**CDS**

Irene Sánchez – [702692@unizar.es](mailto:702692@unizar.es)

Fidel Reviriego – 716678@unizar.es

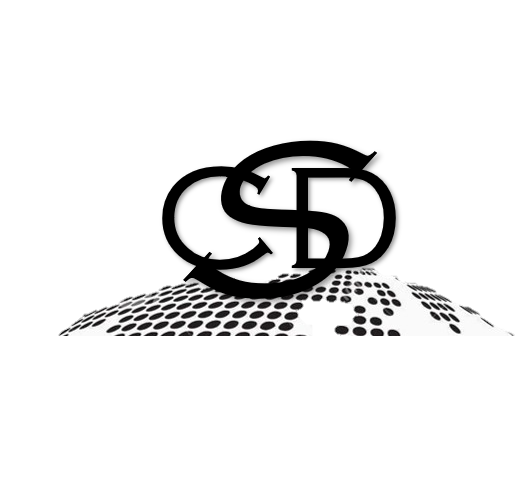
Alberto Mur – 697589@unizar.es

David Sáez – 704130@unizar.es

Fernando Landa – [701791@unizar.es](mailto:701791@unizar.es)

Carlos Tierno – 705548@unizar.es

Mengdie Zhou – 700991@unizar.es



Proyecto MEWAT

Plan de gestión, ANÁLISIS, DISEÑO Y MEMORIA del proyecto

ÍNDICE

[1. Introducción 2](#_Toc507699990)

[2. Organización del proyecto 2](#_Toc507699991)

[3. Plan de gestión del proyecto 3](#_Toc507699992)

[3.1. Procesos 3](#_Toc507699993)

[3.1.1. Procesos de inicio del proyecto 3](#_Toc507699994)

[3.1.2. Procesos de ejecución y control del proyecto 3](#_Toc507699995)

[3.1.3. Procesos técnicos 3](#_Toc507699996)

[3.2. Planes 4](#_Toc507699997)

[3.2.1. Plan de gestión de configuraciones 4](#_Toc507699998)

[3.2.2. Plan de construcción y despliegue del software 4](#_Toc507699999)

[3.2.3. Plan de aseguramiento de la calidad 4](#_Toc507700000)

[3.2.4. Calendario del proyecto y división del trabajo 5](#_Toc507700001)

[4. Análisis y diseño del sistema 6](#_Toc507700002)

[4.1. Análisis de requisitos 6](#_Toc507700003)

[4.2. Diseño del sistema 7](#_Toc507700004)

[Anexo I. Glosario 9](#_Toc507700005)

[Anexo III. Otros anexos que se consideren necesarios 9](#_Toc507700006)

### Introducción

En este proyecto se va a desarrollar una aplicación que permita la reproducción de música vía streaming. El proyecto comprenderá la creación y el mantenimiento de la aplicación móvil Android con versión 3.0 o superior y una versión web.

La aplicación en cuestión, permite al usuario el acceso a listas de canciones que se encuentren en los servidores, las cuales podrán estar ordenadas y listadas por artistas, estilos musicales, álbumes y título de canción. Tambien se permite al usuario la posibilidad de subir canciones de forma privada. Por otro lado, la aplicación tendrá una visión de red social, por lo que el usuario tiene la posibilidad de seguir a otro usuario, de esta manera podrás ver toda su información publica. Ademas de todas las funcionalidades que permite un reproductor de musica.

La aplicación contará con un servidor en el que se almacenan las canciones. El objetivo del proyecto es la creación de una aplicación de alcance nacional, principalmente orientada a usuarios familiarizados con el sistema Android.

El plazo de entrega del proyecto será de 4 meses aproximadamente y tendrá un coste total de 16000 euros con IVA incluido.El 1 de junio se realizara la entrega final del proyecto, en la cual se entragará la aplicación ya desarrollada, los ficheros necesarios para el despliegue y los manuales necesarios.

### Organización del proyecto

El equipo estará compuesto por 4 roles diferentes para la coordinación del proyecto, estos roles serán:

encargados de las bases de datos y servidor: David Sáez y Alberto Mur. Encaragados para el diseño de la de la aplicación móvil: Mengdie Zhou. Encargados del diseño de la interfaz Web: Carlos Tierno. Los desarrolladores de la aplicación móvil: Fernando Landa. Desarrolladores de la interfaz Web: Fidel Riveriego. Y por ultimo directora del proyecto y coordinandora de la pruebas: Irene Sánchez.

### Plan de gestión del proyecto

## Procesos

### Procesos de inicio del proyecto

Para el servidor, se hará uso de una raspberry pi 3.

Los dispositivos donde se realizarán las pruebas sera mediante emuladores y dispositivos físicos de pruebas.

Por último, las tecnologías usadas serán:

* + Como servidor de aplicaciones: Apache Tomcat
  + Como SGBD: MySQL
  + Como plataforma de despliegue: Android
  + Como lenguaje de programación: Java
  + Como lenguaje para la versión Web: jsp y html
  + Como lenguaje de estilo: CSS
  + Como lenguaje de creación/manipulación de tablas: SQL
  + Como librería de búsqueda: Apache Lucene

Cada integrante del proyecto deberá tener un manejo básico de estas tecnologías, ante posibles problemas del manejo de alguna, se solucionará con el seguimiento de algún tutorial o la lectura de la documentación.

### Procesos de ejecución y control del proyecto

* Para la comunicación interna se hará uso de un grupo de WhatsApp para dudas rápidas o concretas. Además, semanalmente habrá una reunión, en la cual estará presente todo el equipo, la cual servirá para exponer problemas surgidos, debatir ideas etc. Por último, se llevará un control de todas las reuniones mediante la redacción de un documento, el cual contendrá todos los temas hablados, los problemas y sus soluciones, esto permite que no haya posibles discusiones en el futuro. Por último, también existe la posibilidad del uso del correo electrónico para un tema particular, en el cual no tenga que intervenir todo el equipo del proyecto.
* El reparto de las tareas a cada integrante del grupo se realizará semanalmente en la reunión mencionada anteriormente, de esta manera todos los integrantes saben qué hace el resto de integrantes y se tiene un control más exhaustivo del proyecto.
* Los temas de gestión del equipo se realizan en las reuniones, poniendo cada integrante su punto de vista de forma ordenada.
* Las entregas del proyecto al cliente se realizarán a través de diversos hitos acordados con el cliente. Algunos de estos hitos podrían ser la entrega de un prototipo de la aplicación, o de una aplicación de muestra con funcionalidad mínima y códigos fuente. La entrega final incluiría la aplicación construida, y la documentación necesaria para el despliegue de esta.

### Procesos técnicos

* Las herramientas que se van a hacer uso se diferencian en 3 bloques según el objetivo:
  + Para el desarrollo tanto móvil como web se hará uso de las herramientas: Android Studio, Eclipse y/o Sublime Text, según las necesidades.
  + Para el despliegue de la aplicación se hará uso de Apache Tomcat y mysql instalados en una Raspberry Pi 3
  + A la hora de probar el sistema Eclipse, Android Studio y dispositivos Android físicos pertenecientes al equipo del proyecto.

## Planes

### Plan de gestión de configuraciones

* Convenciones de nombres: todos los archivos serán llamados de forma que el nombre sea una descripción del fichero, además cada fichero deberá tener una cabecera con su descripción.
* Todos los integrantes del proyecto estarán encargados de la puesta en marcha, apoyo al equipo, revisión de commits, copias de seguridad, control de las versiones entregadas a cliente...
* Como recursos se utilizará GitHub como controlador de versiones. Todos los miembros poseerán permisos para realizar modificaciones en los mismos.
* Todo el código oficial estará en GitHub, cada vez que un componente del equipo haga un cambio, este será debidamente especificado en el commit y se avisa al resto del equipo. El cambio mencionado anteriormente, solo se realizará cuando el código compile y funcione todo correctamente

### Plan de construcción y despliegue del software

* El software se va a construir usando comandos de terminal para la construcción del fichero .war . Para generar el fichero APK se usara Android Studio. No habrá script de construcción automatizada.

Al tener todo el proyecto en GitHub, con la versión mas reciente la cual funciona correctamente, cada vez que cualquier integrante desee hacer pruebas, compilarlo o integrarlo, podrá hacerlo con la certeza de que todo debe funcionar. Hasta que la parte del servidor y bases de datos no este terminada no se podrá comprobar de forma completa, solo se podrá ver el funcionamiento de parte del sistema.

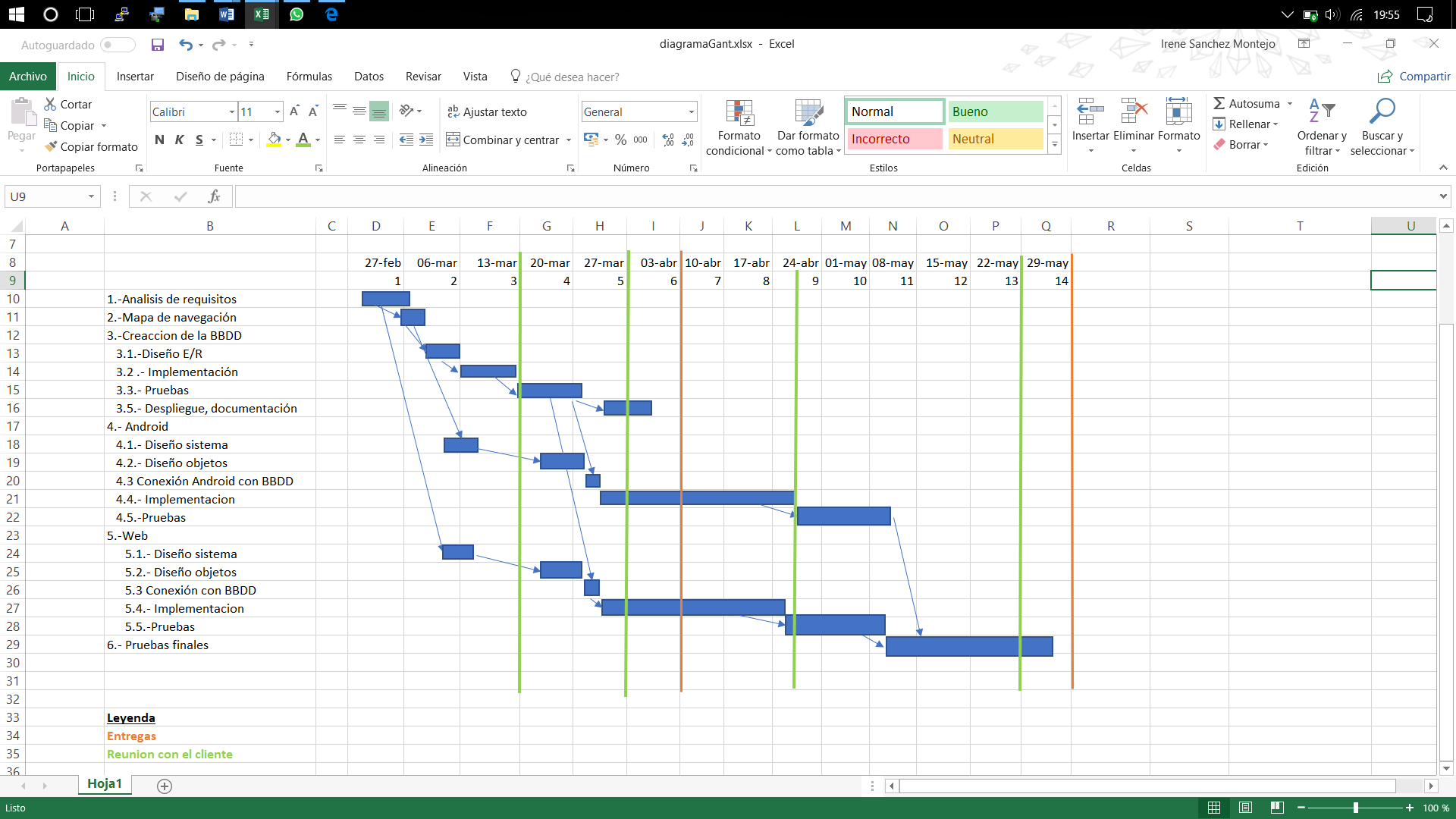
* Mediante el fichero .war y el apk que se entregan al cliente, serán los necesarios para el despliegue ypara la instalación.
  + - El fichero war se despliega en la venta de administración de Apache Tomcat
    - El fichero apk se instala en el dispositivo Android en el que se quiera usar la aplicación

Ademas de un manual donde figuran las instrucciones y restricciones a seguir para instalar el sistema.

### Plan de aseguramiento de la calidad

* Siempre se ejecutará y realizará las pruebas antes de realizar un “commit” con la última actualización del proyecto validado.
* Cuando se considere completa, la versión se testeará en los dispositivos móviles de los desarrolladores y los navegadores de los equipos de los desarrolladores, los cuales servirán para hacer pruebas finales.

### Calendario del proyecto y división del trabajo

 El diagrama de Gantt:

Para el desarrollo de Android, se ha decidido que el trabajo lo realizarán 5 de los miembros del

equipo (Irene Sánchez, Fidel Riveriego, Mengdie Zhou, Fernando Landa, Carlos Tierno). El diseño e

implementación de la BBDD y el servicio web será desarrollado por los dos miembros restantes

(Alberto Mur y David Saez).

Ademas, cada integrante será encargadado de otras partes del proyecto, como se ha mencionando en el punto 2.

Las tareas se reparten a cada uno de los integrantes para aprovechar mejor el trabajo en paralelo. De manera que el punto 1 se ha hecho conjuntamente entre todos lon integrantes del grupo, habiendo hecho las reuniones necesatias para las mismas.

El punto 2, junto con los prototipadfos de la GUI han sido realizados por: Mengdie Zhou.

El punto 3.1 han sido realizados por: Alberto Mur y David Sáez.

3.2: se realizará en paralelo todos los integrantes del grupo, de manera que se pretende tener el trabajo lo antes posible, no obstante, estas implementaciones podrán ser actualizadas porsteriormente por las funcionalidades que puedan surgir en el momento de realizar las implementaciones de las a`licaciones-.

3.3: debido a que en este momento se ha podido ir avanzando en otros puntos del proyecto, tales como el diseño del sistema Web y de la aplicación del móvil, se dejará a un único integrante para que realice las pruebas oportunas, Alberto Mur.

3.4: Despliegue y Documentacion: puesto que habrá que indicar las instrucciones de instalación del servidor junto que la BBDD, se necesitará documentarlo, que lo realizará Alberto Mur.

4. En este punto y en todos los subdivisiones, serán encargador por Fernando Landa, Carlos Tierno y Mengdie Zhou. En paralelo con quienes realizarán el punto 5, que serán: David Sáez, Fidel Riveriego e Irene Sáncehz.

### Análisis y diseño del sistema

## Análisis de requisitos

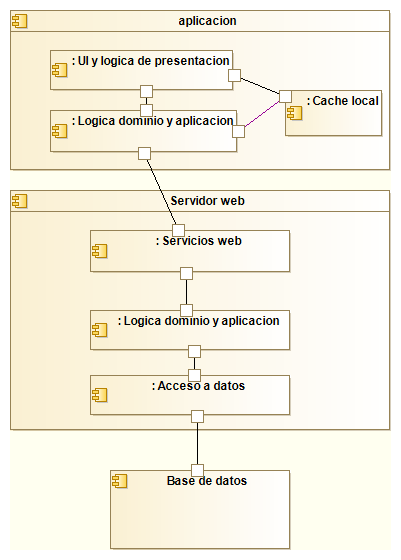
Se presenta una tabla de requisitos funcionales que el sistema debe presentar, en ella se incluye las funcionalidades básicas de funcionamiento. Por otra parte, hay que tener en cuenta las propiedades no funcionales del sistema, que se presentarán en la segunda tabla de requisitos no funcionales.

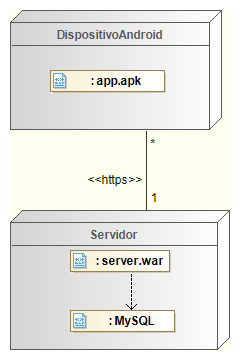
|  |  |
| --- | --- |
| **Requisitos funcionales** | |
| **Código** | Descripción |
| **RF1** | El usuario puede crear cuenta y loguearse con ella. |
| **RF2** | El usuario puede subir una canción |
| **RF3** | El sistema tendrá las funcionalidades básicas de red social: ser amigos, visualizar el estado de un amigo en la lista, compartir canción. |
| **RF4** | El usuario puede escuchar canciones, tanto las que ofrece el servidor como las que ha subido él mismo. |
| **RF5** | El usuario puede crear listas de reproducción de canciones. |
| **RF6** | El usuario puede realizar búsquedas de canciones por cualquier método, ya sea por título, por nombre de artista, por álbum … |
| **RF7** | El sistema constará de las funcionalidades básicas de reproducción de música: pausar, reanudar, avanzar, retroceder … |
| **RF8** | El sistema constará distintos métodos de reproducir las canciones de una lista: aleatorio, en orden, en bucle … |
| **RF9** | La reproducción de las canciones se hace vía streaming. |
| **RF10** | El sistema deberá sincronizar una cuenta de usuario en distintos dispositivos. |

|  |  |
| --- | --- |
| **Requisitos no funcionales** | |
| **Código** | Descripción |
| **RNF1** | El sistema constará de dos interfaces: interfaz de sistema Web e interfaz de aplicación móvil. |
| **RNF2** | El sistema solo permite reproducción de archivos en formato: mp3, Ogg, acc |

## Diseño del sistema

* Diagramas arquitecturales (de módulos, de componentes y conectores, de distribución), patrones de diseño y estilos arquitecturales que se aplicarán. Las interfaces (de módulos y de componentes) son especialmente importantes. También lo son los protocolos de comunicación entre componentes.





En cuanto a los patrones de diseño, se va a usar el patrón facade en el servidor web para definir los servicios del subsistema de manera que ofrezca una interfaz unificada realizando de esta manera una abstracción de los detalles internos.

Respecto a la arquitectura, será de tres capas separando el cliente móvil que será ligero de manera que disponga de la interfaz gráfica y una pequeña parte de lógica de la aplicación, el servidor web que contiene la mayor parte de la lógica de dominio y aplicación y la base de datos que para ahorrar costes se desplegará en la misma máquina que el servidor web, aunque en el futuro se podría instalar en una máquina distinta sin necesidad de realizar muchos cambios.

Otra opción que se tuvo en mente fue usar una arquitectura en dos capas, pero se descartó debido a que el cliente móvil tendría que ser pesado y siguiendo la primera opción en futuras versiones se puede expandir el proyecto de manera que también sea aplicación web utilizando la API que ofrece el servidor web mientras que con dos capas habría que volver a diseñarlo.

Los protocolos de comunicación serán HTTPS para tener conexión segura mediante cifrado entre los clientes móviles y el servidor dado que la conexión se realiza mediante Internet y por tanto no se puede elegir un protocolo sin cifrar como http. Por otra parte, dado que el servidor web y la base de datos se despliegan en la misma máquina física el acceso a los datos se realiza mediante JDBC. En caso de instalación de la base de datos en otra máquina física en futuras versiones, la conexión se podría realizar mediante Intranet.

* Tecnologías elegidas: El lenguaje de programación elegido es Java dado que es un lenguaje habitual de desarrollo en Android, está bien documentado y tiene bastantes librerías para diversos componentes (JDBC para acceso a base de datos, Lucene para implementar búsquedas,) y además todo el equipo de desarrollo tiene experiencia con este lenguaje.
* Otros aspectos técnicos de interés (p.ej. sí hay base de datos si va a ser SQL o NoSQL, si hay una API Web va a ser RESTful o no, si algunas de las operaciones van a ser asíncronas o no, si va a ser una aplicación móvil o de escritorio será nativa o se van a usar tecnologías web, cómo se van a considerar los requisitos de seguridad o de prestaciones, cómo y dónde se harán las instalaciones y despliegues etc.)

La base de datos elegida es SQL usando el sistema gestor MySQL y se ha elegido debido a que se quiere conseguir realizar consultas más complejas y tener consistencia en los datos por lo que a pesar de que una base de datos tipo NoSQL habría aportado escalabilidad horizontal, se ha optado por SQL puesto que además el equipo ya contaba con experiencia en el diseño y puesta en marcha de bases de datos relaciones con MySQL.

En cuanto a la API Web que ofrece el servidor, va a ser RESTful dado que se va a usar HTTP para obtener datos y generar operaciones entre los clientes móviles y el servidor. También se optó por usar SOAP pero no se seleccionó dado que limita a datos con formato XML mientras que con RESTful también son válidos otros formatos como JSON y HTML por lo que en ese aspecto ofrece más flexibilidad.

La aplicación móvil será nativa puesto que, aunque se evaluó la opción de que usase tecnologías web con lo que sería compatible con cualquier sistema operativo se prefirió desarrollar solo para el sistema Android para aprovechar las funcionalidades del dispositivo y tener mayor visibilidad en el mercado si en el futuro se decide incluir la aplicación en Google Play. Además de esta manera se puede ofrecer cierta funcionalidad si no se dispone de conexión a Internet mientras que con tecnologías web no sería posible.

# Anexo I. Glosario

Explicación de acrónimos y términos técnicos utilizados.

# Anexo III. Otros anexos que se consideren necesarios