A1 Report

Step 1

Using external libraries make it easier and faster to implement certain features. If the project did not utilize Apache CLI to gather the command line options, it would have been much more time consuming to create a class manually that performs the same feature. This also holds true for the Log4J library, since it reduces the amount of work needed to implement that specific feature. Overall, it allows us to focus on the goal instead of focusing on smaller, non-important features.

Step 2

A logging approach provides a way to track where certain lines of code are executed. For example, we can easily tell from which method of the class (in this case Main) the code was executed from, and it also indicates what type of logging it is, such as plain info, or an error that needs to be addressed. One disadvantage of logging is that it may not be useful when you need to print a line to the user for them to read, and you don’t want all the extra information to go with it.

Step 3

Many features were added to the Kanban board for this project. It was difficult to identify things that are only visible to the end-user due to the technical complexity of this project, however, the business logic specification of the project provided a general description of what is to be expected of this project, which outlines all the features that should be visible to the end-user. As a result, the business specification was used heavily in the determination of features to work on.

This ensured that the features only model visible value to the user, because the business specifications do describe the “how”, or “why” each feature is implemented, but “what” features need to be implemented. Also, the user is not concerned about the technical process involved in achieving a certain feature, so that was also taken into consideration when labeling the names for each new “issue” in the Kanban board. For example, the feature “Verify path provided by the user” does not describe the process in which it would occur, it simply identifies that this is what to expect as one of the utility features in the script.

Step 4

The version of the program under the”mvp” tag on the GitHub repository represents the minimum viable product for this project. It is considered viable, because it achieves the main functionality of the program, which is to provide instructions to solve a maze. It is minimal because it does so in the most simplest form: Solving a straight maze where all you need to do is move forward.