	for FLAC files		
Select song by artist, album, genre	Create new Activities and make them work properly	2nd April – 6th April	4th April
Add files to playlist	Create a custom playlist	9th April – 13th April	10th April
Code clean-up	Cleaning the code, adding javadoc, etc.	16th April – 20th April	20th April

As can be seen, there was no clear tendency to lag behind with features or to finish them ahead of schedule. The most time-consuming features seem to be those connected with jFLAC, whereas the ones connected with Android required less time as most of the aspects which had to do with the OS were related with the user interface; the only exception was the seek bar feature. It was a blend of jFLAC and Android development which had to be abandoned due to the time constraints of the project.

Overall, the strategy chosen proved to be quite successful, especially judging by the working software and the amount of features which have been developed. It could be possible to have accomplished more, of course, though that would depend on several factors, such as how much earlier would the implementation stage have been started or if more insight into both Android and jFLAC had been acquired prior to developing this program. Another possibility of improving this strategy would be to add another developer so that the application could be pair-programmed. I strongly believe pair-programming would make this implementation strategy work much better and accomplish much more.

4.2 Implementation of Notable Problems

This subsection will describe in great detail the most important, most interesting, as well as most difficult, problems which I came across while implementing my design. It will include pieces of code in order to highlight some of the solutions. It will also clearly state whether the implementation worked, how well it worked and if the feature which required it was developed on time and finished.

4.2.1 File Browsing

To my surprise, the first major problem (albeit not even half as difficult to solve as the ones that came later) I came across to create a file browser for my music player as a part of the "Select song by file" feature. It was quite important to implement it because selecting a song by file is the most basic form of selection any user might need as choosing a file does not require any kind of interaction with the file itself; the only thing needed is a screen with a file browser so that the file can be clicked. This might seem as an easy task but it is worth knowing there are applications on the Google Play whose sole purpose is being file browsers. Thus, my application could, at least to a certain degree as it would not be a very advanced file browser, be treated as two different applications merged into one.

The first thing I did was to carry out a small research into Android-specific solutions to see if there already is any kind of object which would help me easily display a list of files and sub-directories in the current directory as well as move up and down in the folder hierarchy. There was no such convenience, however, therefore I had to implement this all by myself.