

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

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.

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 del proyecto se resumen en: “Alta” o “Añadir”, que son las encargadas de crear/registrar un elemento del módulo correspondiente, “Modificar” que permite hacer cambios en nuestro caso sobre la propia cuenta del usuario o la lista de canciones, “Baja” o “Borrar”, que consiste en borrar la información de la base de datos y finalmente “Mostrar” que como bien indica su nombre, será la función encargada de mostrar lo que corresponda.

Usuario:

- **Dar de alta:** Registrar al usuario en la base de datos.
- **Dar de baja:** eliminar la cuenta.
- **Log in:** Acceso a cuenta ya existente en la base de datos.
- **Log out:** Salir de la cuenta.
- **Modificar:** cambiar los datos que el usuario desee.
- **Mostrar:** para mostrar la lista de usuarios.
- **Añadir canción:** Añade una canción, si no existía ya, a la lista de canciones

Recomendación:

- **Crear:** El usuario crea una recomendación. En esta función, el sistema le pide al usuario una serie de datos, los que incluyen la canción que desea recomendar, una sección de la canción a mostrar cómo preview con duración de 15 segundos, así como una descripción para la recomendación.
- **Borrar:** Se elimina la recomendación.
- **Canciones:**
 - **Mostrar:** Enseña dicha lista de canciones.

Comentarios:

- **Crear:** El usuario crea un comentario bajo recomendación.
- **Borrar:** El usuario elimina el comentario.

Playlist:

- **Crear:** Se crea una playlist con un nombre que el usuario desee, y al crear se establece si se trata de una playlist pública o privada
- **Borrar:** elimina la playlist
- **Mostrar:** muestra las playlist del usuario
- **Mostrar playlist del sistema:** Lo mismo que el mostrar, solo que de una playlist creada por el sistema

Estadísticas:

- **Mostrar estadísticas:** Crea una lista de estadísticas de un usuario o canción según lo pedido por el usuario.

Administrador:

- **Dar de alta a sponsor:** Registrar un sponsor en la base de datos.
- **Dar de baja a sponsor:** Eliminar un sponsor
- **Mostrar:** Encargado de buscar y mostrar a los administradores.

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 nuestra aplicación, de modo que quede explicado tanto el funcionamiento de nuestra interfaz, como de las funciones y el sistema.

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

Dar de alta a sponsor:

CASO DE USO #1	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 del producto.
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. 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 #2	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.
Precondiciones	El nombre 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 #3	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

CASO DE USO #3	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.

CASO DE USO #4	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.

CASO DE USO #5	Hacer 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.

	<ol style="list-style-type: none"> 5. Si la descripción es inadecuada o sobrepasa el máximo de caracteres (b). 6. Se publica la recomendación. 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.

CASO DE USO #6	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, privandola de su visibilidad.
Postcondición si fallo	— — —
Actores principales	Usuario
Actores secundarios	Administrador. 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	<ol style="list-style-type: none"> a. regresa al punto 1.

3.3. Requisitos de Rendimiento

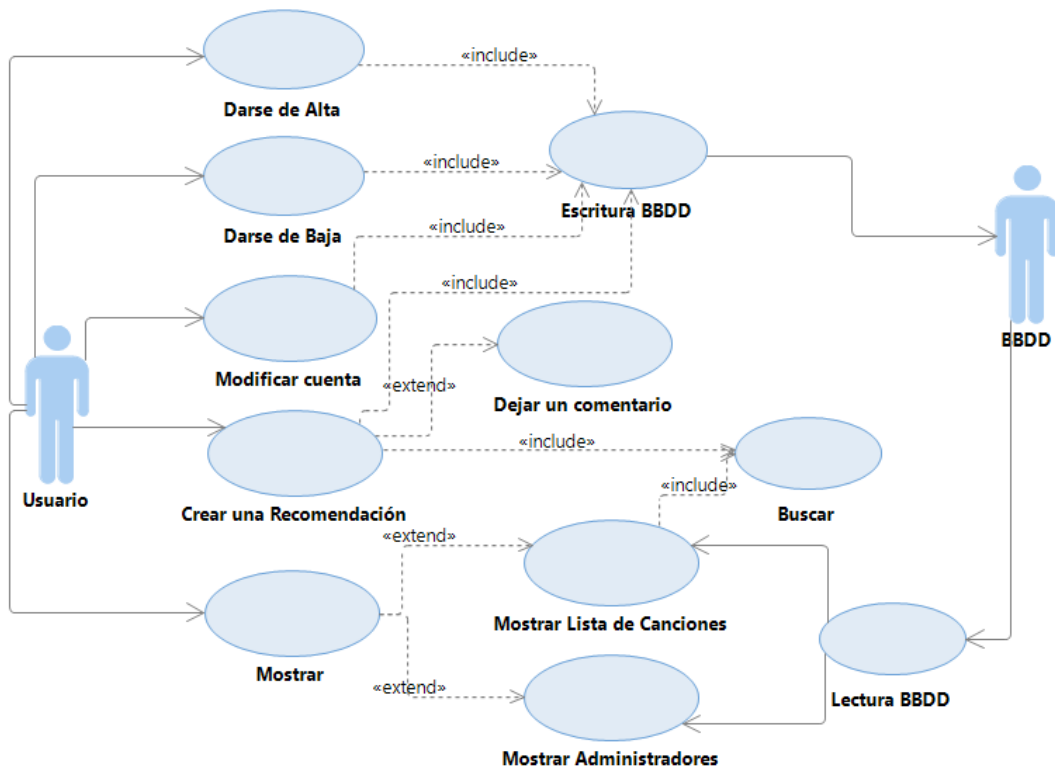
3.4. Atributos del Sistema

4. Apéndices

4.1. Diagramas de Casos de Uso

4.1.1. Gestión de Usuarios

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.



4.1.2. Gestión de Administrador

Los administradores tienen una funcionalidad más simple que la de los usuarios dentro de la aplicación, ya que estos no realizan recomendaciones ni comentarios, si no más bien se encargan de los sponsors y de acceder a la lista de usuarios.

