

5 Testing

This section of the report will provide details to the testing of the software whose implementation was described in the previous chapter. It will explain the overall approach to testing and its two remaining sub-sections will cover the areas of black-box testing and white-box testing.

5.1 Overall Approach to Testing

Testing a music player is not an easy task as the application's main purpose is to play music and it is highly unlikely it would be possible to create software tests to check if a device's speakers produce sound. Thus, perhaps the most important tests for such a program are functional tests. Of course, white-box testing was also performed, though to a smaller degree than acceptance testing, especially due to the fact that the hand-in date for the project was quite pressing at the time these tests were being created and carried out.

5.2 White-Box Testing

White-box testing was performed two-fold: firstly, a set of Android JUnit tests was created. Their task was to test the most basic units of each class, to prove their methods work correctly. All of the unit tests are located in the `android.test` package, which currently contains only two classes: `AlbumTest` and `SongTest`. These are the test classes for `Album` and `Song` classes, and they contain some test cases for methods belonging to said classes. Unfortunately, it was not possible create more unit tests due to the pressing deadline of the project.

Moreover, most of the white-box testing was carried out "on-the-fly" by utilising the `Log.e` method which prints out to the Android's error log. This was the best way of checking if a method returned correct values or if the implementation of an algorithm worked correctly.

All in all, the white-box testing was quite helpful in the process of developing this project, though it was most of the time only to determine where a problem lies after a functional test failure. If there was more time given to create this application, more tests would surely be written as well, especially Android-specific tests. However, learning to design and implement such tests would require much more time, hence this idea had to be abandoned.

5.3 Black-Box Testing

Black-box testing, also known as functional testing and acceptance testing, was the more important approach in this project as the music player did not require any very complex algorithms but was rather function-oriented. Thus, the best way of determining if the application worked as intended was to simply run it and try to exercise its different features – e.g. try to select a song and play it to determine if the output can be heard.

Thanks to the fact that the functional testing can be done even by people who do not know anything about programming, such testing of this music player could be performed using two slightly different approaches. The first one was carried out during the development of the application – a feature would be implemented after which the