

TABLA DE ESPECIFICACIÓN DEL PROBLEMA DE INGENIERÍA DE SOFTWARE, identificando los siguientes elementos

CLIENTE	NEO TUNES
USUARIO	consumidores y productores
REQUERIMIENTOS FUNCIONALES	<ol style="list-style-type: none"> 1. Registrar usuarios productores, artistas y creadores de contenido. 2. Registrar usuarios consumidores, estándar y premium. 3. Registrar canciones y podcasts. 4. Crear una lista de reproducción. 5. Editar una lista de reproducción 6. Compartir una lista de reproducción. 7. Simular la reproducción de una canción o podcast (estándar y premium). 8. Comprar una canción. 9. Generar informes con los datos registrados:.
CONTEXTO DEL PROBLEMA	Una empresa de danesa quiere entrar en el mercado con su aplicación de streaming de música y contenido de audio
REQUERIMIENTOS NO FUNCIONALES	contemplar la creación futura de otros tipos de usuario, tanto consumidores como productores.

Tabla de análisis de requerimientos funcionales

Nombre o identificador	R1- Registrar usuarios productores, artistas y creadores de contenido.		
Resumen	Se debe poder registrar artistas y creadores de contenido con todos los datos necesarios		
Entradas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	nickname	String	debe ser unico
	id	String	
	name	String	

	type	String	debe ser una opcion valida en el menu
	url	String	
Actividades generales necesarias para obtener los resultados	se deben pedir los datos necesarios, verificar que no esté registrado el nickname y crear un objeto		
Resultado o postcondición	imprimir un mensaje de éxito o error y guardarlos como un objeto en el arraylist de usuarios		
Salidas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	msg	String	

Nombre o identificador	R2- Registrar usuarios consumidores, estándar y premium.		
Resumen	Se debe poder registrar consumidores estandar y premium con todos los datos necesarios		
Entradas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	nickname	String	
	lvl	String	Debe ser un a opcion valida
	id	String	
Actividades generales necesarias para obtener los resultados	se deben pedir los datos necesarios, verificar que no esté registrado el nickname y crear un objeto		
Resultado o postcondición	imprimir un mensaje de éxito o error y guardar los datos del usuario en el arraylist de usuarios		
Salidas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	msg	String	

Nombre o identificador	R3- Registrar canciones y podcasts.		
Resumen	Se deben poder agregar canciones a los usuarios productores artistas		
Entradas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	name	String	
	álbum/description	String	
	albumURL/imageURL	String	
	duration	int	
	choice	String	Debe ser una opcion valida(1 o 2)

	artistName/creatorName	String	Debe ser un productor registrado
	price	double	solo si es una cancion
	genre/cat	int	
Actividades generales necesarias para obtener los resultados	se deben pedir los datos necesarios y crear un objeto		
Resultado o postcondición	imprimir un mensaje de éxito o error y guardar los datos del audio en el arraylist de audios		
Salidas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	msg	String	

Nombre o identificador	R4- Crear una lista de reproducción.		
Resumen	Se debe crear una lista de reproducción con canciones o podcasts		
Entradas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	name	String	
Actividades generales necesarias para obtener los resultados	crear el objeto playlist y una matriz que se va a usar para crear el codigo identificador		
Resultado o postcondición	Almacenar los datos y mandar un mensaje de éxito o error		
Salidas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	msg	String	

Nombre o identificador	R5- Editar una lista de reproducción.		
Resumen	Se debe poder editar una lista de reproducción		
Entradas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	name	String	debe ser el nombre de una playlist ya creada
	audio	String	debe ser un audio ya creado
Actividades generales necesarias para obtener los resultados	Verificar que la playlist exista y que el audio exista para añadirlo, o que ya esté en la playlist para eliminarlo		

Resultado o postcondición	imprimir mensaje de confirmación		
Salidas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	msg	String	

Nombre o identificador	R6- Compartir una lista de reproduccion		
Resumen	Se debe generar un codigo de la playlist recorriendo una matriz con numeros aleatorios en cierto recorrido de la matriz dependiendo del tipo de audio que hay en la playlist		
Entradas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	name	String	debe ser una playlist existente
Actividades generales necesarias para obtener los resultados	Se debe crear una matriz con numeros aleatorios, buscar la playlist y verificar que tipo de audio hay en la playlist para definir cual recorrido se va a hacer		
Resultado o postcondición	imprimir mensaje de confirmación con el codigo numerico generado		
Salidas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	msg	String	

Nombre o identificador	R7- Simular la reproducción de una canción o podcast (estándar y premium)		
Resumen	El programa debe simular que se esta reproduccionado un audio		
Entradas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	name	String	debe ser un audi registrado en el programa
	nickname	String	debe ser un usuario registrado
Actividades generales necesarias para obtener los resultados	Verificar que el audio exista y ver que tipo de usuario es para saber si se van a reproducir anuncios		
Resultado o postcondición	Un mensaje simulando que se esta reproduciendo la cancion, y si es un usuario estandar reproducir un anuncio si se cumplen las condiciones para reproducir anuncios. Ademas se deben actualizar los datos de reproducciones totales de esa cancion, de tiempo reproducido de un usuario y de un artista y demas datos.		
Salidas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	msg	String	

Nombre o identificador	R8- Comprar una canción		
Resumen	Los consumidores pueden elegir comprar una canción y esta se guardará en su objeto de usuario		
Entradas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	nickname	String	Debe ser un usuario existente
	name	String	Debe ser un audioregistrado
Actividades generales necesarias para obtener los resultados	Revisar que el usuario esté registrado y que tipo de usuario es, y que la canción exista. Almacenar los datos de la canción guardada en el objeto del consumidor con todos los datos necesarios, y actualizar el número de veces comprada.		
Resultado o postcondición	Imprimir un mensaje de confirmación		
Salidas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	msg	String	

Nombre o identificador	R9- Generar informes con los datos registrados:		
Resumen	<p>Se deben generar los siguientes informes:</p> <p>Para cada tipo de audio, canciones y podcast, informar el acumulado total de reproducciones en toda la plataforma.</p> <p>Informar el género de canción más escuchado (nombre y número de reproducciones) para un usuario específico y para toda la plataforma y su número de reproducciones.</p> <p>Informar la categoría de podcast más escuchada (nombre y número de reproducciones) para un usuario específico y para toda la plataforma.</p> <p>De cada uno de los integrantes del Top 5 de artistas y del Top 5 de creadores de contenido en la plataforma, informar el nombre y número de reproducciones totales.</p> <p>De cada uno de los integrantes del Top 10 de canciones y del Top 10 de podcast, informar el nombre, género o categoría y número total de reproducciones.</p> <p>De cada género, informar el número de canciones vendidas y el valor total de ventas (\$).</p> <p>De la canción más vendida en la plataforma, informar el número total de ventas y el valor total de venta (\$)</p>		
Entradas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	option	String	Debe ser una opción válida de el menú de los informes
	nickname	String	Debe ser un usuario registrado.

Actividades generales necesarias para obtener los resultados	Desplegar un menu con las opciones de generacion de datos y dependiendo de la opcion seleccionada, recolectar los datos necesarios		
Resultado o postcondición	Imprimir un mensaje con los datos pedidos		
Salidas	Nombre entrada	Tipo de dato	Condición de selección o repetición
	msg	String	

Tabla de trazabilidad

Requerimiento Funcional	Nombre de la Clase	Nombre del método
ReqFunc001 Registro de usuarios productores	Clase NeoTunesApp (paquete ui)	registerProducers()
	Clase Controler	registerProducers()
	Clase Controler	isRegistered()
	Clase Controler	users.add()
ReqFunc002 Registro de usuarios consumidores	Clase NeoTunesApp (paquete ui)	registerConsumers()
	Clase Controler	registerConsumers()
	Clase Controler	isRegistered()
	Clase Controler	users.add()
ReqFunc003 Registro de canciones y podcast.	Clase NeoTunesApp (paquete ui)	registerAudio()
	Clase Controler	registerAudio()
	Clase Controler	audios.add()
ReqFunc004 Crear una playlist.	Clase NeoTunesApp (paquete ui)	createPlaylist()
	Clase Controler	createPlaylist()
	Clase Controler	playlists.add()
ReqFunc005 Editar una playlist	Clase NeoTunesApp (paquete ui)	editPlaylist()
	Clase Controler	editPlaylist()
	Clase Controler	playlists.get().add() playlists.get().remove();

ReqFunc006 Compartir una lista de reproduccion	Clase NeoTunesApp (paquete ui)	sharePlaylist()
	Clase Controler	sharePlaylist() exists()
	Clase Playlist	getName() generateCode()
ReqFunc007 Simular la reproducción de una canción o podcast (estándar y premium)	Clase NeoTunesApp (paquete ui)	playAudio()
	Clase Controler	playAudio() isAvailable()
	Clase Consumer	addPodcastListen() addSongListen()
	Clase Standard	getReproductions() addReproductions()
	Clase Adds	play()
ReqFunc008 Comprar una canción	Clase NeoTunesApp (paquete ui)	buySong()
	Clase Controler	buySong()
	Clase Consumers	buySong()
	Clase Song	buy()
	Clase Premium Clase Standard	buySong()
ReqFunc009 Generar informes con los datos registrados:	Clase NeoTunesApp (paquete ui)	generateInfo()
	Clase Controler	totalReproductions() bestGenre() bestCat() top5() top10Audios() genreInfo() mostSoldSong() isRegistered()
	Clase Audio	getReproductions() getName()
	Clase Song	getNumberOfSales() getTotalSold()
	Clase Consumer	getFavGenre() getFavCat()
	Clase Producer	getNickname() getReproductions()

Diagrama de Clase:

