

Software Requirements Specification (SRS)

BestTaste



Por: Teresa De La Vega, Alejandro Garrido, Jorge Martín, Roi López,
Francisco Adrián Fonta, Álvaro Tejedor, Jhon Fredy Jaramillo y
Samuel Martínez

Índice

- 1. Introducción**
 - 1.1. Propósito
 - 1.2. Ámbito del Sistema
 - 1.3. Definiciones, Acrónimos y Abreviaturas
 - 1.4. Referencias
 - 1.5. Visión General del Documento
- 2. Descripción general**
 - 2.1. Perspectiva del Producto
 - 2.2. Funciones del Producto
 - 2.3. Características de los Usuarios
 - 2.4. Restricciones
 - 2.5. Suposiciones y Dependencias
 - 2.6. Requisitos Futuros
- 3. Requisitos Específicos**
 - 3.1. Interfaces Externas
 - 3.1.1. Interfaces de Usuario
 - 3.1.2. Interfaces de Administradores
 - 3.2. Funciones
 - 3.2.1. Módulo de usuarios
 - 3.2.2. Módulo recomendaciones
 - 3.2.3. Módulo Comments
 - 3.2.4. Módulo Estadísticas
 - 3.2.5. Módulo PlayLists
 - 3.2.6. Módulo Administradores
 - 3.3. Requisitos de Rendimiento
 - 3.4. Atributos del Sistema
- 4. Apéndices**
 - 4.1. Diagramas de Casos de Uso
 - 4.1.1. Gestión de Usuarios
 - 4.1.2. Gestión de Administrador
 - 4.2. Diagramas de funciones:

1. Introducción

El presente documento de Especificación de Requisitos de Software le permitirá a los lectores un entendimiento de las características más relevantes de **BestTaste**.

1.1. Propósito

Una SRS tiene como propósito proporcionar una descripción clara y completa acerca del funcionamiento del producto de software según los requisitos del cliente para que así, los desarrolladores se encarguen de realizarlo adecuadamente para que una serie de usuarios lo utilicen.

En este caso, este documento tiene como finalidad definir las características y los conceptos y propósitos de BestTaste enfocado al personal encargado de su desarrollo y mantenimiento. Está dirigido tanto a las partes involucradas en el desarrollo del proyecto, lo cual incluye a desarrolladores y clientes, como a cualquiera de los usuarios que usen el producto.

1.2. Ámbito del Sistema

Hoy en día hay miles de millones de canciones, muchas de las cuales no obtienen mucho reconocimiento, pero que realmente podrían ser disfrutadas por muchos. Es por ello que este proyecto se trata de una red social llamada **BestTaste**, la cual busca ayudar a sus usuarios a encontrar música acorde con sus gustos. A partir de recomendaciones hechas por el mismo u otros usuarios, el usuario de la aplicación podrá escuchar una previsualización o *preview* de 15 segundos de la canción, para así encontrar la que más acorde esté a sus gustos.

Contaremos, además, con sponsors, los cuales serán **dados de alta o de baja** por un administrador. Estos publican anuncios en la aplicación, los cuales aparecerán cada 5 recomendaciones para el usuario casual, pero proporcionaremos la posibilidad de suscribirse a **BestTaste+** por el precio de 9,99 al mes para quitar los anuncios y, además, poder hacer más recomendaciones en un mismo día, encima de otras ventajas.

Con este sistema se podrá **dar de alta, de baja, modificar, listar y buscar** a los clientes o a las canciones correspondientes, y **crear, borrar** las recomendaciones del usuario, además de **crear playlists** con las recomendaciones de otros usuarios. Aunque hace falta aclarar que dichas playlists no serán más que una recopilación de las previsualizaciones de 15 segundos de cada canción, pero que permitirán que el usuario tenga un acceso más directo a estas mismas.

Nuestro propósito con este software es agilizar la búsqueda de música ya que, conociendo los gustos del usuario, otros con similares podrán dar a conocer canciones que si no fuese por dicha recomendación, nunca se hubiese conocido.

1.3. Definiciones, Acrónimos y Abreviaturas

- **SRS:** Especificación de Requisitos del Software(Software Requirements Specification)
- **IEEE:** Institute of Electrical and Electronics Engineers
- **login:** Iniciar sesión
- **logout:** Cerrar sesión
- **Java:** Es un lenguaje de programación ampliamente utilizado para codificar aplicaciones web

- **Preview:** previsualización, generalmente 15 segundos de una canción
- **Playlist:** lista de reproducción.
- **Text:** se refiere a un texto

1.4. Referencias

Como referencias para el desarrollo de este documento, hemos utilizado diferentes fuentes. En primer lugar, el estándar de modelo “SRS IEEE 830”, además de otros documentos que utilizan el modelo mencionado anteriormente. También hemos usado como referencia la aplicación *Spotify* y *X*.

<https://twitter.com/home>

<https://www.fdi.ucm.es/profesor/gmendez/docs/is0809/ieee830.pdf>

<https://www.spotify.com>

1.5. Visión General del Documento

El documento lo podemos ver dividido en una serie de secciones:

- La sección de introducción, la cual introduce tanto al propio documento como a la aplicación **BestTaste**. Se muestra la finalidad de las SRS así como una breve descripción de la aplicación que ofrecemos.
- La sección de la descripción general, que profundiza en en la descripción del sistema, así como sus restricciones y funciones orientadas al usuario final.
- La sección de requisitos específicos, esta sección está más orientada a desarrolladores y programadores ya que emplea términos técnicos.
- La sección de diagramas, también orientada principalmente a los desarrolladores ya que incluye los diagramas de casos de uso.

2. Descripción general

En este apartado vamos a introducir los requisitos del programa, sin contemplar los detalles de estos.

2.1. Perspectiva del Producto

Este software se trata de un software autónomo e independiente, el cual únicamente requiere de sus recursos y de un sistema operativo sobre el que ser ejecutado, el cual debe ser compatible con Java.

BestTaste se llevará a cabo a través de una base de datos que recogerá toda la información tanto del cliente y los sponsors como de las canciones, a la cual tendrán acceso los administradores para que así puedan encargarse de dar de alta y baja a canciones y anuncios.

Finalmente, este producto se desarrollará a través de una interfaz gráfica que se podrá usar mediante una pantalla táctil, o en general, en un dispositivo que permita manipular toda la información. Será necesario establecer una conexión red para acceder a la misma.

2.2. Funciones del Producto

Las funciones principales de esta aplicación son aquellas que nos permitan administrar los usuarios, las recomendaciones y los sponsors. Así, estos podrán añadirse, eliminarse o modificarse en la base de datos. Por ello, distinguiremos entre las funciones de los usuarios y las de los administradores.

Las funciones del usuario consisten en la creación de una cuenta y el uso que le puede dar a la aplicación. Es por ello que empezamos especificando las funciones de darse de alta y de baja de los usuarios:

- **Darse de alta:** El usuario proporciona sus datos entre los cuales introduce un nombre de usuario único, introduce una contraseña, debe ser confirmada escribiéndola una segunda vez. Si los datos son válidos, el usuario estará dado de alta en la aplicación.
- **Darse de baja:** De la misma forma que el usuario puede crear una cuenta, puede también eliminarla. Si el usuario lo desea, puede darse de baja, dejando el nombre de usuario libre para que otros puedan usarlo. Se le preguntará al usuario el motivo de la baja, aunque será un campo no obligatorio, e inmediatamente se borrará la cuenta de la base de datos.
- **Log in:** Ahora que el usuario tiene una cuenta, podrá iniciar sesión y cerrar las veces que desee. El inicio de sesión o *log in* simplemente comprobará que el usuario y la contraseña coinciden y son correctos y le iniciará sesión en la aplicación.
- **Log out:** El usuario puede cerrar la sesión.
- **Modificar su cuenta:** El usuario solicita sus datos para así poder actualizarlos. Luego es notificado de sus modificaciones y se confirma en la base de datos.

En cuanto al uso que le puede dar a la aplicación, las funcionalidades son:

- **Recomendación:** Antes de realizarla, hay que asegurarse de que el usuario no sobrepasa las recomendaciones máximas diarias. El usuario debe seleccionar una canción a recomendar entre una lista proporcionada de canciones, y añadir una breve descripción. Una vez hecho se publica.
- **Borrar recomendaciones:** La recomendación puede también ser borrada en cualquier momento por el usuario, pero no se modificará el número de recomendaciones disponibles. Se borra desde un menú desplegable que aparece en cada recomendación.
- **Crear comentarios:** El usuario puede dejar comentarios bajo una recomendación sin limitaciones.
- **Borrar comentario:** El usuario puede borrar comentarios.
- **Crear playlist de favoritos:** En el mismo menú desplegable donde se borra la recomendación, está la opción de añadirla a favoritos también. Esta opción es aplicable tanto para sus propias recomendaciones como para las publicadas por otros usuarios. El usuario le da a esta opción, se le pide que la guarde en un *playlist*, y si no tiene, deberá crear una en la cual se le pida el nombre y si quiere que sea privada o pública, y se añadirá si no lo estaba ya antes.
- **Acceso a playlist de favoritos:** Se introduce el nombre de la playlist y se muestra la lista.
- **Crear playlist:** El usuario puede crear playlist.
- **Acceso a playlist:** A parte de la playlist de favoritos mencionada anteriormente, existe otra con recomendaciones creadas por el sistema basadas en los favoritos del usuario, y a la que se accede desde el perfil del usuario.

- **Borrar playlist:** El usuario puede borrar la playlist.
- **Buscar una canción:** el usuario puede introducir el nombre de la canción o el artista, y le saldrá recomendaciones hechas por otros usuarios de la canción, en el caso de que es lo que busca, o recomendaciones de canciones pertenecientes al artista.
- **¿No encuentras tu canción favorita?:** Esta opción se basa en la posibilidad de que el usuario pueda añadir canciones a la lista de canciones existente que ofrece el sistema. Se deberá comprobar, por supuesto, que la canción no está ya en la base de datos, y se pasará en forma de solicitud a los administradores.
- **Estadísticas:** Existen dos tipos de estadísticas a las que puede acceder el usuario:
 - *Las de usuarios:* Tendrá que especificar el nombre de usuario del que desea ver las estadísticas
 - *Estadísticas de una canción:* Tendrá que especificar de qué canción quiere ver las estadísticas y se mostrará el número de recomendaciones hechas sobre la canción.

Por otro lado, están las funcionalidades del administrador:

- Tiene acceso a la **lista de usuarios** de la aplicación.
- **Dar de alta a sponsors:** Los administradores reciben una oferta de sponsor, la validan, la guardan en la base de datos y le dan de alta.
- **Dar de baja a sponsors:** Piden la información al sponsor, comprueban que está en la base de datos y si no lo está, confirma con el cliente que los datos proporcionados son los correctos, y si existe, lo elimina.
- **Dar de alta a canción:** Reciben el nombre de la canción, confirman que existe, la meten en la lista de canciones, y confirman que se ha introducido
- **Dar de baja a canción:** Confirma que existe la canción, la borra de la lista de canciones y confirma que se ha borrado con éxito.

2.3. Características de los Usuarios

El sistema cuenta con dos tipos de usuario:

- Administradores: Los cuales son los encargados de la administración de la red social, así como de dar de alta o baja a los sponsors que previamente contactaron con los dueños de la red social, es decir, nuestro cliente. Además, los administradores tendrán acceso a la documentación de los clientes, así como de sus recomendaciones y modificaciones.
- Usuario: Los cuales son los que le van a dar uso a la aplicación. Son los que crean, eliminan recomendaciones, además de dejar comentarios. También crearán y borrarán las playlists de canciones, y tendrán la capacidad de añadir nuevas canciones a la lista general.

2.4. Restricciones

Las posibles restricciones que pueden aparecer en el proyecto son las siguientes:

- Restricciones de tiempo: posible limitación en cuanto a la fecha de entrega debido al poco tiempo para completar el proyecto.
- Restricciones tecnológicas: experiencia media en Java de varios miembros del equipo, lenguaje con el que se va a desarrollar la aplicación.

- Restricciones de recursos humanos: limitación de miembros del equipo en cuanto a la disponibilidad debido al alto nivel de exigencia de tareas diarias de todos en otros ámbitos.

2.5. Suposiciones y Dependencias

Los requisitos y las restricciones descritas en este documento están sujetas a posibles cambios en un futuro. Por ello, es necesario que las fases de análisis queden bien documentadas, y que se defina una fase de mantenimiento del sistema.

Además, para el correcto funcionamiento del sistema, será necesario tener instalado eclipse o cualquier otro programa capaz de soportar 'Java', y tener acceso a internet. También será necesario disponer de sistemas Linux y Windows para la base de datos, ya que son los sistemas soportados por SQL (Structured Query Language).

2.6. Requisitos Futuros

Los requisitos serán implementados en función de las necesidades del cliente, por lo que él deberá ser capaz de adaptarse a cambios futuros. Algunos ejemplos de esto son:

- La posibilidad de añadir canciones enteras a una playlist en vez de únicamente tener un preview.
- Mejoras tecnológicas.
- Capacidad de migración a otras tecnologías.
- Un sistema de reportes para que los usuarios puedan llamar la atención a usuarios que no están haciendo un uso adecuado de la aplicación.

Si esto en algún momento se vuelve realidad, se tendrá que hacer una estimación del coste de la acción, además de dar formación al equipo si hace falta.

3. Requisitos Específicos

En esta sección profundizaremos sobre nuestro sistema, de modo que quede explicado tanto el funcionamiento de nuestra interfaz, como de las funciones y el sistema.

3.1. Interfaces Externas

Lo primero que desarrollaremos en este apartado son las interfaces del sistema.

Detallaremos tanto lo que ve el usuario al entrar, como al darle uso a **BestTaste**, además de la perspectiva del administrador. El color principal de **BestTaste** se basa en distintos tonos de verde, como se verá más adelante, y desde toda interfaz salvo las principales, existirá la posibilidad de volver para atrás mediante una flecha situada arriba a la izquierda.

3.1.1. Interfaces de Usuario

Dividimos según las funciones del usuario.

- Interfaz de inicio:

Al iniciar el programa, lo primero que verá el usuario será una ventana que le pedirá al usuario sus credenciales (Usuario y contraseña) para acceder al sistema. A su vez, debajo de la contraseña, habrá una opción de "¿Aún no tienes cuenta?" la cual será para usuarios nuevos. Si se le da a esta opción, saltará a la interfaz de crear cuenta.

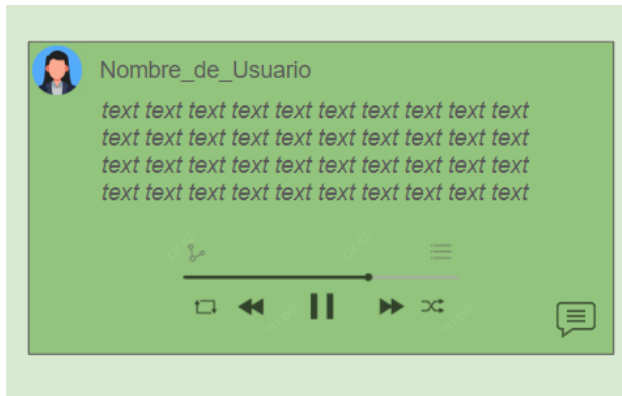
- Interfaz crear cuenta:

La siguiente interfaz que desarrollaremos es la de crear cuenta. Como se mencionó anteriormente, el usuario será enviado a esta ventana si le da a "¿Aún no tienes cuenta?". El usuario verá una ventana parecida a la interfaz de inicio, solo que esta vez, en vez de sus

credenciales, se le pedirá que rellene una serie de datos (nombre completo, correo, contraseña, etc.) y finalmente se le pedirá que confirme los datos y acepte. Ahora ve el sistema principal.

- Interfaz principal:

Una vez iniciada la sesión el usuario es enviado a esta interfaz. El layout es similar al de twitter. A la izquierda tendrá 3 opciones, perfil, estadísticas y crear recomendaciones, en el medio tendrá las recomendaciones de otros usuarios que estarán estructuradas una de tras de la otra, y que individualmente serán de la siguiente manera:



El text será la descripción de la canción añadida por el usuario, cada descripción tendrá el nombre de usuario y el icono de perfil que ha escogido el usuario correspondiente y la opción de darle play al segmento de 15 segundos de la canción. El icono de tres líneas una debajo de la otra será donde el usuario podrá añadir a favoritos una recomendación, y el icono de conversación abajo a la derecha será para dejar un comentario bajo la recomendación. Cada 5 recomendaciones habrá un anuncio que tendrá el siguiente formato:



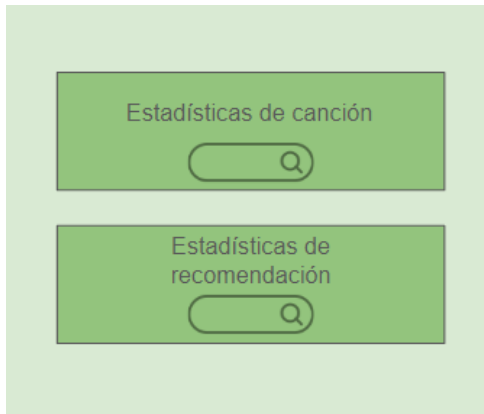
Por último, arriba a la derecha tendrá un icono de búsqueda que podrá usar para encontrar recomendaciones y canciones específicas.

- Interfaz perfil:

Se trata de la interfaz donde el usuario podrá acceder a su información de cuenta, además de sus recomendaciones publicadas y por último a sus playlists. Se trata de una interfaz que está centrada. Arriba tiene la información a modificar, su foto de perfil, su nombre de usuario, correo electrónico, etc. Diferenciado por un breve espacio, están las recomendaciones del usuario, y por último, en dicho breve espacio, pegado a la derecha, está la opción "darse de alta en **BestTaste+**".

- Interfaz estadísticas:

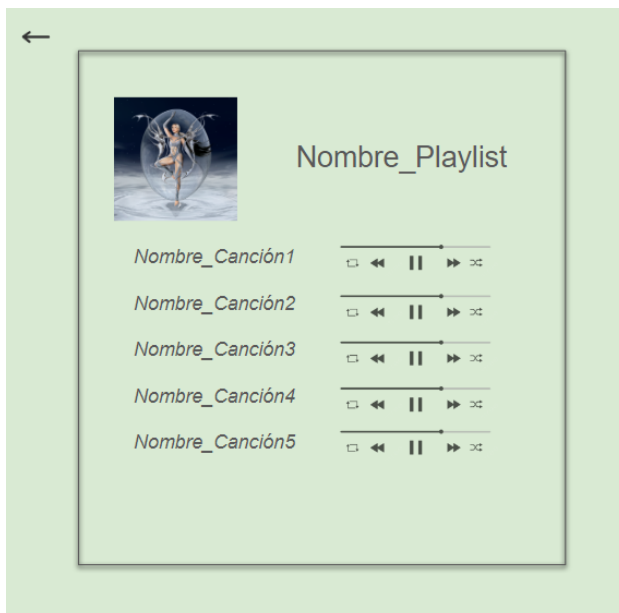
Al darle a esta opción el usuario pasará a una página con dos opciones de la siguiente manera:



Dependiendo de donde escriba el usuario lo que busca, se le abrirán las estadísticas de canción o de una recomendación. Las estadísticas estarán ordenadas según su tipo:

- De canciones: tendrá en el centro una gráfica que muestre el número de recomendaciones que se han hecho de esa canción a lo largo del año, debajo a la izquierda el número de recomendaciones totales y número de usuarios individuales que han hecho una recomendación sobre dicha canción.
- De recomendaciones: Mostrará el número de interacciones totales con la recomendación, el número de comentarios, y el usuario que hizo la recomendación.
- Interfaz playlists:

Cuando el usuario entra en su perfil y accede a su playlist, esta interfaz estará organizada de la siguiente manera:



Así, el usuario tendrá una foto personalizada según el género de canciones más guardado, el nombre de la playlist que le ha puesto y al lado la posibilidad de re escuchar los 15 segundos de la canción que venían en la recomendación que añadió a favoritos.

3.1.2. Interfaces de Administradores

Los administradores tendrán una interfaz muy distinta a la de los usuarios. La página principal tendrá la misma estructura, solo que en esta, los administradores no tendrán la opción de añadir una recomendación ni acceder a estadísticas. Por ello, la página principal sí que mostrará recomendaciones realizadas por usuarios además de anuncios, y la opción de buscar, pero a la izquierda, la única opción que tendrán será de “gestión”.

- Interfaz de gestión:

Se trata de la ventana que tendrá el administrador para cumplir su función, que es dar de alta o baja a un sponsor, además de tener acceso a la lista de usuarios. Esta interfaz es una página con tres opciones, “Añadir un sponsor”, “Dar de baja a un sponsor” y “Ver lista de usuarios”.

- Interfaz añadir sponsor:

Si el administrador le da a añadir sponsor, será enviado a una página donde se le pedirá una serie de datos. El nombre del sponsor, la descripción que el sponsor manda cuando solicita estar en el sistema, el logo de la marca que se pondrá arriba a la izquierda en el anuncio, así como el video publicitario, el cual el propio sistema ajustará a la plantilla de videos sponsor, la cual aparece más arriba en el apartado.

- Interfaz dar de baja a un sponsor:

Cuando el administrador le de a la opción de dar de baja, se le enviará a una página donde se le pida tanto el nombre del sponsor, como el ID único que se le asignó dicho sponsor cuando el sistema lo creó. Tendrá debajo de la información solicitada dos opciones, “aceptar” y “cancelar”, si le da a aceptar, el sistema buscará dicho sponsor y lo eliminará, mostrando un mensaje de éxito, o de error si no se encontró.

- Interfaz ver lista de usuarios:

Esta opción llevará al administrador a una página donde se muestran todos los usuarios pertenecientes a la plataforma, así como sus correos, e IDs únicos que se les asignó al crear la cuenta.

3.2. Funciones

3.2.1. Módulo de usuarios

Iniciar sesión:

CASO DE USO #4	Iniciar sesión
Objetivo, descripción	El usuario introduce sus datos de inicio de sesión para acceder a la aplicación
Entradas	El usuario introduce su nombre de usuario junto a su contraseña
Precondiciones	La cuenta previamente debe de estar registrada
Salidas	Se muestra un mensaje si se ha podido realizar el inicio de sesión o no

Postcondición si éxito	Se informa al usuario de que el inicio de sesión ha sido correcto y se muestra la interfaz principal del usuario
Postcondición si fallo	Se informa al usuario de que el usuario o la contraseña que ha introducido no son válidas y se vuelve a la interfaz de inicio de sesión
Actores principales	Usuario
Actores secundarios	Base de datos
Secuencia principal	<ol style="list-style-type: none"> 1. Introducir nombre de usuario junto a la contraseña 2. Comprobar que el usuario y la contraseña son correctos. Si no son correctos: S1 3. Se informa al usuario acerca del éxito del inicio de sesión. El usuario accede a la aplicación
Secuencia alternativa	S1. Los datos introducidos no son correctos. Se informa al usuario del error, debe introducir los datos correctamente.

Cerrar sesión:

CASO DE USO #5	Cerrar sesión
Objetivo, descripción	El usuario cierra su sesión de la aplicación pulsando en la pestaña “Cerrar sesión”
Entradas	-----
Precondiciones	La cuenta previamente debe de estar registrada
Salidas	Se muestra un mensaje si se ha podido cerrar sesión o no
Postcondición si éxito	Se informa al usuario de que el cierre de sesión ha sido correcto y se muestra la interfaz principal de la aplicación
Postcondición si fallo	Se informa al usuario de que no se ha podido cerrar sesión y se muestra la interfaz de inicio de sesión
Actores principales	Usuario

Actores secundarios	Base de datos
Secuencia principal	<ol style="list-style-type: none"> 1. Pulsar pestaña “Cerrar sesión” 2. Comprobar que el usuario previamente ha iniciado sesión. Si no se ha iniciado S1 3. Se informa al usuario acerca del cierre del inicio de sesión. El usuario accede a la interfaz de bienvenida de la aplicación.
Secuencia alternativa	S1. El cierre de sesión no se ha podido realizar. Se informa al usuario del error, se accede a la interfaz principal del usuario.

Darse de alta en BestTaste+:

Caso de Uso #6	Darse de alta en BestTaste+
Objetivos, descripción	El usuario tiene la posibilidad de darse de alta como usuario BestTaste+ para así eliminar anuncios y poder realizar más recomendaciones.
Entradas	Los datos bancarios del usuario.
Precondiciones	Que el usuario no esté suscrito ya a BestTaste+ y que tenga una cuenta bancaria válida.
Salidas	El usuario actualiza su perfil a uno de BestTaste+ , teniendo ahora más recomendaciones y sin anuncios, además de otros beneficios.
Postcondiciones si éxito	Se muestra un mensaje de éxito al usuario y se le pasa a la página actualizada con lo mencionado anteriormente.
Postcondiciones si fallo	Se muestra un mensaje de error y el motivo del error, ya sea por fallo en los datos bancarios, o en que el usuario ya era parte de BestTaste+ .

Actores principales	Usuario
Actores secundarios	Base de datos
Secuencia principal	<ol style="list-style-type: none"> 1. El usuario le da al botón de “Unirse a BestTaste+” (sí ya forma parte, S-1) 2. Se piden los datos bancarios del usuario, y que acepte los términos y condiciones, y se comprueba que los datos sean correctos (S-2 si no lo son) 3. Se muestra un mensaje de éxito al suscribirse al usuario 4. Este pasa a formar parte de los usuario BestTaste+
Secuencia alternativa	<p>S-1: Se muestra un mensaje de error por ya pertenecer a BestTaste+</p> <p>S-2: Se informa al usuario de que ha habido un fallo al introducir los datos bancarios, y se le redirecciona devuelta a la página de su perfil</p>

3.2.2. Módulo recomendaciones

Crear una recomendación:

CASO DE USO #1	Crear una recomendación
Objetivo, descripción	El usuario realiza una recomendación de una canción.
Entradas	Selección de la canción deseada a recomendar.
Precondiciones	El usuario debe tener disponibles recomendaciones, en este caso no haber sobrepasado el límite de recomendaciones diarias que se le proporcionan. Además, debe haber una breve descripción en la recomendación.
Salidas	<ol style="list-style-type: none"> 1. Publicación de la recomendación. 2. Aviso del rechazo al intentar hacer la recomendación.
Postcondición si éxito	Publicará la recomendación, dando visibilidad a otros usuarios de la misma.
Postcondición si fallo	Salta un aviso con el por qué no se ha podido realizar la recomendación.

Actores principales	Usuario
Actores secundarios	Administrador. Base de datos.
Secuencia principal	<ol style="list-style-type: none"> 1. Selección de una canción. 2. Pulsar la opción de “Recomendar” de dicha canción. 3. Si el usuario ha superado su límite de recomendaciones diarias (a). 4. El usuario agrega la descripción de la recomendación. 5. Si la descripción es inadecuada o sobrepasa el máximo de caracteres (b). 6. Se publica la recomendación y se reduce el número de recomendaciones restantes al usuario. 7. Finaliza visualizandose la recomendación publicada.
Secuencia alternativa	<ol style="list-style-type: none"> a. No se puede avanzar con la recomendación. Se avisa al usuario que ha sobrepasado su límite de recomendaciones diarias, luego vuelve al punto 1. b. Salta un aviso de que la descripción no es adecuada o sobrepasa el límite de caracteres, luego vuelve al punto 4.

Borrar una recomendación:

CASO DE USO #2	Borrar recomendación
Objetivo, descripción	El usuario borra una recomendación, realizada anteriormente por el mismo.
Entradas	Seleccionar la recomendación la cual se quiere borrar.
Precondiciones	El usuario debe tener al menos una recomendación publicada por sí mismo.
Salidas	Aviso del éxito al borrar la recomendación.
Postcondición si éxito	Se elimina la recomendación tanto de la página como de la base de datos.

Postcondición si fallo	Se muestra un mensaje detallando que no se ha podido borrar la recomendación.
Actores principales	Usuario
Actores secundarios	Base de datos.
Secuencia principal	<ol style="list-style-type: none"> 1. Selección de la recomendación. 2. Pulsar la opción de “Eliminar” de dicha recomendación. 3. Salta un aviso para confirmar la acción de borrado. 4. Si no acepta (a). 5. Si acepta se procede a borrar la recomendación. 6. Regresa al apartado de recomendaciones.
Secuencia alternativa	a. regresa al punto 1.

Mostrar lista de canciones:

CASO DE USO #3	Mostrar lista de canciones
Objetivo, descripción	El usuario solicita al sistema mostrar la lista de canciones. El sistema muestra la lista de canciones.
Entradas	Ninguna.
Precondiciones	Ninguna.
Salidas	Se muestra al usuario una lista con las canciones.
Postcondición si éxito	Se muestra la lista de canciones al usuario.
Postcondición si fallo	No se muestra la lista de canciones.
Actores principales	Usuario.
Actores secundarios	Base de datos
Secuencia principal	<ol style="list-style-type: none"> 1- Se comprueba si hay canciones en la lista. 1- Se muestran las canciones

Secuencia alternativa	1- No hay canciones en la lista
------------------------------	---------------------------------

3.2.3. Módulo Comments

Crear comentario:

CASO DE USO #1	Crear comentario
Objetivo, descripción	El usuario selecciona una recomendación para escribir un comentario sobre ella.
Precondiciones	Que haya una recomendación si quiero escribir.
Salidas	Se muestra el comentario.
Postcondición si éxito	Escribo el comentario.
Postcondición si fallo	No escribo el comentario.
Actores principales	Usuario y Administrador
Actores secundarios	Base de datos
Secuencia principal	1- Seleccionar la recomendación. 2- Si existe S-1. 3- Si no existe S-2.
Secuencia alternativa	S-1- Escribo el comentario S-2- No escribo, porque no existe dicha recomendación

Borrar un comentario:

CASO DE USO #2	Borrar comentario
Objetivo, descripción	El usuario borra un comentario.

Precondiciones	
Salidas	Elimino el comentario.
Postcondición si éxito	Si existe dicho comentario lo elimino.
Postcondición si fallo	Si no existe dicho comentario, lo notifico.
Actores principales	Usuario y Administrador
Actores secundarios	Base de datos
Secuencia principal	1- Busco dicho comentario. 2- Si existe S-1. 3- Si no existe S-2.
Secuencia alternativa	S-1- Lo elimino. S-2- Notifico que no existe.

3.2.4. Módulo Estadísticas

3.2.5. Módulo PlayLists

Crear playlist:

CASO DE USO #1	Crear playlist
Objetivo, descripción	El usuario crea una playlist.
Precondiciones	La playlist no debe existir ya en el sistema.
Salidas	Playlist nueva.
Postcondición si éxito	Se crea la playlist.
Postcondición si fallo	Se informa de que la playlist ya existe.

Actores principales	Usuario
Actores secundarios	Base de datos
Secuencia principal	<p>1- Se pide al usuario un nombre para la playlist. Si este nombre ya está ocupado S-1.</p> <p>2- Se crea la playlist tanto en el sistema como en la base de datos.</p>
Secuencia alternativa	S-1- Se informa de que la playlist ya existe en el sistema.

Borrar una playlist:

CASO DE USO #2	Borrar playlist
Objetivo, descripción	El usuario tiene la opción de borrar una playlist que creó previamente.
Precondiciones	La playlist debe existir previamente en el sistema.
Salidas	Se elimina la playlist del sistema.
Postcondición si éxito	Se informa al usuario de que la playlist se ha eliminado correctamente.
Postcondición si fallo	Se notifica al usuario de que ha habido un error al borrar la playlist y el motivo.
Actores principales	Usuario
Actores secundarios	Base de datos
Secuencia principal	<p>1- Se pide al usuario el nombre de la playlist que desea borrar. Si no existe S-1</p> <p>2- Se elimina la playlist tanto del sistema como de la base de datos.</p>

Secuencia alternativa	S-1- Notifico que no existe.
------------------------------	------------------------------

Crear playlist de favoritos:

CASO DE USO #3	Crear playlist de favoritos
Objetivo, descripción	El usuario busca una canción para añadirla a una playlist.
Precondiciones	Una precondición es que busques una canción para añadirla.
Salidas	Añades tu canción a la playlist.
Postcondición si éxito	Encuentro la canción que quiero añadir, la playlist ya existe y la añadido a esa playlist.
Postcondición si fallo	Intento añadir la una canción, no la encuentro entonces notifico de que no existe dicha canción que intento añadir.
Actores principales	Usuario y Administrador
Actores secundarios	Base de datos
Secuencia principal	1- Busco dicha canción. 2- Si existe S-1. 3- Si no existe S-2. 4- El usuario añade la canción a la playlist
Secuencia alternativa	S-1- Busco la playlist para añadirla, y si esta no existe la creo y la añadido. S-2- Notifico de que no existe dicha canción.

Mostrar playlist de favoritos:

CASO DE USO #4	Mostrar playlist de favoritos
Objetivo, descripción	El usuario busca una playlist para mostrarla.
Precondiciones	Una precondición es que exista dicha playlist.
Salidas	Mostramos la playlist.
Postcondición si éxito	Si existe dicha playlist la mostramos.
Postcondición si fallo	Si no existe dicha playlist, informo de que no existe.
Actores principales	Usuario y Administrador
Actores secundarios	Base de datos
Secuencia principal	1- Busco dicha playlist. 2- Si existe S-1. 3- Si no existe S-2.
Secuencia alternativa	S-1- Muestro la playlist. S-2- Notifico de que no existe dicha playlist.

Mostrar playlist del sistema (solo para usuarios BestTaste+):

CASO DE USO #5	Mostrar playlists del sistema
Objetivo, descripción	El usuario solicita al sistema mostrar todas las playlists pertenecientes a BestTaste+ . El sistema muestra las playlists.
Entradas	Ninguna.
Precondiciones	Ser miembro de BestTaste+ .
Salidas	Se muestra al usuario una lista con todas las playlists de BestTaste+ .

Postcondición si éxito	Se muestran las playlist al usuario.
Postcondición si fallo	No se muestra ninguna playlist.
Actores principales	Administrador y usuario.
Actores secundarios	Base de datos
Secuencia principal	1- Se comprueba si el usuario es miembro de BestTaste+ . 2- Se muestran las playlists.
Secuencia alternativa	1- No es miembro de BestTaste+ . 2- No hay playlists disponibles para el usuario.

3.2.6. Módulo Administradores

Mostrar lista de usuarios:

CASO DE USO #1	Mostrar lista de usuarios
Objetivo, descripción	El administrador solicita al sistema una lista de todos los usuarios, pudiendo establecer un filtro. El sistema muestra bajo las condiciones establecidas los usuarios que las cumplan.
Entradas	El administrador proporciona los siguientes datos (opcional): Condiciones para el filtro.
Precondiciones	No hay.
Salidas	Se muestra al administrador una lista con todos los usuarios que cumplan el filtro.
Postcondición si éxito	Se es mostrada la lista de usuarios al administrador.
Postcondición si	No se le muestra ninguna lista al administrador.

fallo	
Actores principales	Administrador y usuario
Actores secundarios	Base de datos
Secuencia principal	1- Se pide mostrar por pantalla la lista de usuarios. 2- Se establece de manera voluntaria un filtro. 3- Si hay al menos un usuario que lo cumpla, se muestran.
Secuencia alternativa	S-1- Error, no existen usuarios. S-2- Error, no existen usuarios que cumplan la condición establecida en el filtro.

Dar de alta a sponsor:

CASO DE USO #2	Dar de alta a sponsor
Objetivo, descripción	El administrador mete los datos necesarios de un sponsor para poder crearlo en el sistema. Este lee los datos proporcionados y le da de alta para que ahora su anuncio salga por la app.
Entradas	Nombre del sponsor, tipo de producto, video de 15 segundos promocionando el producto y una breve descripción.
Precondiciones	Ninguno de los campos de entrada deben estar vacíos. El sponsor debe ser apropiado para una audiencia 13+. El video no debe mostrar nada inapropiado.
Salidas	-
Postcondición si éxito	Se añade el sponsor a la base de datos con un ID único. Se empieza a mostrar el video proporcionado junto a la breve descripción a los usuarios.

Postcondición si fallo	No se añade el sponsor a la base de datos. El sponsor posiblemente ya existiese en la base de datos. Se informa de ello.
Actores principales	Administrador.
Actores secundarios	Base de datos.
Secuencia principal	<ol style="list-style-type: none"> 1. Pedir los datos 2. Se introducen 3. Se comprueba que sean válidos. Si no lo son: S-1 4. Se almacena en la base de datos 5. Se empieza a mostrar el anuncio en la aplicación
Secuencia alternativa	S-1: Error al leer los datos. Datos existen ya en la base de datos

Dar de baja a sponsor:

CASO DE USO #3	Dar de baja a sponsor
Objetivo, descripción	El administrador mete los datos existentes del sponsor al sistema. Este lee los datos proporcionados y le da de baja y su anuncio deja de mostrarse en la aplicación.
Entradas	Nombre del sponsor e ID único.
Precondiciones	El sponsor debe existir en la base de datos
Salidas	Mensaje de éxito en la eliminación del sponsor
Postcondición si éxito	Se elimina al sponsor de la base de datos. Se deja de mostrar el video proporcionado junto a la breve descripción a los usuarios.
Postcondición si fallo	El sponsor continúa formando parte de la base de datos, o no existía para empezar. Se informa de ello.
Actores principales	Administrador.
Actores secundarios	Base de datos.

Secuencia principal	<ol style="list-style-type: none"> 1. Pedir los datos 2. Se introducen 3. Se comprueba que sean válidos. Si no lo son: S-1 4. Se elimina de la base de datos 5. Se deja de mostrar el anuncio en la aplicación
Secuencia alternativa	S-1: Error al leer los datos. Datos no existen en la base de datos

Mostrar lista de administradores:

Caso de uso #4	Mostrar lista:
Objetivos, descripciones	Mostrar la lista de administradores para que el usuario tenga una manera de ver quienes son los admins en caso de que necesite contactar con ellos
Entradas	-
Precondiciones	Que exista algún admin en la base de datos
Salidas	Una lista de todos los administradores de la aplicación
Postcondición si éxito	Se muestra una lista al usuario de todos los administradores, junto a una manera de contactar con ellos por si fuera necesario.
Postcondición si fallo	Un mensaje de error por la falta de un administrador en la base de datos.
Actores principales	Administradores, Usuarios
Actores secundarios	Base de datos
Secuencia principal	<ol style="list-style-type: none"> 1. Selección de la opción de ver lista de administradores 2. Comprobación de la existencia de dicha lista (Si no existe, S-1) 3. Salida de la lista con los datos correspondientes
Secuencia alternativa	S-1: Mensaje de error ya que no existe ningún administrador

3.3. Requisitos de Rendimiento

Aunque si que es verdad que al principio nuestra aplicación no contará con millones de usuarios, sí que esperamos tener unos miles. Por esto, nuestra aplicación contará con los suficientes recursos para soportarlo. El número de transacciones por segundo será también muy elevado, siendo este alrededor de cien por segundo.

Tomando todo esto en cuenta, nuestra aplicación estará hecha para soportar alrededor de los cien mil usuarios, y mil transacciones por segundo, y a medida que nuestra aplicación crezca, aumentaremos estos números a millones de usuarios y la cantidad de transacciones a cien mil por segundo.

Todo esto lo hemos determinado basándonos en la frecuencia de uso de nuestra aplicación, la cual será elevada, y la cantidad de registros que se espera almacenar, que serán cientos de miles.

3.4. Atributos del Sistema

Para la fiabilidad de la aplicación nos aseguraremos de hacer varios casos de prueba para así garantizar el buen funcionamiento de la aplicación, sin ningún tipo de errores. De cara a la seguridad, estableceremos un sistema de login con contraseña para evitar cualquier tipo de problemas.

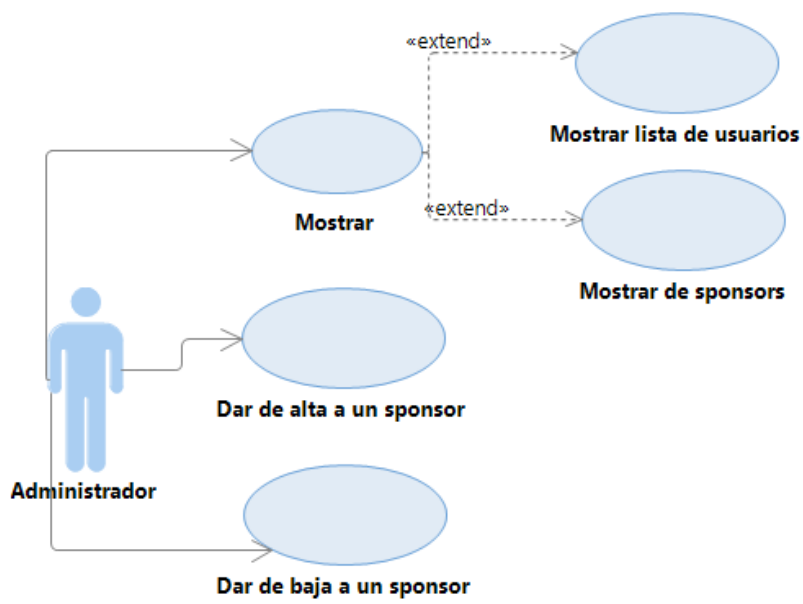
Buscamos tener una aplicación que sea fiable, segura y mantenible, y todo esto lo aseguraremos mirando que toda acción se realice de manera correcta, almacenándose, además, en la base de datos, asegurándonos también de que toda información personal del usuario se guarde de manera encriptada en la base de datos para poder protegerlos frente a ataques externos, y manteniendo en todo momento una buena documentación para que a la hora de mantener la aplicación, o de la entrada de nuevos desarrolladores, todos los requisitos queden claros.

4. Apéndices

4.1. Diagramas de Casos de Uso

4.1.1. Gestión de Usuarios

Los usuarios son los que usan las distintas funcionalidades del sistema. Es decir, estos son los que crearán las recomendaciones, los comentarios, los que buscan canciones, ven su lista y además, hacen sugerencias de canciones a añadir a dicha lista. También son los que tienen acceso a las estadísticas, todo esto encima de, por supuesto, darse de alta o baja en la aplicación.



4.2. Diagramas de funciones: