Criterion A: Planning

Word Count: 484

DELIVERABLE 1: THE SCENARIO:

The client, Mr. xx, is secondary design technology at ISL. The client saw simulations and challenges as the best and most engaging way to teach students about Java coding and enhance their problem-solving skills and process.

The client's problem is that he wants to simulate the 2021 Perseverance rover landing on Mars so that the client can use it as a Java teaching tool for young programmers, a challenge to be more precise, with a learning focus on if conditionals. The client wishes that "Ideally, you would create a mars rover spaceship as such which the students would be able to code themselves and it would run against the simulation to see if they coded it correctly and if they do things at the wrong stage then the rover would crash." (Client, 2021). The client also specified some extensions for the simulation such as; " take into consideration the weather when the rover landed at mars. Therefore every time the simulation would have different winds... it would be really nice if the simulation could have some graphics of what is going on." (Client, 2021).

DELIVERABLE 2: RATIONALE FOR THE PROPOSED SOLUTION:

I believe that OOP Java programming will help me solve Mr. xx's problem since it would be an efficient way to solve it. I could use composition and inheritance to make the process of coding more efficient and shorter. The aim will be to create the perfect simulation of the Mars rover landing, which later the students will import to their Intellij project and compare their simulation to mine. The client also added; "I wonder if it would be easy to have just a paper to go supplement with it." (Client, 2021) Therefore, the students will also have a printable help documentation in which they would read all the information necessary to complete the simulation and use it as a helpful guide throughout the challenge.

According to the solution and requirements demanded by the client, I believe that my solution would be the best fit coded in Java instead of another language because of;

- The client is personally asking for the challenge in Java since it is the language he teaches and the language in which the students will learn how to code.
- Java supports encapsulation, composition, inheritance, polymorphism, and abstraction (Altvater, 2017), making it an OOP language, therefore an excellent language to code the solution to Mr. xx's problem.

- GUI (Graphical User Interface) in Java is an easy-to-use visual experience builder for Java applications. (Hartman, 2021).
- Java also supports different platforms such as MacOs and Windows meaning more compatibility for Client’s student.

In terms of the simulation, I believe that starting from the De-Spin stage would be ideal since it is the first stage at which the students would be able to insert functions and start to affect the rover's landing. The simulation would later end in the Surface Operation stage, meaning that the simulation would have 16 different stages, each one different from the other. I will call this challenge “Perseverance Landing Simulation Challange”.

STATING THE SUCCESS CRITERIA

1. Code/GUI has all Landing Stages
    a. De-spin
    b. Cruise Balance Masses Ejected
    c. Entry Interface Point
    d. Guidance Start
    e. Heading Alignment
    f. Begin SUFR
    g. Parachute Deploy
    h. Heat Shield Separation
    i. TRN Image Acquisition Begins
    j. TRN Valid Solution
    k. Backshell Separation
    l. Descent Stage Throttle Down
    m. Rover Separation
    n. Touchdown - Flyaway
    o. Descent Stage Engine Cutoff
    p. Surface Operation
2. Include a GUI which live feeds the simulation of the landing
    a. Basic layout
        i. Includes a NASA Logo
        ii. Logo of the Perseverance Rover 2020
        iii. Frame of the GUI
    b. Include a ImageView of mars rover including the main parts
        i. Heat Shield
        ii. Main Rover
        iii. Thruster Attachment
        iv. Parachute
        v. Backshell
        vi. Thrusters
    c. Include texts which live feeds the crucial information regarding the Perseverance rover
        i. Current Stage
        ii. Distance from landing site
        iii. Velocity
        iv. Altitude
        v. Touchdown time
        vi. Time until next phase
    d. Terrain
        i. Fly terrain
        ii. Landing terrain

3. Include a Main class in code where the user can code their solution to the simulation and run to test it
4. Have a randomized mars terrain
    a. Mars terrain will be constituted of 2d Array of randomized size
5. Simulation will include a manual/document including:
    a. Stages of landing with basic information such as the altitude, distance, velocity, the time until the next phase of landing, and the time to the rover touching the ground:
        i. De-spin
        ii. Cruise Balance Masses Ejected
        iii. Entry Interface Point
        iv. Guidance Start
        v. Heading Alignment
        vi. Begin SUFR
        vii. Parachute Deploy
        viii. Heat Shield Separation
        ix. TRN Image Acquisition Begins
        x. TRN Valid Solution
        xi. Backshell Separation
        xii. Descent Stage Throttle Down
        xiii. Rover Separation
        xiv. Touchdown - Flyaway
        xv. Descent Stage Engine Cutoff
        xvi. Surface Operation
        xvii. Function (success Criteria 2)
    b. Functions
        i. Release parachute
        ii. Initiate thrusters
        iii. Release heat shield
        iv. Separate backshell
        v. Release mass
        vi. Initiate Camera
        vii. Initiate cables lowering
        viii. Cut cables