

CONCEPTUALIZACIÓN ENTREGA 2

En primer lugar, se le dará opción para que pueda registrarse, iniciar sesión o salir.

Si presiona la opción de registrarse, aparecerán otras 3 opciones; registrarse como usuario, como administrador o como profesor.

- Registrarse como usuario: tendrá que ingresar todos los datos correspondientes. El gmail y nickname tendrán que ser registrados hasta que no haya ningún usuario registrado con los mismos que usted ingrese, esto lo hará el método CheckGmail() y CheckUsername() de la clase Gate. Estos dos y la contraseña, no pueden ser datos vacíos, en ese caso se volverán a pedir hasta que se registre correctamente. En caso de estar todos los datos correctos, el método SingUser() de la clase Gate añadirá al nuevo usuario registrado a la listas de los usuarios (users), esta también de la clase Gate. Todos los usuarios, al momento de registrarse no son Premium.
- Registrarse como administrador: tendrá que ingresar el código de administrador que es "123", en caso de registrarlo mal, volverá al menú anterior. Luego, se le pide su gmail, nombre de usuario (los cuales se pedirán hasta que no coincida con los de otro administrador, esto lo hará el método CheckGmailA() y CheckUsernameA() de la clase Gate) y contraseña, las que, al igual que en registrarse como usuario, no pueden ser vacías. En caso de estar todo correcto, el método SingAdmin() de la clase Gate añadirá al nuevo administrador registrado a la lista de administradores (managers), esta también de la clase Gate.
- Registrarse como profesor: tendrá que ingresar el código de profesor que es "321", en caso de registrarlo mal, volverá al menú principal. Luego, se le piden sus datos correspondientes, y al igual que en los otros dos registros, gmail y nombre de usuario no pueden existir previamente, gracias a el método CheckGmailP() y checkUsernameP() de la clase Gate y, al igual que contraseña, no pueden ser vacíos. Se le pide además que ingrese el curso del cual quiere ser profesor. Si todo está registrado correctamente, el método SingTeacher() de la clase Gate, añadirá al nuevo profesor registrado a la lista de profesores (teachers), esta también de la clase Gate.

Si presiona la opción de iniciar sesión, aparecerán otras 4 opciones; iniciar sesión como usuario, administrador, profesor o volver atrás.

- Iniciar sesión como usuario: se le pedirá que registre su nombre de usuario o gmail y contraseña. El método LogAsUser() de la clase Gate verificará que exista un usuario con esos datos, si es que existe se abrirá un nuevo menú.
 - (1) Pestaña buscar: Se abrirá un nuevo menú.
 - (0) Buscador genérico: el método GenericSearch() de la clase MediaPlayer da la opción de buscar por el filtro que usted desee, ingresa una palabra y se buscará todo lo relacionado a esa palabra, ya sea video, canción, clase, karaoke, playlist, album o serie y luego el método Play() de la clase MediaPlayer reproducirá lo que usted haya seleccionado del GenericSearch().
 - (1) Buscar Videos: el método Search("1") de la clase MediaPlayer da la opción de buscar por criterios, mientras más criterios se van agregando, menos videos van apareciendo, ya que es más específica la búsqueda. Luego, el método Play() de la clase MediaPlayer reproducirá el video seleccionado en el método Search("1"). El "1" es específicamente para los videos.
 - (2) Buscar Canción: el método Search("2") de la clase MediaPlayer da la opción de buscar por criterios, mientras más criterios se van agregando,

menos canciones van apareciendo porque es más específica la búsqueda. Luego, el método Play() de la clase MediaPlayer reproducirá la canción seleccionada en el método Search("2"). El "2" es específicamente para las canciones.

(3) Buscar Serie: el método Search("3") de la clase MediaPlayer da la opción de buscar por criterios, mientras más criterios se van agregando, menos series van apareciendo porque es más específica la búsqueda. Luego, el método Play() de la clase MediaPlayer reproducirá la serie seleccionada en el método Search("3"). El "3" es específicamente para las series.

(4) Buscar Playlist: el método Search("4") de la clase MediaPlayer da la opción de buscar por criterios, mientras más criterios se van agregando, menos playlists van apareciendo porque es más específica la búsqueda. Luego, el método Play() de la clase MediaPlayer reproducirá la playlist seleccionada en el método Search("4"). El "4" es específicamente para las Playlists.

(5) Buscar Album: el método Search("5") de la clase MediaPlayer da la opción de buscar por criterios, mientras más criterios se van agregando, menos albums van apareciendo porque es más específica la búsqueda. Luego, el método Play() de la clase MediaPlayer reproducirá el álbum seleccionado en el método Search("5"). El "5" es específicamente para album.

(6) Buscar Karaoke: el método Search("6") de la clase MediaPlayer da la opción de buscar por criterios, mientras más criterios se van agregando, menos karaokes van apareciendo porque es más específica la búsqueda. Luego, el método Play() de la clase MediaPlayer reproducirá el karaoke seleccionado en el método Search("6"). El "6" es específicamente para Karaoke.

(7) Buscar Clase: el método Search("7") de la clase MediaPlayer da la opción de buscar por criterios, mientras más criterios se van agregando, menos clases van apareciendo porque es más específica la búsqueda. Luego, el método Play() de la clase MediaPlayer reproducirá la clase seleccionada en el método Search("7"). El "7" es específicamente para las clases (Lessons).

(8) Buscar y reproducir añadiendo a la cola: el método Play() de la clase MediaPlayer que recibe el usuario y te genera una lista de canciones temporal, te ofrece todas las canciones de la base de datos y te permite ir añadiéndolas a tu lista y reproducirlas.

(9) Para volver al menú: vuelves al menú del usuario.

(2) Pestaña información: Se abrirá un nuevo menú.

(1) Conocer información de una canción: se abrirá un nuevo menú.

(1) Ver calificación de una canción: el método GetQualification() de la clase MediaPlayer necesita que se busque una canción para ver su calificación, por lo que este método recibe la canción que se buscó en el método Search("2") y muestra la calificación de la canción seleccionada.

(2) Ver información de una canción: el método GetMetadata() de la clase MediaPlayer necesita que se busque una canción que se buscó en el método Search("2") y muestra la información de la canción seleccionada.

(3) Ver información intrínseca de una canción: el método GetIntrinsicInformation() de la clase MediaPlayer necesita que se

busque una canción para ver su información intrínseca, por lo que se busca una canción con el método Search("2") y se muestra la información intrínseca de la canción seleccionada.

(4) Ver información de la plataforma de la canción: el método GetPlatformInformation() de la clase MediaPlayer, recibe una canción seleccionada por el método Search("2") y me muestra la información de la plataforma de la canción.

() Presionar cualquier tecla para volver al menú de la pestaña información.

(2) Conocer la información de un video: se abrirá un nuevo menú.

(1) Ver calificación de un video: el método GetQualification() de la clase MediaPlayer necesita que se busque un video para ver su calificación, por lo que este método recibe el video que se buscó en el método Search("1") y muestra la calificación del video seleccionado.

(2) Ver información de un video: el método GetMetadata() de la clase MediaPlayer necesita que se busque un video que se buscó en el método Search("1") y muestra la información del video seleccionado.

(3) Ver información intrínseca de un video: el método GetIntrinsicInformation() de la clase MediaPlayer necesita que se busque un video para ver su información intrínseca, por lo que se busca un video con el método Search("1") y se muestra la información intrínseca del video seleccionado.

(4) Ver información de la plataforma de un video: el método GetPlatformInformation() de la clase MediaPlayer, recibe un video seleccionado por el método Search("1") y me muestra la información de la plataforma del video.

() Presionar cualquier tecla para volver al menú de la pestaña información.

(3) Volver al menú principal: vuelve al menú del usuario.

(3) Pestaña comunidad: Se abrirá un nuevo menú.

(1) Seguir a: se abrirá un nuevo menú.

(1) Seguir a usuario: el método Follow("Usuario", lista de usuario, un usuario, una lista de profesores) de la clase MediaPlayer recibe "Usuario" y es la clave para se pueda seleccionar un usuario al que le quiero ver las playlist. Luego, muestra todas las playlist del usuario, seleccionas una playlist y después muestra las canciones de la playlist seleccionada y si al usuario le gusta, la puede añadir a favoritos y si no puede seguir buscando usuarios. Si el usuario no es premium y quiere seguir un usuario que tenga playlist públicas no podrá.

(2) Seguir a un Album: el método Follow("Album", lista de usuario, un usuario, una lista de profesores) de la clase MediaPlayer recibe "Album" y es la clave para se pueda seleccionar un album al que le quiera ver las canciones. Si le gusta, lo puede añadir a favoritos y si no, puede seguir buscando otros album.

(3) Seguir a una Playlist: el método Follow("Playlist", lista de usuario, un usuario, una lista de profesores) de la clase MediaPlayer recibe "Playlist" y es la clave para se pueda seleccionar una playlist a la que quiera ver las canciones o videos y si le gustó la puede añadir a

favoritos y si no, puede seguir buscando otras playlist, pero si no es premium y quiere seguir una playlist que sea pública, no podrá.

(4) Seguir a un Artista: el método Follow("Artista", lista de usuario, un usuario, una lista de profesores) de la clase MediaPlayer recibe "Artista" y es la clave para se pueda seleccionar un artista al que le quiera ver las canciones, videos, karaokes y si le gustó lo puede añadir a favoritos y si no, puede seguir buscando otros artistas.

(5) Seguir una Serie: el método Follow("Series", lista de usuario, un usuario, una lista de profesores) de la clase MediaPlayer recibe "Series" y es la clave para se pueda seleccionar una serie a la que le quiera ver los videos y si le gustó, la puede añadir a favoritos y si no, puede seguir buscando otras series.

(6) Seguir a un profesor: el método Follow("Teacher", lista de usuario, un usuario, una lista de profesores) de la clase MediaPlayer recibe "Teacher" y es la clave para se pueda seleccionar un profesor al que le quiera ver los videos de sus clases y si le gustó, lo puede añadir a favoritos y si no, puede seguir buscando otros profesores.

() Presionar cualquier tecla para volver al menú de la pestaña comunidad.

(2) Editar perfil: se abre un nuevo menú.

(1) Hacerse Premium: el método MemberShip() de la clase Gate hace que un usuario pase de no ser Premium a convertirse en Premium, pide los datos del usuario al que quiere convertir en Premium y luego, pide los datos de tu tarjeta y cuando finaliza el atributo membershipType pasa de "no pago" a "pago".

(2) Dejar de ser Premium: el método NoMembership() de la clase Gate le da la opción al usuario de pasar de ser Premium a no Premium, el cual pide que se ingresen los datos de usuario y el atributo membershipType pasa de "pago" a "no pago".

(3) Cambiar contraseña: el método ChangePassword() de la clase Gate pide que se ingresen los datos del usuario al que se le quiere cambiar la contraseña y luego le pide que ingrese la nueva contraseña y llama al método ChangePassword2() que cambia la contraseña antigua por la nueva.

(4) Editar Perfil: el método ChangeProfile() de la clase Gate, pide que se ingrese un usuario y da las opciones de cambiar el nombre de usuario, nombre, apellido, edad, país, ciudad, calle o código postal y luego cuando el usuario cambia todo lo que quería cambiar se presiona la opción de actualizar.

() Presionar cualquier tecla para volver al menú de la pestaña comunidad.

(3) Calificar o descargar un archivo multimedia: se abre un nuevo menú.

(1) Calificar canción: el método Qualify() de la clase MediaPlayer recibe una canción seleccionada por Search("2") y permite calificar la canción del 1 al 5.

(2) Calificar video: el método Qualify() de la clase MediaPlayer recibe un video seleccionado por Search("1") y permite calificar un video del 1 al 5.

(3) Descargar canción: si el usuario no es Premium no se podrá descargar una canción. El método Download() de la clase MediaPlayer recibe una canción buscada por el método Search("2") y recibe el usuario que desea descargar la canción, busca la ruta de la canción y las copia y pega en el archivo de música.

() Presionar cualquier tecla para volver al menú de la pestaña comunidad.

(4) Volver al menú principal: vuelve al menú del usuario.

(4) Pestaña Playlist: se abre un nuevo menú.

(1) Crear o editar Playlist de canciones: el método CreatePlaylistS() de la clase MediaPlayer recibe un usuario y "a" o "e" (a para agregar y e para eliminar), le pide al usuario el nombre de la playlist y luego los datos de las canciones que quiere añadir o eliminar de la playlist y la palabra "agregar" hace que se lance el evento OnAddSong() en la clase User y se van agregando las canciones, y la palabra "eliminar" hace que se lance el evento OnDeleteSong() en la clase User y se van eliminando las canciones. Si no es Premium no podrá crear Playlist públicas.

(2) Crear o editar Playlist de videos: el método CreatePlaylistV() de la clase MediaPlayer recibe un usuario y "a" o "e" (a para agregar y e para eliminar), le pide el nombre de la playlist y luego los datos de los videos que quiere añadir o eliminar de la playlist y la palabra "agregar" hace que se lance el evento OnAddVideo() en la clase User y se van agregando los videos, y la palabra "eliminar" hace que se lance el evento OnDeleteSong() en la clase User y se van eliminando los videos. Si no es Premium no podrá crear Playlist públicas.

(3) Ordenar Playlist: se abre un nuevo menú.

(1) Para canción: se abre el siguiente menú.

(1) Ordenar alfabéticamente

(2) Ordenar por fecha

(3) Ordenar por duración

(4) Ordenar por calificación

() Cualquier carácter para volver.

Luego se le preguntará (1) ordenar de mayor a menor o () de menor a mayor. Si es "1" up1 será true y si no será false.

Luego se le pide la playlist a ordenar y luego el método PlayListStarter("song", nombre playlist, up1, "Opción anterior", user) de la clase MediaPlayer actúa dependiendo de la opción anterior. Este método hace que se lance el evento OnOrderBy() en la clase Playlist que ordena las canciones dependiendo de la "Opción anterior".

(2) Para video: se abre el siguiente menú.

(1) Ordenar alfabéticamente

(2) Ordenar por fecha

(3) Ordenar por duración

(4) Ordenar por calificación

() Cualquier carácter para volver.

Luego se le preguntará (1) ordenar de mayor a menor o () de menor a mayor. Si es "1" up1 será true y si no será false.

Luego se le pide la playlist a ordenar y luego el método PlayListStarter("video", nombre playlist, up1, "Opción anterior", user) de

la clase MediaPlayer actúa dependiendo de la opción anterior. Este método hace que se lance el evento OnOrderBy() en la clase Playlist que ordena los videos dependiendo de la "Opción anterior".

() Cualquier tecla para volver al menú de pestaña playlist.

(4) Ordenar Album: se abre el siguiente menú.

(1) Ordenar alfabéticamente

(2) Ordenar por fecha

(3) Ordenar por duración

(4) Ordenar por calificación

() Cualquier carácter para volver.

Luego se le preguntará (1) ordenar de mayor a menor o () de menor a mayor. Si es "1" upa1 será true y si no será false.

Luego se le pide el Album a ordenar y luego el método AlbumStarter(nombre serie, upa1, "opción anterior") de la MediaPlayer actúa dependiendo de la "opción anterior" Este método hace que se lance el evento OnOrderByA() en la clase Album que ordena las canciones dependiendo de la "opción anterior".

(5) Crear lista de canciones recomendada: el método CreateRecommendedListS() de la clase MediaPlayer genera una lista revisando las listas de los usuarios que la persona sigue o canciones a las que les dio like o que agregó a favoritos, etc y si no tiene ninguna actividad, le muestra 15 canciones aleatorias y luego tiene la opción de agregarla a sus playlist como privada o pública,. Las públicas solo las podrán agregar los usuarios Premium.

(6) Crear lista de videos recomendada: el método CreateRecommendedListV() de la clase MediaPlayer genera una lista revisando las listas de los usuarios que la persona sigue o videos a los que les dio like o que agregó a favoritos, etc y si no tiene ninguna actividad, le muestra 15 canciones aleatorias y luego tiene la opción de agregarla a sus playlist como privada o pública. Las públicas solo las podrán agregar los usuarios Premium.

(7) Volver al menú de usuario.

(5) Pestaña Follows:

(1) Reproducir Playlist: busca dentro de las Playlist que sigues y debes seleccionar una y el método Play() de la clase MediaPlayer permite escuchar la Playlist.

(2) Reproducir Playlist de tus usuarios: muestra la lista de los usuarios y permite seleccionar el que quiera ver sus playlist y el método Play() de la clase MediaPlayer permite escuchar la Playlist.

(3) Reproducir canción de Album de tus seguidos: se muestran los album y se selecciona uno y luego muestra sus canciones y se selecciona una y el método Play() de la clase MediaPlayer la reproduce.

(4) Reproducir canción o video de tus artistas seguidos: se abre un nuevo menú.

(1) Reproducir videos del artista: se selecciona el artista del que se desea reproducir su video, luego se selecciona el video y el método Play() de MediaPlayer lo reproduce.

- (2) Reproducir canciones del artista: se selecciona el artista del que se desea reproducir su canción y luego se selecciona la canción y el método Play() de MediaPlayer lo reproduce.
 - () Volver al menú de crear lista recomendada.
 - (5) Reproducir series seguidas: se selecciona la serie que se quiere reproducir y el método Play() de MediaPlayer lo reproduce.
 - (6) Reproducir clases de sus profesores seguidos: se selecciona el profesor al que se le quieren reproducir sus clases, luego se selecciona su clase y esta se reproduce gracias al método Play() de MediaPlayer.
 - (7) Volver al menú del usuario
- (6) Atrás: vuelve al menú principal
- Iniciar sesión como administrador: se le pide que ingrese el código de administrador y su contraseña. El método LogAsAdmin() de la clase Gate verificará que exista un administrador con esos datos, si es que existe se abrirá un nuevo menú.
 - (1) Importar canción: el método ImportSong() de la clase Admin mostrará todas sus canciones que usted tenga (necesario que estén en .wav) y usted podrá seleccionar una y poner los datos y restricciones correspondientes y se añadirá a la lista Songs de la clase MediaPlayer, junto con el artista y el album seleccionado.
 - (2) Importar video: el método ImportVideo() de la clase Admin mostrará todos los videos que usted tenga (necesario que estén en .mp4) y usted podrá seleccionar uno y poner los datos y restricciones correspondientes y se añadirá a la lista Videos de la clase MediaPlayer.
 - (3) Importar karaoke: el método ImportKaraoke() de la clase Admin mostrará todas las canciones que usted tenga (necesario que estén en .wav) y usted debe ingresar la ruta del archivo .txt que contenga la letra de la canción, luego se añadirá el Karaoke a la lista Karaoke de la clase MediaPlayer.
 - (4) Añadir o eliminar video existente a una Serie: se le preguntará si quiere añadir (Add) o eliminar (Delete) un video, tendrá que poner todos los datos y luego el método VideoSerieStarter() de la clase MediaPlayer. Dependiendo de lo que quiere hacer se lanzará el evento OnAddVideoSerie() o OnDeleteVideoSerie() de la clase Admin que agrega o elimina el video a la serie correspondiente.
 - (5) Cambiar su contraseña: el método ChangePassword() de la clase Gate te pide los datos del administrador al que se le quiere cambiar la contraseña y luego se le pide la nueva contraseña y llama al método ChangePassword2() que cambia la contraseña antigua por la nueva.
 - (6) Atrás: volver al menú principal.
- Iniciar como profesor: se le pide el código de profesor ("321") su Gmail y su contraseña el método LogAsTeacher() de la clase Gate verificará que exista un profesor con esos datos, si es que existe se abrirá un nuevo menú.
 - (1) Importar clase: el método ImportLesson() de la clase Teacher mostrará todos los videos que usted tenga (necesario que estén en .mp4) y usted podrá seleccionar uno y poner los datos y restricciones correspondientes y se añadirá a la lista Lessons de la clase MediaPlayer.
 - (2) Editar su perfil: el método ChangeTeacher() de la clase Gate y se le dará la opción de cambiar su nombre de usuario o curso.
 - (3) Atrás: volver al menú principal

Todos nuestros elementos son serializables.