# Tarea - Identificación del problema y análisis de requerimientos

## Caso de Estudio:

|  |  |
| --- | --- |
| Cliente | Compañía NeoTunes |
| Usuario | * Clientes de la plataforma (oyentes y creadores) |
| Requerimientos funcionales | RF1-Registrar usuarios productores, artistas y creadores de contenido.  RF2-Registrar usuarios consumidores, estándar y premium.  RF3-Registrar canciones y podcasts.  RF4-Crear una lista de reproducción.  RF5-Editar una lista de reproducción.  RF6-Compartir una lista de reproducción.  RF7-Simular la reproducción de una canción o podcast para los usuarios estándar y premium  RF8-Comprar una canción.  RF9-Para cada tipo de audio, canciones y podcast, informar el acumulado total de reproducciones en toda la plataforma.  RF10-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.  RF11-Informar la categoría de podcast más escuchada (nombre y número de reproducciones) para un usuario específico y para toda la plataforma.  RF12-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.  RF13-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.  RF14-De cada género, informar el número de canciones vendidas y el valor total de ventas ($).  RF15-De la canción más vendida en la plataforma, informar el número total de ventas y el valor total de venta ($). |
| Contexto del problema | La compañía NeoTunes necesita una aplicación para ingresar al mercado musical, por lo cual se va a diseñar un prototipo de aplicación que funcione para que los usuarios de la plataforma puedan crear diferentes audios o comprarlos. Además de esto, los usuarios que compren canciones serán dueños de la música que adquieran.  Los usuarios pueden ser de tipo estándar o premium cuya diferencia principal radica en la cantidad de música que pueden comprar o las listas de reproducción que creen. Mientras que los usuarios premium pueden comprar canciones y crear listas de reproducción de forma ilimitada, los usuarios estándar tienen un límite para esto. |
| Requerimientos no funcionales | RNF1- La aplicación debe ser escalable (Debe permitir agregar nuevos tipos de usuarios o audios en el futuro). |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF1-Registrar usuarios productores, artistas y creadores de contenido. | | |
| Resumen | Dados los datos de los usuarios productores el sistema deberá validar que no exista un usuario con el mismo id y posteriormente agregará el usuario según su tipo. | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| nickname | String |  |
|  | id | String |  |
| Fecha de vinculación | GregorianCalendar |  |
| Nombre | String |  |
| Url | String |  |
| Actividades generales necesarias para obtener los resultados | El sistema verifica que el id del usuario no este repetido, y validara que tipo de usuario productor es (Artista o creador de contenido), posteriormente agregara el usuario a la lista de usuarios | | |
| Resultado o Postcondición | El usuario productor agregado en la lista de Usuarios. | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF2-Registrar usuarios consumidores, estándar y premium. | | |
| Resumen | Dados los datos de un usuario consumidor, el sistema deberá verificar que el id ingresado no exista y si es el caso agregarlo a la lista de usuarios. | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| nickName | String |  |
|  | id | String |  |
|  | Fecha de vinculacion | GregorianCalendar |  |
| Actividades generales necesarias para obtener los resultados | El sistema verifica que el usuario no haya sido registrado previamente, después de eso, validara que tipo de usuario consumidor es (Estándar-premium) y lo agregara a la lista de usuarios. | | |
| Resultado o Postcondición | Un mensaje de confirmación acerca de la adicipon del usuario. | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| message | String |  |
| Nombre o identificador | RF3-Registrar canciones y podcasts. | | |
| Resumen | Dados los datos de un audio, el sistema valida que tipo de audio es y que su nombre no este repetido, posteriormente lo agrega a la lista de audios del sistema | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| nombre | String |  |
| url | String |  |
| duracion | double |  |
| Numero de veces reproducida | Int |  |
| Album | String |  |
| Genero | Int |  |
| Valor de venta | Double |  |
| Numero de ventas | Int |  |
| descripcion | String |  |
| Categoria | int |  |
| Actividades generales necesarias para obtener los resultados | El sistema verifica que el nombre del audio no este registrado previamente, posteriormente, valida que tipo de audio se esta registrando y lo agrega a la lista de audios del sistema | | |
| Resultado o Postcondición | Un mensaje que contiene la información de la adición del audio | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| Message | String |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF4-Crear una lista de reproducción. | | |
| Resumen | Dado los datos iniciales de una lista de reproducción dados por un usuario consumidor, el sistema valida que tipo de usuario consumidor es y en caso de ser un usuario estándar el sistema verifica que tenga espacio suficiente para agregar la lista de reproducción. | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| nombre | String |  |
|  | Canciones | Audio |  |
| Actividades generales necesarias para obtener los resultados | El sistema verifica que tipo de usuario es, para posteriormente validar que el usuario tenga el espacio disponible para agregar la playlist | | |
| Resultado o Postcondición | Un mensaje con la información sobre la adición de la lista de reproduccion | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| message | String |  |
| Nombre o identificador | RF5-Editar una lista de reproducción. | | |
| Resumen | Dado el nombre de una lista de reproducción ingresado por un usuario de tipo consumidor el sistema debe verificar que la lista exista y posteriormente, editarla con los datos dados por el usuario. | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| nombre | String |  |
|  | IdUsuario | String |  |
| Actividades generales necesarias para obtener los resultados | El sistema busca el usuario con el id indicado y busca en su lista de playlist si la playlist indicada existe, posteriormente permite que el usuario cambie información de su lista de reproduccion | | |
| Resultado o Postcondición | Un mensaje de confirmación sobre el cambio en la playlist | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| message | String |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF6-Compartir una lista de reproducción. | | |
| Resumen | Dado el id de un Usuario consumidor y el nombre de una playlist, se debe mostrar en pantalla el código que identifica a la playlist | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| nombre | String |  |
|  | id | String |  |
| Actividades generales necesarias para obtener los resultados | El sistema busca al usuario con el id indicada, y cuando lo encuentra busca la playlist con el nombre indicado y cambia la información de esta por nueva información indicada por el usuario. | | |
| Resultado o Postcondición | Mensaje de confirmacion | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| message | String |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF7-Simular la reproducción de una canción o podcast para los usuarios estándar y premium | | |
| Resumen | Dado el id de un usuario Consumidor, y el nombre de una canción el sistema verifica que el usuario haya comprado la canción y la reproducirá de acuerdo a el tipo de usuario que sea | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| id | String |  |
|  | nombre | String |  |
| Actividades generales necesarias para obtener los resultados | El sistema debe verificar que el usuario exista y que sea una instancia de consumidor, posteriormente verifica que tenga en su lista de canciones la cncion que debe reproducir y después de validar que tipo de usuario es, reproduce la canción con las condiciones impuestas. | | |
| Resultado o Postcondición | El mensaje con la reproducción de la canción. | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| message | int |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF8-Comprar una canción. | | |
| Resumen | Dado el id de un usuario consumidor y el nombre de una canción, el sistema debe verificar que el usuario exista y la canción también, para después de esto asociar la canción con el usuario correspondientes después de verificar el tipo de usuario y dado el caso el limite de canciones establecido. | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| nickName | String |  |
|  | name | String |  |
| Actividades generales necesarias para obtener los resultados | El sistema verifica que exista el usuario y la cancion indicada, posteriormente valida el tipo de usuario que es y si aplica el limite de canciones que puede comprar | | |
| Resultado o Postcondición | La cancion agregada a la lista de canciones del usuario | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF9-Para cada tipo de audio, canciones y podcast, informar el acumulado total de reproducciones en toda la plataforma. | | |
| Resumen | El sistema busca para cada tipo de audio la información sobre el acumulado de reproducciones y los informa al usuario. | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |
| Actividades generales necesarias para obtener los resultados | El sistema verifica para cada tipo de audio el acumulado de reproducciones y lo muestra al usuario. | | |
| Resultado o Postcondición | Un mensaje con la información solicitada. | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| message | String |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF10-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. | | |
| Resumen | Dado el nickname de un usuario o la indicación por parte del usuario, el sistema verifica según la instrucción el genero mas reproducido y lo informa | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| nickname | String | Si el usuario indica buscar el genero mas repetido para un usuario especifico |
| Actividades generales necesarias para obtener los resultados | El sistema verifica que el usuario indicado existe (si aplica) y después busca entre los géneros existentes el que tenga la mayor reproducción. | | |
| Resultado o Postcondición | Un mensaje por pantalla con la información solicitada. | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| Message | String | Si al menos uno de los generos tiene una reproducción. |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF11-Informar la categoría de podcast más escuchada (nombre y número de reproducciones) para un usuario específico y para toda la plataforma. | | |
| Resumen | Dado el nickname de un usuario o la indicación por parte del usuario, el sistema verifica según la instrucción la categoría mas reproducida y lo informa | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| nickname | String | Si se indica buscar la categoría mas reproducida para un usuario especifico |
| Actividades generales necesarias para obtener los resultados | El sistema verifica que el usuario indicado existe (si aplica) y después busca entre las categorias existentes la que tenga la mayor reproducción. | | |
| Resultado o Postcondición | Un mensaje en pantalla con la información solicitada. | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| message | String | Si al menos una de las categorías presenta al menos una reproduccion |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF12-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. | | |
| Resumen | El sistema verifica los 5 artistas y los 5 creadores de contenido con la mayor cantidad de reproducciones e informa el nombre y número total de reproducciones. | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |
| Actividades generales necesarias para obtener los resultados | El sistema verifica entre la colección de artistas los que tengan numero de reproducciones por tipo e informa de cada uno su nombre y acumulado de reproducciones. | | |
| Resultado o Postcondición | Un mensaje en pantalla con la información requerida. | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| message | String | Si al existe al menos un usuario productor y tiene al menos una reproduccion |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF13-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. | | |
| Resumen | El sistema busca las 10 canciones y los 10 podcast con la mayor cantidad de reproducciones e informa de ellos la información requerida | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |
| Actividades generales necesarias para obtener los resultados | El sistema busca en la colección de audios, por cada tipo de audio los 10 con la mayor cantidad de reproducciones, posteriormente informa el nombre, generi o categoría(según corresponda) y el acumulado de reproducciones. | | |
| Resultado o Postcondición | Un mensaje en pantalla con la información solicitada. | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF14-De cada género, informar el número de canciones vendidas y el valor total de ventas ($). | | |
| Resumen | EL sistema verifica por cada genero la cantidad de canciones vendidas y el acumulado de las ventas. Posteriormente, muestra esta información al usuario. | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |
| Actividades generales necesarias para obtener los resultados | El sistema verifica en la colección de audios los audios que sean una instancia de cancion y cuenta por cada genero cuantas canciones han sido vendidas y el total de estas ventas. Esta información es mostrada al usuario. | | |
| Resultado o Postcondición | Un mensaje en pantalla con la información solicitada. | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| Message | String | Si al menos existe una cancion y ha sido vendida una vez. |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | RF15-De la canción más vendida en la plataforma, informar el número total de ventas y el valor total de venta ($). | | |
| Resumen | El sistema busca entre la colección de audios entre los audios que sean canciones cual es la cancion mas vendida y muestra la información solicitada. | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |
| Actividades generales necesarias para obtener los resultados | El sistema busca en la colección de audios aquellos que sean instancia de cancion y posteriormente verifica la cancion mas vendida e informa el numero de ventas y el valor total de las ventas. | | |
| Resultado o Postcondición | Un mensaje en pantalla con la información solicitada. | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| message | String | Si existe al menos una cancion y ha sido comprada al menos una vez |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador |  | | |
| Resumen |  | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |
| Actividades generales necesarias para obtener los resultados |  | | |
| Resultado o Postcondición |  | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| RF1-Registrar usuarios productores, artistas y creadores de contenido. | NeoTunesApp  ui | showMenu() |
| NeoTunesApp  ui | executeOption() |
| NeoTunesApp  ui | addProductor() |
|  | NeoTunesController  model | addUser() |
|  | NeoTunesController  model | searchUser() |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| RF2-Registrar usuarios consumidores, estandar y premium. | NeoTunesApp  ui | showMenu() |
| NeoTunesApp  ui | executeOption() |
| NeoTunesApp  ui | addConsumer() |
|  | NeoTunesController  model | addUser() |
|  | NeoTunesController  model | searchUser() |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| RF3-Registrar canciones y podcast. | NeoTunesApp  ui | showMenu() |
| NeoTunesApp  ui | executeOption() |
| NeoTunesApp  ui | addProductor() |
|  | NeoTunesController  model | existProducer() |
|  | NeoTunesController  model | showArtist() |
|  | NeoTunesController  model | searchArtist() |
|  | NeoTunesController  model | showContentCreator() |
|  | NeotunesController  model | searchContentCreator() |
|  | NeoTunesController  model | addUser()  (Sobrecargado) |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| RF4-Crear una lista de reproduccion. | NeoTunesApp  ui | showMenu() |
| NeoTunesApp  ui | executeOption() |
| NeoTunesApp  ui | addProductor() |
|  | NeoTunesController  Model | existConsumer() |
|  | NeoTunesController  Model | existAudios() |
|  | NeoTunesController  Model | showConsumer() |
|  | NeoTunesController  Model | addPlaylistToConsumer() |
|  | NeoTunesController  model | addAudioToPlaylist() |
|  | NeoTunesController  model | searchUser() |
|  | NeoTunesController  model | searchAudio() |
|  | *Consumer*  model | *addPlaylist()* |
|  | *Consumer*  *model* | *addAudioToPlaylist()* |
|  | *Consumer*  model | *searchPlaylist()* |
|  | *Estándar*  *model* | *isAvailablePlaylist()* |
|  | Playlist  model | addAudio() |
|  | Playlist  model | *searchAudio()* |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| RF5-Editar una lista de reproduccion. | NeoTunesApp  ui | showMenu() |
| NeoTunesApp  ui | executeOption() |
| NeoTunesApp  ui | addProductor() |
|  | NeoTunesController  Model | existConsumer() |
|  | NeoTunesController  model | existPlaylist() |
|  | NeoTunesController  model | searchUser() |
|  | *Consumer*  model | *existPlaylist()* |
|  | NeoTunesController  model | editPlaylist() |
|  | NeoTunesController  model | ShowConsumer() |
|  | NeoTunesController  model | showPlaylist() |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |