

Mars Lander Game

Report Name	Project Outline
Author (User Id)	Ryan Pargeter (ryp9)
Supervisor (User Id)	Laurence Tyler (lgt)
Module	CS39440
Degree Scheme	G400 (Computer Science)
Date	February 8, 2019
Revision	1.0
Status	Release

1 Project description

The Mars Lander Game project will develop an android application that is designed to be a fun way to learn about how our current technologies deal with the entry descent and landing on mars.

We currently have a game called Lunar Lander that does something similar for the moon. The moon has no atmosphere and so just uses basic thrusters to land, fortunately because of the lack of atmosphere it can't burn up, so the entry, descent and landing systems are very simple. the lunar game is also very basic (2D, essentially landing on a line). It is also a web-based game and so it is difficult to use on mobile devices and doesn't store any of your information e.g. best score etc. Another problem with the Lunar Game is that it does not appear to be based off the real lunar gravitational pull. I will have a realistic representation of mars this will make it more fun to play, as we will build up speed on entry and we will have to do things like detach heat shields during entry. It also relies on an internet connection, so if we are going to use it on the move we may not always have.

The Mars Lander Game will be an android application so that it can be used on the move. It does not rely on an internet connection to run as it will be stored locally. It will be fully 3d to make it more like an actual Martian landing mission. I will base the Lander of current robotic missions and I will use their ideas for entry, descent and landing [1]. The first stage of landing will be Entry. This will involve simulating a realistic Martian atmosphere. This Stage will involve having a heat shield facing the correct direction so that we don't burn up in the atmosphere. I will have some controls to keep the lander straight so that there is a game element to this stage the more of the (non heat shielded part) lander that is exposed to the heat the quicker the lander will burn up, I will have some sound effects to show when the lander is very damaged.

The next stage is to deploy a parachute and detach the heat shield, the key to this stage is timing if we do it too early then we are going to be going too fast for our parachute and if we do it too late we will not have enough fuel to slow down enough for a safe landing.

Finally we will have our lander covered in bouncing gas bags and we will use thrusters to slow the landing once the user has decided we are going slow enough we detach the tether to allow the lander covered in bouncing gas bags to hit Martian atmosphere, if we are going too fast then we will smash or bounce unpredictably if the user does everything correctly then they will get points based on if they have landed on a designated area worth more points (bulls-eye) and also how much fuel they have left and how little damage they have done to their lander.

once they have done this the system will calculate their points and display them in a leader board which will be stored locally. It is important to note that this is the basic idea of the project if I have more time I will attempt to have different entry, descent and landing systems but that will have to be decided later in the project depending on progress (described in next paragraph).

After the start screen the user will be able to make a choice of the type of EDL system and the type of lander/rover they wish to land with. The weight of the lander will determine which landing systems will be successful e.g. for heavy landers a dead-beat gas bag will not work as it will roll over this will end with mission failure. There are many different types of EDL, bouncing gas bags is the easiest to actually land but it is the hardest to know where we are going end up at the end. Things like a sky crane are much more complex but it has extremely accurate landing (actually has automatic landing site detection on board).

2 Proposed tasks

- **Research on how to do 3d Modeling on android** - I have already done some research into this and it seems that OpenGL ES [2] will be the best for this as I have done some basic

OpenGL in the past and it is built into android studio which is where I will be developing my app. I have done android development in a module last semester and so I will not need to do any research in that area. I will also need to look into what I should actually make the models with as OpenGL is just an open library that allows us to contact out graphics card, I could manually program the vertices but that would be exceptionally difficult and do I will need some form of software to help me create my 3d models before something like OpenGL can interact with them (blender will be a logical choice as I have done it in a graphics module in 2nd year). After some prior experience in OpenGL I think I will be able to work with this library alone without using anything on top of it, the only thing I will use to aid OpenGL is Blender to create my objects as manually coding them will be almost impossible.

- **Deciding on and setting up Version Control and Continues Integration Systems** - I will need to have a version control system so I can easily store my repository and access it from anywhere with internet it is also suitable because my project supervisor could access it to check on progress. I will need to think about using a continuous integration system such as Jenkins I am not sure my build will be big enough to warrant using something like this and i am not sure if I will need to run all my tests daily on a system like this because i don't think it will take that long to run them all manually.
- **User Interface** - I will need to create a leader board and a start game page so the user can decide on a course of action instead of being forced straight into playing the game.
- **3d modeling and interaction** - this will be a large piece of work involving developing each stage of EDL and it will be the look and feel of the game.
- **Martian atmosphere research and implementation** - I will need to do research on the Martian atmosphere and how it will affect the mars lander I will then need to apply this physics system to my game. The key to this will be determining how quickly the lander will fall towards mars and at what speeds the lander will burn up. This will be the main gaming element as it gives the user a way to loose and achieve.
- **Entry, Descent and Landing research and implementation** - Gas Bags won't work if we land too quickly and dead beat bags will tip if we are too heavy or land at a severe angle and things like a sky crane rely entirely on smart consumption of fuel. These will need to be researched and the advantages and disadvantages implemented in my game.
- **testing** - I will need to test my project regularly I will not do Test Driven Development, but I will create a unit test for each part of functionality (class) and I will do user testing regularly and at the end of the project.
- **meetings and diary** - I will need a weekly meeting with my supervisor and a project diary that I will write in every day so that I can keep track of my progress and make it easier to talk about and write up my work. I will also follow Scrum like methodology where i will plan week long sprints.
- **Preparation for demonstrations** I will have two demonstrations to prepare for and I will need to plan for these by writing a script and preparing my application for demonstration.

3 Project deliverables

- **Mid-Project Demonstration** - At this point in development I intend to have completed the main user interface and I want my model lander to be able to simulate simply falling

towards mars. I will write a script talking about the progress of my project and a demonstration of the functionality I have got so far.

- **The Application** - This will include everything necessary to run the app, It will include the user interface, 3d models and background code to run the physics systems. I will split this into User Interface delivery (the main page and general look and feel), 3D models (the Lander and the Martian surface), Physics system (gravity and atmosphere) and the 3d model movement and interaction (the users controls of the lander and the effect of the designed gravity on the lander).
- **Research Reports** - I will need to research Entry, Descent and Landing systems and the physics of mars. These are essential pieces of work to develop my games and will need to be documented thoroughly. I will also need to review OpenGL and how it will be used to help me create my game, this will include a discussion on what i have had to use in conjunction with OpenGL e.g. Blender.
- **Tests** - I will need to have assigned tests for users to run to ensure everything is tested and work to the users expectations. these tests should be very generic e.g. burn up in atmosphere, this is so that the user needs to navigate themselves to the solution to determine how user friendly it all is.
- **Final Report** - this will be the big report at the very end of the project, it will discuss the process of achieving my application (design steps and choices) it will also contain any research information I have collected and references I have used.
- **Final Demonstration** - This will be much like the Mid-Project Demonstration but it will be longer and will involve using (hopefully) my fully functioning game and some discussion on the design decisions I made during the creation of this app. I will write a script describing all of my game functionality.

Bibliography

[1] - <https://marsmobile.jpl.nasa.gov/mer/mission/timeline/edl/steps/> I used this to confirm my ideas for entry descent and landing, as it has approximations on distances, I will find this extremely useful in the designing of my app

[2] - <https://developer.android.com/guide/topics/graphics/opengl> I found this whilst searching for a way to 3d model in android, using this link I found out I could use OpenGL on android. I already know a bit of OpenGL, so it seems like a natural choice.