PROGRAMMING DESIGN REPORT

Name: Chuong Ho Dac Thanh

ID number: 101921623

Task to be submitted: 7.3: Program Design

This report aims to outline the most important elements of the program in task 6.3 (GUI Player Music).

In general, the program can be divided into four main modules: LoadApp(), DrawApp(), UpdateApp() and PlaySound(), which respectively loads the data, draws all visual components into the graphical window, updates the data by user's responses in the graphics window and plays the music whenever a music is requested. The in-use data structure is a variable record that contains information about albums, playlists and current opened Album or current opened Playlist as well as the current opened track.

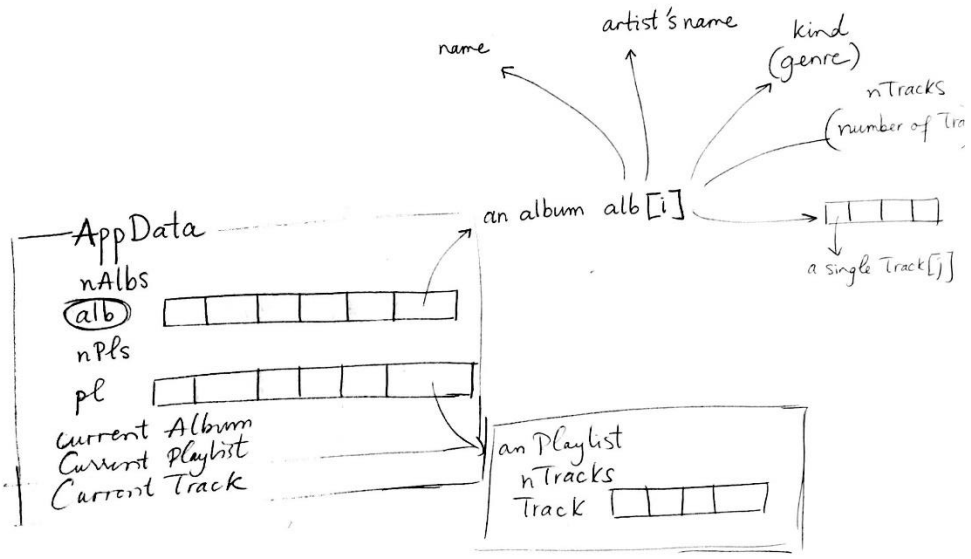

Figure: Data used in the GUI music player program

The LoadApp() procedure calls out an external data file in .DAT format and imports all the data, i.e number of Albums, array of Album, number of Playlists, array of Playlists and current Album/Playlist and current Track, into the main record AppData. Every subsequent execution of the program is based on this loaded-in record.

The DrawApp() procedure draws all elements into the graphics windows. This procedure includes a number of procedures and each of these is responsible for more specific component of the graphics window. For example, the DrawImages() procedure draws every image that represents its album, while DrawTracks draws all the tracks corresponding to the currently opened album or currently opened playlist.

The UpdateApp(), possibly the most complicating part of the program, deals with all changes inside the program. Based on ProcessEvents() provided by Swingame, it continuously updates current album and current playlist and current track by checking the user's mouse input with the help of the function ButtonClicked. An important design decision is that if an album is opened by the user, then current playlist, which has type Pointer, must be nil, and vice versa. The design allows the program to update the list of tracks to be shown on screen properly.

The GUI music program has been further extended by allowing users to create their own playlists that they are free to choose the tracks to be included. After clicking the Create Playlist Button, the Creating Playlist mode is turned on that enables the newly created playlist to be updated as long as the user clicks on his favorite tracks. After that, the user should click on the button Finish creating to turn the program into the normal mode.

Finally, the PlaySound() is responsible for opening the requested music. The track is only called out to be played if there is no music currently playing. This design helps prevent the program from playing the chosen track overlappingly, which can potentially lead to no music being played at all.