**Names:**

* Danna Valentina López Muñoz-**A00395625**
* Leidy Daniela Londoño Candelo- **A00392917**
* Isabella Huila Cerón-**A00394751**

|  |  |
| --- | --- |
| Client | Neo-Tunes Company |
| User | Anyone who downloads the app |
| Functional requirements | **R1**: Register producer users of type artists and content creators.  **R2:** Register consumer users of type standard and premium.  **R3:** Register songs and podcasts.  **R4:** Create a playlist  **R5:** Edit a playlist  **R6:** Share a playlist  **R7:** Simulate playback of a song or podcast  **R8:** Buy a song  **R9:** Report the total accumulated reproductions on the entire platform for each type of audio, songs, and podcast.  **R10:** Inform the genre of the most listened song for a specific user and for the entire platform.  **R11:** Inform the most listened podcast category for a specific user and for the entire platform.  **R12:** Report the name and number of total views for the members of the Top 5 artists and Top 5 content creators.  **R13:** Report the name, genre or category and total number of reproductions of each audio of the Top 10 songs and the Top 10 podcast.  **R14:** Report the number of sold songs and the total value of sales for each genre.  **R15:** Report the number of sales and the total sales value of the best-selling song on the platform.  **R16:** Show ads to Standard Consumer users. |
| Problem context | * Develop a software prototype that allows Neo-Tunes to enter the market for streaming music and audio content. The company's business model has two focuses, user subscriptions and the sale of songs through the platform. |
| No functional requirements | **RN1:** the system must be secure. |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R1**: Register producer users of type artists and content creators. | | |
| Resume | The system must allow the register of producer users, they have a name that identifies them, an id, the date of the profile creation, an URL to the image of them photo and a list of songs or podcasts that they created. | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| name | String | Every user has a name. This can’t be repeated |
| id | String | Every player has an identification number. This can’t be repeated |
| date | LocalDate | Date of bonding to the app |
| url | String | It’s the image of the profile of the user |
| songs/podcast | Song/Podcast | If the user is an Artist, she/he has a list of songs and if is a Content creator then have a list of podcasts. |
| General activities necessary to obtain the results | Save the inputs and all player data | | |
| Result or postcondition | Register a producer user | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Mention if the user was created or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R2**: Register consumer users of type standard and premium. | | |
| Resume | The system must allow the register of consumer users, they have a name that identifies them, an id, the date of the profile creation and a list of playlists with audios (song and podcast). | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| name | String | Every user has a name. This can’t be repeated |
| id | String | Every player has an identification number. This can’t be repeated |
| date | LocalDate | Date of bonding to the app |
| playlists | PlayList | If the user is a Standard, she/he has a limit of thirty playlists and if is a Premium then has unlimited playlists. |
| General activities necessary to obtain the results | Save the inputs and all player data | | |
| Result or postcondition | Register a consumer user | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Mention if the user was created or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R3**: Register songs and podcasts. | | |
| Resume | The system must allow the register of audios the type of song or podcast, they both have a name that identifies them, an URL to a distinctive image, a time of duration and a number of reproductions. The songs also have an album, a genre, a price, and the times each one was sold. The podcasts also have a description, a category and they can NOT be sold. | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| name | String | Every audio has a name |
| url | String | It sends to the distinctive image of the audio |
| duration | double | The time that the audio lasts |
| reproduction | int | The number of reproductions that the audio has |
| album | String | It’s the album to witch a song belongs |
| genre | Genre | The possible genres of the songs are rock, pop, trap and house |
| price | double | The cost of the song |
| timesSold | int | The mount of times that a consumer user buys a song |
| description | String | A little explanation of the podcast |
| category | Category | The possible categories of the podcasts are politics, entertainment, videogames and fashion |
| General activities necessary to obtain the results | Save the inputs and all audios’ data | | |
| Result or postcondition | Register an audio | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Mention if the audio was created or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R4**: Created a playlist | | |
| Resume | The system must allow the register of playlists that belong to the consumer users, they have a name that identifies them, a code that is generate for the system when the playlist is share and a list of audios that are add when the user edit it. | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| name | String | Every playlist has a name |
| General activities necessary to obtain the results | Save the playlist data | | |
| Result or postcondition | Register a playlist | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Mention if the playlist was created or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R5**: Edit a playlist | | |
| Resume | To add or deleted audios of the playlists, selected the name of the consumer user that have the playlist, the name of the playlist to edit and if want to add or deleted an audio | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| nameUser | int | It guarantees that the user is the type of consumer |
| namePlaylist | int | It’s the playlist to edit in case that the user has more than one playlist |
| selection | int | It’s the decision of the user of add or deleted a audio |
| General activities necessary to obtain the results | Save the inputs to edit the playlist | | |
| Result or postcondition | A playlist edited | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Mention if the playlist was edited or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R6**: Share a playlist | | |
| Resume | To generate the code of 16 characters of a playlist it’s use the numbers of a matrix of size 6x6 that is randomly filled with numbers from 0 to 9. A playlist that just has songs take the 16 numbers with a route in form of N. A playlist that just has podcasts take the 16 numbers with a route in form of T. A playlist that has both kind of audios take the 16 numbers traversing the matrix from bottom to top and from right to left when the sum of the variables of the matrix is an odd number greater than one. | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| nameUser | int | It guarantees that the user is the type of consumer |
| namePlaylist | int | It’s the playlist to share and add the attribute code |
| General activities necessary to obtain the results | Save the inputs to share the playlist | | |
| Result or postcondition | A playlist shared and the code of it | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Mention if the playlist was share and the resulting code or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R7**: Simulate playback of a song or podcast | | |
| Resume | It’s simulated for console the playback of an audio. Each time this is done, the number of plays of the selected song is increased by 1. | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| nameUser | int | It guarantees that the user is the type of consumer to reproduces the song |
| nameSong | int | The song that user want listen |
| General activities necessary to obtain the results | Save the inputs to playback the audio | | |
| Result or postcondition | A song reproduced | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Mention if the audio was reproduced or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R8**: Buy a song | | |
| Resume | It’s simulated for console the playback of an audio. Each time this is done, the number of plays of the selected song is increased by 1. | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| nameUser | int | It guarantees that the user that is going to buy is the type of consumer |
| nameSong | int | The song that user want buy |
| confirmation | int | After to show the price of the song to the user, he/she selected if still wants to buy it or not |
| General activities necessary to obtain the results | Save the inputs to buy the song | | |
| Result or postcondition | The purchase of a song | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Mention if the song was bought or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R9**: Report the total accumulated reproductions on the entire platform for each type of audio, songs, and podcast | | |
| Resume | The user selected the type of audio that wants to know the accumulated reproductions in the platform and the system calculated it | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| typeAudio | int | The type of audio (song or audio) that wants to know the reproductions |
| General activities necessary to obtain the results | Save the inputs to count the reproductions of the audios | | |
| Result or postcondition | Accumulated reproductions of all each type of audio | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Show the number of the accumulated reproductions of the whole audios or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R10**: Inform the genre of the most listened song for a specific user and for the entire platform | | |
| Resume | It’s selected if wants to know the most listened song in the system or by a user, if wants to know about a specific user then is selected the name of the consumer user | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| selection | int | The user type if wants to know genre the most reproduced song of the system or the users |
| nameUser | int | The position of the user in case that wants to know the genre of the most reproduced song of this |
| General activities necessary to obtain the results | Save the inputs to know the genre of most reproduced song in every case | | |
| Result or postcondition | Show the genre of the song | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Show the most reproduced song or if there was an error in the process of calculated |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R11**: Inform the most listened podcast category for a specific user and for the entire platform | | |
| Resume | It’s selected if wants to know the most listened podcast in the system or by a user, if wants to know about a specific user then is selected the name of the consumer user | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| selection | int | The user type if wants to know category of the most reproduced podcast of the system or the users |
| nameUser | int | The position of the user in case that wants to know the category of the most reproduced podcast of this |
| General activities necessary to obtain the results | Save the inputs to know the category of the most reproduced podcast in every case | | |
| Result or postcondition | Show the category of the podcast | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Show the most reproduced podcast or if there was an error in the process of calculated |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R12**: Report the name and the number of total views for the members of the top 5 artists and top 5 content creators | | |
| Resume | The system sums the views of every song that belong to an artist or podcast that belong to a content creator and compare this number in every user of the same type to order them from bigger to smaller | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| tUser | int | Selected if wants to know top 5 of the artists or the content creators |
| General activities necessary to obtain the results | Sums the views of the songs or podcast of every producer user and organize the users according with this number, then show name and the number of views of the total sum of the first five of both kind of user producer | | |
| Result or postcondition | Top 5 of producer users in their categories (artist or content creator) | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Show name and total views of the user of the top 5 in their respective type or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R13**: Report the name, genre or category and total number of reproductions of each audio of the Top 10 of song and the Top 10 of podcast | | |
| Resume | The system compares the views of every song or podcast and order everyone according with this number | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| tAudio | int | Selected if wants to know the Top 10 of songs or podcasts |
| General activities necessary to obtain the results | Save the inputs to know the name, genre or category and the reproductions of every audio in their top 10 | | |
| Result or postcondition | Top 10 of every type of audio (songs or podcasts) | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Show the name, category/genre and reproductions of every song/podcast in the top 10 or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R14**: Report the number of sold songs and the total value of sales for each genre. | | |
| Resume | The user selected the genre that wants to know the value of sales reproductions in the platform and the system calculated it | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
| genre | int | The genre that wants to know the value of sales |
| General activities necessary to obtain the results | Save the inputs to count the sales of the songs for genre | | |
| Result or postcondition | Accumulated value of the whole sale of songs of the same genre and the number of songs of this genre | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Show the number of songs and the value of the sales for genre or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R15**: Report the number of sales and the total sales value of the best-selling song in the platform. | | |
| Resume | The system compares the number of sales of each song and save the one with the highest number and calculated total sale value of that song | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
|  |  |  |
| General activities necessary to obtain the results | Save the inputs to know the most sale song and total value of the sale of this | | |
| Result or postcondition | The total value of the song most sale | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Show the most sale song name and the total sales value or if there was an error in the process |

|  |  |  |  |
| --- | --- | --- | --- |
| Name or identifier | **R16**: Show ads to Standard Consumer users | | |
| Resume | Every time that a standard consumer user reproduces two songs or one podcast, the system shows an ad to him/her | | |
| Inputs | Input name | Datatype | Selection or repetition condition |
|  |  |  |
| General activities necessary to obtain the results | Save the inputs to show the ads to the Standard Consumer users | | |
| Result or postcondition | Show ads to a standard consumer user every two song or one podcast reproduced | | |
| Outputs | Output name | Datatype | Selection or repetition condition |
| creationMessage | String | Show the ad or if there was an error in the process |

|  |  |  |
| --- | --- | --- |
| **Requerimiento Funcional** | **Nombre de la Clase** | **Nombre del método** |
| ReqFunc001  R1: Register producer users, artists, and content creators. | Clase Artist | Artist(....) |
| Clase Creator | Creator(…) |
| Clase NeoTunesController | registerProducer(..) |
| Clase NeoTunesManager | registerProducer() |
| ReqFunc002  A2: Register consumer, standard and premium users. | Clase Standard | Standard(..) |
| Clase Premium | Premium(..) |
| Clase NeoTunesController | registerConsumer(..) |
| Clase NeoTunesManager | registerConsumer() |
| ReqFunc003  A3: Register songs | Clase Song | Song(…) |
| Clase NeoTunesController | registerSong(..) |
| Clase NeoTunesController | toStringGener() |
| Clase NeoTunesManager | registerAudio() |
| ReqFunc003  A3: Register podcasts | Clase Podcast | registerPodcast(..) |
| Clase NeoTunesController | toStringCategory() |
| Clase NeoTunesController | registerPodcast(..) |
| Clase NeoTunesManager | registerAudio() |
| ReqFunc004  A4: Create a playlist | Clase Playlist | Playlist(..) |
| Clase NeoTunesController | registerPlaylist(..) |
| Clase NeoTunesManager | createPlaylist() |
| ReqFunc005  A5: Edit a playlist | Clase Standard | getPlaylist() |
| Clase Playlist | addAudio(..) |
| Clase Premium | getPlaylist() |
| Clase NeoTunesController | addElementToPlaylist(..) |
| Clase Playlist | deleteAudio(..) |
| Clase Playlist | setName(..) |
| Clase NeoTunesController | editNameofPlaylist(..) |
| Clase NeoTunesController | editPlaylist(..) |
| Clase User | getNickName() |
| Clase NeoTunesController | showUser() |
| Clase Audio | getName() |
| Clase NeoTunesController | showAudios() |
| Clase Standard | showAudios(..) |
| Clase Premium | showAudios(..) |
| Clase NeoTunesController | showAudiosPlaylist(…) |
| Clase NeoTunesController | showPlaylistOfUser(..) |
| Clase NeoTunesManager | editPlaylist() |
| Clase NeoTunesController | deleteElementToPlaylist(..) |
| ReqFunc006  R6: Share a playlist | Clase NeoTunesController | generateMatrix() |
| Clase NeoTunesController | printMatrix(..) |
| Clase NeoTunesController | generateCodeN(…) |
| Clase NeoTunesController | generateCodeT(…) |
| Clase NeoTunesController | generateCodeImpar(…) |
| Clase  Standard | getPlaylist() |
| Clase Playlist | getAudios() |
| Clase  Premium | getPlaylist() |
| Clase NeoTunesController | isPlaylistOfPodcast(..) |
| Clase NeoTunesController | isPlaylistOfSong(..) |
| Clase NeoTunesController | sharePlaylist(..) |
| Clase NeoTunesManager | sharePlaylist() |
| ReqFunc007  R7:  Simulate playing a song or podcast (standard and premium). | Clase Audio | play() |
| Clase NeoTunesController | startAdvertisement() |
| Clase Consumer | getAudioHistory() |
| Clase NeoTunesController | reproduceAudio(..) |
| Clase NeoTunesManager | playbackController() |
| ReqFunc008  R8: Buy a song. | Clase Premium | toBuy () |
| Clase Standar | toBuy () |
| Clase Shop | Shop(..) |
| Clase Consumer | getShopping() |
| Clase Song | setNumTimeSold |
| Clase Song | getNumTimeSold() |
| Clase NeoTunesController | buySong(..) |
| Clase NeoTunesController | valueSong(..) |
| Clase NeoTunesManager | buySong() |
| ReqFunc009  R9: Report the total accumulated reproductions on the entire platform for each type of audio, songs, and podcast. | Clase NeoTunesController | reportSongViews() |
| Clase Song | getNumPlays() |
| Clase reportPodcastViews() | reportPodcastViews() |
| Clase Podcast | getNumPlays() |
| Clase NeoTunesManager | reportAudioViews() |
| ReFunc0010  R10: Inform the genre of the most listened song for a specific user and for the entire platform. | Clase NeoTunesController | mostGenreUser(..) |
| Clase Consumer | mostGenreUser() |
| Clase NeoTunesController | mostGenreSystem() |
| Clase NeoTunesController | showConsumer() |
| Clase NeoTunesManager | mostGenreUser() |
| Clase NeoTunesManager | mostGenreSystem() |
|  |  |
| ReqFunc0011  Inform the most listened podcast category for a specific user and for the entire platform. | Clase NeoTunesController | mostCategoryUser() |
| Clase NeoTunesController | mostCategorySystem() |
| Clase Consumer | mostCategoryUser() |
| Clase NeoTunesController | showConsumer() |
| Clase NeoTunesManager | mostCategorySystem() |
| Clase NeoTunesManager | mostCategoryUser() |
|  |  |
| ReqFunc0012  R12: Report the name and number of total views for the members of the Top 5 artists and Top 5 content creators. | Clase NeoTunesController | top5() |
| Clase Producer | getNumberReproductions() |
| Clase NeoTunesManager | showTop5Producers() |
| ReqFunc0013  R13: Report the name, genre or category and total number of reproductions of each audio of the Top 10 songs and the Top 10 podcast. | Clase NeoTunesController | top10() |
| Clase Song | getNumPlays() |
| Clase Song | getCategory() |
| Clase NeoTunesManager | showTop10Audios() |
| Clase Song | getName() |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |