Theresa Pham

CSC 415: Software Engineering

Assignment 4

Open Source Software: Analysis & Design

How Requirements Are Met

Modularity & Encapsulation of Data:

    The program will be modular in how it splits up the classes not only by the MVC design pattern though use of Entity, Boundary, and Controller classes, but also by splitting the classes to achieve high cohesion and low coupling where any controller classes and functions within them have one task they are trying to achieve and are only taking parameters from other classes that are needed for use. Because much of the data within the program will need to be able to access each other, much of the data and functions are made public for easy accessibility, though some of the functions within the GeneratePlaylist class are made private so that the algorithms within the functions cannot be tampered with through other classes.

Elegance and Efficiency of Algorithms:

    As the project is, there is no effective way or searching for an album. The user must either find the URI of an album and paste that in, or drag the album cover in from Spotify. For the creation of a searching functionality, albums specifically will be searched for based on keywords. When an album is selected from the search results, it will parse the JSON of the album for its URI in order to match the existing functionality of the project and display the countries on the map. In addition, because Spotify's

database includes thousands upon thousands of songs, it is important to be able to sort through those songs quickly and efficiently. A merge sort is a quick way to sort large amounts of data in which most of it does not need to be stored, which can be made more complex by narrowing the genre or examining the user's Spotify preferences if connected.

Appropriateness of Data Structures:

A priority queue implemented as a heap structure is very appropriate for this application as heap structures are ideal when comparisons need to be made between nodes in order to generate playlists either by popularity or by other factors, and playlists list songs in a first-in-first-out manner like the queue structure.