



**Proyecto UPBEAT**  
**Grupo 03. Barbara Liskov**

Plan de gestión, análisis y memoria del proyecto

Alejandro Ruiz Sumelzo  
Alejandro Piedrafitra Barrantes  
Álvaro Santamaría De la Fuente  
Fernando Navarro Zarralanga  
José Félix Yagüe Royo  
Víctor Pérez Sanmartín  
Sergio Torres Castillo



**Universidad**  
**Zaragoza**

24 de febrero de 2020

## Introducción

# Índice

	Página
<b>1. Organización del proyecto</b>	<b>4</b>
<b>2. Plan de gestión del proyecto</b>	<b>4</b>
2.1. Procesos . . . . .	4
2.1.1. Procesos de inicio del proyecto . . . . .	4
2.1.2. Procesos de ejecución y control del proyecto . . . . .	4
2.1.3. Procesos técnicos . . . . .	5
2.2. Planes . . . . .	6
2.2.1. Plan de gestión de configuraciones . . . . .	6
2.2.2. Plan de construcción y despliegue del software . . . . .	7
2.2.3. Plan de aseguramiento de la calidad . . . . .	7
2.2.4. Calendario del proyecto y división del trabajo . . . . .	8
<b>3. Análisis y diseño del sistema</b>	<b>11</b>
3.1. Análisis de requisitos . . . . .	11
3.2. Diseño del sistema . . . . .	11

## 1. Organización del proyecto

Integrante	Puesto	Responsabilidades
Alejandro Ruiz Sumelzo	Director del proyecto. Coordinador y desarrollador del grupo de back-end. Encargado de la documentación del análisis y diseño del sistema.	Responsable de redactar algunas actas en reuniones con el profesor. Control de la distribución de trabajo (elaboración de calendario) y revisión de esfuerzos. Desarrollador de modelos, repositorios y controladores de la API. Encargado del despliegue del back end sobre el servidor.
Alejandro Piedrafita Barrantes	Desarrollador de apoyo para el grupo de back-end	Realización de tareas de gestión (edición de memoria y otros documentos). Desarrollador de modelos, repositorios y controladores de la API. Diseño del sistema mediante diagramas.
Víctor Pérez Santmartín	Desarrollador de apoyo para el grupo de back-end	Responsable de redactar algunas actas en reuniones con el profesor. Realización de tareas de gestión (edición de memoria y otros documentos). Desarrollador de modelos, repositorios y controladores de la API. Encargado del diseño e implementación de la base de datos.

## 2. Plan de gestión del proyecto

### 2.1. Procesos

#### 2.1.1. Procesos de inicio del proyecto

#### 2.1.2. Procesos de ejecución y control del proyecto

El estándar a utilizar en el código, puesto que va a ser una aplicación web, se utilizará TypeScript. Los responsables de realizar la puesta en marcha serán los encargados de la parte Front-end y de la parte Back-end. La creación de copias de seguridad y semejantes se realizarían de manera automática gracias a GitHub. El repositorio que se creará con todos los archivos referentes al proyecto se encontrará en GitHub, para que todos los integrantes del proyecto puedan acceder fácilmente a los archivos. Además, se usará el Issue Tracker de GitHub para la gestión de incidencias. El proyecto estará dividido en varios repositorios: uno específico para Front-end, otro para Back-end, y la memoria. Para conseguir que no se modifique el mismo fichero por dos personas al mismo tiempo y evitar problemas, cada equipo tendrá más sub-ramas de desarrollo, por ejemplo, una

para cada miembro del equipo, que serán actualizadas con cambios no siempre funcionales y cuando sean más estables se volcarán a la rama de desarrollo principal. En la rama principal de cada uno de los repositorios sólo podrá haber una versión funcional del sistema, que antes de ser subida será sometida a diferentes test automáticos, entre los que se incluirán test para comprobar la estabilidad del sistema (pruebas de sobrecarga) y test que revisarán las acciones disponibles para comprobar los requisitos que se han resuelto. Para que lo desarrollado en cada uno de estos repositorios pase al repositorio funcional, cada líder de las respectivas partes revisará el código actualizado y si todo está correcto se considerará válido. Todos los componentes del equipo son capaces de modificar los ficheros de los repositorios excepto en el de las versiones, el cual solo podrán subir archivos y modificarlos los líderes del Front-End y el Back-End.

### **2.1.3. Procesos técnicos**

## 2.2. Planes

### 2.2.1. Plan de gestión de configuraciones

La convención de nombres utilizadas para nombrar los distintos archivos sería la siguiente:

NOMBRE	TIPO	VERSIÓN	REVISIÓN
--------	------	---------	----------

Las versiones solo se modificarán cada vez que se produzcan cambios suficientemente importantes, como por ejemplo la implementación de una nueva funcionalidad. Cada vez que se cree una nueva versión, pero sus cambios sean menores, como resolución de errores, se modificará su número de revisión, pero no de versión. Se crearán ficheros de documentación que permita ir recopilando toda la información referente a los cambios. Además, en los ficheros de documentación en los que se expliquen las diversas funcionalidades que tiene la aplicación y que errores se han ido resolviendo, cuando estos sean de una nueva versión o revisión solo se ofrecerá la información sobre los cambios que existan entre esta y la versión o revisión anterior, pero siempre que se cambie la versión se documentarán los cambios respecto a la primera revisión de la versión anterior (p.ej. La versión 2.1 solo contendrá las novedades respecto a la versión 2.0, pero la versión 3.0 contendrá todos los cambios que hayan sucedido desde la versión 2.0 aunque la mayoría se hayan documentado ya en las revisiones).

2.2.2. Plan de construcción y despliegue del software

2.2.3. Plan de aseguramiento de la calidad

#### 2.2.4. Calendario del proyecto y división del trabajo

En la primera iteración del proceso de diseño nos centraremos en desarrollar las funcionalidades principales del sistema, mientras que en la segunda iteración se corregirán todos los errores encontrados en la primera, se implementarán las funcionalidades secundarias y se afinará el diseño de la página web y de las aplicaciones móviles para que sean más agradables al usuario.

Para la primera iteración, se planea permitir la creación, edición y borrado de clientes con sus credenciales básicas: nombre de usuario, nombre real, correo, contraseña. Unido a esto, comprobar si las entidades Artista y Usuario se crean y borrar correctamente. También se permitirá la subida de canciones por parte de los artistas; estas canciones serán visibles en la aplicación y podrán ser reproducidas (al igual que los podcasts). Los álbumes estarán disponibles con su descripción y podrán ser consultados, reproduciendo cada una de sus canciones.

Para la segunda iteración se finalizarán los requisitos que, por falta de tiempo, no pudieron ser completados en la primera y se añadirán funcionalidades al sistema. Estas funcionalidades son: añadir canciones a la lista de reproducción de un usuario, permitir información adicional en los perfiles de usuario (como puede ser una foto de perfil, una descripción, etc.), además de poder seguirse entre dos usuarios. Se permitirá la búsqueda y filtrado de determinadas canciones y/o álbumes por unos determinados parámetros, al igual que utilizar un ecualizador en la aplicación web con el uso de *banners*.



# ABRIL 2020

SUNDAY	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY	SATURDAY
			1	2	Proyecto Upbeat 3 Back: lógica del login y registro, modificación del perfil.	Proyecto Upbeat 4 Web y móvil: vistas de login y registro, modificación del perfil.
5	Proyecto Upbeat 6 Primera versión de la web y la aplicación, que permita a un cliente registrarse, iniciar sesión y modificar el perfil.	7	8	9	Proyecto Upbeat 10 Back: lógica para reproducir una canción.	Proyecto Upbeat 11 Web y móvil: ver y reproducir una canción, pantalla de inicio.
12	Proyecto Upbeat 13 Pruebas conjuntas de las funcionalidades y corrección de errores.	14	Proyecto Upbeat 15 Entrega de la segunda versión de la memoria.	16	Proyecto Upbeat 17 Back: lógica para crear cantantes y subir canciones.	Proyecto Upbeat 18 Web y móvil: vistas para crear cantantes y subir canciones.
19	Proyecto Upbeat 20 Pruebas conjuntas de las funcionalidades y corrección de errores.	21	22	23	Proyecto Upbeat 24 Web y móvil: vistas para ver y escuchar un álbum con una o varias canciones, además de la información del mismo.	Proyecto Upbeat 25 Back: lógica de un álbum con una o varias canciones, además de la información del mismo.
26	Proyecto Upbeat 27 Pruebas conjuntas de las funcionalidades y corrección de errores.	28	29	Proyecto Upbeat 30 Despliegue de la lógica y BD en servidor en la nube.		

# MAYO 2020

SUNDAY	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY	SATURDAY
					Proyecto Upbeat 1 Web y móvil: vistas para crear listas de reproducción y canciones favoritas.	Proyecto Upbeat 2 Back: lógica de creación de las listas de reproducción y canciones favoritas.
3	Proyecto Upbeat 4 Pruebas conjuntas de las funcionalidades y corrección de errores.	5	6	Proyecto Upbeat 7 Despliegue de la lógica y BD en servidor en la nube.	Proyecto Upbeat 8 Web y móvil: vistas para seguir a otros usuarios y listas de reproducción.	Proyecto Upbeat 9 Back: lógica para seguir a otros usuarios y listas de reproducción.
10	Proyecto Upbeat 11 Pruebas conjuntas de las funcionalidades y corrección de errores.	12	13	Proyecto Upbeat 14 Despliegue de la lógica y BD en servidor en la nube.	Proyecto Upbeat 15 Web: equalizador y banners, aplicación final.	Proyecto Upbeat 16 Back: banners, lógica y BD final.
Proyecto Upbeat 17 Despliegue de la lógica y BD en servidor en la nube. Pruebas y depuración.	Proyecto Upbeat 18 Pruebas y depuración.	Proyecto Upbeat 19 Pruebas y depuración.	Proyecto Upbeat 20 Pruebas y depuración.	Proyecto Upbeat 21 Pruebas y depuración.	Proyecto Upbeat 22 Pruebas y depuración.	Proyecto Upbeat 23 Pruebas y depuración.
Proyecto Upbeat 24 Pruebas y depuración.	Proyecto Upbeat 25 Pruebas y depuración.	Proyecto Upbeat 26 Pruebas y depuración.	Proyecto Upbeat 27 Pruebas y depuración.	Proyecto Upbeat 28 Pruebas y depuración.	Proyecto Upbeat 29 Pruebas y depuración.	Proyecto Upbeat 30 Pruebas y depuración.

### 3. Análisis y diseño del sistema

#### 3.1. Análisis de requisitos

#### 3.2. Diseño del sistema