

# Appendix

---

<b>Bibliography</b>	<b>2</b>
Criterion A: Planning	<b>3</b>
<b>Interview 1: The Problem</b>	<b>3</b>
<b>Interview 2: The Solution + Success Criteria</b>	<b>4</b>
<b>Criterion E: Client Interview</b>	<b>7</b>
Interview 3: Hittin The Criteria Success	7
Target Audience Evidence:	8
Student Manual:	9

## Bibliography

<https://www.guru99.com/java-swing-gui.html>

<https://stackify.com/oops-concepts-in-java/>

# Criterion A: Planning

## Interview 1: The Problem

Client: So what I've got really is that I want to have a programming challenge for students, okay and the challenge is for them to try and land the mars rover on mars. It should simulate the real mars rover landing, so there is a lot of the stuff right now which is at the NASA website, and it allows us to get an idea of the stages in which it goes through, for example when it is at a certain altitude at a certain speed the supersonic parachute will come out and the next stage would be to deploy the parachute, start up the rockets look for a palace to land, and look at all these different stages. 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 whether they coded it correctly and if they do things at the wrong stage then the rover would crash. As extensions possibly what would be really cool is if the simulation would take into consideration the weather when the rover landed at mars, therefore every time the simulation would be different winds. And on top of that, it would be really nice if the simulation could have some graphics of what is going on.

Me: So like Tetris, Pac-Man kind of graphics.

Client: So like a 2D view of the stages as it is coming down. Would be more than enough even like the old-style space invader where you can see the ground coming, just so there is a visual representation as well. But that is the kind of a nice extra, if that is not possible then a console output would be representative of what they would see at NASA when they see the data coming in from the rover

Me: Ok so that would be like a text saying "The mars has landed"?

Client: Yes, it would also give the current statistics, so at the speed, they are moving current weather conditions, and then it would say "parachute deployed" or it would say "lost communication" which would give the indication that the rover has crashed.

Me: Ok thank you.

## Interview 2: The Solution + Success Criteria

Me: The solution would be through the lenses of Object Oriented Programming. Including one world, one actor and different children within that actor, where the actor would inherently different methods. I was thinking that one of the success criteria would be for the code to have comments in order for the students to use as guide.

Client: Your code is some type of guide. I wonder if it would be easy to have just a paper to go supplement with it. Like a little bit of documentation to tell them the type of documentation they can use.

Me: Ok so some type of manual or guide for them to use as the simulation occurs. Like a normal document including a manual and a guide.

Client: Ok so a document including the functions they can use.

Me: Ok so an optional success criteria that could work would also as you said before; have a 2D graphical display so that the user can see once they run their own simulation.

Client: Like a 2D graphical, that would be great if we had that. However, if that is not possible then like a console representation, so it looks like the kind of console you would expect to see at NASA.

Me: And then I was actually going to ask you what other success criteria I should include?

Client: Ok so how many stages would the simulation include?

Me: I don't really know at the moment; but it would start from the de-spin stage as we talked about, to the final landing stage.

Client: Ok so maybe we can expand on that one and write them up, and then check which stages should be included in the simulation and those off.

Me: And then maybe a success criteria could be talking about real weather conditions in the scene.

Client: So yeah, to use the NASA API of the weather conditions on Mars, to influence the weather. The thing is that with that idea, every time they try it; is not always the same condition, so the conditions need to be randomized , so they can't just write a static code that they could use in order to land always on the same space with the same coordinates. So another thing we could talk about is, having like randomized coordinates, and randomized

weather then.... I mean if they are going to write their code in intellij they could always import the project to their library.

Me: Also notifications of current status would be necessary for the user to know what is going on and the conditions of the rover. So like crash, landing successfully ect.

Client: So the user is notified, or maybe the user can request status. So if we are going to keep it real then we could also have something like if the rover has lost connection then the user would get a notification of "...lost signal..." which means that the rover has crashed.

Me: Ok.

Client: So the only other thing that I guess we need to think about is, what functions would the user be able to use.

Me: That would tie in with the stages.

Client: Yes and also like we just talked about, check status. And just from the top of my head like, release parachute.

Me: Yes, also, initiate thrusters, release heat shield. Ect. Yes Ill have a look

Client: Ok have a think through, so we have the different stages that the rover goes through and the different actions which the user can use.

Me: Also in the manual I could divide, in case the students need help. Like a help guide of the solution, but however I could use a solution, like in this stage you should do this ect.

Client: I think that is pretty much it. Shouldn't it. Ye I suppose what you could do to expand what you could do. Like what each function would do, however some of them should be self-explanatory.

Me: Like I could do that, I'll look into it. Like having a little box on the side of the main code where it shows what happens as the rover is falling.

Client: Yes and also a little status box which includes information, and when it crashes everything goes dark and little message shows "signal lost". I think that if I were to use this in the lesson as well it would be essential for "if " conditionals.

Me: Ok thank you, is it ok if I turn it off?

Client: Yeah yeah.

## Criterion E: Client Interview

### Interview 3: Hittin The Criteria Success

Client: So going from the first success criteria, really happy with the GUI I think it has properly exceeded my expectations on what could be achieved on the time you had to work on the project. Yeah I think the graphics you have used are really good. I think it looks authentic. I like the idea of the addition of the wind. It is a really good addition to give it some momentum. I also really like the thrusters that are kicking on and off, as well as the parachute and other animations you have put in. So yeah really happy with the GUI. Um great you have included all the stages that's perfect and you have 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. 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. This student code gets run in a while loop?

Me: Yeah

Client: So the student does not have to worry about this at this stage which is good because I really just want them to focus on the if statements. I think you have exposed the method well for the user and some of this there is nothing that the user has to do except keep the thrust on right?

Me: Yeah

Client: I think this really nice and clear for the user gives them a clear and nice idea of how to use if statements. I would be really interested to use this in class to see how much fun they get from seeing it crash or succeed. 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. Um documentation, I think you have put a lot of good work for this um again it needs more testing, how do you think it went on testing?

Me: I think it went well but I think it was missing more methods and more description of the landing stages. I also think that the main thing missing is the explanation of the TRN Image Acquisition Begins where the user has to search for the 9 zero's on the 2D arrays.

Client: Yes I also believe that adding um a slider where the user can control the velocity of the simulation so they can go faster on certain parts or slower when necessary.

Me: Yeah that would be a good idea.

## Target Audience Evidence:

Figure 1: Students solving 2D Array for TRN Image Acquisition (Author's Own, 2022)

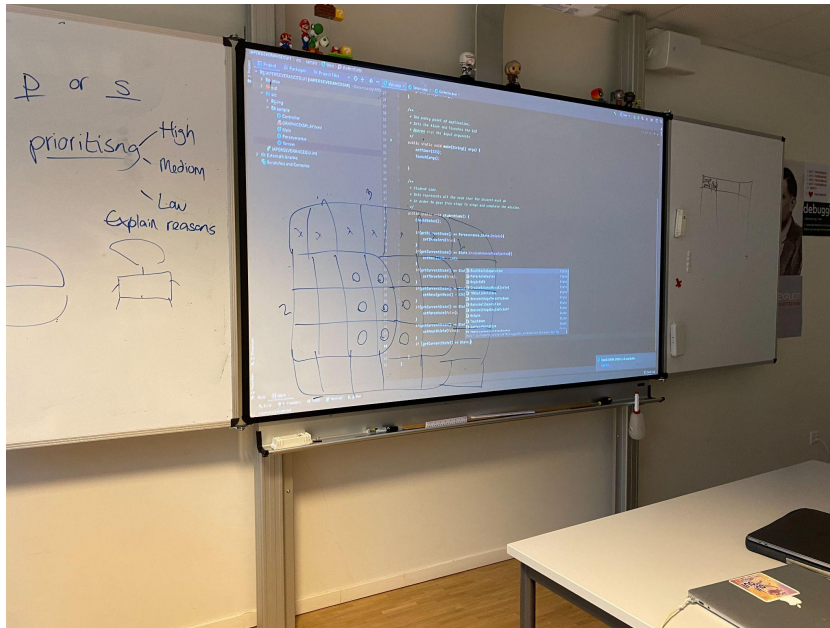


Figure 2: Students running GUI (Author's Own, 2022)

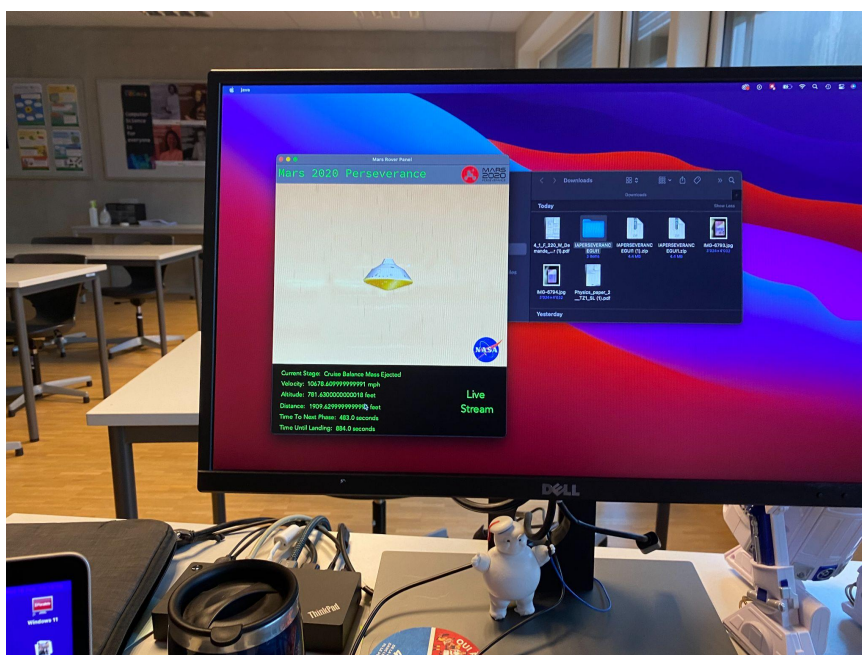




Figure 3: Student writing on studentCode() (Author's Own, 2022)



### Student Manual:

Link To User Manual:

<file:///Users/vicentefeliu/Downloads/IA%20Perseverance%20Student%20Handbook.pdf>