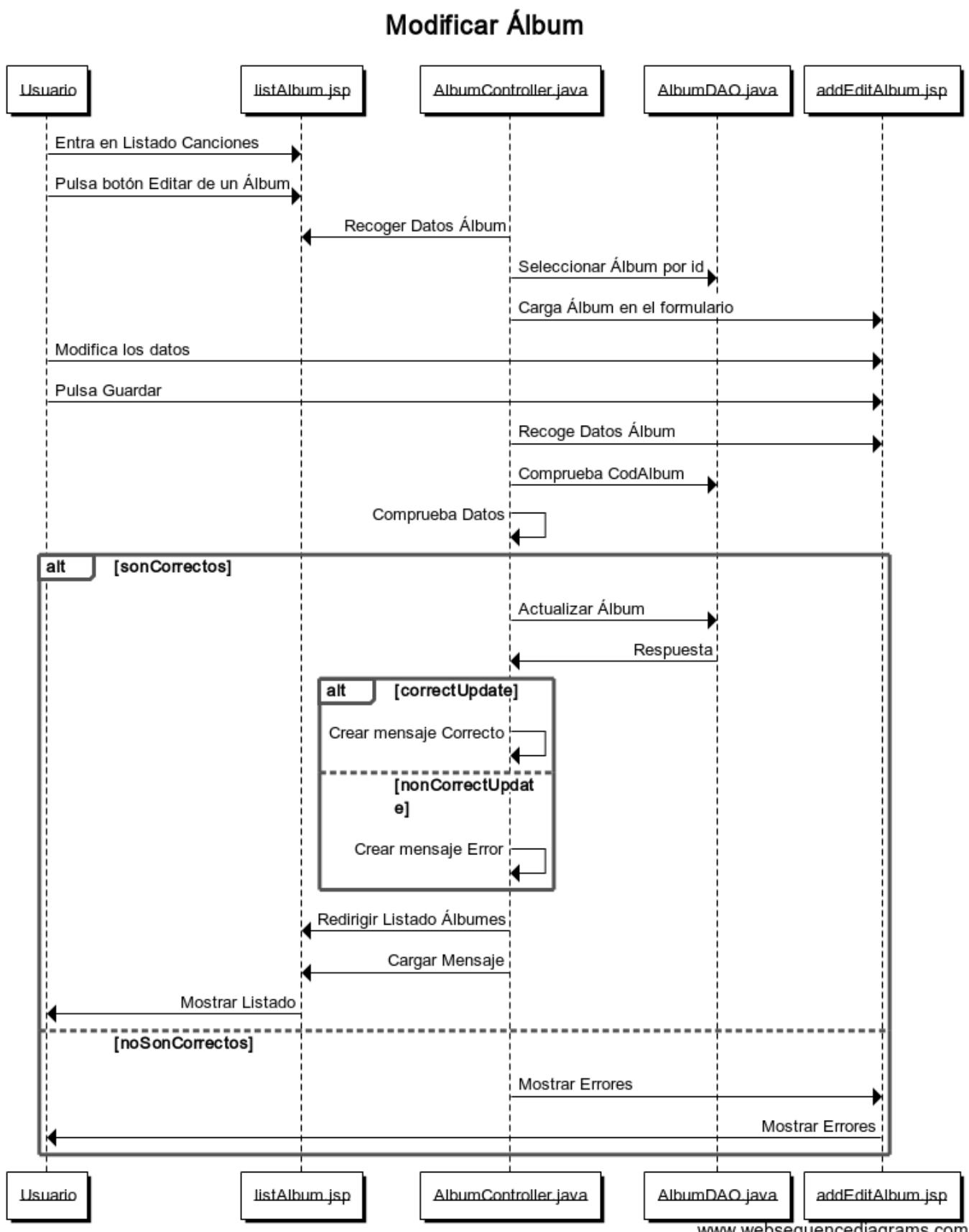


4.2.4.1.4. Modificar Usuario (Intranet) (Apartado 4.2.2.1.4.)

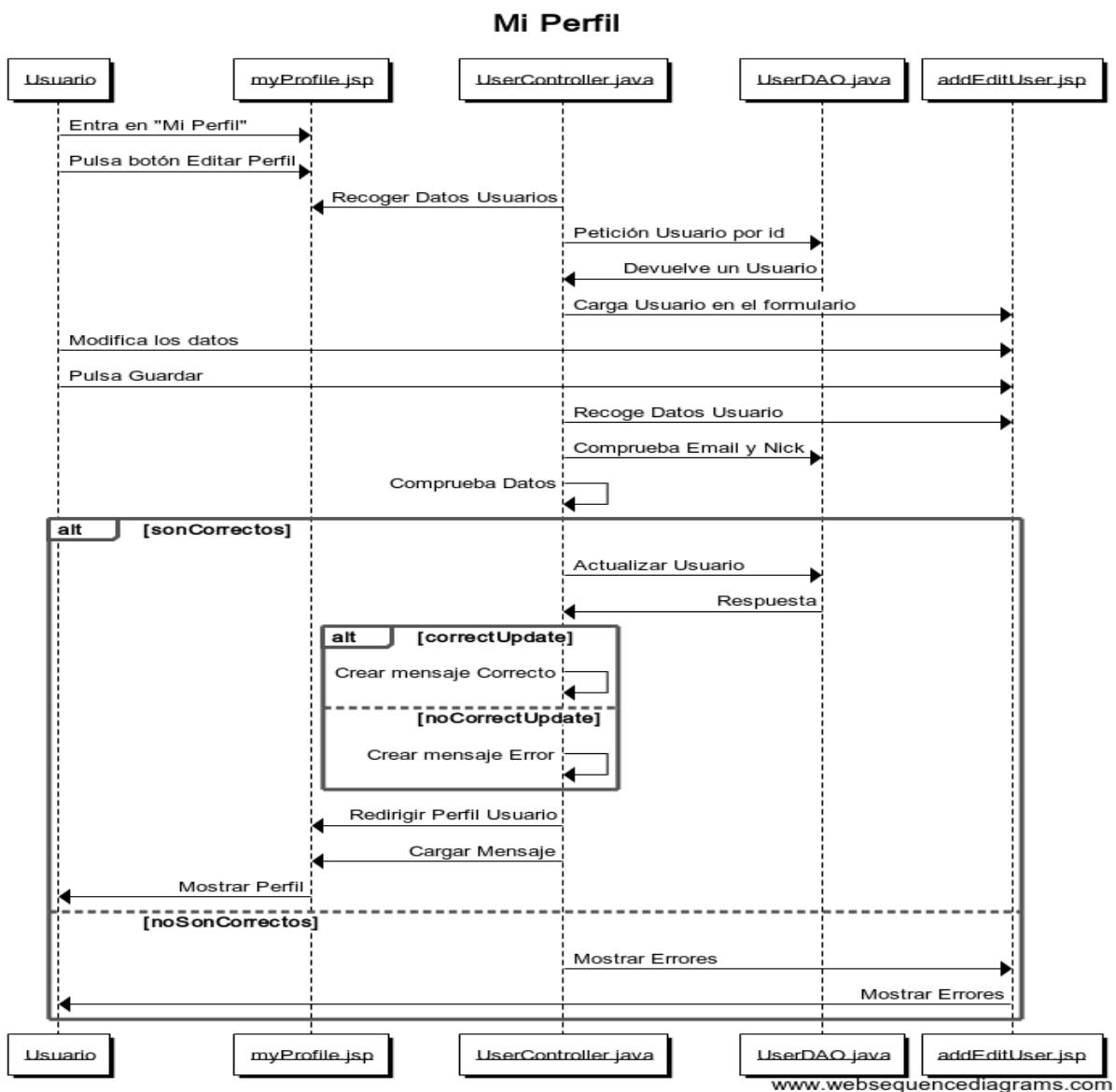
Modificar Usuario



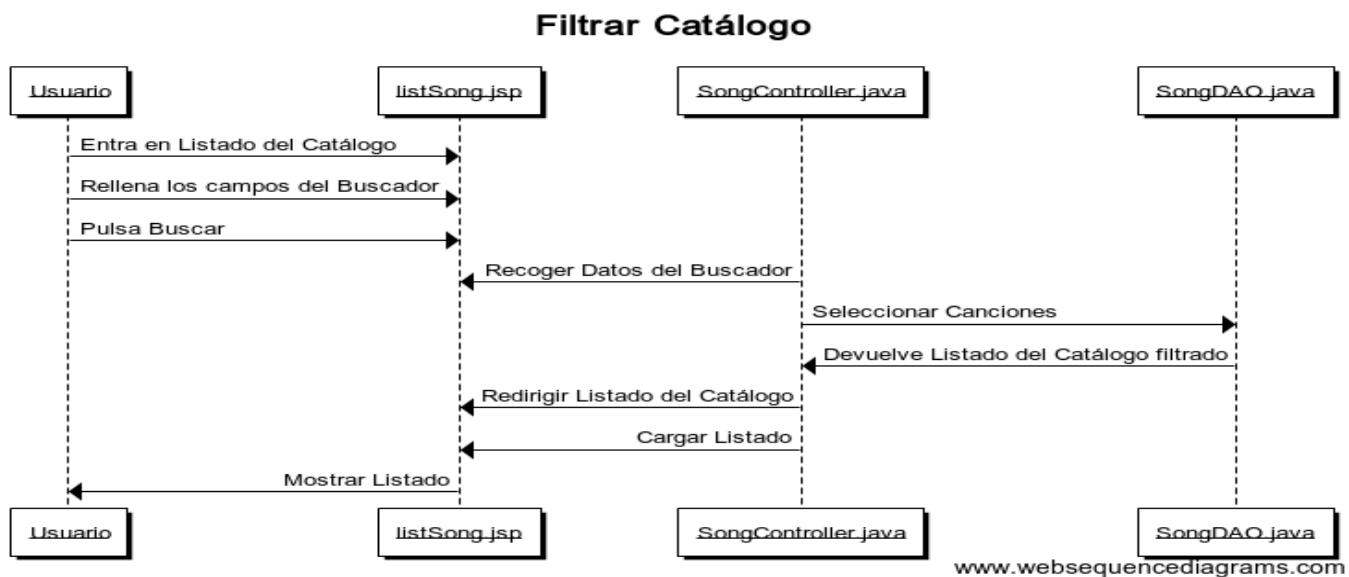
4.2.4.1.12. Modificar Álbum (Intranet) (Apartado 4.2.2.1.12.)



4.2.4.2.4. Mi Perfil (Intranet) (Apartado 4.2.2.2.4.)



4.2.4.2.5. Filtrar Catálogo (Intranet) (Apartado 4.2.2.2.5.)



4.3. Análisis y Diseño de Interfaz de Usuario

4.3.1. Intranet

A Web Page
[A Web Page](#) https://

Intranet Music Selector

Nick/Email:

Password:

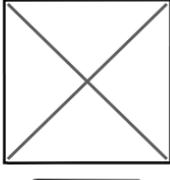


Imagen Álbum

Login

A Web Page
[A Web Page](#) https://

Catálogo Musical

Añadir Canción

ID	Título	Autor	Género	Seleccionar
1	Back in Black	ACDC	Rock	<input type="checkbox"/>
2	Propuesta Indecente	Romeo Santos	Salsa	<input type="checkbox"/>
3	Vuelve	Beret	Pop	<input type="checkbox"/>
4	I'm yours	Jason Mroz	Jazz	<input type="checkbox"/>
5	Vive la vida	ColdPlay	Rock alternativo	<input type="checkbox"/>
6	Carolina	M Clan	Rock en Español	<input type="checkbox"/>

Catálogo Musical

A Web Page
[A Web Page](#) https://

Canción

Título:

Autor:

Álbum:

Género:

Canción:

Vídeo:

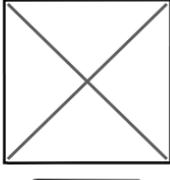


Imagen Álbum

Crear/Editar Canción

A Web Page
[A Web Page](#) https://

Listado de Usuarios

Añadir Usuario

ID	Nick	Nombre	Apellido	Seleccionar
1	DayB	David	Ruiz	<input type="checkbox"/>
2	Adriancamon	Adrián	Ruiz	<input type="checkbox"/>
3	Sky	Cielo	Sia	<input type="checkbox"/>
4	Asuna	Aria	James	<input type="checkbox"/>
5	Kirito	Kazuto	Kirigaya	<input type="checkbox"/>

Listado Usuarios

A Web Page
[A Web Page](#) https://

Usuario

Nick:

Nombre:

Password:

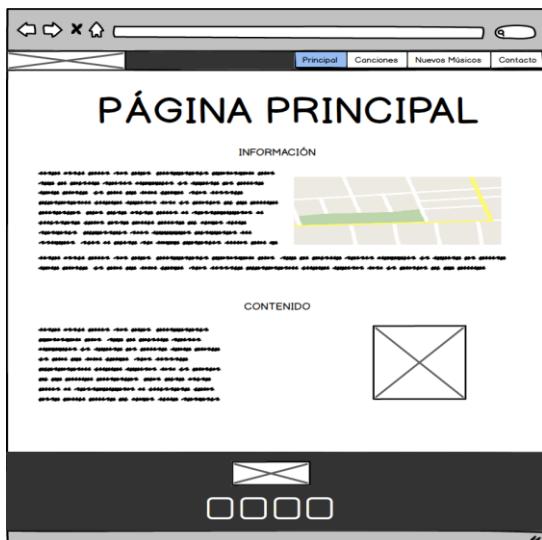
Rep Password:

Email:

Administrador

Crear/Editar Usuario

4.3.2. WebApp



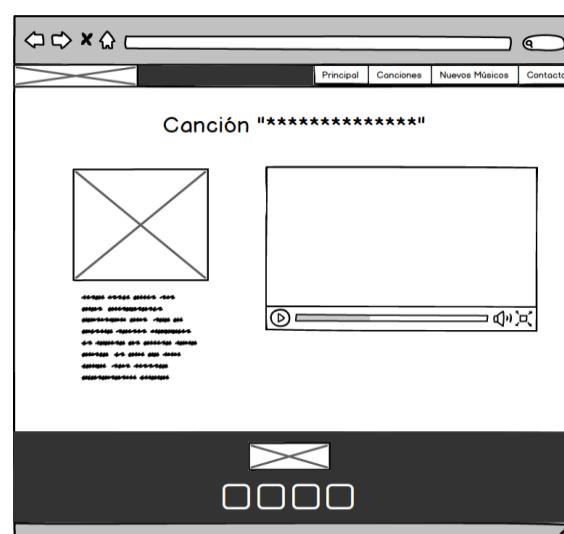
Pantalla de Inicio



Pantalla de Registro



Listado Canciones



Información Canción



Pantalla Contacto

4.4. Diseño de la Arquitectura de la Aplicación

4.4.1. Tecnologías y Herramientas usadas y descripción de las mismas

Los programas usados para la realización del Proyecto MusicSelector son los siguientes:

- **IntelliJ IDEA:** IntelliJ IDEA es un entorno de desarrollo integrado (IDE) destinado principalmente a Java. Este entorno se usa especialmente para el desarrollo de programas. Es desarrollado por una compañía llamada JetBrains que formalmente se llamó IntelliJ. Está disponible en dos ediciones: Community Edition que tiene licencia de Apache 2.0 y una edición comercial conocida como Ultimate Edition. Ambos pueden usarse para crear software que pueda venderse. Lo que hace que IntelliJ IDEA sea tan diferente de sus contrapartes es su facilidad de uso, flexibilidad y su diseño sólido.
- **SQL Developer:** Oracle SQL Developer es la herramienta gráfica gratuita que proporciona Oracle para desarrollar, ejecutar consultas o scripts SQL, tanto DML como DDL, sobre bases de datos Oracle, etc. Además, en la aplicación de MusicSelector se ha decidido usar procedimientos almacenados que permiten, de una forma ordenada, realizar diversas acciones sobre las tablas de la BBDD (Inserts, obtención de Datos, realización de actualizaciones de información sobre registros, etc). El lenguaje utilizado para el desarrollo de los procedimientos es PL/SQL. Otro dato de importancia acerca de la seguridad de la aplicación es que nuestra app programada en java realiza las llamadas a la BBDD a partir de Prepared Statement lo que simplifica el riesgo de que se introduzca un Script malicioso en nuestro código.
- **DataModeler:** Oracle SQL Developer DataModeler es una herramienta gráfica libre que permite realizar las tareas de modelado de datos. Mediante el uso de Oracle SQL Developer DataModeler los usuarios pueden crear de manera gráfica un modelo relacional de una base de datos que deseemos o requiramos por algún motivo en específico, permitiendo así mayor velocidad de comprensión.
- **Spring:** Spring Framework es un marco de aplicación e inversión de control para la plataforma Java. Las características principales del marco pueden ser utilizadas por cualquier aplicación Java, pero existen extensiones para construir aplicaciones web sobre la plataforma Java EE (Enterprise Edition). Aunque el marco no impone ningún modelo de programación específico, se ha popularizado en la comunidad Java como una adición o incluso reemplazo del modelo Enterprise JavaBeans (EJB). Cabe decir que Spring Framework es de código abierto.
- **JSP:** JavaServer Pages (JSP) es una tecnología que ayuda a los desarrolladores de software a crear páginas web dinámicas basadas en HTML y XML, entre otros tipos de documentos. JSP es similar a PHP, pero usa el lenguaje de programación Java. Para desplegar y correr JavaServer Pages, se requiere un servidor web compatible con contenedores servlet como Apache Tomcat o Jetty.
- **SpringBoot:** Según el sitio oficial del proyecto, se puede definir Spring Boot de la siguiente manera: "Es una solución para crear aplicaciones basadas en Spring de una manera rápida, autónoma y con características deseables para producción". Al momento, resulta de mucho interés contar con herramientas que apoyen al desarrollo rápido de aplicaciones. Y ésta es la motivación principal para presentar esta primera entrada sobre el tema. Entre las principales características de Spring Boot se encuentran:
 - Contenedores Java embebidos: Tomcat o Jetty
 - Soporte para la automatización con Maven y Gradle
 - Configuración sugerida para iniciar rápidamente con un proyecto (Starters)
 - Configura automáticamente Spring, cuando sea posible
 - Características listas para producción: métricas, seguridad, verificación del estatus, externalización de configuración, etc.
 - No genera código y no requiere configuración XML

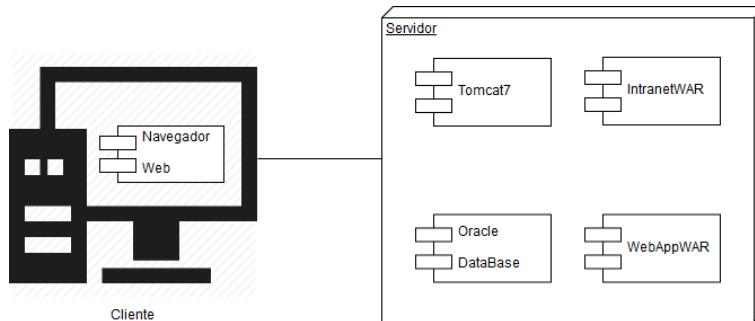
- **Maven:** Maven es una herramienta de software para la gestión y construcción de proyectos Java que tiene un modelo de configuración de construcción basado en un formato XML. Maven utiliza un Project Object Model (POM) para describir el proyecto de software a construir, sus dependencias de otros módulos y componentes externos, y el orden de construcción de los elementos. Viene con objetivos predefinidos para realizar ciertas tareas claramente definidas, como la compilación del código y su empaquetado.



4.4.2. Arquitectura y Componentes de la Aplicación

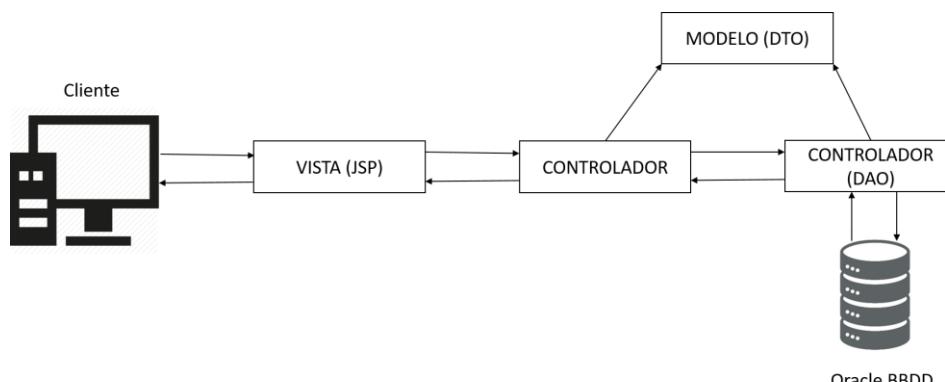
4.4.2.1. Diagrama de Componentes

En servidor elegido para desplegar las aplicaciones intranet.war y webapp.war del proyecto MusicSelector será un Apache Tomcat 8. Una vez desplegadas las aplicaciones estas tendrán que tener conexión con la Base de Datos de Oracle.

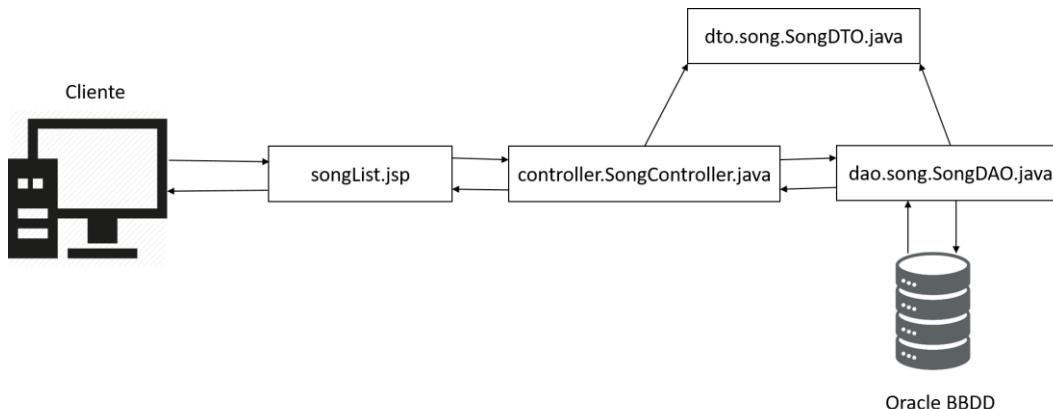


4.4.2.2. Arquitectura de la Aplicación

En la imagen que tenemos a continuación podemos observar la arquitectura de la aplicación. Los usuarios se conectan a través del navegador a la aplicación al acceder donde se les visualizará la interfaz (.jsp). Una vez el usuario realice una acción, esta será recogida en el controlador. Los datos obtenidos por el controlador principal serán parseados como un objeto utilizando los modelos (dto) diseñados. Después de haber generado el objeto pertinente, este será enviado al controlador secundario (dao) donde se realizará la conexión con la base de datos y se realizarán la petición solicitada. Los datos devueltos de la base de datos serán enviados al controlador desde donde se le mostrarán al usuario en la vista.



A continuación, voy a explicar un proceso específico donde se muestre a qué paquete afectaría cada caso.



En el ejemplo anterior, un usuario está buscando canciones en la Sección de Catálogo Musical de la WebApp. El usuario se encuentra en la pantalla del listado de canciones (songList.jsp) donde se le muestra el listado de todas las canciones almacenadas en el catálogo. Después de haber rellenado el formulario de Busqueda el usuario pulsa el botón “Buscar”. El controlador “SongController.java” recoge los datos del formulario y los parsea en objeto de tipo CatalogDTO (“Modelo”). Este objeto se envía a la clase SongDAO.java que se encargará de conectarse a la base de datos donde se ejecutará el procedimiento almacenado que devuelve un cursor con las canciones que cumplan las condiciones establecidas por el usuario. El cursor es devuelto a la clase SongDAO.java donde es parseado como un Listado de CatalogDTO (“Modelo”) que es devuelto al SongController.java. El SongController.java pone los datos en el modelo de la vista (“songList.jsp”) y se la muestra a los datos con los datos cargados.

4.5. Estructura de las Aplicaciones

En la imagen de la derecha se muestra la estructura de la Intranet y de la WebApp del Proyecto MusicSelector (La imagen es solo de la Intranet. Sin embargo, la estructura de las dos aplicaciones del proyecto MusicSelector es la misma). Estas Aplicaciones se encuentran divididas en 4 paquetes principales y una carpeta donde se colocarán las interfaces. A continuación voy a describir brevemente cada uno de los paquetes.

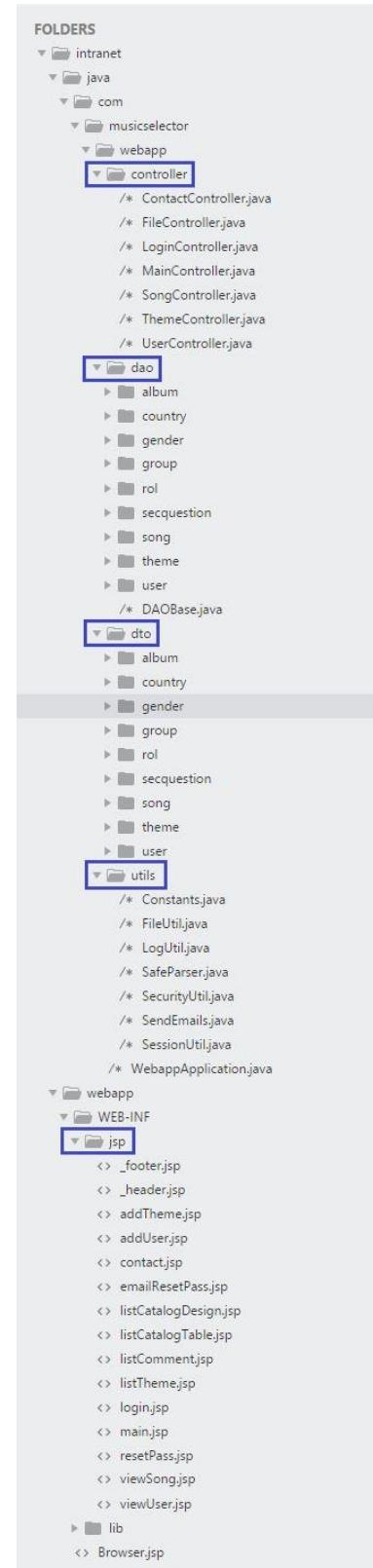
Paquete “controller”: En este paquete se encuentran las clases encargadas de realizar las tareas de las diferentes páginas de la aplicación. Desde el controlador se recogen la información de la vista para así generar nuevos objetos utilizando los modelos establecidos. Además, es el encargado de realizar las funciones correspondientes en cada caso sobre los objetos generados.

Paquete “dto”: En este paquete se encuentran los modelos de los objetos. En cada uno de los modelos se definen los atributos de estos objetos. Además, gracias a la utilización de Spring se han implementado etiquetas sobre estos atributos para permitir una más fácil validación de los datos al ser enviados desde el formulario.

Paquete “dao”: En este paquete se encuentran las clases encargadas de realizar la conexión con la base de datos. Además, se implementan los métodos necesarios para poder realizar las llamadas a los procedimientos almacenados definidos en la base de datos. Todos los métodos encargados de realizar funciones sobre la base de datos utilizan consultas parametrizadas para tener un mayor control y seguridad a la hora de tratar los datos.

Paquete “utils”: En este paquete se encuentran las clases encargadas de realizar funciones genéricas que van a ser utilizadas en varias clases. Por ejemplo: Envío de correos, Guardar variables en sesión, Gestión de Ficheros, etc.

Carpeta “jsp”: En esta carpeta se encuentran los archivos .jsp que hacen referencias las interfaces de la aplicación. Spring permite enviar objetos utilizando un modelo de datos clave valor. Estos objetos son recogidos en la interfaz permitiendo utilizar los datos tanto para controlar la información a visualizar como para mostrar los atributos de dicho objeto.



5. Documento de Implementación e Implantación del Sistema

5.1. Pruebas

Para poder probar las diferentes secciones de las aplicaciones se ha diseñado un documento de “Planificación y registro de casos de prueba” donde se han descrito las diferentes pruebas a realizar en la aplicación para comprobar su correcto funcionamiento. Además, se ha realizado una Tabla de Datos para cada una de uno de los formularios de las aplicaciones mostrando los distintos resultados de un mismo proceso según los datos que se introduzcan.

5.1.1. Documento de “Planificación y registro de casos de prueba”

CASOS DE PRUEBAS					
ID	App	Apartado	Descripción	Precondiciones	Resultados Esperados
1	Intranet	Login	Un usuario con rol Admin se intenta logear	Registrar Usuario con rol Admin	Entrar en la App
2	Intranet	Login	Un usuario con rol Follower se intenta logear	Registrar Usuario con rol Follower	Mostrar Error y no dejar entrar
3	Intranet	Login	Pulsar "Recuperar Contraseña" e introducir un correo de un Usuario registrado	Registrar Usuario	Envío de Email al Correo
4	Intranet	Login	Pulsar "Recuperar Contraseña" e introducir un correo de un Usuario NO registrado		Mostrar Error
5	Intranet	Login	Acceder a la Url que te han enviado al correo	Caso de Prueba 3	Acceso Pantalla Reset Pass
6	Intranet	Login	Acceder a la Url que te han enviado al correo por 2º vez	Caso de Prueba 5	Mostrar Error (url no válidad).
7	Intranet	Main	Logout	Caso de Prueba 1	Desconectar de la Intranet
8	Intranet		Acceder "Pantalla Principal" desde el menú superior	Caso de Prueba 1	Acceder a la Pantalla Principal
9	Intranet		Acceder "Sección Canciones" desde el menú superior	Caso de Prueba 1	Acceder a la Sección de Canciones
10	Intranet		Acceder "Sección Álbumes" desde el menú superior	Caso de Prueba 1	Acceder a la Sección de Álbumes
11	Intranet		Acceder "Sección Artistas" desde el menú superior	Caso de Prueba 1	Acceder a la Sección de Artistas
12	Intranet		Acceder "Sección Usuario" desde el menú superior	Caso de Prueba 1	Acceder a la Sección de Usuarios
13	Intranet	Main	Pulsar sobre el la imagen de "Sección de Canciones"	Caso de Prueba 1	Acceder a la Sección de Canciones
14	Intranet	Main	Pulsar sobre el la imagen de "Sección de Álbumes"	Caso de Prueba 1	Acceder a la Sección de Álbumes
15	Intranet	Main	Pulsar sobre el la imagen de "Sección de Artistas"	Caso de Prueba 1	Acceder a la Sección de Artistas
16	Intranet	Main	Pulsar sobre el la imagen de "Sección de Usuarios"	Caso de Prueba 1	Acceder a la Sección de Usuarios
17	Intranet	Sección Canciones	Realizar Búsqueda con todos los campos		Filtro Correcto del Catálogo
18	Intranet	Sección Canciones	Pulsar botón limpiar		Resetéa los campos del Formulario
19	Intranet	Sección Canciones	Pulsar botón Añadir Canción		Se abre pantalla formulario Canción
20	Intranet	Sección Canciones	Pulsar botón Editar Canción	Añadir una Canción	Se abre pantalla formulario Canción con los datos de la canción Cargados
21	Intranet	Sección Canciones	Pulsar botón Eliminar Canción		Elimina la canción
22	Intranet	Sección Canciones	Pulsar botón Exportar excel	Añadir una Canción	Se exporta correctamente
23	Intranet	Sección Canciones	Pulsar botón Exportar csv	Añadir una Canción	Se exporta correctamente
24	Intranet	Sección Canciones	Pulsar botón Exportar txt	Añadir una Canción	Se exporta correctamente
25	Intranet	Formulario Canción	Pulsamos sobre Añadir Canción en el Listado y rellenamos todos los datos del formulario y pulsamos guardar		Se guarda correctamente
26	Intranet	Formulario Canción	Pulsamos sobre Añadir Canción en el Listado y rellenamos algunos datos del formulario y pulsamos guardar (dejar título, fecha...)		Muestra error en el Formulario
27	Intranet	Formulario Canción	Pulsamos sobre Añadir Canción en el Listado y rellenamos formulario con datos de una canción ya existente y pulsamos guardar (título y grupo)		Muestra error en el Formulario
28	Intranet	Formulario Canción	Pulsamos sobre Añadir Canción en el Listado y subir un audio en .mp3		Se guarda y reproduce correctamente
29	Intranet	Formulario Canción	Pulsamos sobre Añadir Canción en el Listado y subir un video en .mp4		Se guarda y reproduce correctamente
30	Intranet	Formulario Canción	Pulsamos sobre Añadir Canción en el Listado y pulsamos volver		Regresamos al Listado de Canciones
31	Intranet	Sección Álbumes	Realizar Búsqueda con todos los campos		Filtro Correcto del Catálogo
32	Intranet	Sección Álbumes	Pulsar botón limpiar		Resetéa los campos del Formulario
33	Intranet	Sección Álbumes	Pulsar botón Añadir Álbum		Se abre pantalla formulario Álbum
34	Intranet	Sección Álbumes	Pulsar botón Editar Álbum	Añadir una Álbum	Se abre pantalla formulario Álbumes con los datos del Álbum Cargados
35	Intranet	Sección Álbumes	Pulsar botón Eliminar Álbum		Elimina el Álbum
36	Intranet	Sección Álbumes	Pulsar botón Exportar excel	Añadir una Álbum	Se exporta correctamente
37	Intranet	Sección Álbumes	Pulsar botón Exportar csv	Añadir una Álbum	Se exporta correctamente
38	Intranet	Sección Álbumes	Pulsar botón Exportar txt	Añadir una Álbum	Se exporta correctamente

39	Intranet	Formulario Álbumes	Pulsamos sobre Añadir Álbum en el Listado y rellenamos todos los datos del formulario y pulsamos guardar		Se guarda correctamente
40	Intranet	Formulario Álbumes	Pulsamos sobre Añadir Álbum en el Listado y rellenamos algunos datos del formulario y pulsamos guardar (dejar nombre, fecha...)		Muestra error en el Formulario
41	Intranet	Formulario Álbumes	Pulsamos sobre Añadir Álbum en el Listado y rellenamos formulario con datos de un Álbum ya existente y pulsamos guardar (Nombre y grupo)		Muestra error en el Formulario
42	Intranet	Formulario Álbumes	Pulsamos sobre Añadir Álbum en el Listado y subir una cover con dimensiones (500 * 500)		Se guarda correctamente
43	Intranet	Formulario Álbumes	Pulsamos sobre Añadir Álbum en el Listado y subir una imagen de unas dimensiones distintas a las recomendadas		La imagen se deforma
44	Intranet	Formulario Álbumes	Pulsamos sobre Añadir Álbum en el Listado y pulsamos volver		Regresamos al Listado de Álbumes
45	Intranet	Sección Artistas	Realizar Búsqueda con todos los campos		Filtro Correcto del Catálogo
46	Intranet	Sección Artistas	Pulsar botón limpiar		Resetéa los campos del Formulario
47	Intranet	Sección Artistas	Pulsar botón Añadir Artista		Se abre pantalla formulario Artista
48	Intranet	Sección Artistas	Pulsar botón Editar Artista	Añadir una Artista	Se abre pantalla formulario Artista con los datos del Artista Cargados
49	Intranet	Sección Artistas	Pulsar botón Eliminar Artista		Elimina la Artista
50	Intranet	Sección Artistas	Pulsar botón Exportar excel	Añadir una Artista	Se exporta correctamente
51	Intranet	Sección Artistas	Pulsar botón Exportar csv	Añadir una Artista	Se exporta correctamente
52	Intranet	Sección Artistas	Pulsar botón Exportar txt	Añadir una Artista	Se exporta correctamente
53	Intranet	Formulario Artistas	Pulsamos sobre Añadir Artista en el Listado y rellenamos todos los datos del formulario y pulsamos guardar		Se guarda correctamente
54	Intranet	Formulario Artistas	Pulsamos sobre Añadir Artista en el Listado y rellenamos algunos datos del formulario y pulsamos guardar (dejar nombre, fecha...)		Muestra error en el Formulario
55	Intranet	Formulario Artistas	Pulsamos sobre Añadir Artista en el Listado y rellenamos formulario con datos de un Artista ya existente y pulsamos guardar (Nombre)		Muestra error en el Formulario
56	Intranet	Formulario Artistas	Pulsamos sobre Añadir Artista en el Listado y pulsamos volver		Regresamos al Listado de Artistas
57	Intranet	Sección Usuarios	Realizar Búsqueda con todos los campos		Filtro Correcto del Catálogo
58	Intranet	Sección Usuarios	Pulsar botón limpiar		Resetéa los campos del Formulario
59	Intranet	Sección Usuarios	Pulsar botón Añadir Usuario		Se abre pantalla formulario Usuario
60	Intranet	Sección Usuarios	Pulsar botón Editar Usuario	Añadir una Usuario	Se abre pantalla formulario Usuario con los datos del Usuario Cargados
61	Intranet	Sección Usuarios	Pulsar botón Eliminar Usuario		Elimina el Usuario
62	Intranet	Sección Usuarios	Pulsar botón Exportar excel	Añadir una Usuario	Se exporta correctamente
63	Intranet	Sección Usuarios	Pulsar botón Exportar csv	Añadir una Usuario	Se exporta correctamente
64	Intranet	Sección Usuarios	Pulsar botón Exportar txt	Añadir una Usuario	Se exporta correctamente
65	Intranet	Formulario Usuarios	Pulsamos sobre Añadir Usuario en el Listado y rellenamos todos los datos del formulario y pulsamos guardar		Se guarda correctamente
66	Intranet	Formulario Usuarios	Pulsamos sobre Añadir Usuario en el Listado y rellenamos algunos datos del formulario y pulsamos guardar (dejar título, fecha...)		Muestra error en el Formulario
67	Intranet	Formulario Usuarios	Pulsamos sobre Añadir Usuario en el Listado y rellenamos formulario con datos de una Usuario ya existente y pulsamos guardar (Nick y Email)		Muestra error en el Formulario
68	Intranet	Formulario Usuarios	Pulsamos sobre Añadir Usuario en el Listado e introducir valores incorrectos (Fecha con letras)		Muestra error en el Formulario
69	Intranet	Formulario Usuarios	Pulsamos sobre Añadir Usuario en el Listado y subir una imagen con dimensiones (500 * 500)		Se guarda correctamente

70	Intranet	Formulario Usuarios	Pulsamos sobre Añadir Usuario en el Listado y subir una imagen de unas dimensiones distintas a las recomendadas		La imagen se deforma
71	Intranet	Formulario Usuarios	Pulsamos sobre Añadir Usuario en el Listado y pulsamos volver.		Regresamos al Listado de Usuarios
72	WebApp	Login	Un usuario con rol Admin se intenta logear	Registrar Usuario con rol Admin	Entrar en la App
73	WebApp	Login	Un usuario con rol Follower se intenta logear	Registrar Usuario con rol Follower	Entrar en la App
74	WebApp	Login	Intentar hacer Login sin haber dado de Alta al Usuario		Mostrar mensaje de Error
75	WebApp	Login	Pulsar "Recuperar Contraseña" e introducir un correo de un Usuario registrado	Registrar Usuario	Envío de Email al Correo
76	WebApp	Login	Pulsar "Recuperar Contraseña" e introducir un correo de un Usuario NO registrado		Mostrar Error
77	WebApp	Login	Acceder a la Url que te han enviado al correo		Acceso Pantalla Reset Pass
78	WebApp	Login	Acceder a la Url que te han enviado al correo por 2º vez		Mostrar Error (url no válidad).
79	WebApp	Login	Pulsar botón Dar de Alta		Se abre pantalla formulario Usuario
80	WebApp	Login	Pulsamos sobre Dar de Alta y rellenamos todos los datos del formulario y pulsamos guardar		Se guarda correctamente
81	WebApp	Login	Pulsamos sobre Dar de Alta y rellenamos algunos datos del formulario y pulsamos guardar (dejar nick, edad...)		Muestra error en el Formulario
82	WebApp	Login	Pulsamos sobre Dar de Alta y rellenamos formulario con datos de un Usuario ya existente y pulsamos guardar (Nick y Email)		Muestra error en el Formulario
83	WebApp	Login	Pulsamos sobre Dar de Alta e introducir valores incorrectos (Fecha con letras)		Muestra error en el Formulario
84	WebApp	Login	Pulsamos sobre Dar de Alta y subir una imagen con dimensiones (500 * 500)		Se guarda correctamente
85	WebApp	Login	Pulsamos sobre Dar de Alta y subir una imagen de unas dimensiones distintas a las recomendadas		La imagen se deforma
86	WebApp	Login	Pulsamos sobre Dar de Alta en el Listado y pulsamos volver.		Regresamos a la Pantalla Principal
87	WebApp		Acceder "Pantalla Principal" desde el menú superior		Acceder a la Pantalla Principal
88	WebApp		Acceder "Sección Catálogo" desde el menú superior		Acceder a la Sección de Catálogo
89	WebApp		Acceder "Sección Músicos Nóveles" desde el menú superior		Acceder a la Sección de Músicos Nóveles
90	WebApp		Acceder "Sección Foro" desde el menú superior		Acceder a la Sección de Foro
91	WebApp		Acceder "Sección Mi Perfil" desde el menú superior		Acceder a la Sección de Mi Perfil
92	WebApp		Acceder "Sección Mis Videos" desde el menú superior		Acceder a la Sección de Mis Videos
93	WebApp		Acceder "Sección Contacto" desde el menú superior		Acceder a la Sección de Contacto
94	WebApp	Pantall Principal	Pulsar Botón Acceso Catálogo		Acceder a la Sección de Catálogo
95	WebApp	Pantall Principal	Pulsar Botón Acceso Músicos Nóveles		Acceder a la Sección de Músicos Nóveles
96	WebApp	Sección Catálogo	Utilizar el filtro de la Sección Catálogo		Se filtra correctamente
97	WebApp	Sección Catálogo	Utilizar el Modo Reproducción en el filtro de la Sección Catálogo		Cambia al Modo Reproductor
98	WebApp	Sección Catálogo	Sobre el botón Información de una canción		Se accede a la pantalla Información de una canción
99	WebApp	Sección Catálogo	Se reproducen las canciones en el Modo Reproductor		Se reproduce la canción correctamente
100	WebApp	Sección Músicos Nóveles	Utilizar Buscador Músicos Jóvenes		Se realiza la búsqueda correctamente
101	WebApp	Sección Músicos Nóveles	Probar a visualizar Videos Musicales		Los videos se reproducen correctamente
102	WebApp	Sección Mi Perfil	Acceder a la Sección Mi Perfil	Caso de Prueba 72/73	Se cargan los datos del Usuario Correctamente
103	WebApp	Sección Mi Perfil	Pulsar el botón Editar en la Sección Mi Perfil	Caso de Prueba 103	Se abre pantalla formulario Usuario cargando los datos del Usuario Loggeado

104	WebApp	Sección Mi Perfil	Pulsamos sobre Editar perfil y rellenamos todos los datos del formulario y pulsamos guardar		Se guarda correctamente
105	WebApp	Sección Mi Perfil	Pulsamos sobre Editar perfil y modicamos datos del formulario dejando algunos campos vacíos y pulsamos guardar (dejar nick, edad...)		Muestra error en el Formulario
106	WebApp	Sección Mi Perfil	Pulsamos sobre Editar perfil y rellenamos formulario con datos de un usuario ya existente y pulsamos guardar (Nick y Email)		Muestra error en el Formulario
107	WebApp	Sección Mi Perfil	Pulsamos sobre Editar perfil e introducir valores incorrectos (Fecha con letras)		Muestra error en el Formulario
108	WebApp	Sección Mi Perfil	Pulsamos sobre Editar perfil y subir una imagen con dimensiones (500 * 500)		Se guarda correctamente
109	WebApp	Sección Mi Perfil	Pulsamos sobre Editar perfil y subir una imagen de unas dimensiones distintas a las recomendadas		La imagen se deforma
110	WebApp	Sección Mi Perfil	Pulsamos sobre Editar perfil y pulsamos volver.		Nos envía a la Pantalla Principal
111	WebApp	Sección Foro	Accedemos a la sección Foro	Caso de Prueba 72/73	Nos lleva al listado de Temas
112	WebApp	Sección Foro	Pulsamos sobre el botón Añadir Tema		Nos redirige al Fomulario del Tema
113	WebApp	Sección Foro	Rellenamos datos del tema y pulsamos Publicar		Se guarda correctamente
114	WebApp	Sección Foro	No rellenamos los datos del Tema y Pulsamos guardar		Se muestra un Mensaje de Error
115	WebApp	Sección Foro	Rellenamos formulario búsqueda temas		Nos filtra correctamente
116	WebApp	Sección Foro	Damos a eliminar Tema	Solo Usuario Administrador	Se elimina correctamente
117	WebApp	Sección Foro	Accedemos al listado de comentarios del Tema		Se muestran los comentarios publicados sobre ese tema
118	WebApp	Sección Foro	Rellenamos un comentario y le damos a Publicar		Se publica el comentario
119	WebApp	Sección Foro	No rellenamos el comentario y le damos a publicar		Se muestra mensaje de error
120	WebApp	Sección Foro	Le damos a Responder y le damos a Publicar		Se elimina correctamente
121	WebApp	Sección Foro	Damos a eliminar Comentario	Solo Usuario Administrador	Se elimina
122	WebApp	Sección Contacto	Rellenar Mensaje de Contacto y Asunto (Sin estar Loggeado)		Se envía comentario
123	WebApp	Sección Contacto	Rellenar Mensaje de Contacto y Asunto (Estando Loggeado)		Muestra error en el Formulario de Contacto

5.1.2. Pruebas Funcionales

Para demostrar que el sistema cumple con las especificaciones indicados anteriormente (Apartado 3.1) se han realizado pruebas funcionales de las aplicaciones. Para mostrar estas pruebas funcionales se ha utilizado el siguiente diseño de tabla donde se presenta la información necesaria para la comprensión de las pruebas.

Requisito Funcional "Nombre"	
Precondición	"Precondición"
Datos de Entrada	"Datos de Entrada"
Datos de Salida	<ol style="list-style-type: none"> 1. Caso 1. OK 2. Caso 2. OK 3. Caso 3. OK 4. Caso 4. OK 5. Caso 5. OK

Las pruebas específicas realizadas para comprobar el funcionamiento de las aplicaciones pueden verse en el Anexo 8.

5.2. Instalación y Configuración

Se adjuntarán los archivos .war de la Intranet y de la WebApp del Proyecto de MusicSelector. Estos archivos tendrán que ser desplegados en el entorno de producción junto a los Scripts de Base de Datos. Todos los archivos se enviarán al cliente en un CD junto al Manual de Instalación y al Manual de Usuario de las Aplicaciones.

5.3. Manual de Usuario

En este capítulo de la documentación voy a representar una simulación de recorrido por las aplicaciones con el fin de mostrar las distintas pantallas y características de las mismas. Durante la descripción de las pantallas iré informando acerca de las funciones que se pueden realizar en cada una de estas para después explicarlas de forma más específica. En primer lugar, comenzaré hablando de la Intranet y posteriormente hablaré de la WebApp.

En las imágenes que se irán adjuntando en los Manuales de Usuario se van a ir marcando las funcionalidades de cada una de las pantallas con un rectángulo verde. El resto de mensajes, valores y acciones serán marcados con un rectángulo azul.

5.3.1. Intranet

5.3.1.1. Login

La primera pantalla que nos encontramos en la intranet es la Pantalla de Login (Imagen 1). En esta pantalla los Usuarios tendrán accesos a dos funcionalidades. Estas funcionalidades son el “Login”, que da acceso al interior de la Intranet, y “Recuperar Contraseña”, que permite a un Usuario registrado restablecer su contraseña si la ha olvidado.

- **Login:** Para poder loggearnos en la aplicación deberemos introducir el Nick/Email y la Contraseña en los campos del formulario de la Pantalla de Login y pulsar el botón “Login”. Si los datos introducidos no se corresponden con los de Usuario registrado que tenga rol “Administrador” entonces se mostrará un mensaje de error. Sin embargo, si los datos introducidos corresponden con los de un Usuario Administrador registrado entonces se permite el acceso a la Intranet.
- **Recuperar Contraseña:** Si hemos olvidado nuestra contraseña podremos recuperarla pulsando el botón “recuperar contraseña” de la Pantalla de Login. Una vez pulsado el botón la aplicación nos redirigirá a una pantalla donde tendremos que introducir el email con el que nos dimos de Alta en la aplicación y pulsar enviar. Si el email introducido no existe o es incorrecto nos mostrará un mensaje de error. Si el email introducido es correcto y pertenece a un Usuario registrado en la aplicación se nos enviará un correo al email introducido con una Url para poder restablecer la contraseña.

La URL que se nos enviará al correo es de un solo uso. Si intentamos acceder más de una vez a la misma URL nos redirigirá al Login de la Intranet indicándonos que esa dirección ya ha caducado.

Una vez que nos haya llegado el correo con la URL para recuperar la contraseña al email accedemos a ella. Esta URL nos cargará información sobre nuestra cuenta (Nick, Nombre, Apellidos...) y nos permitirá introducir una nueva contraseña. Al pulsar el botón “restablecer contraseña” la nueva

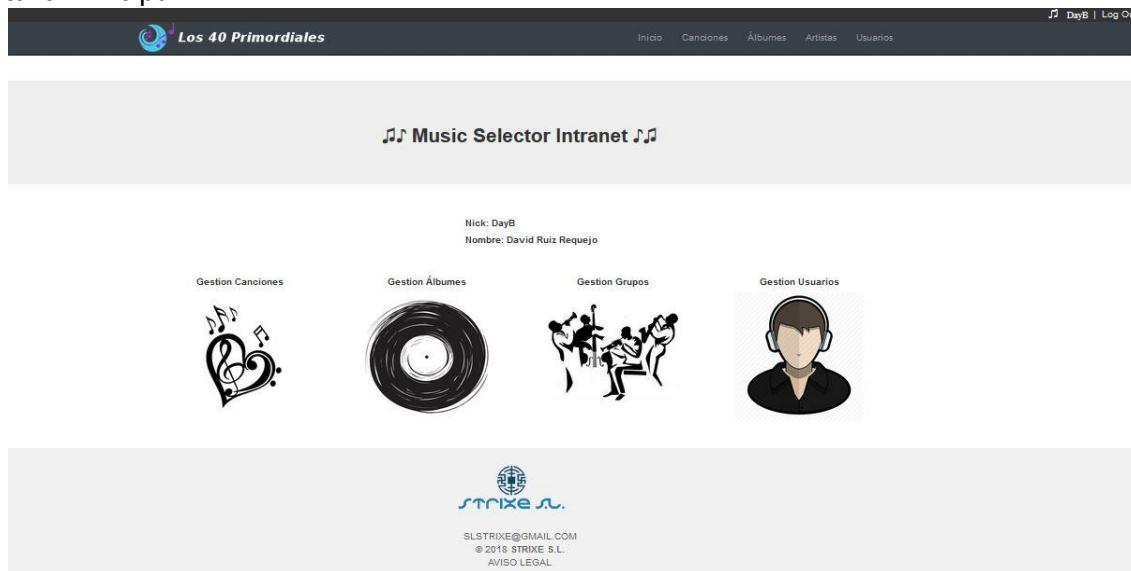
contraseña será validada. Si la contraseña introducida no cumple las pautas definidas se mostrará un mensaje de error en el formulario. Si la contraseña es correcta será actualizada y seremos redirigidos a la Pantalla de Login.

5.3.1.2. Página Principal

Una vez nos hayamos Loggeado en la aplicación accederemos a la Pantalla Principal de la Intranet. En esta pantalla vemos por primera vez la estructura de las páginas de la aplicación. Antes de continuar con la descripción de las funcionalidades que podemos realizar en la Página Principal vamos a hacer una breve descripción de la estructura de las páginas y de cada una de sus partes. Las páginas se encuentran divididas en 3 partes principales.

- **Cabecera:** En la cabecera podemos ver un menú superior que nos permite acceder a las diferentes secciones de la aplicación (Canciones, Álbumes, Artistas y Usuarios). Además, nos permite realizar el Logout de la aplicación.
- **Contenido:** Es el apartado central de las páginas donde se encuentra el contenido.
- **Pie de Página:** Se encuentra en la parte inferior de las páginas de la Web y contiene información sobre la empresa fabricante del producto.

Tras haber descrito la estructura de las páginas de la Intranet vamos a continuar con la explicación de la Pantalla Principal.



En la página principal nos aparecerán algunos datos de nuestra cuenta (Nick, Nombre y Apellidos). Además, habrá imágenes que nos redirigirán a las diferentes secciones de la aplicación (Canciones, Álbumes, Artistas, Usuarios).

5.3.1.3. Sección Canciones

Después de la Pantalla principal tenemos la primera de las 4 secciones de la Intranet: la sección de Canciones. En esta sección, los usuarios Administradores podrán añadir, editar, eliminar, buscar canciones y exportar los datos de las mismas.

Título	Artista	Año	Género	Acciones
Back in Black	AC/DC	1970	ROCK	
Beat It	Michael Jackson	1982	POP	
Carolina	M Clan	2001	BLUES	
Como te atreves a volver	Morat	2016	POP	
Highway To Hell	AC/DC	1979	ROCK	
Laura no esta	Nek	1997	POP	
Pan y Mantequilla	Efecto Pasillo	1997	POP	
Play Love	Zartcort	2013	RAP	

First < 1 > Last

- Añadir Canciones:** Para poder añadir una nueva canción al catálogo musical tenemos que pulsar el botón “Añadir” que se encuentra sobre el listado de canciones. Al pulsar este botón se abrirá el formulario de Canciones donde deberemos introducir la información de la nueva canción.

Insertar Canción

Título:	<input type="text"/>	Video Canción:	<input type="button" value="Examinar..."/> No se ha seleccionado ningún archivo.
Artista:	<input type="text" value="Coldplay"/>	Buscar Álbum	
Álbum:	<input type="text"/>		
Género:	<input type="text" value="BLUES"/>		
Año:	<input type="text"/>		

Audio Canción:
 No se ha seleccionado ningún archivo.

Guardar Volver

Cuando lo hayamos hecho, pulsamos el botón “Guardar” que se encargará de Validar los datos de la canción, es decir, comprobará que los datos introducidos sean los correctos y que la canción que queremos añadir no esté ya insertada en el catálogo musical. Si se produce algún error en el proceso de validación nos devolverá al formulario de Canciones indicándonos que errores hemos cometido. Si los datos introducidos son correctos, la canción será añadida al catálogo musical y se nos redirigirá al listado de canciones donde se nos mostrará un mensaje indicándonos que la canción se ha añadido correctamente.

- Editar Canciones:** Si queremos editar la información de una canción ya existente tenemos que pulsar sobre el botón “editar” de la canción en el listado de canciones. Una vez pulsemos el botón se nos

abrirá la pantalla “Editar canción” con los datos de la canción en el formulario.

Después de haber modificado los datos de la canción y pulsemos el botón “guardar” se realizarán las mismas validaciones que hemos explicado en al insertar la canción. Si se produce algún error nos mostrará un mensaje indicándonoslo. Si los nuevos datos son correctos se actualizarán los datos de la canción y se nos redirigirá al listado de canciones donde se nos mostrará un mensaje indicándonos que la canción se ha actualizado correctamente.

- Eliminar Canciones:** Si queremos eliminar una canción del catálogo musical tenemos que pulsar sobre el botón “eliminar” de la canción en el listado de canciones. Cuando pulsamos ese botón nos saldrá un mensaje de confirmación. Si pulsa la opción “Aceptar” la canción será eliminada.
- Buscar Canciones:** Sobre el listado de canciones podemos ver un buscador que nos va a permitir filtrar el listado de las canciones. Para ello tenemos que introducir datos en el buscador y pulsar el botón “buscar”. El filtrado de álbumes se podrá realizar por título, género, época y artista.
- Exportar Datos Listado Canciones:** La Aplicación nos permite exportar los datos que haya en el listado en ese momento a tres formatos distintos al pulsar alguno de los botones de exportación que se encuentran sobre el listado de canciones: .xlsx, .csv y .txt.

5.3.1.4. Sección Álbumes

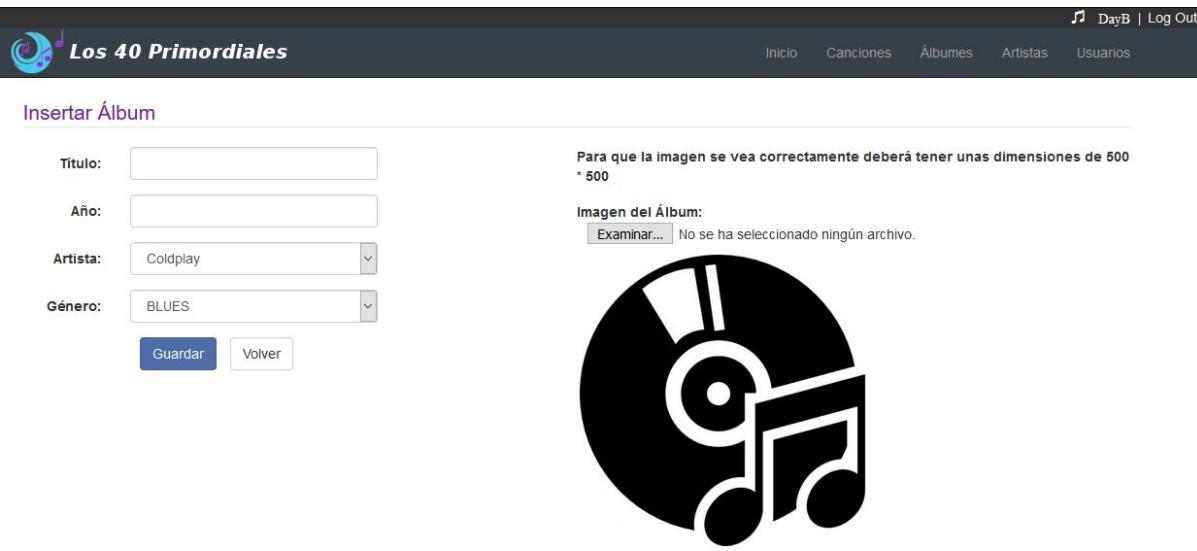
Una vez definidos todos los procesos de la Sección de Canciones vamos a pasar a la Sección de Álbumes. En esta sección podremos realizar gestiones sobre los álbumes existentes o añadir álbumes nuevos.

Título	Artista	Género	Fecha Salida	Acciones
Back in Black	AC/DC	ROCK	2003	
Backstreet's Back	Backstreet Boys	POP	1997	
El misterioso caso de...	Efecto Pasillo	POP	2013	
Formula, Vol. 2	Romeo Santos	BACHATA	2013	
Highway To Hell	AC/DC	ROCK	1979	
Nek Lei, gli amici e tutto il resto	Nek	POP	1997	
News of the World	Queen	ROCK	1977	
Out of Our Heads	The Rolling Stones	ROCK	1965	

- Añadir Álbum:** Para poder añadir un nuevo álbum tendremos que pulsar sobre el botón “Añadir” que hay sobre el listado de álbumes. Al pulsar sobre el botón se nos abrirá el formulario de álbumes.

En el formulario se podrán introducir el título del álbum, la fecha de salida, el artista al que pertenece, el género del álbum y la carátula del mismo.

Una vez introducidos los datos pulsamos el botón “guardar”. Al pulsar este botón se realizarán las validaciones de los datos. Estas validaciones consistirán en comprobar que los datos introducidos sean los correctos y que el álbum que queremos añadir no esté ya insertado. Si se produce algún error en el proceso de validación nos devolverá al formulario de álbumes es indicándonos que errores hemos cometido. Si los datos introducidos son correctos el álbum será añadido y se nos redirigirá al listado de álbumes donde se nos mostrará un mensaje indicándonos que el álbum se ha añadido correctamente.



The screenshot shows the 'Insertar Álbum' (Insert Album) form. It includes fields for Title, Year, Artist (Coldplay), and Genre (BLUES). There is a note about image dimensions (500x500 pixels) and a file input for the album cover image. A large decorative graphic of a vinyl record with musical notes is centered on the page.

- Editar Álbum:** Si queremos editar la información de un álbum ya existente tenemos que pulsar sobre el botón “editar” del álbum en el listado de álbumes. Una vez pulsemos el botón se nos abrirá la pantalla “Editar Álbum” con los datos del álbum en el formulario.

Después de haber modificado los datos del álbum y pulsemos el botón “guardar” se realizarán las mismas validaciones que hemos explicado en al insertar el álbum. Si se produce algún error nos mostrará un mensaje indicándonoslo. Si los nuevos datos son correctos se actualizarán los datos del álbum y se nos redirigirá al listado de álbumes donde se nos mostrará un mensaje indicándonos que el álbum se ha actualizado correctamente.

- Eliminar Álbum:** Si queremos eliminar un álbum musical tenemos que pulsar sobre el botón “eliminar” del álbum en el listado de álbumes. Cuando pulsemos ese botón nos saldrá un mensaje de confirmación. Si pulsa la opción de Aceptar el álbum será eliminada. No obstante, si hay alguna canción asociada al álbum que queremos borrar nos dará error al intentar eliminarlo mostrándonos un mensaje de información en la parte superior de la pantalla.
- Buscar Álbumes:** Sobre el listado de álbumes podemos ver un buscador que nos va a permitir filtrar el listado de los álbumes. Para ello tenemos que introducir datos en el buscador y pulsar el botón “buscar”. El filtrado de álbumes se podrá realizar por “nombre”, “género” y “artista”.
- Exportar Datos Listado Álbumes:** La Aplicación nos permite exportar los datos que haya en el listado en ese momento a tres formatos distintos al pulsar alguno de los botones de exportación que se encuentran sobre el listado de álbumes: .xlsx, .csv y .txt.

5.3.1.5. Sección Artistas

La siguiente sección que nos vamos a encontrar es la Sección de Artistas. En esta sección podremos realizar gestiones sobre los Artistas o “Dar de Alta” a nuevos.

Nombre	Fecha Origen	Género	Nación	Acciones
AC/DC	1973	HEAVY METAL	Australia	
Backstreet Boys	1993	POP	Reino Unido	
Coldplay	1996	POP	Reino Unido	
Efecto Pasillo	2007	POP	España	
Iron Maiden	1975	HEAVY METAL	Reino Unido	
Led Zeppelin	1968	ROCK	Reino Unido	
M Clan	2001	ROCK	España	
Mecano	1981	POP	España	

- Añadir Artista:** Para poder añadir un nuevo artista tendremos que pulsar sobre el botón “Añadir” que hay sobre el listado de Artistas. Al pulsar sobre el botón se nos abrirá el formulario de Artistas. En el formulario se podrán introducir el nombre del artista, la fecha de origen, el género del artista, nacionalidad e información adicional del artista.

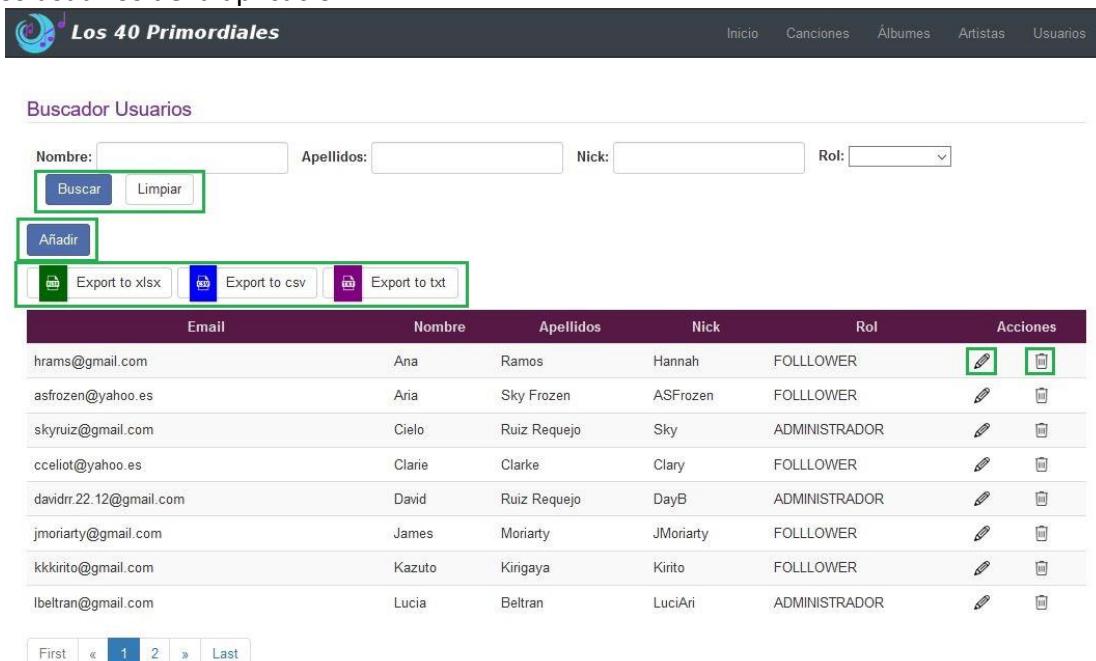
Una vez introducidos los datos pulsamos el botón “guardar”. Al pulsar este botón se realizarán las validaciones de los datos. Estas validaciones consistirán en comprobar que los datos introducidos sean los correctos y que el artista que queremos añadir no esté ya registrado en la emisora. Si se produce algún error en el proceso de validación nos devolverá al formulario de Artistas indicándonos

que errores hemos cometido. Si los datos introducidos son correctos el artista será añadido y se nos redirigirá al listado de Artistas donde se nos mostrará un mensaje indicándonos que el artista se ha añadido correctamente.

- **Editar Artista:** Si queremos editar la información de un artista ya existente tenemos que pulsar sobre el botón “editar” del artista en el listado de Artistas. Una vez pulsemos el botón se nos abrirá la pantalla “Editar Artista” con los datos del artista en el formulario. Después de haber modificado los datos del artista y pulsemos el botón “guardar” se realizarán las mismas validaciones que hemos explicado en al insertar el artista. Si se produce algún error nos mostrará un mensaje indicándonoslo. Si los nuevos datos son correctos se actualizarán los datos del artista y se nos redirigirá al listado de Artistas donde se nos mostrará un mensaje indicándonos que el artista se ha actualizado correctamente.
- **Eliminar Artista:** Si queremos eliminar un artista musical tenemos que pulsar sobre el botón eliminar del artista en el listado de artistas. Cuando pulsemos ese botón nos saldrá un mensaje de confirmación, si pulsa la opción de Aceptar el artista será eliminada. No obstante, si hay alguna canción o álbum asociada al artista que queremos borrar nos dará error al intentar eliminarlo mostrándonos un mensaje de información en la parte superior de la pantalla.
- **Buscar Artista:** Sobre el listado de artistas podemos ver un buscador que nos va a permitir filtrar el listado de los artistas. Para ello tenemos que introducir datos en el buscador y pulsar el botón buscar. El filtrado de artistas se podrá realizar por nombre y por género.
- **Exportar Datos Listado Artista:** La Aplicación nos permite exportar los datos que haya en el listado en ese momento a tres formatos distintos al pulsar alguno de los botones de exportación que se encuentran sobre el listado de artistas: .xlsx, .csv y .txt.

5.3.1.6. Sección Usuarios

Para finalizar este Manual vamos a hablar de la Sección de Usuarios. En esta sección se lleva a cabo la gestión de los usuarios de la aplicación.

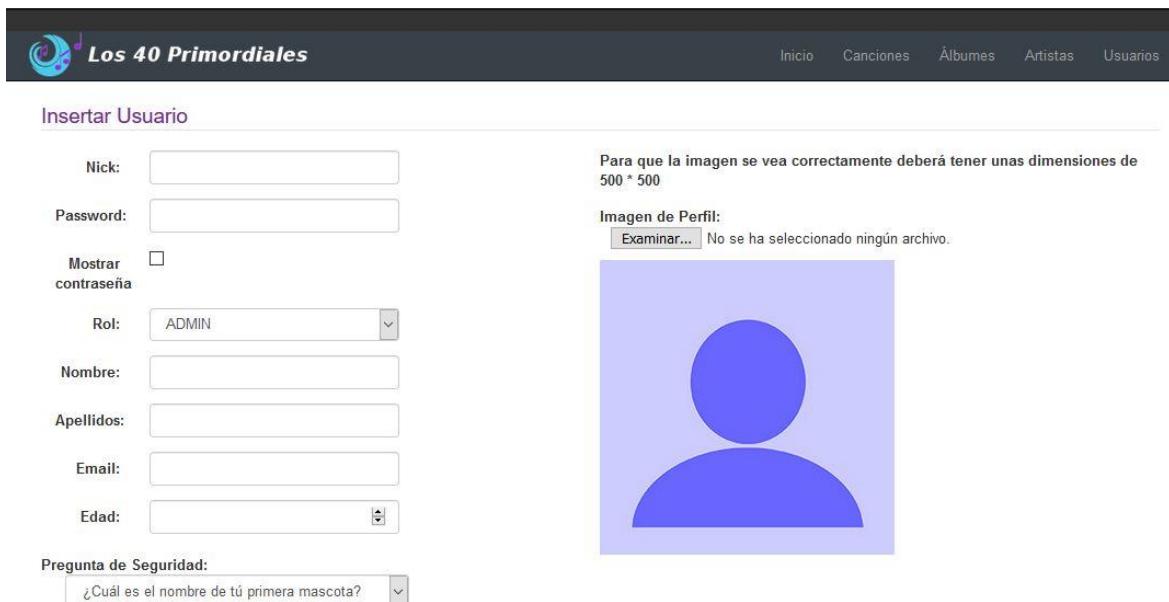


The screenshot shows the 'User Management' section of the application. At the top, there is a navigation bar with the logo 'Los 40 Primordiales' and links for 'Inicio', 'Canciones', 'Álbumes', 'Artistas', and 'Usuarios'. Below the navigation bar is a search form titled 'Buscador Usuarios' with fields for 'Nombre', 'Apellidos', 'Nick', and 'Rol', along with 'Buscar' and 'Limpiar' buttons. There is also an 'Añadir' button. Below the search form are three export buttons: 'Export to xlsx', 'Export to csv', and 'Export to txt'. The main area displays a table of users with columns: Email, Nombre, Apellidos, Nick, Rol, and Acciones. The table contains the following data:

Email	Nombre	Apellidos	Nick	Rol	Acciones	
hrams@gmail.com	Ana	Ramos	Hannah	FOLLOWER		
astrozen@yahoo.es	Aria	Sky Frozen	ASFrozen	FOLLOWER		
skyruiz@gmail.com	Cielo	Ruiz Requejo	Sky	ADMINISTRADOR		
cceliot@yahoo.es	Clarie	Clarke	Clary	FOLLOWER		
davidr.22.12@gmail.com	David	Ruiz Requejo	DayB	ADMINISTRADOR		
jmorarity@gmail.com	James	Moriarty	JMorarity	FOLLOWER		
kkkirito@gmail.com	Kazuto	Kirigaya	Kirito	FOLLOWER		
lbeltran@gmail.com	Lucia	Beltran	LuciAri	ADMINISTRADOR		

At the bottom, there is a pagination control with buttons for 'First', '<', '1', '2', '>', and 'Last'.

- **Alta Usuario:** Para poder dar de alta un nuevo usuario tendremos que pulsar sobre el botón “Añadir” que hay sobre el listado de usuarios. Al pulsar sobre el botón se nos abrirá el formulario de usuarios. En el formulario se podrán introducir el nombre, apellidos, Nick, password, email, pregunta seguridad, respuesta seguridad e imagen de perfil.



The screenshot shows the 'Insertar Usuario' (Insert User) form. It includes fields for Nick, Password, Show password checkbox, Role (set to ADMIN), Name, Surname, Email, Age, Security Question (What was the name of your first pet?), and a placeholder for Profile Picture. The profile picture placeholder is a blue silhouette of a person.

Una vez introducidos los datos pulsamos el botón “guardar”. Al pulsar este botón se realizarán las validaciones de los datos. Estas validaciones consistirán en comprobar que los datos introducidos sean los correctos. Además, se comprobará que el Nick y el email introducidos no pertenezcan a un usuario ya dado de alta en la aplicación. Si se produce algún error en el proceso de validación nos devolverá al formulario de usuarios indicándonos que errores hemos cometido. Si los datos introducidos son correctos el usuario será añadido y se nos redirigirá al listado de usuarios donde se nos mostrará un mensaje indicándonos que el usuario se ha añadido correctamente.

- **Editar Usuario:** Si queremos editar la información de un usuario ya existente tenemos que pulsar sobre el botón “editar” del usuario en el listado de usuarios. Una vez pulsemos el botón se nos abrirá la pantalla “Editar Usuario” con los datos del usuario en el formulario. Después de haber modificado los datos del usuario y pulsemos el botón “guardar” se realizarán las mismas validaciones que hemos explicado en al insertar el usuario. Si se produce algún error nos mostrará un mensaje indicándonoslo. Si los nuevos datos son correctos se actualizarán los datos del usuario y se nos redirigirá al listado de usuarios donde se nos mostrará un mensaje indicándonos que el usuario se ha actualizado correctamente.
- **Eliminar Usuario:** Si queremos eliminar un usuario tenemos que pulsar sobre el botón “eliminar” del usuario en el listado de usuarios. Cuando pulsemos ese botón nos saldrá un mensaje de confirmación, si pulsa la opción de Aceptar el usuario será eliminado. Una vez eliminemos un usuario se le enviará un mensaje a su correo electrónico indicándole que ha sido dado de baja de la aplicación.
- **Buscar Usuario:** Sobre el listado de usuarios podemos ver un buscador que nos va a permitir filtrar el listado de los usuarios. Para ello tenemos que introducir datos en el buscador y pulsar el botón buscar. El filtrado de usuarios se podrá realizar por nombre, apellidos, Nick y rol.

- **Exportar Datos Listado Usuario:** La Aplicación nos permite exportar los datos que haya en el listado en ese momento a tres formatos distintos al pulsar alguno de los botones de exportación que se encuentran sobre el listado de usuarios: xlsx, csv y txt.

5.3.2. WebApp

Después de haber descrito la Intranet, vamos a dar paso a la WebApp. La página web consta de 4 apartados, a los cuales se pueden acceder desde el menú de navegación situado en la parte superior derecha de la pantalla. Las páginas de la web tendrán una estructura similar a la de la intranet. La estructura es la siguiente:

- **Cabecera:** En la cabecera podemos ver un menú superior que nos permite acceder a las diferentes secciones de la aplicación (Inicio, Catálogo Musical, Mi Perfil, Foro, Contactos). Algunas de las secciones solo se mostrarán si el Usuario se encuentra loggeado en la aplicación (Mi Perfil, Foro)
- **Contenido:** Es el apartado central de las páginas, donde se encuentra el contenido.
- **Pie de Página:** Se encuentra en la parte inferior de las páginas de la Web y contiene los accesos a las diferentes redes sociales de “Los 40 Primordiales”.

A continuación, pasare a hablar acerca del contenido de cada una de las páginas.

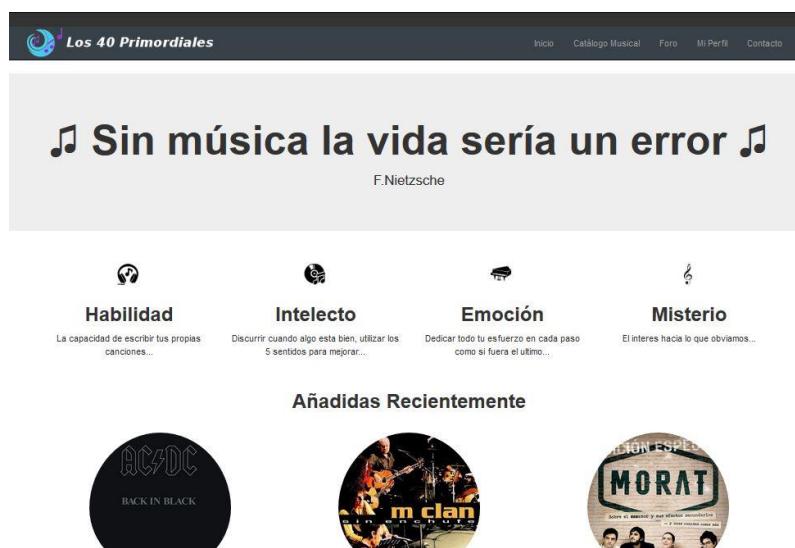
5.3.2.1. Login

- **Login:** Para poder loggearnos en la aplicación deberemos introducir el Nick/Email y la Contraseña en los campos del formulario de la pantalla de Login y pulsar el botón “Login”. Si los datos introducidos no se corresponden con los de un usuario registrado se mostrará un mensaje de error. Sin embargo, si los datos introducidos corresponden con los de un “Usuario Administrador” registrado entonces se realizará el login en la web.
- **Recuperar Contraseña:** Si hemos olvidado nuestra contraseña podremos recuperarla pulsando el botón “recuperar contraseña” de la pantalla de Login. Una vez pulsado el botón la aplicación nos redirigirá a una pantalla donde tendremos que introducir el email con el que nos dimos de alta en la aplicación y pulsar “enviar”. Si el email introducido no existe o es incorrecto nos mostrará un mensaje de error. Si el email introducido es correcto y pertenece a un usuario registrado en la aplicación se nos enviará un correo al email introducido con una Url para poder restablecer la contraseña.
- **Alta Usuario:** Para poder dar de alta un nuevo usuario tendremos que pulsar sobre el botón “Dar de Alta”. Al pulsar sobre el botón se nos abrirá el formulario de usuarios. En el formulario se podrán introducir el nombre, apellidos, Nick, password, email, pregunta de seguridad, respuesta de seguridad e imagen de perfil.

Una vez introducidos los datos pulsamos el botón “guardar”. Al pulsar este botón se realizarán las validaciones de los datos. Estas validaciones consistirán en comprobar que los datos introducidos sean los correctos. Además, se comprobará que el Nick y el email introducidos no pertenezcan a un usuario ya dado de alta en la aplicación. Si se produce algún error en el proceso de validación nos devolverá al formulario de usuarios indicándonos que errores hemos cometido. Si los datos introducidos son correctos el usuario será añadido y se nos redirigirá a la pantalla de Login.

5.3.2.2. Página Inicio

Al entrar a la WebApp, la primera página que nos encontramos es la Página de Inicio. Desde ella tenemos acceso a información de la Emisora de Radio de “Los 40 Primordiales” y a información acerca del contenido que el usuario va a poder ver en la página. Además, se podrán ver las últimas canciones que han sido añadidas al catálogo musical.



♪ Sin música la vida sería un error ♪
F.Nietzsche

Habilidad
La capacidad de escribir tus propias canciones...

Intelecto
Discurrir cuando algo está bien, utilizar los 5 sentidos para mejorar...

Emoción
Dedicar todo tu esfuerzo en cada paso como si fuera el último...

Misterio
El interés hacia lo que obviamos...

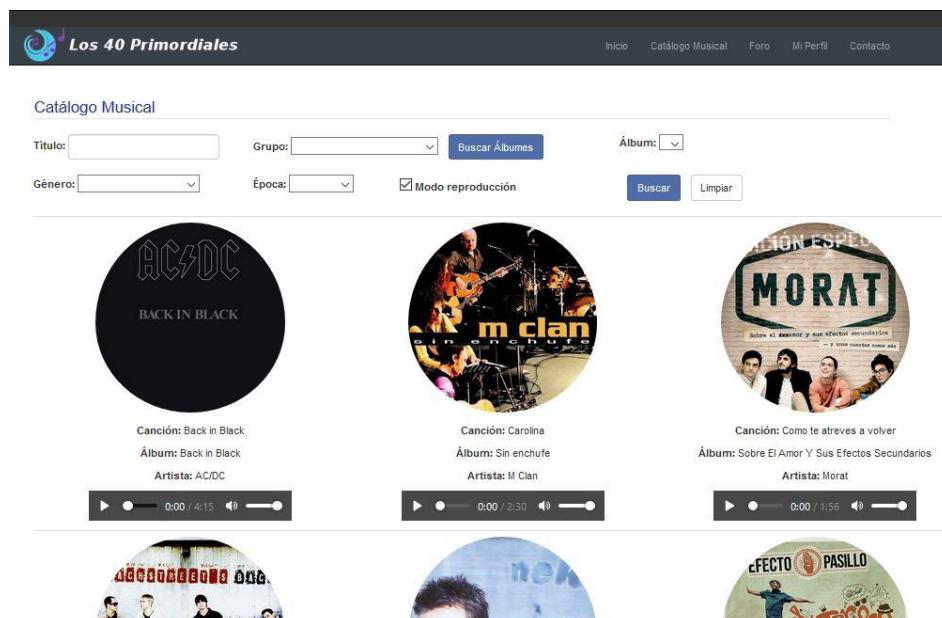
Añadidas Recientemente

- 
- 
- 

5.3.2.3. Catálogo Musical

El Catálogo Musical es el apartado principal de la aplicación donde los usuarios podrán acceder al contenido musical de la emisora. Para mejorar la usabilidad de la aplicación se ha decidido añadir un buscador sobre el listado de canciones. Este buscador nos permitirá filtrar las canciones por Título, Artista, Álbum, Género y Periodo. Además, se han implementado dos modalidades de Catálogo diferentes. Estas modalidades son:

- Modo Básico:** El modo básico consiste en la representación del listado de canciones utilizando una tabla. Los datos disponibles desde este modo serán: Título, Artista, Álbum, Género y Periodo. También habrá un botón que nos permitirá visualizar la ficha de la canción.
- Modo Reproducción:** En este modo las canciones se visualizarán de manera más amigable para el usuario permitiendo que éste reproduzca la canción desde el listado.



Catálogo Musical

Título: Grupo: Buscar Álbumes
Álbum:
Género: Época: Modo reproducción
Buscar Limpiar

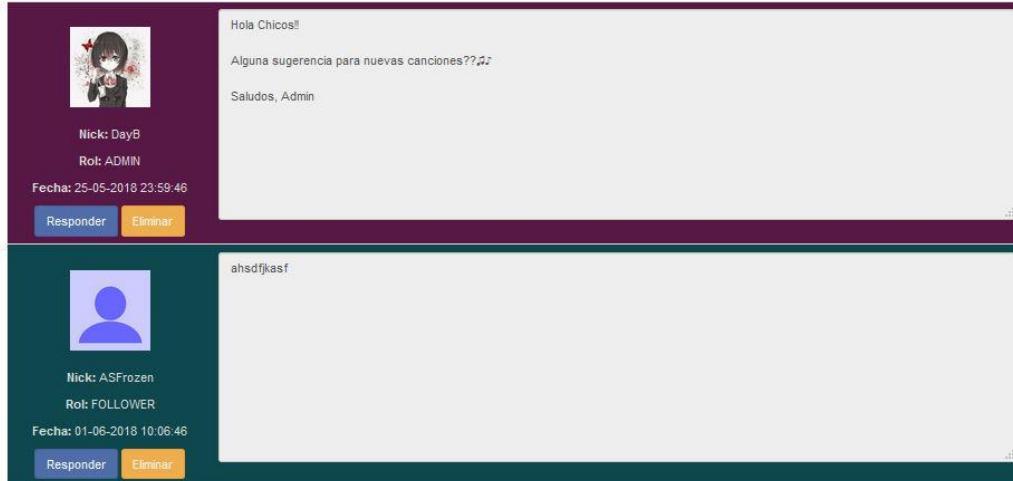
 Canción: Back in Black. Álbum: Back in Black Artista: AC/DC	 Canción: Carolina Álbum: Sin enchufe Artista: M Clan	 Canción: Como te atreves a volver Álbum: Sobre El Amor Y Sus Efectos Secundarios Artista: Morat
 0:00 / 4:15	 0:00 / 2:30	 0:00 / 1:56

5.3.2.4. Foro

Esta es una de las secciones que solo estará disponible para aquellos usuarios que estén loggeados. En esta sección los usuarios podrán comentar temas ya existentes o añadir nuevos.

- Añadir Nuevo Tema:** Para poder añadir nuevos temas a la aplicación tendremos que pulsar sobre el botón “Añadir” que se encuentra en sobre el listado de temas. Al pulsar este botón se abrirá el formulario del Tema. Una vez hayamos llenado el Título del Tema y el Comentario pulsaremos sobre “Publicar” para añadir el nuevo tema.
- Eliminar Tema:** Esta opción solo estará disponible para los usuarios administradores. Estos usuarios podrán pulsar el botón “Eliminar” del tema. Al pulsar este botón mostrará un mensaje de confirmación. Si se pulsa “Aceptar” el tema será eliminado.
- Listado Comentarios:** Sobre un tema los usuarios podrán acceder al listado de los comentarios de ese tema. En este listado se visualizarán todos los comentarios ordenados por fecha. En cada uno de los comentarios se mostrará el Nick del Usuario que lo ha escrito, su rol en la aplicación, su imagen de perfil si la hubiera, la fecha de publicación del comentario y el comentario.
- Publicar Comentario:** En la parte inferior del listado de comentarios se encuentra el Formulario de Comentarios donde los usuarios podrán escribir sus comentarios para que sean publicados en el tema.
- Responder Comentario:** En la parte inferior de los comentarios se encuentra el botón “Responder”. Al pulsar sobre este botón se cargará el comentario seleccionado para así poder responder al mismo.
- Eliminar Comentario:** Esta opción solo estará disponible para los usuarios administradores. Junto al botón “Responder” se mostrará el botón “Eliminar” que les permitirá eliminar comentarios publicados.

♪ Tema "Nuevas Canciones" ♪



The screenshot shows a forum interface with a dark purple header and a light purple footer. The main content area has a white background.

Header (Top Left): Shows a user profile picture of a girl with black hair and red flowers, followed by the text: "Nick: Day6", "Rol: ADMIN", and "Fecha: 25-05-2018 23:59:46". Below this are two buttons: "Responder" (blue) and "Eliminar" (orange).

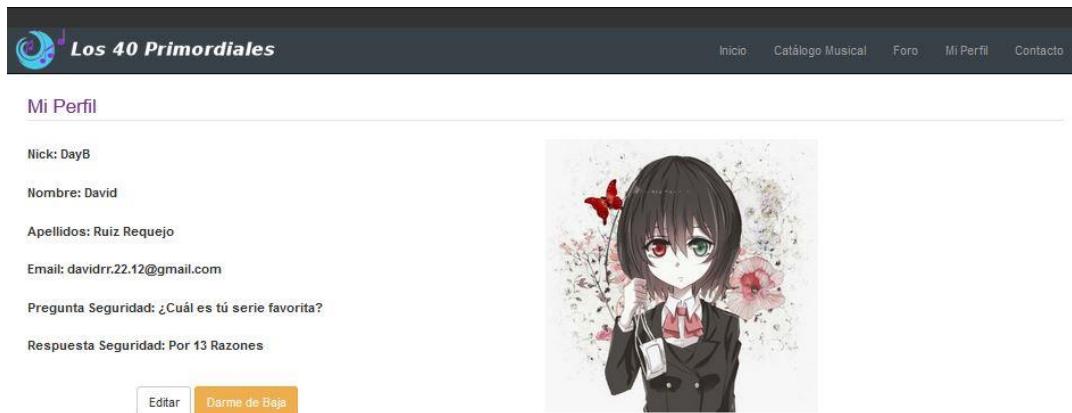
Header (Top Right): Displays the title of the topic: "Hola Chicos!!" and a message: "Alguna sugerencia para nuevas canciones??♪♪". Below this is another message: "Saludos, Admin".

Comment 1 (Top): Shows a user profile picture of a girl with a blue gradient, followed by the text: "Nick: ASFrozen", "Rol: FOLLOWER", and "Fecha: 01-06-2018 10:06:46". Below this are two buttons: "Responder" (blue) and "Eliminar" (orange). The comment text itself is "ahsdjfasf".

Comment 2 (Bottom): Shows a user profile picture of a blue gradient circle, followed by the text: "Nick: ASFrozen", "Rol: FOLLOWER", and "Fecha: 01-06-2018 10:06:46". Below this are two buttons: "Responder" (blue) and "Eliminar" (orange). The comment text itself is "ahsdjfasf".

5.3.2.5. Mi Perfil

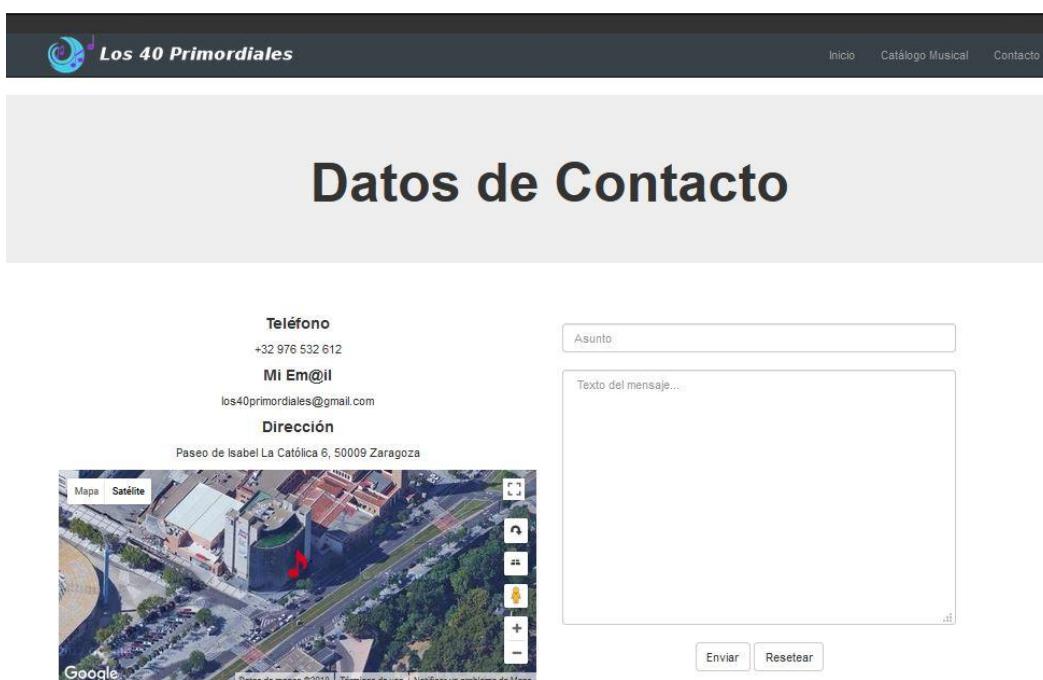
En esta sección de la aplicación los usuarios podrán personalizar su perfil. Para ello tendremos que pulsar el botón “Editar”. Una vez pulsemos el botón se nos abrirá la pantalla “Editar Usuario” con los datos del usuario en el formulario.



Después de haber modificado los datos del usuario y de pulsar el botón “guardar” se realizarán las mismas validaciones que hemos explicado en al dar de alta el usuario. Si se produce algún error nos mostrará un mensaje indicándonoslo. Si los nuevos datos son correctos se actualizarán los datos del usuario y se nos redirigirá a la pantalla principal de la aplicación.

5.3.2.6. Contacto

Para finalizar nos encontramos con la pantalla de “contacto” en la que el usuario podrá transmitir sus consejos y sugerencias a los administradores de la web por vía email. En esta pantalla, además, el usuario tendrá datos de contacto como el teléfono, el correo de la empresa y la dirección de la sede (esta dirección estará visualizada en un mapa). Para poder enviar los mensajes los usuarios deberán estar loggeados en la aplicación



6. Documento de Cierre

6.1. Resultados Obtenidos y Conclusiones

6.1.1. Resultados Obtenidos

Mirando atrás, al comienzo del proyecto, y viendo las expectativas que tenía puestas en él, debo decir que estoy contento con el trabajo realizado. Estoy satisfecho ya que he conseguido superar muchas de las adversidades que me he ido encontrando durante el transcurso del mismo.

He de destacar, que una de las mayores dificultades que me he encontrado ha sido la utilización SpringBoot junto con Maven para la realización del proyecto. Al no tener conocimiento base sobre el Framework Spring me fue difícil empezar con el proyecto, sin embargo, una vez superada esa primera etapa, he conseguido mejores resultados de los que pretendía en un primer momento.

Debido al corto plazo que hemos tenido para poder realizar el proyecto ha habido funcionalidades de las que estaban previstas en un primer momento que no ha sido posible realizar. No obstante, la emisora ha quedado satisfecha con el trabajo realizado y no se descarta que en un futuro próximo opten por encargarnos ciertas mejoras y nuevos desarrollos para las aplicaciones.

6.1.2. Conclusiones

Aunque el proyecto conjunto de MusicSelector, es decir, Intranet y WebApp, no han salido tal y como se habían planificado en un primer momento, creo que realizar este proyecto me ha servido para madurar como programador. He podido descubrir las dificultades asociadas a desarrollar un proyecto tan ambicioso desde 0, comenzando por el Análisis del mismo, hasta el diseño y la programación. Me ha enseñado lo importante que es dedicarle tiempo a realizar una base firme para que no se produzcan pérdidas de un tiempo limitado, ya que éste suele ser establecido por el cliente. Y lo más importante, me he dado cuenta de que, cuando hablamos de programación, pocas cosas son imposibles. Sólo hay que dedicarle tiempo, ser constante, saber buscar la solución y darte cuenta de que, en la mayoría de las ocasiones, ha habido antes otra persona con el mismo problema o con un problema muy similar al tuyo.

En resumen, aunque no haya podido terminar este proyecto como me gustaría, tengo la firme intención de seguir mejorándolo hasta llegar al punto en el que cumpla los requisitos expuestos al comienzo.

6.2. Diario de Bitácora

Como diario de bitácora se tendrán en cuenta los commits de los proyectos realizados en GitHub. A continuación, adjunto las direcciones de los proyectos en GitHub:

https://github.com/DaviDRR2212/MS_Intranet

https://github.com/DaviDRR2212/MS_WebApp

7. Bibliografía

Direcciones Base

Libro Clean Code

<https://www.mkyong.com/spring-boot/spring-boot-jdbc-oracle-database-commons-dbcp2-example/>
<https://spring.io/guides/gs/handling-form-submission/>

Otras Direcciones

<http://www.baeldung.com/spring-mvc-form-tutorial>
<http://acodigo.blogspot.com.es/2017/04/spring-mvc-validar-formularios.html>
<http://www.forosdelweb.com/f45/spring-mvc-como-obtener-variable-para-iterar-vista-jsp-1051048/>
<http://www.jeromejaglale.com/doc/java/spring/mvc>
<https://medium.com/@milansavaliyaz/spring-boot-hello-world-example-with-jsp-view-7ffed2ae931d>
<https://hellokoding.com/spring-boot-hello-world-example-with-jsp/>
<https://blogs.oracle.com/geertjan/spring-boot-scenario-with-netbeans-ide>
<https://code.i-harness.com/es/q/1375ce2>
<https://www.logicbig.com/tutorials/spring-framework/spring-data-access-with-jdbc/connect-oracle.html>
<https://www.logicbig.com/tutorials/spring-framework/spring-data-access-with-jdbc/spring-call-stored-procedure.html>
<https://manthapavankumar.wordpress.com/2013/10/14/calling-oracle-stored-procedure-using-spring-jdbc/>
<https://www.adictosaltrabajo.com/tutoriales/spring-email/>
<http://jvmhub.com/2015/08/02/spring-boot-with-thymeleaf-tutorial-part-2-forms-with-validation/>
<https://o7planning.org/en/11655/create-a-user-registration-application-with-spring-boot-spring-form-validation#a14002325>
<https://o7planning.org/en/11669/spring-boot>
<https://stackoverflow.com/questions/2422468/how-to-upload-files-to-server-using-jsp-servlet>
<https://spring.io/guides/gs/validating-form-input/>
<https://stackoverflow.com/questions/3303675/how-to-make-an-input-type-button-act-like-a-hyperlink-and-redirect-using-a-get-r>
<https://www.htmlcinco.com/nuevos-tipos-de-input-en-html5/>
<http://javasampleapproach.com/spring-framework/perform-form-validation-spring-boot>
<https://stackoverflow.com/questions/8781558/neither-bindingresult-nor-plain-target-object-for-bean-name-available-as-request>
<http://www.jtech.ua.es/j2ee/publico/spring-2012-13/sesion05-apuntes.html>
<https://www.mkyong.com/spring-mvc/spring-mvc-dropdown-box-example/>
<https://htmlcolorcodes.com/es/>

8. Anexos

8.1. Contrato y Pliego de Condiciones (Apartado 3.6)



CONTRATO DE APLICACIÓN

En _____, a ___ de ____ de 20___

REUNIDOS

D David Ruiz Requejo, mayor de edad, con DNI 73021215 X y domicilio en Zaragoza, actuando en nombre y representación de Strixe S.L. inscrita en el Registro Mercantil de Zaragoza con domicilio social en Zaragoza, actuando en su calidad de Programador, en posesión de poderes suficientes para este acto. (El Desarrollador)

D XXXXXXXXXXXXXX, mayor de edad, con DNI XXXXXXXXXX y domicilio en XXXXXXXXXXXXXXXXXXXXXX, actuando en nombre y representación de la emisora "Los 40 primordiales" inscrita en el Registro Mercantil de. XXXXXXXXXXXXXX, en calidad de XXXXXXXXXXXXXX, en posesión de poderes suficientes para este acto. (El cliente)

MANIFIESTAN

Que las partes están interesadas en formalizar el presente contrato; que poseen suficientes poderes para la firma del mismo; que se reconocen capacidad legal necesaria para poder llevar a cabo la celebración y declaran expresamente que actúan de forma libre, voluntaria y no viciada.

EXPONEN

El cliente está interesado en que el prestador lleve a cabo el diseño y desarrollo de un proyecto que incluye una aplicación de escritorio o aplicación web que se utilizará como intranet de la emisora, y una aplicación web, conforme a las necesidades específicas indicadas en el Anexo I del presente contrato.

Que el prestador tiene como objeto el diseño y desarrollo de las aplicaciones arriba mencionadas, y que cuenta con los medios necesarios para ello; que las características encargo son las indicadas en el Anexo I por el cliente; que ambas partes aceptan cumplir con sus respectivas obligaciones.

En relación a lo anteriormente expuesto, las partes otorgan el presente contrato que se regirá por las siguientes

CLAUSULAS

I. OBJETO DEL CONTRATO

El presente contrato regula la prestación de servicio de diseño y programación solicitado por el cliente. Las aplicaciones se acogerán en todo caso a las categorías, diseño y contenidos indicados en el Anexo I de este contrato.

II. PROPIEDAD INTELECTUAL

El prestador garantiza al cliente que todo el material utilizado para el desarrollo del proyecto, así como el resultado obtenido, es un producto original que no vulnera ninguna ley o derechos de terceros, en especial los referidos a propiedad industrial e intelectual.

En caso de ser el cliente el encargado de proporcionar los contenidos (gráficos, textos, vídeos, categorías...), éste se hace responsable de cualquier tipo de reclamación de terceros en relación a la titularidad de dichos contenidos, eximiendo de toda responsabilidad al prestador.

III. OBLIGACIONES DEL PRESTADOR

El prestador se compromete a desarrollar el presente proyecto bajo las directrices del cliente, ajustándose a los términos indicados por éste en el Anexo I del presente contrato y conforme a las mejores prácticas existentes en el mercado, así como con la máxima diligencia posible.

Una vez aceptadas las características del proyecto, pueden producirse variaciones en el diseño y/o contenido del mismo a petición del cliente. Salvo que conlleven una variación sustancial del Proyecto Inicial no supondrá aumento del precio, considerando variación sustancial toda aquello que suponga un aumento del tiempo total de trabajo estimado inicialmente superior al 25%.

El prestador se compromete a finalizar el desarrollo en el plazo acordado siempre que la otra parte haya colaborado activamente en el desarrollo del mismo (aceptando los diseños, entregando los contenidos, etc)

IV. OBLIGACIONES DEL CLIENTE

El cliente se obliga a realizar el pago del precio en los términos indicados en la cláusula VI del presente contrato.

El cliente se obliga a mantener un contacto constante con el prestador entregando en tiempo y forma los contenidos del proyecto (Textos, imágenes, videos, categorías...), la aceptación del diseño y cualquier otra necesidad que requiera el prestador para poder finalizar el proyecto. En cualquier caso, se atenderá a lo dispuesto respecto a propiedad intelectual en la cláusula II.

En caso de depender la entrega de contenidos de un tercero seleccionado por el cliente, éste deberá indicarlo en el apartado comunicaciones y comprometiéndose a responder de los posibles retrasos que pudieran darse

V. COMUNICACIONES

Las partes se obligan a comunicarse toda la información que pudiera ser necesaria para el correcto desarrollo del proyecto. Toda comunicación entre las partes relativa al presente contrato se realizará por escrito o telefónicamente. A efectos de comunicaciones y/o notificaciones las partes designan:

Prestador

Domicilio en _____, con número de Fax _____, correo electrónico _____ y
Teléfono _____

Cliente

Domicilio en _____, con número de Fax _____, correo electrónico _____ y
Teléfono _____

Tercero

Domicilio en _____, con número de Fax _____, correo electrónico _____ y
Teléfono _____

Cualquier cambio de domicilio o dirección de contacto deberá ser comunicado a la otra parte por escrito con una antelación mínima de 5 días hábiles.

VI. DURACIÓN Y PRECIO

El presente contrato entrará en vigor el día 1 de marzo de 2018 y tendrá una duración de 3 meses, siendo posible, a petición del prestador, añadir 15 días adicionales para llevar a cabo la entrega del proyecto, sin ello suponer repercusión económica alguna.

El precio a abonar por parte del cliente como pago por la prestación del servicio prestado equivale a 8.000 € (ocho mil euros), añadiendo a dicha cantidad el IVA correspondiente (16%)

Dicho precio será abonado de la siguiente forma:

- 2.000 € (dos mil euros), que corresponden al 20 % del precio, serán abonados en el momento de la firma del presente contrato.
- El 80 % restante, equivalente a 6.000 € (seis mil euros), serán abonados en el momento de finalización del proyecto.

El pago del 80 % final será llevado a cabo siempre que el prestador de por finalizado el proyecto y en cualquier caso cuando se cumpla el plazo de entrega y el cliente no haya entregado o colaborado activamente en la entrega de contenidos, validación de colores y diseño o entrega de las aplicaciones encargadas.

El pago será realizado mediante transferencia Bancaria en el número de cuenta XXXXXXXXXXXXXXXX.

XXXX

VIII. CONFIDENCIALIDAD Y PROTECCIÓN DE DATOS

Las partes contratantes declaran conocer y cumplir la legislación europea y española sobre protección de datos, comprometiéndose a tratar los datos personales obtenidos durante el desarrollo del proyecto de acuerdo con dicha normativa.

Se informa al cliente que sus datos quedarán almacenados en un fichero, automatizado o no, con las únicas finalidades de informarle sobre novedades y nuevos proyectos en los que se encuentre trabajando la empresa prestadora, así como para el mantenimiento de la relación contractual.

Se informa al afectado que puede ejercer los derechos de acceso, rectificación, cancelación y oposición de sus datos de carácter personal solicitándolo por escrito y acompañando una fotocopia del DNI en la dirección del prestador indicada en el presente contrato.

Toda la información relativa al proyecto, así como a los datos de carácter personal, tendrá el carácter de confidencial por lo que las partes deberán guardar el mayor secreto respecto a las mismas, garantizando en todo caso el no acceso por parte de terceros a dicha información.

XI. NO COMPETENCIA

El cliente se compromete a no realizar proyectos de equivalente o análoga naturaleza para un tercero, ni iniciar a partir del presente desarrollo una nueva línea de negocio relacionada con los proyectos objeto de este contrato. Se considerará que el cliente incurre en dicha circunstancia siempre que opere directamente o indirectamente mediante otra empresa en la que disponga de participación social, o que actúe como mero asesor o colaborador y que en definitiva obtenga como resultado un proyecto igual, semejante o con la misma finalidad a la ofrecida por el prestador.

El incumplimiento del anterior compromiso llevará aparejada una penalización equivalente a 2.000 €, sin perjuicio y de las indemnizaciones que correspondiesen por los daños y perjuicios causados al prestador.

XI. EXTINCIÓN

Además de por las causas generales del Derecho, este contrato se extinguirá:

- a. Por el transcurso del mismo.
- b. Por ser declarados en situación de suspensión de pagos, quiebra o concurso de acreedores cualquiera de las partes.
- c. Por incumplimiento de las obligaciones estipuladas en el presente contrato.

Las partes aceptan expresamente el sometimiento a las penalizaciones económicas indicadas a continuación siempre que se rescinda el contrato de forma previa a la finalización del proyecto:

XII JURISDICCIÓN Y LEGISLACIÓN APLICABLE

Todas las cuestiones litigiosas sobre el presente contrato mercantil, quedarán regidas por la legislación española, específicamente por lo dispuesto en el Código Mercantil, y en su defecto, por las disposiciones españolas del Código de Comercio, Leyes Especiales, usos mercantiles y con carácter supletorio, por el Código Civil.

En cualquier caso, será obligatorio que en caso de conflicto las partes intenten previamente resolver la cuestión de mutuo acuerdo, sometiéndose en su caso a los Juzgados y Tribunales de Zaragoza que por orden correspondan.

ANEXO I: PROYECTO TÉCNICO

Descripción:

El proyecto va a consistir en el desarrollo de dos aplicaciones una de escritorio y una web o dos aplicaciones de escritorio. La primera de estas (app de escritorio o app web) servirá como intranet para los trabajadores de la emisora en ella podrán gestionar el catálogo musical perteneciente a la misma pudiendo hacerlo de una manera más eficiente a como se hacía antes. La segunda aplicación será una página web para los usuarios suscritos a la emisora donde se podrá encontrar contenido musical, visual... al cual no se podía acceder desde la radio.

Plazos establecidos:

El plazo de entrega se fija en el día **10 de junio de 2018**. Además, se adjunta una tabla con los plazos marcados para la finalización de cada una de las tareas que comprende el proyecto:

TAREA	FCH INICIO	FCH FIN
T1. Descripción de Proyecto	-	25/03/2018
T2. Documento Acuerdo de Proyecto	26/03/2018	08/04/2018
T3. Documento Análisis y Diseño del Primer Sprint	09/04/2018	22/04/2018
T3.1. Diseño Estructura BBDD	09/04/2018	13/04/2018
T3.2. Diseño Interfaz Mockup	14/04/2018	15/04/2018
T3.3. Aprender Spring	15/04/2018	22/04/2018
T4. Primer Sprint (Aplicación Intranet)	23/04/2018	08/05/2018
T4.1. Sección Usuarios (Login, Registro, Editar)	23/04/2018	26/04/2018
T4.2. Sección Catálogo (Añadir, Eliminar, Editar, Lista Reproducción)	26/04/2018	06/04/2018
T5. Documento análisis y diseño del segundo sprint	09/05/2018	20/05/2018
T5.1. Diseño Interfaz Mockup	09/05/2018	12/05/2018
T6. Segundo Sprint	21/05/2018	01/06/2018
T6.1. Sección Usuarios (Login, Registro, Editar)	21/05/2018	22/05/2018
T6.2. Sección Catálogo (Listado, Filtro, Información, Lista Reproducción, Exportar)	22/05/2018	24/05/2018
T6.3. Sección Información (Contacto)	24/05/2018	01/06/2018
T7. Documento de instalación, documento de manual de usuario y documento de cierre	01/06/2018	03/06/2018

Formato de entrega del trabajo:

El proyecto se entregará al cliente por duplicado. Cumpliendo el compromiso inicial, se generará un disco óptico con el contenido necesario para instalar las aplicaciones en el servidor.

La documentación, con la descripción de las aplicaciones, y los manuales de usuario, además de estar incluida en formato .pdf en el disco, se entregará, también por duplicado, en formato papel.

8.2. Scripts Base de Datos**8.2.1. Creates**

```
-- Generado por Oracle SQL Developer Data Modeler 4.0.3.853
-- en: 2018-05-04 23:26:02 CEST
-- sitio: Oracle Database 11g
-- tipo: Oracle Database 11g

CREATE TABLE MSTB_ALBUM
(
    idAlbum      INTEGER NOT NULL ,
    title        VARCHAR2 (50 CHAR) NOT NULL ,
    releaseDate  VARCHAR2 (10 CHAR) NOT NULL ,
    gender       VARCHAR2 (60 CHAR) NOT NULL ,
    cover        VARCHAR2 (50 CHAR) ,
    idGroup      INTEGER NOT NULL
);
ALTER TABLE MSTB_ALBUM ADD CONSTRAINT MSTB_ALBUM_PK PRIMARY KEY ( idAlbum ) ;
ALTER TABLE MSTB_ALBUM RENAME COLUMN gender to IDGENDER;
ALTER TABLE MSTB_ALBUM MODIFY IDGENDER NUMBER;
ALTER TABLE MSTB_ALBUM MODIFY releaseDate NUMBER;
ALTER TABLE MSTB_ALBUM ADD CONSTRAINT MSTB_TITLE_UN UNIQUE ( title ) ;
ALTER TABLE MSTB_ALBUM ADD codAlbum VARCHAR2(200 CHAR) NOT NULL;
ALTER TABLE MSTB_ALBUM ADD CONSTRAINT MSTB_ALBUM_UN UNIQUE ( codAlbum ) ;

CREATE TABLE MSTB_COMMENT
(
    idComment      INTEGER NOT NULL ,
    COMMENTARY     VARCHAR2 (400 CHAR) ,
    fchComment     DATE NOT NULL ,
    MSTB_THEME_idTopic INTEGER NOT NULL ,
    MSTB_USER_idUser  INTEGER NOT NULL
);
ALTER TABLE MSTB_COMMENT ADD CONSTRAINT MSTB_COMMENT_PK PRIMARY KEY ( idComment ) ;

CREATE TABLE MSTB_GENDER
(
    idGender      INTEGER NOT NULL ,
    gender        VARCHAR2 (20 CHAR) NOT NULL
);
ALTER TABLE MSTB_GENDER ADD CONSTRAINT MSTB_GENDER_PK PRIMARY KEY ( idGender ) ;

CREATE TABLE MSTB_COUNTRY
(
    idCountry     INTEGER NOT NULL ,
    country       VARCHAR2 (50 CHAR) NOT NULL
);
ALTER TABLE MSTB_COUNTRY ADD CONSTRAINT MSTB_COUNTRY_PK PRIMARY KEY ( idCountry ) ;

CREATE TABLE MSTB_GROUP
(
    idGroup      INTEGER NOT NULL ,
    name         VARCHAR2 (40 CHAR) NOT NULL ,
    
```

```
gender          VARCHAR2 (60 CHAR) NOT NULL ,
originDate     VARCHAR2 (10 CHAR) NOT NULL ,
additionalInfo VARCHAR2 (200 CHAR)
) ;
COMMENT ON COLUMN MSTB_GROUP.idGroup
IS
'Id del Grupo/Solo musical' ;
COMMENT ON COLUMN MSTB_GROUP.name
IS
'Nombre del grupo' ;
COMMENT ON COLUMN MSTB_GROUP.gender
IS
'Generos de la música del grupo' ;
COMMENT ON COLUMN MSTB_GROUP.originDate
IS
'Fecha de fundación' ;
ALTER TABLE MSTB_GROUP ADD CONSTRAINT MSTB_GROUP_PK PRIMARY KEY ( idGroup ) ;
ALTER TABLE MSTB_GROUP RENAME COLUMN gender to IDGENDER;
ALTER TABLE MSTB_GROUP MODIFY IDGENDER NUMBER;
ALTER TABLE MSTB_GROUP ADD IDCOUNTRY NUMBER;
ALTER TABLE MSTB_GROUP MODIFY originDate NUMBER;
ALTER TABLE MSTB_GROUP ADD codGroup VARCHAR2(200 CHAR) NOT NULL;
ALTER TABLE MSTB_GROUP ADD CONSTRAINT MSTB_GROUP_UN UNIQUE ( codGroup ) ;

CREATE TABLE MSTB_NSONG
(
idNSong      INTEGER NOT NULL ,
codSong      VARCHAR2 (50 CHAR) NOT NULL ,
title        VARCHAR2 (50 CHAR) NOT NULL ,
instrument   VARCHAR2 (50 CHAR) NOT NULL ,
audio        VARCHAR2 (50 CHAR) ,
video        VARCHAR2 (50 CHAR) ,
idUser       INTEGER NOT NULL
) ;
ALTER TABLE MSTB_NSONG ADD CONSTRAINT MSTB_NSONG_PK PRIMARY KEY ( idNSong ) ;
ALTER TABLE MSTB_NSONG ADD CONSTRAINT MSTB_CODSONG_UN UNIQUE ( codSong ) ;

CREATE TABLE MSTB_PROFILEGALLERY
(
idImage      INTEGER NOT NULL ,
image        VARCHAR2 (500 CHAR)
) ;
ALTER TABLE MSTB_PROFILEGALLERY ADD CONSTRAINT MSTB_PROFILEGALLERY_PK PRIMARY KEY (
idImage ) ;

CREATE TABLE MSTB_ROLUSER
(
idRol        INTEGER NOT NULL ,
kindUser     VARCHAR2 (20 CHAR) NOT NULL
) ;
ALTER TABLE MSTB_ROLUSER ADD CONSTRAINT MSTB_ROLUSER_PK PRIMARY KEY ( idRol ) ;

CREATE TABLE MSTB_SECQUESTION
(
idSecQuestion INTEGER NOT NULL ,
question      VARCHAR2 (60 CHAR) NOT NULL
) ;
ALTER TABLE MSTB_SECQUESTION ADD CONSTRAINT MSTB_SECQUESTION_PK PRIMARY KEY (
idSecQuestion ) ;

CREATE TABLE MSTB_SONG
(
```

```

create or replace PACKAGE MSPKG_GENDER AS

    TYPE cursorRetorno IS REF CURSOR;

    /*
        Script: LIST_GENDERS
        Description: Procedimiento que devuelve el Listado de los géneros
    */
    PROCEDURE LIST_GENDERS (
        cGENDER OUT cursorRetorno);

END MSPKG_GENDER;
/
create or replace PACKAGE BODY MSPKG_GENDER AS

    /*
        Script: LIST_GENDERS
        Description: Procedimiento que devuelve el Listado de los géneros
    */
    PROCEDURE LIST_GENDERS (
        cGENDER OUT cursorRetorno)
    IS
    BEGIN
        OPEN cGENDER FOR
            SELECT *
            FROM MSTB_GENDER;
        EXCEPTION
        WHEN OTHERS THEN
            IF cGENDER%ISOPEN THEN
                CLOSE cGENDER;
            END IF;
            RAISE;
    END LIST_GENDERS;

END MSPKG_GENDER;

```

8.3. Requisitos Funcionales y No Funcionales

8.3.1. Requisitos Funcionales

8.3.1.1. Intranet

RF-01 ~ Login (Intranet)			
Tipo	Funcional	Prioridad	Baja
Código	RF-01	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Login Usuarios (Intranet)		
Descripción Requisito	Para entrar a la aplicación de Intranet se tendrá que acceder a utilizando el usuario y contraseña de un usuario registrado con el rol de Administrador. El usuario podrá realizar el login introduciendo en el formulario el Nick o el email y la contraseña.		

RF-02 ~ Logout (Intranet)			
Tipo	Funcional	Prioridad	Baja
Código	RF-02	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Logout (Intranet)		
Descripción Requisito	Habrá un botón en el menú principal que permitirá al usuario cerrar la sesión.		

RF-03 ~ Recuperar Datos Usuarios (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-03	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Recuperar Datos Usuarios (Intranet)		
Descripción Requisito	Si el usuario olvida su contraseña podrá recuperarla con su email. El usuario deberá Introducir su correo de electrónico y se le enviará al correo un enlace con una dirección para poder restablecer la contraseña. Esta dirección tendrá un único uso.		

RF-04 ~ Envío de Correos (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-04	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Envío de Correos (Intranet)		
Descripción Requisito	Se enviará correos a los usuarios.		

RF-05 ~ Estructura Páginas (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-05	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Estructura Páginas (Intranet)		
Descripción Requisito	Las diferentes páginas de la intranet tendrán el siguiente formato: <ul style="list-style-type: none"> • Cabecera • Contenido • Pie de Página 		

RF-06 ~ Estructura Cabecera (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-06	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Estructura Cabecera (Intranet)		
Descripción Requisito	La cabecera se encontrará formada por: <ul style="list-style-type: none"> • Logo de la Emisora • Nombre de la emisora • Otros datos (Nick usuario loggeado, Cerrar Sesión) • Menú Aplicación 		

RF-07 ~ Estructura Pie de Página (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-07	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Estructura Pie de Página (Intranet)		
Descripción Requisito	El pie de página contendrá la siguiente información: <ul style="list-style-type: none"> • Nombre de la Compañía, es decir, Strixe S.L. • Email Compañía 		

- | | |
|--|-----------------------|
| | • Logo de la Compañía |
|--|-----------------------|
- Logo de la Compañía

RF-08 ~ Estructura Contenido (Intranet)

Tipo	Funcional	Prioridad	Alta
Código	RF-08	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Estructura Contenido (Intranet)		
Descripción Requisito	<p>Se han definido las diferentes partes de la que va a estar formada la intranet. Los apartados que va a contener son los siguientes:</p> <ul style="list-style-type: none"> • Sección Canciones • Sección Álbumes • Sección Artistas • Sección Usuarios 		

RF-09 ~ Estructura Sección Canciones (Intranet)

Tipo	Funcional	Prioridad	Alta
Código	RF-09	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Estructura Sección Canciones (Intranet)		
Descripción Requisito	<p>Se han definido los diferentes apartados de los que va a estar formados la Sección Canción. Los apartados serán los siguientes:</p> <ul style="list-style-type: none"> • Añadir Canción • Modificar Canción • Eliminar Canción • Listado de Canciones • Filtro de Canciones • Exportar Información 		

RF-10 ~ Estructura Sección Álbum (Intranet)

Tipo	Funcional	Prioridad	Alta
Código	RF-10	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Estructura Sección Álbum (Intranet)		
Descripción Requisito	<p>Se han definido los diferentes apartados de los que va a estar formados la Sección Álbum. Los apartados serán los siguientes:</p> <ul style="list-style-type: none"> • Añadir Álbum • Modificar Álbum • Eliminar Álbum • Listado de Álbumes • Filtro de Álbumes • Exportar Información 		

RF-11 ~ Estructura Sección Artista (Intranet)

Tipo	Funcional	Prioridad	Alta
Código	RF-11	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Estructura Sección Artista (Intranet)		
Descripción Requisito	<p>Se han definido los diferentes apartados de los que va a estar formados la Sección Artista. Los apartados serán los siguientes:</p> <ul style="list-style-type: none"> • Añadir Artista 		

	<ul style="list-style-type: none"> • Modificar Artista • Eliminar Artista • Listado de Artistas • Filtro de Artistas • Exportar Información
--	--

RF-12 ~ Estructura Sección Usuarios (Intranet)

Tipo	Funcional	Prioridad	Alta
Código	RF-12	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Estructura Sección Usuarios (Intranet)		
Descripción Requisito	<p>Se han definido los diferentes apartados en los que va a estar divididos la sección usuarios. Los apartados serán los siguientes:</p> <ul style="list-style-type: none"> • Añadir Usuario (RF-03) • Eliminar Usuario • Modificar Usuario • Listar Usuarios • Filtro de Usuarios • Exportar Información 		

RF-13 ~ Añadir Canción (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-13	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Añadir Canción (Intranet)		
Descripción Requisito	<p>Se permitirá al usuario añadir canciones al catálogo rellenando un formulario. Se van a describir los datos que van a ir asociados a una canción. Los datos van a ser los siguientes:</p> <ul style="list-style-type: none"> • idSong: id de la Canción. (Obligatorio) • codSong: Código único de la canción. (Obligatorio) • title: Título de la canción. (Obligatorio) • idGroup: id del Artista. (Obligatorio) • idAlbum: id del Álbum al que pertenece. • period: Epoca. (Obligatorio) • idGender: id del Género de la Canción. (Obligatorio) • audio: path mp3 de la canción. • video: path video de la canción. 		

RF-14 ~ Editar Canción (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-14	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Editar Canción (Intranet)		
Descripción Requisito	Se permitirá al usuario administrador modificar los datos de la canción seleccionada en el listado.		

RF-15 ~ Eliminar Canción (Intranet)

Tipo	Funcional	Prioridad	Baja
-------------	-----------	------------------	------

Código	RF-15	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Eliminar Canción (Intranet)		
Descripción Requisito	Se permitirá al usuario eliminar canciones del catálogo.		

RF-16 ~ Listado Canciones (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-16	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Listado Catálogo (Intranet)		
Descripción Requisito	Se mostrará un listado de las canciones que hay en el catálogo.		

RF-17 ~ Filtrar Canciones (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-17	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Filtrar Catálogo (Intranet)		
Descripción Requisito	En el listado se permitirá al usuario administrador filtrar por los campos de título, artista, periodo y género. De esta forma se facilitará y optimizará su trabajo para la emisora.		

RF-18 ~ Exportar Listado Canciones (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-18	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Exportar Listado Canciones (Intranet)		
Descripción Requisito	Los usuarios podrán exportar el listado de canciones a .xlsx, .csv y .txt. (Se intentará a pdf)		

RF-19 ~ Añadir Álbum (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-19	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Añadir Álbum (Intranet)		
Descripción Requisito	Se permitirá al usuario añadir álbumes de canciones al catálogo rellenando un formulario. Se van a describir los datos que van a ir asociados a un Álbum. Los datos van a ser los siguientes:		<ul style="list-style-type: none"> • idAlbum: id del Álbum. (Obligatorio) • codAlbum: Código único del Álbum. (Obligatorio) • name: Nombre del Álbum. (Obligatorio) • idGroup: id del Artista. (Obligatorio) • releaseDate: Año de Salida. (Obligatorio) • idGender: id del Género del Álbum. (Obligatorio) • cover: Path de la portada del Álbum

RF-20 ~ Editar Álbum (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-20	Aplicación	Intranet

Etapa Implementación	1º Sprint		
Nombre Requisito	Editar Álbum (Intranet)		
Descripción Requisito	Se permitirá al usuario administrador modificar los datos de la Álbum seleccionada en el listado.		

RF-21 ~ Eliminar Álbum (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-21	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Eliminar Canción (Intranet)		
Descripción Requisito	Se permitirá al usuario eliminar Álbunes del catálogo desde el listado de Álbunes.		

RF-22 ~ Listado Álbunes (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-22	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Listado Álbunes (Intranet)		
Descripción Requisito	Se mostrará un listado de las Álbunes que hay en el catálogo.		

RF-23 ~ Filtrar Álbunes (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-23	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Filtrar Álbunes (Intranet)		
Descripción Requisito	En el listado se permitirá al usuario administrador filtrar por los campos de nombre, artista y género. De esta forma se facilitará y optimizará su trabajo para la emisora.		

RF-24 ~ Exportar Listado Álbum (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-24	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Exportar Listado Álbumes (Intranet)		
Descripción Requisito	Los usuarios podrán exportar el listado de Álbumes a .xlsx, .csv y .txt. (Se intentará a pdf)		

RF-25 ~ Añadir Artista (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-25	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Añadir Artista (Intranet)		
Descripción Requisito	Se permitirá al usuario administrador añadir canciones al catálogo rellenando un formulario. Se van a describir los datos que van a ir asociados a una canción. Los datos van a ser los siguientes: <ul style="list-style-type: none"> • idGroup: id de la Canción. (Obligatorio) • codGroup: Código único del Artista. (Obligatorio) 		

	<ul style="list-style-type: none">• name: Título de la canción. (Obligatorio)• origindate: id del Artista. (Obligatorio)• idCountry: id del Álbum al que pertenece.• idGender: id del Género de la Canción. (Obligatorio)
--	--

RF-26 ~ Editar Artista (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-26	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Editar Artista (Intranet)		
Descripción Requisito	Se permitirá al usuario administrador modificar los datos de la canción seleccionada en el listado.		

RF-27 ~ Eliminar Artista (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-27	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Eliminar Artista (Intranet)		
Descripción Requisito	Se permitirá al usuario eliminar canciones del catálogo desde el listado de canciones.		

RF-28 ~ Listado Artista (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-28	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Listado Artista (Intranet)		
Descripción Requisito	Se mostrará un listado de las canciones que hay en el catálogo.		

RF-29 ~ Filtrar Artista (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-29	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Filtrar Artista (Intranet)		
Descripción Requisito	En el listado se permitirá al usuario administrador filtrar por los campos de nombre, género. De esta forma se facilitará y optimizará su trabajo para la emisora.		

RF-30 ~ Exportar Listado Artista (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-30	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Exportar Listado Artistas (Intranet)		
Descripción Requisito	Los usuarios podrán exportar el listado de Artista a .xlsx, .csv y .txt. (Se intentará a pdf)		

RF-31 ~ Dar de Alta Usuarios (Intranet)

Tipo	Funcional	Prioridad	Alta
Código	RF-31	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Dar de Alta Usuarios (Intranet)		
Descripción Requisito	<p>Se permitirá a los usuarios administradores poder dar de Alta desde la Intranet a nuevos usuarios para las Aplicaciones del Proyecto MusicSelector. Estos usuarios tendrán los siguientes datos:</p> <ul style="list-style-type: none"> • idUser: id del usuario. (Obligatorio) • nick: Nick del usuario. (Obligatorio) • password: Contraseña del usuario. (Obligatorio) • idSecurityQuestion: Id de la pregunta de seguridad. (Obligatorio) • securityAnswer: Respuesta de la pregunta de seguridad. (Obligatorio) • name: Nombre del usuario. (Obligatorio) • surname: Apellidos del usuario. (Obligatorio) • email: Email del usuario. (Obligatorio) • profileImage: Imagen de perfil del usuario. • idRol: Rol que tiene el Usuario (Obligatorio) 		

RF-32 ~ Roles

Tipo	Funcional	Prioridad	Baja
Código	RF-32	Aplicación	-
Etapa Implementación	1º Sprint		
Nombre Requisito	Roles		
Descripción Requisito	<p>En la aplicación va a haber definidos 2 Roles para los usuarios.</p> <ul style="list-style-type: none"> • Rol ADMIN (Administrador), este rol es para aquellos usuarios que van a estar a cargo de la aplicación. • Rol FOLLOWER, este rol es para aquellos usuarios que se suscriben a la aplicación 		

RF-33 ~ Modificar Usuario (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-33	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Modificar Usuario (Intranet)		
Descripción Requisito	Los usuarios administradores podrán modificar los datos de los usuarios.		

RF-34 ~ Eliminar Usuario (Intranet)

Tipo	Funcional	Prioridad	Baja
Código	RF-34	Aplicación	Intranet
Etapa Implementación	1º Sprint		
Nombre Requisito	Eliminar Usuario (Intranet)		
Descripción Requisito	Los usuarios podrán modificar los datos de su perfil.		

RF-35 ~ Listado Usuario (Intranet)

Nombre Requisito	Eliminar Comentario (WebApp)
Descripción Requisito	Los Usuarios administradores podrán eliminar comentarios

RF-63 ~ Contacto (WebApp)

Tipo	Funcional	Prioridad	Baja
Código	RF-63	Aplicación	WebApp
Etapa Implementación	2º Sprint		
Nombre Requisito	Contacto (WebApp)		
Descripción Requisito	Los usuarios podrán enviar información sobre nuevo contenido a añadir a la web o sugerencia y dudas para/sobre la misma.		

8.3.2. Requisitos No Funcionales

Requisito: RNF-01 ~ Lenguaje Programación Base

Tipo	No Funcional	Prioridad	Alta
Código	RNF-01	Aplicación	-
Fecha Impl.	-	Fecha Límite Impl.	-
Nombre Requisito	Lenguaje Programación Base		
Descripción Requisito	La aplicación estará desarrollada en el lenguaje de Java por lo que al ser una aplicación Web se deberá utilizar el Framework de Spring.		

Requisito: RNF-02 ~ Usabilidad

Tipo	No Funcional	Prioridad	Alta
Código	RNF-02	Aplicación	-
Fecha Impl.	-	Fecha Límite Impl.	-
Nombre Requisito	Usabilidad		
Descripción Requisito	Las aplicaciones constarán de interfaces amigables e intuitivas para facilitar su uso a los usuarios.		

Requisito: RNF-03 ~ Seguridad

Tipo	No Funcional	Prioridad	Alta
Código	RNF-03	Aplicación	-
Fecha Impl.	-	Fecha Límite Impl.	-
Nombre Requisito	Seguridad		
Descripción Requisito	Se mejorará la seguridad de las aplicaciones cifrando las contraseñas. Se tendrá en cuenta la Ley de Protección de datos al guardar en la aplicación datos sensibles de los usuarios.		

Requisito: RNF-04 ~ Estructura de la Aplicación

Tipo	No Funcional	Prioridad	Alta
Código	RNF-04	Aplicación	-
Fecha Impl.	-	Fecha Límite Impl.	-
Nombre Requisito	Patrón de la Aplicación		
Descripción Requisito	Las aplicaciones se realizarán basándose en el patrón de MVC (Model, View, Controller).		

Requisito: RNF-05 ~ Estructura Código

Tipo	No Funcional	Prioridad	Alta
Código	RNF-05	Aplicación	-
Fecha Impl.	-	Fecha Límite Impl.	-
Nombre Requisito	Estructura Código		
Descripción Requisito	<p>Junto a las aplicaciones se desarrollará una documentación del código (JavaDoc). Las clases, métodos y resto de elementos deben tener adjuntada la siguiente información:</p> <ul style="list-style-type: none"> • Nombre • Descripción • Versión • Si es un método, además <ul style="list-style-type: none"> ○ Parámetros Entrada y Salida 		

Requisito: RNF-06 ~ Utilización de Base de Datos

Tipo	No Funcional	Prioridad	Alta
Código	RNF-06	Aplicación	-
Fecha Impl.	-	Fecha Límite Impl.	-
Nombre Requisito	Utilización de Base de Datos		
Descripción Requisito	<p>Las aplicaciones constarán de una base de datos relacional en la que se guardará tanto el contenido musical de la emisora como los datos personales de los usuarios registrados en la misma. La Base de Datos será Oracle para el desarrollo de la misma se utilizará la herramienta de SQLDeveloper</p>		

Requisito: RNF-07 ~ Compatibilidad

Tipo	No Funcional	Prioridad	Alta
Código	RNF-07	Aplicación	-
Fecha Impl.	-	Fecha Límite Impl.	-
Nombre Requisito	Compatibilidad		
Descripción Requisito	Las aplicaciones web se podrán utilizar en diferentes Navegadores.		

Requisito: RNF-08 ~ Facilidad de Mantenimiento

Tipo	No Funcional	Prioridad	Alta
Código	RNF-08	Aplicación	-
Fecha Impl.	-	Fecha Límite Impl.	-
Nombre Requisito	Facilidad de Mantenimiento		
Descripción Requisito	<p>La aplicación se desarrollará intentando que en el momento en el que se encuentren las fallas o se deseé implementar mejoras en la aplicación, los cambios necesarios para estas correcciones o mejoras se pueda realizar de una manera ágil y sencilla.</p>		

Requisito: RNF-09 ~ Fácil Conocimiento

Tipo	No Funcional	Prioridad	Alta
Código	RNF-09	Aplicación	-
Fecha Impl.	-	Fecha Límite Impl.	-
Nombre Requisito	Fácil Conocimiento		

Descripción Requisito	Al tener una interfaz intuitiva es más fácil para el usuario recordar los pasos a seguir para explotar todas las funcionalidades que permite realizar la aplicación.
------------------------------	--

Requisito: RNF-10 ~ Diversidad de Navegadores			
Tipo	No Funcional	Prioridad	Alta
Código	RNF-10	Aplicación	-
Fecha Impl.	-	Fecha Límite Impl.	-
Nombre Requisito	Diversidad de Navegadores		
Descripción Requisito	Las aplicaciones web podrán ser utilizadas desde los principales navegadores (Mozilla Firefox, Internet Explorer y Google Chrome). Sin embargo, solo se garantizará el correcto funcionamiento desde las últimas versiones de los mismos.		

8.4. Diagramas de Secuencia (Apartado 4.2.2.)

8.4.1.1. Intranet

8.4.1.1.1. Login (Intranet)

DS-01 ~ Login (Intranet)			
D. Secuencia:	Login (Intranet)	Código:	DS-01
Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este diagrama de secuencia se muestran los objetos que van a ser necesarios y las acciones que van a realizar los mismos en un orden cronológico para la realización del Login en la Intranet del Proyecto MusicSelector		
Listado Objetos: de	<ul style="list-style-type: none"> • Usuario (cliente que se conecta a nuestra aplicación) • Login.jsp (Interfaz del Login) • LoginController.java (Clase encargada del control del Login) • UserDAO (DAO en la que se encuentran los métodos que afectan a la clase UserDTO) • Main.jsp (Interfaz de la Pantalla Principal) 		
Proceso:	Al conectarse a la dirección de la Intranet a los usuarios les aparecerá un formulario para poder loggearse. Desde esta pantalla no se permitirá el registro de los usuarios ya que esta aplicación será solo para uso interno de la web. Una vez introducidos los datos en la pantalla de logeo se comprobará que coincidan con los de algún usuario administrador ya registrado. Si esto es así, se permitirá el acceso, si no, se mostrará un mensaje de error indicando que los datos no son correctos.		

8.4.1.1.2. Recuperar Contraseña (Intranet)

DS-02 ~ Recuperar Contraseña (Intranet)			
D. Secuencia:	Login (Intranet)	Código:	DS-02

Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este diagrama de secuencia se muestran los objetos que van a ser necesarios y las acciones que van a realizar los mismos en orden cronológico para la realizar el proceso de Recuperar Contraseña en la Intranet del Proyecto MusicSelector		
Listado de Objetos:	<ul style="list-style-type: none"> • Usuario (cliente que se conecta a nuestra aplicación) • login.jsp (Interfaz del Login) • emailResetPass.jsp (Interfaz de Enviar Pass correo=) • SendEmail.java • resetPass.jsp • LoginController.java (Clase encargada del control del Login) • UserDAO (DAO en la que se encuentran los métodos que afectan a la clase UserDTO) 		
Proceso:	<p>El usuario se encuentra en la pantalla de Login. Para comenzar el proceso de recuperar contraseña el Usuario tiene que pulsar sobre el botón “Recuperar Contraseña”. Una vez pulsado este botón se le redirigirá a la pantalla de emailResetPass.jsp donde el usuario tendrá que introducir un email. Al darle al botón “Enviar” le llegará un mensaje al correo electrónico con una URL, de un único uso, que lo llevará a la pantalla resetPass.jsp. En esta pantalla se le pedirá al usuario que introduzca una contraseña. Cuando haya escrito la nueva contraseña y le dé a “guardar” el controlador recogerá la contraseña y actualizará los datos del usuario en la base de datos. Al terminar el proceso el usuario será redirigido al Login. Si el usuario intenta volver a acceder a la url que se le ha enviado al correo se le redirigirá al login mostrándole un mensaje de error.</p>		

8.4.1.1.3. Dar de Alta Usuarios (Intranet)

DS-03 ~ Alta Usuario (Intranet)			
D. Secuencia:	Alta Usuario (Intranet)	Código:	DS-03
Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este diagrama de secuencia se muestran los objetos que van a ser necesarios y las acciones que van a realizar los mismos en un orden cronológico para dar de alta un Usuario desde la Intranet del Proyecto MusicSelector		
Listado de Objetos:	<ul style="list-style-type: none"> • Usuario (cliente que se conecta a nuestra aplicación) • Registre.jsp (Interfaz del Login) • RegistreController.java (Clase encargada del control del Login) • UserDAO (DAO en la que se encuentran los métodos que afectan a la clase UserDTO) • Main.jsp (Interfaz de la Pantalla Principal) 		

Proceso:	El usuario introduce los datos en el formulario de Usuario (“Registre.jsp”). Una vez pulse el botón “guardar” se mandarán los datos al Controlador (“RegistreController.java”) donde se comprobarán que los datos introducidos sean correctos. Además, se encarga de conectar con la BBDD (“UserDAO”) para saber si existe algún usuario con el Nick o el Email introducido. Si al validar los datos alguno de estos es erróneo se devuelve al formulario mostrando los mensajes de error consecuentes. Si los datos introducidos son correctos se da de alta al usuario en la aplicación y se devuelve al usuario al Inicio. (“Main.jsp”)
-----------------	--

8.4.1.1.4. Modificar Usuario (Intranet)

DS-04 ~ Modificar Usuario (Intranet)			
D. Secuencia:	Modificar Usuario (Intranet)	Código:	DS-04
Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este diagrama de secuencia se muestran los objetos que van a ser necesarios y las acciones que van a realizar los mismos en un orden cronológico para poder Modificar los Datos de un Usuario desde la Intranet del Proyecto MusicSelector		
Listado de Objetos:	<ul style="list-style-type: none"> • Usuario (cliente que se conecta a nuestra aplicación) • listUser.jsp (Interfaz del Listado Usuario) • UserController.java (Clase encargada del control del Login) • addEditUser.jsp (Interfaz Formulario Usuario) • UserDAO (DAO en la que se encuentran los métodos que afectan a la clase UserDTO) 		
Proceso:	El usuario administrador se encuentra en el listado de Usuarios. Aquí pulsa sobre el botón “Editar” de la fila del usuario que quiere modificar. Una vez pulsado este botón accedemos al controlador que recoge los datos del Usuario seleccionado en un objeto UserDTO y nos redirige al formulario de Usuario cargando los datos del mismo. Una vez el Usuario termine de realizar los cambios pulsará sobre el botón “Guardar”. Esto hará que los datos sean recogidos en el controlador y validados. Si se han introducido datos incorrectos se mostrará un mensaje de error; si no, se actualizarán los datos del Usuario. El usuario será devuelto al Listado de Usuarios tras realizar el proceso.		

8.4.1.1.5. Eliminar Usuarios (Intranet)

DS-05 ~ Eliminar Usuario (Intranet)			
D. Secuencia:	Eliminar Usuario (Intranet)	Código:	DS-05

Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este diagrama de secuencia se muestran los objetos que van a ser necesarios y las acciones que van a realizar los mismos en un orden cronológico para Eliminar un Usuario desde la Intranet del Proyecto MusicSelector		
Listado de Objetos:	<ul style="list-style-type: none"> • Usuario (cliente que se conecta a nuestra aplicación) • listUser.jsp (Interfaz del Listado Usuario) • UserController.java (Clase encargada del control del Login) • UserDAO (DAO en la que se encuentran los métodos que afectan a la clase UserDTO) 		
Proceso:	<p>El usuario administrador se encuentra en el listado de Usuarios. Aquí pulsa sobre el botón “Eliminar” de la fila del usuario que quiere eliminar. Una vez pulsado este botón accedemos al controlador que recogerá la id del usuario a eliminar para así eliminarlo desde la BBDD. El usuario será devuelto al Listado de Usuarios tras realizar el proceso.</p>		

8.4.1.1.6. Filtra Usuarios (Intranet)

DS-06 ~ Filtrar Usuarios (Intranet)

D. Secuencia:	Filtrar Usuarios (Intranet)	Código:	DS-06
Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este diagrama de secuencia se muestran los objetos que van a ser necesarios y las acciones que van a realizar los mismos en un orden cronológico para Filtrar Usuarios en la Intranet del Proyecto MusicSelector		
Listado de Objetos:	<ul style="list-style-type: none"> • Usuario (cliente que se conecta a nuestra aplicación) • listUser.jsp (Interfaz del Listado Usuario) • UserController.java (Clase encargada del control del Login) • UserDAO (DAO en la que se encuentran los métodos que afectan a la clase UserDTO) 		
Proceso:	<p>El usuario administrador se encuentra en el listado de Usuarios. Sobre el listado de Usuario hay un buscador donde el Usuario puede introducir información sobre el usuario que quiere buscar. Una vez haya introducido los datos y le haya dado a “buscar” desde el controlador se recogerán esos datos. Una vez obtenidos los datos se realiza una consulta a BBDD obteniendo un listado de usuarios que cumplen las condiciones expuestas por el usuario. El controlador obtiene este listado y lo envía a la interfaz donde se visualizan los datos.</p>		

8.4.1.1.7. Añadir Canciones (Intranet)

DS-07 ~ Añadir Canciones (Intranet)

D. Secuencia:	Añadir Canciones (Intranet)	Código:	DS-07
Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este diagrama de secuencia se muestran los objetos que van a ser necesarios y las acciones que van a realizar los mismos en un orden cronológico para poder Añadir una Canción al Catálogo Musical en la Intranet del Proyecto MusicSelector		
Listado de Objetos:	<ul style="list-style-type: none"> • Usuario (cliente que se conecta a nuestra aplicación) • listSong.jsp (Interfaz del Listado de Canciones) • SongController.java (Clase encargada del control de Canciones) • SongDAO (DAO en la que se encuentran los métodos que afectan a la clase SongDTO) • addEditSong.jsp (Interfaz del Formulario de Canciones) 		
Proceso:	<p>El usuario se encuentra en el listado de canciones. El usuario pulsa sobre el botón “añadir”. Una vez pulsado este botón el controlador nos redirige al formulario de Canciones. Una vez que el usuario haya llenado los datos de la canción pulsa sobre el botón “guardar”. Al pulsar “guardar”, los datos de la canción son recogidos y validados en el controlador, además, se comprueba que la canción no esté ya introducida en la aplicación. Si los datos son correctos se añade la canción al catálogo y se devuelve al usuario al listado de canciones. Si se produce algún error se devuelve al formulario de canción indicando los errores ocurridos.</p>		

8.4.1.1.8. Modificar Canciones (Intranet)

DS-08 ~ Modificar Canciones (Intranet)			
D. Secuencia:	Modificar Canciones (Intranet)	Código:	DS-08
Implementación:	Sprint 1	Aplicación:	Intranet
Descripción Corta:	En este diagrama de secuencia se muestran los objetos que van a ser necesarios y las acciones que van a realizar los mismos en un orden cronológico para poder Modificar la información de las canciones del catálogo del Proyecto MusicSelector		
Listado de Objetos:	<ul style="list-style-type: none"> • Usuario (cliente que se conecta a nuestra aplicación) • songList.jsp (Interfaz del Listado de Canciones) • SongController.java (Clase encargada del control de Canciones) • SongDAO (DAO en la que se encuentran los métodos que afectan a la clase SongDTO) • addEditSong.jsp (Interfaz Formulario Canciones) 		
Proceso:	<p>El usuario administrador se encuentra en el listado de Canciones. Aquí pulsa sobre el botón “Editar” de la fila de la canción que quiere modificar. Una vez pulsado este botón accedemos al controlador que recoge los datos de la Canción seleccionado en un objeto SongDTO y nos redirige al formulario “Canción” cargando los datos del objeto. Una vez el Usuario termine de realizar los cambios pulsará sobre el botón</p>		

8.5. Pruebas

8.5.1. Pruebas Funcionales

8.5.1.1. Pruebas para la Intranet

8.5.1.1.1. Prueba RF-02 Login (Intranet)

Prueba RF-02 Login (Intranet)	
Precondición	Desde la pantalla de Login pulsamos el botón de Login
Datos de Entrada	Se rellenan los datos del formulario de Loggeo
Datos de Salida	<ol style="list-style-type: none">Si se rellena el formulario con los datos de un usuario registrado que tiene rol Admin se da paso a la aplicación. OKSi se rellena el formulario con los datos de un usuario registrado que tiene rol Follower se muestra mensaje de error. OKSi se rellena el formulario con los datos de un usuario no registrado se muestra mensaje de error. OKSi no se rellenan todos los campos del formulario se muestra mensaje de error. OK

8.5.1.1.2. Prueba RF-03 Recuperar Datos Usuarios (Intranet)

Prueba RF-03 Recuperar Datos Usuarios (Intranet)	
Precondición	Desde la pantalla de Login pulsamos el botón de Recuperar Contraseña
Datos de Entrada	Se rellenan los datos del formulario de Recuperar Contraseña
Datos de Salida	<ol style="list-style-type: none">Si se rellena el formulario con el email de un usuario registrado se envía un correo con una url para poder establecer una nueva contraseña. OKSi se rellena el formulario con el email de un usuario no registrado se muestra un mensaje de error. OKSi no se rellena el formulario se muestra un mensaje de error. OKSi se accede a la url enviada al correo se envía a la pantalla de regenerar contraseña. OKSi se intenta acceder más de una vez a la url enviada al correo se muestra mensaje de error. OKSi se rellena el formulario de regenerar contraseña con una contraseña correcta se actualiza la contraseña del usuario. OKSi se rellena el formulario de regenerar contraseña con una contraseña incorrecta se muestra mensaje de error. OKSi no se rellena el formulario de regenerar contraseña se muestra mensaje de error. OK

8.5.1.1.3. Prueba RF-13 Añadir Canción (Intranet)

Prueba RF-13 Añadir Canción (Intranet)	
Precondición	Se pulsa el botón Añadir desde el Listado de Canciones
Datos de Entrada	Se rellenan los datos del formulario de Canción
Datos de Salida	<ol style="list-style-type: none"> La pantalla debe cargar los formularios de Géneros y Artistas. OK Al seleccionar un Artista y darle a buscar álbumes la pantalla cargará el desplegable de Álbumes con los Álbumes de ese Artista. OK Si ya existe una canción de un artista con el mismo título se muestra un mensaje de error. OK Si no se rellenan el campo de título se muestra un mensaje de error. OK En el caso de que la validación sea correcta, se inserta la nueva canción en la tabla MSTB_SONG y se vuelve al listado de canciones mostrando un mensaje de que se ha insertado correctamente. OK

8.5.1.1.4. Prueba RF-14 Editar Canción (Intranet)

Prueba RF-14 Editar Canción (Intranet)	
Precondición	Desde el listado de Canciones se pulsa sobre el botón Editar de alguna de las canciones
Datos de Entrada	Se modifican los valores en los campos título, género, artista, álbum, fecha, audio y vídeo.
Datos de Salida	<ol style="list-style-type: none"> En la pantalla se deben llenar los campos con los datos de la canción seleccionada. OK Si ya existe una canción de un artista con el mismo título se muestra un mensaje de error. OK Si no se rellenan el campo de título se muestra un mensaje de error. OK En el caso de que la validación sea correcta, se actualiza la canción en la tabla MSTB_SONG y se vuelve al listado de canciones mostrando un mensaje de que se ha actualizado correctamente. OK

8.5.1.1.5. Prueba RF-15 Eliminar Canción (Intranet)

Prueba RF-15 Eliminar Canción (Intranet)	
Precondición	Desde el listado de Canciones se pulsa sobre el botón Eliminar de alguna de las canciones
Datos de Entrada	
Datos de Salida	<ol style="list-style-type: none"> Se muestra un mensaje de confirmación antes de borrar la canción. OK Si se borra correctamente se indica que se ha borrado. OK

- | | |
|--|--|
| | <ol style="list-style-type: none">3. Si se produce un error al intentar borrar se muestra un mensaje de error. OK |
|--|--|

8.5.1.1.6. Prueba RF-17 Filtro Canciones (Intranet)

Prueba RF-17 Filtro Canciones (Intranet)

Precondición	Se pulsa el botón Buscar del Listado de Canciones
Datos de Entrada	Se rellenan los datos del formulario del Lista
Datos de Salida	<ol style="list-style-type: none">1. Si se cargan el listado se cargan correctamente los desplegables. OK2. Se realiza correctamente el Filtrado de las canciones con los datos llenados en el formulario. OK

8.5.1.1.7. Prueba RF-19 Añadir Álbum (Intranet)

Prueba RF-19 Añadir Álbum (Intranet)

Precondición	Se pulsa el botón Añadir desde el Listado de Álbumes
Datos de Entrada	Se rellenan los datos del formulario de Canción
Datos de Salida	<ol style="list-style-type: none">1. La pantalla debe cargar los formularios de Géneros y Artistas. OK2. Si ya existe un Álbum de un artista con el mismo nombre se muestra un mensaje de error. OK3. Si no se rellenan el campo de título, año de salida se muestra un mensaje de error. OK4. En el caso de que la validación sea correcta, se inserta el nuevo Álbum en la tabla MSTB_ALBUM y se vuelve al listado de Álbumes mostrando un mensaje de que se ha insertado correctamente. OK

8.5.1.1.8. Prueba RF-20 Editar Álbum (Intranet)

Prueba RF-20 Editar Álbum (Intranet)

Precondición	Desde el listado de Álbumes se pulsa sobre el botón Editar de alguna de las canciones
Datos de Entrada	Se modifican los valores en los campos nombre, género, artista, fecha salida, cover.
Datos de Salida	<ol style="list-style-type: none">1. La pantalla debe tener rellenados los campos con los datos del Álbum seleccionado. OK2. Si ya existe un Álbum de un artista con el mismo nombre se muestra un mensaje de error. OK

- | | |
|--|---|
| | <ol style="list-style-type: none"> 3. Si no se rellenan el campo de nombre se muestra un mensaje de error. OK 4. En el caso de que la validación sea correcta, se actualiza el Álbum en la tabla MSTB_ALBUM y se vuelve al listado de Álbumes mostrando un mensaje de que se ha actualizado correctamente. OK |
|--|---|

8.5.1.1.9. Prueba RF-21 Eliminar Álbum (Intranet)

Prueba RF-21 Eliminar Álbum (Intranet)	
Precondición	Desde el listado de Álbumes se pulsa sobre el botón Eliminar de alguna de los Álbumes
Datos de Entrada	
Datos de Salida	<ol style="list-style-type: none"> 1. Se muestra un mensaje de confirmación antes de borrar el Álbum. OK 2. Si se borra correctamente se indica que se ha borrado. OK 3. Si se produce un error al intentar borrar se muestra un mensaje de error. OK

8.5.1.1.10. Prueba RF-23 Filtro Álbumes (Intranet)

Prueba RF-23 Filtro Álbumes (Intranet)	
Precondición	Se pulsa el botón Buscar del Listado de Álbumes
Datos de Entrada	Se rellenan los datos del formulario del Lista
Datos de Salida	<ol style="list-style-type: none"> 1. Si se cargan el listado se cargan correctamente los desplegables. OK 2. Se realiza correctamente el Filtrado de los Álbumes con los datos rellenados en el formulario. OK

8.5.1.1.11. Prueba RF-25 Añadir Artista (Intranet)

Prueba RF-25 Añadir Artista (Intranet)	
Precondición	Se pulsa el botón Añadir desde el Listado de Artistas
Datos de Entrada	Se rellenan los datos del formulario de Artista
Datos de Salida	<ol style="list-style-type: none"> 1. La pantalla debe cargar los formularios de Géneros y Artistas. OK 2. Si no se rellena nombre, se muestra un mensaje de error. OK 3. En el caso de que la validación sea correcta, se inserta el nuevo Artista en la tabla MSTB_GROUP y se vuelve al listado de Artistas mostrando un mensaje de que se ha insertado correctamente. OK

8.5.1.1.12. Prueba RF-26 Editar Artista (Intranet)

Prueba RF-26 Editar Artista (Intranet)	
Precondición	Desde el listado de Artistas se pulsa sobre el botón Editar de alguna de las canciones
Datos de Entrada	Se modifican los valores en los campos nombre, género, fecha origen.
Datos de Salida	<ol style="list-style-type: none"> La pantalla debe tener cumplimentados los campos con los datos del Artista seleccionado. OK Si no se rellenan el campo de nombre se muestra un mensaje de error. OK En el caso de que la validación sea correcta, se actualiza el Artista en la tabla MSTB_GROUP y se vuelve al listado de Artistas mostrando un mensaje de que se ha actualizado correctamente. OK

8.5.1.1.13. Prueba RF-27 Eliminar Artista (Intranet)

Prueba RF-27 Eliminar Artista (Intranet)	
Precondición	Desde el listado de Artistas se pulsa sobre el botón Eliminar de alguna de los Artistas
Datos de Entrada	
Datos de Salida	<ol style="list-style-type: none"> Se muestra un mensaje de confirmación antes de borrar el Artista. OK Si se borra correctamente se indica que se ha borrado. OK Si existen Álbumes o Canciones asociadas al Artista no se podrá eliminar. OK Si se produce un error al intentar borrar se muestra un mensaje de error. OK

8.5.1.1.14. Prueba RF-29 Filtro Artistas (Intranet)

Prueba RF-29 Filtro Artistas (Intranet)	
Precondición	Se pulsa el botón Buscar del Listado de Artistas
Datos de Entrada	Se rellenan los datos del formulario del Lista
Datos de Salida	<ol style="list-style-type: none"> Si se cargan el listado se cargan correctamente los desplegables. OK Se realiza correctamente el Filtrado de los Artistas con los datos rellenados en el formulario. OK

8.5.1.1.15. Prueba RF-31 Dar de Alta Usuario (Intranet)

Prueba RF-31 Dar de Alta Usuario (Intranet)	
Precondición	Se pulsa el botón Añadir desde el Listado de Usuarios