

EXPLICACION SPOTFLEX

1) Funcionamiento General del programa

En esta sección se hablara de la interfaz y los menús, como funcionan y básicamente que hace cada botón/opcion

i) Menú Principal:

- Ingresar como usuario: El usuario puede iniciar su sesión
- Crear Usuario: El usuario creará su sesión

ii) Menú de Sesión: Una vez al ingresar como usuario se le darán varias opciones:

- Elegir Cancion: Al elegir una cancion de la biblioteca de canciones te dará acceso al submenú de esta
- Elegir Video: Al elegir un video de la biblioteca de canciones te dará acceso al submenú de esta
- Subir Cancion: El usuario podrá subir una cancion a Spotflex
- Subir Videos: E usuario podrá subir un video a Spotflex
- Búsqueda General: El usuario podrá buscar canciones, videos, playlists, usuarios y personas escribiendo su nombre, el buscador no es tan estricto con la búsqueda ya que puede buscar sin importar las mayúsculas o si pone parte del nombre (ej.: Si quieres buscar un actor que se llame Carlos puedes buscarlo por CARLOS, carlos, Carl, etc.)
- Búsqueda Personalizada: El usuario podrá buscar canciones y videos por varios filtros ya sea por palabras claves, datos de personas involucradas, duración, calidad, etc. Ademas tiene la posibilidad de buscarlos por el método or que buscara aquellas que cumplen con al menos un criterio o por el método and que los buscara aquellas que cumplan todos los requerimientos (omitiendo en ambos casos los filtros vacíos)
- Playlist: El usuario podrá crear y escuchar playlists, también está la opcion de borrarlas del sistema siempre y cuando el mismo usuario las haya creado
- Favoritos: El usuario podrá ver sus canciones y videos favoritos
- Ranking: El usuario podrá ver el ranking de las mejores canciones y videos
- Me gustaría: El usuario podrá ver canciones que posiblemente no haya escuchado que podrían ser de su agrado
- Foto de Perfil: Haciendo clic en el icono de la foto (arriba de tu nombre de usuario) podrás poner una foto de perfil
- Configuración: Aquí puedes editar los datos del Usuario tales como su nickname, contraseña, tipo de perfil, etc.
- Cerrar Sesión: El usuario cerrara su sesión

- iii) Submenú de Cancion/Video
Acá se mostrara tanto la informacion de la cancion/video como las diversas opciones que puedes hacer con esta
 - Reproducir: Puedes reproducir la cancion/video
 - Descargar: Opcion exclusiva de las canciones, la cancion se descargara en una carpeta llamada Descargas Spotflex en la carpeta bin del programa
 - Comentar: Puedes ver y comentar el video o cancion
 - Agregar a la cola: Puedes agregar la cancion/video a su cola de reproducción respectiva
 - Favorito: Agrega la cancion a favoritos
 - Calificar: Puedes calificar la cancion/video de 1 a 5 estrellas
- iv) Submenú de configuración:
Acá se mostraran las distintas opciones que puede configurar el usuario
 - Cambiar Nombre: Le da al usuario la posibilidad de cambiar el nombre
 - Cambiar Edad: Le da al usuario la posibilidad de cambiar la edad
 - Cambiar Accesibilidad del perfil: Le da al usuario la posibilidad de cambiar su cuenta de privado a público o viceversa
 - Cambiar Tipo de Cuenta: Le da al usuario la posibilidad de ser cambiar el tipo de cuenta (Free a Premium)
 - Cambiar Nickname: Le da al usuario la posibilidad de cambiar el nickname
 - Cambiar Contraseña: Le da al usuario la posibilidad de cambiar su contraseña
 - Ver Informacion del Usuario: Puedes ver tu informacion
 - Borrar Usuario: borrara tu sesión, si eres administrador te da la posibilidad de borrar a cualquier usuario, en caso de que borres tu sesión se te cerrara la sesión.
 - Borrar Cancion/Video: Podrás borrar cualquier archivo multimedia que el usuario haya subido a la plataforma
- v) Filtros de Búsqueda: Acá se mencionaran los posibles filtros de búsqueda para la búsqueda personalizada/avanzada:
 - Informacion: Puedes buscar una canción/video por la descripción de esta
 - Nombre de la persona involucrada: Puedes buscar canciones/videos por nombres o apellidos de personas involucradas
 - Género de la persona involucrada: Puedes buscar canciones/videos por personas de un género (ej.: Buscar todas las canciones hechas por un hombre)
 - Edad de la persona involucrada: Puedes buscar canciones/videos por personas de una edad (también se puede buscar por personas de mayor o menor edad)

- Duración: Puedes buscar una canción o un video por duración, puedes buscar algo igual, mayor o menor al dato ingresado
- Calificación: Puedes buscar una canción/video por su calificación promedio (También puedes buscar algo mayor o menor a esa calificación)
- Calidad: Puedes buscar una canción/video por su calidad (También puedes buscar algo mayor o menor a esa calidad)
- Género: Puedes buscar una canción o un video por algún genero

2) Clase Usuario: Permite la creación de usuarios y también es la que almacena los nombres de usuario y contraseñas para el inicio de sesión

2.1) Atributos:

- i) private string Nombre: Contiene el nombre del usuario
- ii) private int Edad: Contiene la edad del usuario
- iii) private bool Tipo de usuario: Indica si el usuario es público o privado con true o false
- iv) private bool Premium: Indica si el usuario es premium o no con true o false
- v) private string Nickname: Indica el nickname del Usuario
- vi) private string Contraseña: Indica la contraseña del usuario
- vii) private bool Administrador: Indica si el usuario es administrador con true o false
- viii) private string Foto de perfil: Indica el path o directorio de la foto de perfil asociada al usuario
- ix) private Lista de strings GenerosQueSigue: Indica los géneros que le gustan al usuario
- x) private Lista de Personas CantantesQueSigue: Indica los cantantes que sigue el usuario
- xi) private Lista de Canciones Favoritas: Indica las canciones favoritas del Usuario
- xii) private Lista de Videos Favoritas: Indica los videos favoritos del usuario
- xiii) private Lista de Multimedia Recomendaciones: Indica las canciones y videos que le podrían gustar al usuario
- xiv) private Lista de Multimedia UsuariosSeguidores: Indica los Usuarios que siguen a este usuario
- xv) private int Contador: Indica a los usuarios las oportunidades que tienen para ganar el mini juego que les permite convertirse en premium si es que son usuarios free
- xvi) private Lista de Canciones CancionesAgregadas: Indica las canciones que agrego el usuario
- xvii) private Lista de Videos VideosAgregados: Indica los videos que agrego el usuario

- xviii) private Lista de Playlist Playlists Agregadas: Indica las playlists que agrego el usuario
- xix) private Lista de Canciones Cola de canciones: Indica las canciones que están en la cola
- xx) private Lista de Canciones Cola de videos: Indica los videos que están en la cola

2.2) Metodos

- i. Public Void Constructor de Personas: Este creara a las personas con los datos ingresados por el usuario
- ii. Public Void AgregarFotoPerfil(): Ingresara una foto de portada al sistema, preguntándole al usuario el path de esta
- iii. Public List<Canciones> Get_ListaCancionesAgregadas(): Retorna la lista de canciones agregadas
- iv. Public List<Video> Get_ListaVideosAgregados(): Retorna la lista de videos agregadas
- v. Public List<Playlist> Get_ListaPlaylistAgregadas(): Retorna la lista de playlists agregadas
- vi. Public Void Add_Song_To_My_List(Canción canción): Con esta opción el usuario agregara a su lista de canciones agregadas la canción
- vii. Public Void Add_Video_To_My_List(Video video): Con esta opción el usuario agregara a su lista de videos agregadas el video
- viii. Public Void Add_Playlist_To_My_List(Playlist playlist) Con esta opción el usuario agregara a su lista de playlists agregadas
- ix. Public Void Cambiar_Dato_String(int caso, string palabra): Cambiará el nickname, el nombre o la contraseña del usuario según el caso usando la palabra
- x. Public Void Cambiar_Dato_Booleano(int caso): Cambiará el tipo de usuario, si es premium o si es administrador según sea el caso
- xi. Public Void Cambiar_Dato_Edad(int edad): Cambiará la edad del usuario usando el parámetro
- xii. Public Void Lose_BonusGame(): Descontará una oportunidad para jugar al mini juego

3) Controlador de la clase Usuario

3.1) Atributos

- i. List<Usuario> usuarios: El controlador de la clase usuario conoce a todos los usuarios que están en el sistema
- ii. Form1 appform: El controlador conoce a la vista
- iii. ControladorPersona controlador: Este controlador conoce al controlador de persona
- iv. ControladorCancion ctrl_cancion: Este controlador conoce al controlador de cancion
- v. ControladorVideo ctrl_video: Este controlador conoce al controlador de video

3.2) Metodos

- i. Void InicioUsuario(): Este método cargara todos los usuarios que hayan al iniciar el programa
- ii. Bool OnLoginClicked(): Este método permitirá el acceso o no a entrar en tu sesión
- iii. Bool OnDetectPrivate(): Este método te informara si algún usuario es privado o no
- iv. Void OnDeleteSong(): Este método borrara una cancion que hayas subido
- v. Void OnDeleteVideo(): Este método borrara un video que hayas subido
- vi. List<Canciones> OnGetColaCancion: Este método devolverá la cola de canciones de algún usuario
- vii. List<Video> OnGetColaVideo: Este método devolverá la cola de videos de algún usuario
- viii. Bool OnRegisterButtonClicked: Este método analizara si se puede registrar o no un usuario, en el caso de que así sea lo agregara a la lista de usuarios del sistema
- ix. Void OnGetMyFavs: Este método devolverá las canciones y videos favoritas del usuario en una lista
- x. Void OnUserInfo: Este método se encargara de entregarle al form la informacion necesaria para que se muestre en tu perfil (Nick y foto a la esquina superior izquierda)
- xi. Void OnUserCompleteInfo: Este método se encargara de entregarle al form toda la informacion del perfil de tu usuario
- xii. Void OnPhotoChanged: Este método se encargara de cambiar tu foto de perfil
- xiii. Bool OnCheckAdmin: Este método se encargara de verificar si algún usuario es administrador o no
- xiv. Bool OnChangeUserData: Este método se encargara de verificar si es posible cambiar la informacion de tu perfil, de ser así la cambiara

- xv. `Void OnPrivateChange()`: Cambiara tu usuario de público a privado y viceversa
- xvi. `Void OnPremiumChanged()`: Cambiara tu usuario de free a premium y viceversa
- xvii. `Void OnEraseUser()`: Este método borrara algún usuario
- xviii. `Bool OnCanAddSong()`: Este método verificara si es posible que un usuario pueda agregar canciones
- xix. `Bool OnCanAddVideo()`: Este método verificara si es posible que un usuario pueda agregar videos
- xx. `Void OnAddSongToMyList()`: Agrega una cancion a la lista de canciones subidas por el usuario
- xxi. `Void OnAddVideoToMyList()`: Agrega un video a la lista de canciones subidas por el usuario
- xxii. `Void OnAddFavoriteSong()`: Agrega una cancion a la lista de favoritos del usuario
- xxiii. `Void OnAddFavoriteSong()`: Agrega una cancion a la lista de canciones favoritas del usuario
- xxiv. `Void OnAddFavoriteVideo()`: Agrega un video a la lista de videos favoritos del usuario
- xxv. `List<Playlist> OnGetUserPlaylist()`: Devuelve las playlists creadas por el usuario
- xxvi. `Bool OnCheckFavorite()`: Verifica si una cancion o video es de los favoritos del usuario
- xxvii. `Usuario OnGetUsuario()`: Devuelve un usuario en especifico
- xxviii. `Void OnGetUser()`: Devuelve una lista de todos los usuarios que están en el sistema
- xxix. `Void OnSearchedUserInfo()`: Devuelve toda la informacion de un usuario, incluyendo las canciones y videos que subió
- xxx. `List<Usuarios> OnGetPlatformUsers()`: Devuelve toda la informacion de todos los usuarios de la plataforma
- xxxi. `Void OnGeneralUserSearch()`: Devuelve todos los usuarios que contengan en su nombre el string ingresado en la búsqueda
- xxxii. `Void OnCanFollowUser()`: Verificara si es posible o no seguir un usuario
- xxxiii. `Void OnFollowUser()`: Método que permite que un usuario pueda seguir a otro usuario
- xxxiv. `Void OnUnfollowUser()`: Método que permite que un usuario deje de seguir a otro usuario
- xxxv. `Void OnSaveData()`: Método que guarda todos los datos sobre los usuarios, este se activa cada vez que se hace algún cambio
- xxxvi. `Void OnRemoveCancionFav()`: Método que remueve una cancion de favoritos
- xxxvii. `Void OnRemoveVideoFav()`: Método que remueve un video de favoritos
- xxxviii. `Void OnColaCancion()`: Método que agrega una cancion a la cola de canciones del usuario

- xxxix. Void OnColaVideo(): Método que agrega un video a la cola de videos del usuario
- xl. Void OnGetRecomendaciones(): Método que define que las canciones y videos recomendadas para el usuario

4) Clase Padre Multimedia: Esta clase contendrá todo lo que puede tener un archivo multimedia ya sea canción o video, como también lo que puede hacer de por si un archivo de estas características

4.1) Atributos

- i. Protected int Duración: Indica la duración del archivo (se instaló librería NuGet WMP.Lib para la ejecución de este atributo)
- ii. Protected string Título: Indica el título del archivo
- iii. Protected string Fecha_Inclusion: Indica la fecha en la que se incluye el archivo a la plataforma
- iv. Protected long Size: Indica el tamaño del archivo en bytes
- v. Protected string Estudio: Indica el estudio donde proviene el archivo
- vi. Protected string CarpetaArchivo: Indica el directorio del archivo
- vii. Protected string Fecha_Publicacion: Indica la fecha de publicación del archivo
- viii. Protected string Descripción: Indica la descripción del archivo
- ix. Protected string Portada: Indica el directorio de la portada del archivo
- x. Protected int NumeroReproducciones: Indica el número de reproducciones del archivo
- xi. Protected List<string> Comentarios: Indica los comentarios que tiene el archivo
- xii. Protected List<int> Ranking: Indica las calificaciones que tiene el archivo
- xiii. Protected List<string> Genero: Indica los géneros de la canción
- xiv. Protected List<Usuario> Seguidores: Indica los Usuarios que siguen al archivo

4.2) Metodos

- i. Void Get Mean Tier: Te entrega su calificación promedio

5) Clase Canciones: Clase hija de Multimedia, acá se tienen los atributos específicos de la clase canciones (esta clase puede usar todo lo que está en multimedia)

5.1) Atributos

- i. Private string Letra: Indica la letra de la canción
- ii. Private string Álbum: Indica el álbum/disco de la canción
- iii. Private List<Personas> PersonasCancion: Indica los artistas de la canción

5.2) Metodos

- i. Void Descargar Canciones: Descarga la canción a la carpeta de descargas del programa

6) Controlador de la clase Canciones

6.1) Atributos

- i. List<Canciones> canciones: El controlador de la clase canciones conoce a todas las canciones que están en el sistema
- ii. Form1 appform: El controlador conoce a la vista
- iii. ControladorPersona ctrl_persona: Este controlador conoce al controlador de clase persona

6.2) Metodos

- i. Void InicioCanciones(): Este método cargara a todas las canciones que estén en el programa
- ii. IEnumerable<Canciones> OnGetRankingSongs: Este Método entregara una lista de canciones ordenada de mejor a peor ranking
- iii. Void OnRemoveCancion(): Este método removerá una cancion del sistema
- iv. Void OnGetDataSong(): Este método entregara todas las canciones del sistema con su nombre y foto
- v. Void OnShowInfoCancion(): Este método entregara toda la informacion de alguna cancion
- vi. Void OnReproducirCancion(): Este método reproducirá alguna cancion gracias a la dirección de esta
- vii. Void OnDownloadSong(): Este método descargara una cancion en particular
- viii. Void OnRankedSong(): Este método se encargara de calificar una cancion en particular
- ix. Void OnOrSongSearch(): Este método se encargara de buscar todas las canciones que cumplan con al menos un criterio que indico el usuario
- x. Void OnAndSongSearch(): Este método se encargara de buscar todas las canciones que cumplan con todos los criterios que indico el usuario
- xi. Canciones CancionesOnGetSong(): Este método entregara una cancion en particular
- xii. Void OnCommentAdded(): Este método agregara un comentario en alguna cancion
- xiii. List<string> OnGetUsuariosComentan: Este método entrega el nickname de todos los usuarios que comentan
- xiv. List<string> OnGetComentarios: Este método entrega todos los comentarios de alguna cancion
- xv. List<string> OnGetFotoUsuariosComentan: Este método entregara todas las fotos de los usuarios que comentan

- xvi. List<Canciones> OnGetMySongs: Entrega todas las canciones que hizo un artista en específico
- xvii. List<Canciones> OnGetAllSongs: Entrega todas las canciones del sistema
- xviii. Void OnGeneralSongSearch: Entrega las canciones que contengan el string buscado por el usuario
- xix. Void OnChangeAllData: Actualiza la lista de usuarios que comentan y los que califican canciones
- xx. Void OnChangePhotoComent(): Actualiza la lista de fotos de los usuarios que comentan
- xxi. Void OnSaveData(): Guarda todo lo relacionado a canciones, este método se activa ante cualquier cambio con relación a estas

7) Clase Video: Clase hija de Multimedia, acá se tienen los atributos específicos de la clase videos (esta clase puede usar todo lo que está en multimedia)

7.1) Atributos:

- i. Private string Calidad: Indica la calidad del video
- ii. Private List<Personas> PersonaVideo: Indica las personas que están involucradas en la producción del video

8) Controlador de la clase Video

8.1) Atributos

- i. List<Canciones> canciones: El controlador de la clase videos conoce a todas los videos que están en el sistema
- ii. Form1 appform: El controlador conoce a la vista
- iii. ControladorPersona ctrl_persona: Este controlador conoce al controlador de clase persona

8.2) Metodos

- i. Void InicioVideos(): Este método cargara a todos los videos que estén en el programa
- ii. IEnumerable<Video> OnGetRankingVideos: Este Método entregara una lista de Videos ordenada de mejor a peor ranking
- iii. Void OnRemoveVideo(): Este método removerá un Video del sistema
- iv. Void OnGetDataVideo(): Este método entregara todas los Videos del sistema con su nombre y foto
- v. Void OnShowInfoVideo(): Este método entregara toda la informacion de algún Video
- vi. Void OnReproducirVideo(): Este método reproducirá algún Video gracias a la dirección de esta
- vii. Void OnRankedVideo(): Este método se encargara de calificar un Video en particular

- viii. Void OnOrVideoSearch(): Este método se encargara de buscar todas los videos que cumplan con al menos un criterio que indico el usuario
- ix. Void OnAndVideoSearch(): Este método se encargara de buscar todos los videos que cumplan con todos los criterios que indico el usuario
- x. Video VideoOnGetVideo(): Este método entregara un Video en particular
- xi. Void OnCommentAdded(): Este método agregara un comentario en algún Video
- xii. List<string> OnGetUsuariosComentan: Este método entrega el nickname de todos los usuarios que comentan
- xiii. List<string> OnGetComentarios: Este método entrega todos los comentarios de algún Video
- xiv. List<string> OnGetFotoUsuariosComentan: Este método entregara todas las fotos de los usuarios que comentan
- xv. List<Video> OnGetMyVideos: Entrega todas los Video que hizo un director/actor en especifico
- xvi. List<Video> OnGetAllVideos: Entrega todos los Videos del sistema
- xvii. Void OnGeneralVideoSearch: Entrega las Video que contengan el string buscado por el usuario
- xviii. Void OnChangeAllData: Actualiza la lista de usuarios que comentan y los que califican videos
- xix. Void OnChangePhotoComent(): Actualiza la lista de fotos de los usuarios que comentan
- xx. Void OnSaveData(): Guarda todo lo relacionado a Video, este método se activa ante cualquier cambio con relación a estas

9) Clase Playlist: Clase que almacena listas de canciones o videos que luego son guardadas en el almacenamiento

9.1) Atributos:

- i) Private List<Canciones> Playlist_Canciones: Indica las canciones de la playlist
- ii) Private List<Video> Playlist_Videos: Indica los videos de la playlist
- iii) Private string nombre: Indica el nombre de la playlist
- iv) Private List<Usuarios> Seguidores: Indica los usuarios que siguen esta playlist

9.2) Metodos

- i. List<Canciones> MostrarCanciones: Entrega todas las canciones que contiene la playlist
- ii. List<Videos> MostrarVideos: Entrega todas los videos que contiene la playlist

10) Controlador de la clase Playlist

10.1) Atributos

- i. List<Playlists> playlists: El controlador de la clase playlist conoce a todas las playlists que están en el sistema
- ii. Form1 appform: El controlador conoce a la vista

10.2) Metodos

- i. Void InicioPlaylist(): Este método cargara todas las playlists guardadas al iniciar el programa
- ii. List<Playlist> OnGetPlaylists(): Entrega todas las playlists que tiene el sistema
- iii. Void OnGeneralSearchPlaylist(): Entrega todas las playlists que tengan por nombre el string incluido que busca el usuario
- iv. Void OnGeneratePlayList(): Crea la playlist que realiza el usuario
- v. Playlist OnUserPlaylist(): Reproduce una playlist elegida por el usuario
- vi. Void OnDeletePlaylist(): Borra alguna playlist del sistema
- vii. Void OnSaveData(): Guarda todos los datos relacionados a las playlists, se activa cada vez que se realiza algún cambio

11) Clase Persona: Clase que tiene la información de las personas involucradas ya sea en una canción o en un video

11.1) Atributos:

- i. Private string Nombre: Indica el nombre de la persona
- ii. Private string Apellido: Indica el apellido de la persona
- iii. Private int Edad: Indica la edad de la persona
- iv. Private string Sexo: Indica el sexo de la persona
- v. Private string PuestoTrabajo: Indica el puesto de trabajo de la persona
- vi. Private List<Usuario> Seguidores: Indica los usuarios que siguen a la persona

12) Controlador de la clase Personas

12.1) Atributos

- i. List<Personas> playlists: El controlador de las personas conoce a todas las personas que están en el sistema
- ii. Form1 appform: El controlador conoce a la vista

12.2) Metodos

- i. Void InicioPersona(): Este método cargara a todas las personas que se encuentren en el sistema
- ii. List<Personas> OnAddPersonasCancion: Este método agregara las personas que están involucradas a una cancion al sistema
- iii. List<Personas> OnAddPersonasVideo: Este método agregara las personas que están involucradas a un video al sistema
- iv. Void OnCanFollowPerson(): Verificara si es posible o no seguir una persona
- v. Void OnFollowPerson(): Método que permite que un usuario pueda seguir a otra persona
- vi. Void OnGeneralPersonSearch(): Buscara a las personas que tengan en su nombre o su apellido el string buscado por el usuario
- vii. Void OnShowPersonInfo(): Entregara la informacion de la persona como datos personales y trabajos realizados
- viii. Void OnSaveData(): Este método guardara todo lo que tenga que ver con las personas, este método se activa al realizar cualquier cambio

13) Metodos relacionados a eventos en el Forms

13.1) Metodos

- i. Void MostrarInformacionMultimedia(): Este método cargara todas las canciones y videos del sistema a la biblioteca
- ii. Void MisCanciones(): Entrega todas las canciones que subió un usuario
- iii. Void MisVideos(): Entrega todas los videos que subió un usuario
- iv. Void MisCancionesyVideos(): Entrega todas las playlists que son públicas y que son del usuario
- v. Void OnRemoveCancion(): Borra una cancion del sistema
- vi. Void OnRemoveVideo(): Borra un video del sistema
- vii. Void OnLoginButtonClicked(): Evalúa si los datos ingresados por el usuario son correctos para iniciar sesión
- viii. Void OnRegisterButtonClicked(): Evalúa si es posible registrar un usuario con los datos ingresados por este
- ix. Void OnChangeUserData(): Evalúa si es posible cambiar los datos de algún usuario con los datos ingresados por este
- x. Void OnShowUsersBorrar(): Mostrara todos los usuarios que podrá borrar, este método solo lo vera funcionar un usuario administrador
- xi. Void OnChangePrivate(): Cambia la privacidad de algún usuario
- xii. Void OnChangePremium(): Cambia el plan de algún usuario
- xiii. Void OnEraseUser(): Borra algún usuario
- xiv. Void OnUserInfo(): Mostrara la Info del usuario en el perfil
- xv. Void OnUserCompleteInfo(): Mostrara toda la Info del usuario

- xvi. Void OnPhotoChanged(): Método que se encargara de cambiar la foto de perfil
- xvii. Void OnSongClick(): Método que se activa al hacer click a alguna foto de la cancion
- xviii. Void OnVideoClick(): Método que se activa al hacer click a algún video de la cancion
- xix. Void OnMultimediaRanked(): Metodo que califica una cancion o video
- xx. Void OnSongFavorite(): Metodo que selecciona una cancion o video como favoritos
- xxi. Void OnShowComentariosSong(): Metodo que muestra los comentarios de alguna cancion
- xxii. Void OnBusquedaGeneral(): Metodo que se encarga de recibir todos los usuarios, personas, canciones, videos y playlists que se cumplen con el requerimiento general
- xxiii. Void OnBusquedaOr(): Metodo que se encarga de recibir todas las canciones y videos que cumplen con alguno de los filtros
- xxiv. Void OnBusquedaAnd(): Metodo que se encarga de recibir todas las canciones y videos que cumplen con todos los filtros
- xxv. Void OnGetFavorites(): Metodo que busca todas las canciones y videos que están en las favoritas del usuario
- xxvi. Void InformacionCancion(): Metodo que muestra la informacion de la cancion
- xxvii. Void InformacionVideo(): Metodo que muestra la informacion del video
- xxviii. Void InformacionBusquedaGeneral(): Metodo que muestra todas las canciones, videos, playlists, usuarios y personas que se encontraron por el método general
- xxix. Void InformacionBusquedaGeneral(): Metodo que muestra todas las canciones y videos que se encontraron por el método avanzado
- xxx. Void OnPlaylistClickedBySearch(): Metodo que reproduce una playlist encontrada en el buscador
- xxxi. Void ShowMyFavorites(): Metodo que muestra las canciones y videos favoritas del usuario
- xxxii. Void OnSongClickedBySearch(): Metodo que reproduce una cancion encontrada en el buscador
- xxxiii. Void OnVideoClickedBySearch(): Metodo que reproduce un video encontrado en el buscador
- xxxiv. Void OnUserClicked(): Metodo que mostrara la info de un usuario encontrado en el buscador
- xxxv. Void OnPersonClicked(): Metodo que mostrara la info de una persona encontrada en el buscador
- xxxvi. Void ShowSearchedUserInfo(): Metodo que mostrara la Info de un usuario, ciertos datos se mostraran solo si el usuario es publico
- xxxvii. Void ShowSearchedPersonInfo(): Metodo que mostrara la Info de una persona, ciertos datos se mostraran solo si el usuario es publico

- xxxviii. Void OnGeneratePlaylist(): Este método se encargara de mandar los datos necesarios para que se realice una playlist
- xxxix. Void AddCancionPlaylist(): Este método se encarga de agregar una cancion a la playlist
 - xl. Void OnCancionPlay(): Metodo que muestra el reproductor de canciones
 - xli. Void AddVideoPlaylist(): Este método se encarga de agregar un video a la playlist
 - xl.ii. Void OnVideoPlay(): Metodo que muestra el reproductor de videos
 - xl.iii. Void OnVideoFavorite(): Metodo que agrega un video a favoritos
 - xl.iv. Void OnShowComentariosVideos(): Metodo que muestra los comentarios del video
 - xl.v. Void EliminarPlaylist(): Metodo que elimina una playlist del sistema
 - xl.vi. Void OnVideoClickedBySearch(): Este método reproducirá un video encontrado en alguna búsqueda
 - xl.vii. Void OnGetRecomendaciones(): Este método se encargara de obtener todo lo necesario para mostrar las recomendaciones
 - xl.viii. Void OnShowRecomendaciones(): Este método se encargara de mostrar todas las canciones y videos recomendadas