This project was a great challenge that presented us with many learning opportunities. We have incorporated different concepts learned in class and gained a solid understanding of how they can be used in the context of our project. We, therefore, believe that our project does fulfill the standards that are being evaluated in the following ways:

Functionality: - 18

Our game compiles and runs successfully. Once in a while, a GO AWAY error may pop up because of the trivia API we used in our game but it should be fine once you rerun it. All keys and mouse implementations work as explained in the documentation.

Design: - 17

We have submitted the breakdown of our game design in the program specification document. We did not over-engineer our solutions to the problems that came up. Using the N-ary tree to handle administering the questions was particularly clever.

Creativity: - 20

Our game is creative especially because it is not something that has been implemented before. In fact, the feedback we received from our peers in class emphasized this quality of our game.

Sophistication: - 16

Our game is sophisticated in terms of the data structures we have implemented. We used an N-ary tree that helped store data for the questions. We also used an external API to generate random questions for each category. Simulating the flawless movement of the player sprite was sophisticated too.

Broadness: **-** 20

- We used an external library to handle the API response and format the questions to be used from JSON format to Java Primitives/Objects (ArrayList and String) (json-simple-1.1.1). We also used the Buffered Image library to handle the images, Image I/O which are both inbuilt in Java.
- We used Generic nodes that can take any type of data. If we wanted our game to take in integers, we would easily make that modification.
- For file input, our game uses external image files to draw all the images on the screen and the different positions of the sprite.
- We used randomization to generate the questions displayed at random and to shuffle the choices before displaying them to the user.
- We used built-in data structures including ArrayList, Arrays
- We used user-defined structures eg: The N-Ary tree

Code Quality: - 16

We tried to avoid unnecessary repetition of code by using methods to call similar lines of codes which were called in different instances.

We used well-thought variables and class names in our project