Liam Saxon

[Email address]

AT3-Java III

Table of Contents

[What data structures are you using? 2](#_Toc40867286)

[Where are you using hashing techniques? 2](#_Toc40867287)

[What sorting algorithm are you using how this is different from selection 2](#_Toc40867288)

[and bubble sort? 2](#_Toc40867289)

[What search technique are you using? 2](#_Toc40867290)

[What third party libraries are you using? 2](#_Toc40867291)

[Where can I find the documentation for this? 2](#_Toc40867292)

[A mock-up of the GUI. 3](#_Toc40867293)

[What source control are you using? 3](#_Toc40867294)

[What are your coding standards you are enforcing? 3](#_Toc40867295)

[What tests are you going to run? 3](#_Toc40867296)

# What am I making?

I will be going with the default media player with a few minor tweaks to the requirements. For instance, the Data structure I use will be a Doubly linked list instead of a binary tree.

# What data structures are you using?

* Doubly Linked lists

The DLL will be used for storing the songs and the URL’s the URL will be where the song is located on the targeted machine. The DLL allows for traversal through the song list and removal or addition to the list with ease.

# Where are you using hashing techniques?

I will have a login form where the user can provide a Username and password. The hashing will be used on the user’s password to ensure a level of security.

# What sorting algorithm are you using how this is different from selection

# and bubble sort?

I will be using the Merge sort the major difference of this sort is the time complexity compared to the two mentioned above. Insertion and Bubble sort both have terrible time complexity’s, on average they both equal n^ 2.

Merge sort on its best, worst and average all even out at a lovely n\*log(N).

# What search technique are you using?

I will be using a Binary Search for any results that need to be searched.

# What third party libraries are you using?

I will be using Hamcrest. It is a 3rd party library that allows for writing matcher objects allowing ‘match’ rules to be defined declaratively. This works in conjunction with Junit Testing.

# <http://hamcrest.org/JavaHamcrest/>

JFoenix may also be included to have some nicer and easier to use UI elements, depending of how well they mesh with the rest of the assignment.

<https://github.com/jfoenixadmin/JFoenix>

# Where can I find the documentation for this?

The documentation for Hamcrest can be found on their website

<http://hamcrest.org/JavaHamcrest/javadoc/2.2/>

The documentation for JFoenix can be found on their website

http://www.jfoenix.com/documentation.html

# A mock-up of the GUI.

# 

# What source control are you using?

I will be using github to version control my program.

# What are your coding standards you are enforcing?

I will be using the coding conventions outlined in the Oracle document, Java Coding Conventions

Link:

https://www.oracle.com/technetwork/java/codeconventions-150003.pdf

# What tests are you going to run?

I will end up doing a test case table listing and testing individual components with screenshots as proof that the individual components work as needed.