Criterion E: Evaluation

Word Count: 532

DELIVERABLE 1: SUCCESS CRITERIA AND FEEDBACK FROM CLIENT

Criterion	Client	Myself
Code/GUI has all Landing Stages: De-spin Cruise Balance Masses Ejected Entry Interface Point Guidance Start Heading Alignment Begin SURF Parachute Deploy Heat Shield Separation TRN Image Acquisition Begins TRN Valid Solution Backshell Separation Descent Stage Throttle Down Rover Separation Touchdown - Flyaway Descent Stage Engine Cutoff Surface Operation	"I'm really happy with the GUI. I think it has properly exceeded my expectations on what could be achieved on the time you hads to work on the project." (Appendix Criterion E)	The GUI/Code hits all of the landing stages every time it is run without an error. There are a few glitches at the start where it freezes a bit however that is out of my control.
Include a GUI which live feeds the simulation of the landing: ***Basic layout*** Includes a NASA Logo Logo of the Perseverance Rover 2020 Frame of the GUI Include a ImageView of mars rover including the main parts Heat Shield Main Rover	"Also included criteria 2 and I think it was a good idea of including this section here at the bottom and that you have used this style green text it kind of gives it a more of program old style program and that they would use something like that" (Appendix Criterion E)	The user when opening up the GUI has a clear view of all crucial information necessary to perform the challenge in addition this information is updated according to an amount of time given by the user. Furthermore the simulation performs all animations of actions which the rover would perform in real life.

Thruster Attachment Parachute Backshell Thrusters ***Include texts which live feeds the crucial information regarding the Perseverance rover*** Current Stage Distance from landing site Velocity Altitude Touchdown time Time until next phase ***Terrain*** Fly terrain Landing terrain		
Include a Main class in code where the user can code their solution to the simulation and run to test it	"In terms of the stages of the code I am pleased with how the student can query the state of the rover and how simple it is." (Appendix Criterion E)	The user has been provided with a method in the main class where they can code their solution to the simulation and run it with ease.
Have a randomised mars terrain: Mars terrain will be constituted of 2d Array of randomised size	"The 2D array was also a very nice part of your project and it also added more difficulty into the challenge itself which is a good thing." (Appendix Criterion E)	Each time the simulation is run the mars terrain changes in size and the zeroes also change along with it. It is fully randomised.
Simulation will include a manual/document including: ***Stages of landing with basic information such as the altitude, distance, velocity, the time until the next***	"I think you have put a lot of good work for this um again it needs more testing" (Appendix Criterion E)	Personally I think this is good and as the client said there was a lot of effort put into it. However there is more testing to be done in order to perfect this part because when testing it with students there was

phase of landing, and the time to the rover touching the ground: De-spin Cruise Balance Masses Ejected Entry Interface Point Guidance Start Heading Alignment Begin SUFR Parachute Deploy Heat Shield Separation TRN Image Acquisition Begins TRN Valid Solution Backshell Separation Descent Stage Throttle Down Rover Separation Touchdown - Flyaway Descent Stage Engine Cutoff Surface Operation Function (success Criteria 2) Functions Release parachute Initiate thrusters Release heat shield Separate backshell Release mass Initiate Camera Initiate cables lowering		some crucial information missing such as the initial mass of the rover, which is needed in order to complete the mission.

DELIVERABLE 2: RECOMMENDATIONS AND FUTURE IMPROVEMENTS

Pause and continue function (Run a stage then click to continue)

After running a test session (Appendix Criterion E) it was evident that the first feedback received along with improving manual documentation was to include a pause and continue function were the user would be able to pause and unpause the GUI simulation, this would provide the user with more information to see where they are messing up throughout the challenge.

Manual Documentation

As seen by the feedback the biggest problem with the challenge was not the code itself but the manual documentation given to the user. According to various students there was key information missing such as a better explanation of the TRN Image acquisition Begins and small things such as no initial mass of Mars Rover given.

Individual testing of stages instead of all together on the run

Finally, one last feedback given by the students was that it would have been more interactive if instead of running all the simulations at one and testing if the user code works, to have the possibility for the user to run stage by stage of the landing. This would be more time efficient and allow the user to better understand where a mistake could have been done.