Criterion A: Planning

Statement of Problem

My client is Mr. [redacted], a **Robotics teacher** at my school and my advisor is the High School IB Computer Science teacher. In Robotics classes, the teacher assigns exercises for middle school students to do. The assignments are mainly completed on the computer as they work with the school's robotic devices to understand electronics and become interested in computers.

The school has a robotic arm, called the uArm, that isn't in use. The client would like for the students to use the arm and complete exercises with it. However, most of the applications found online weren't safe and simple enough for the students, as the speed and the control application didn't fit the requirements the client hoped for (see Appendix 1).

Thus, after discussing with the client, I proposed to develop a user-friendly application using Java for middle school students when working with the arm (see Appendix 1). This application would have all the features that can be accomplished when working with other electronics in class. Additionally, a safe environment is needed when the students complete the exercises using the robotic arm.

Rationale for Proposed Solution

I have chosen to use the Eclipse Integrated Development Environment with one of the most utilized object-oriented programming languages, Java since I am more familiar with it than Python (see Appendix 1). This technology will create a user-friendly environment and an application that works on different operating systems since students use Windows and macOS at our school. With the use of Java, simple manipulation of files: reading and writing files can be achieved since it's easier to send files of their progress to the teacher rather than filming it (see Appendix 1). Additionally, Eclipse will allow for effective management of code, provides auto-completion and verification of code to successfully create the application.

The GUIs of the application will be built using SceneBuilder and the JavaFx library on Eclipse. FXML, a declarative language, will be used in the project because it requires less code and can more efficiently create aesthetic GUIs, moreover, using the CSS language in the FXML allows customization. Furthermore, it separates the *view* definition from the *controller* and *model* parts of the application, allowing for better management.

Also, I will understand the protocols needed to communicate with the uArm in Java, as this is the purpose of this project. This will be accomplished by using external libraries in Java so that communication between different programming languages isn't needed to save time and reduce complexity (see Appendix 3). For the students, the GUI will allow the communication between the arm and the user to be simple. Additionally, the speed of the arm can be set by the developer to create a safe environment for the user (see Appendix 1). This is done in hopes of encouraging students to become interested in Electronics and Computer Science.

Stating Success Criteria

1. Create an aesthetic graphic user interface that can be easily utilized by middle school students.
2. Establish a connection, whereby communicating with the robotic arm using the application on a computer using Java.
3. The application must allow the user to use an arm, by having a simple yet powerful graphical user interface.
4. The speed of the arm should be slow so that it can be considered safe by the client.
5. The user can use all the functionalities of the arm: moving the tool head to a location and grabbing small objects.
6. The application should allow simple management of a list of exercises that the students will complete.
7. The application should allow the user to easily manage the files of the application.
8. The application should allow the teacher to send exercises to his/her students.

Word Count: 450 (Excluding Titles, Success Criterions, and Parentheses)