Project proposal

Space Travel

Aleksandra Musial

Zami Nizam

Project description

The name of our project is going to be “Space Travel". The theme of our project is to simulate a space where our users can move around and explore some mysterious parts of Space. The main purpose of this project is to demonstrate our knowledge on working in 3D and to make an app for children to make them interested about space and its beauties within its mysteries. This project does not directly relate to our creative research since the one we did was a study on penguins. However, our goal with the creative research project was to educate people about penguins and to point out that there is very less data available online about penguins comparing to other animals. And we plan to educate people about space and solar planet system, stars systems, wormholes, black-holes and other celestial bodies with this project as well. The final output we are looking forward to achieving, is a simulation of our solar system and space travel. As we are planning to let our users travel through different parts of space virtually and learn in brief or detail about various space objects, their formations, what they are made of, their movement, size comparisons of celestial bodies etc. To achieve that we are planning to use 3D graphics to generate a basic skeleton of the whole app. That way we can apply trial and error method until we get an output that looks like it meets most of the aspects to simulate a space realistic enough to make the users excited to go deeper into our simulation. To add further realistic touch to the simulation we plan to add texture to most celestial objects representing planets, some specific stars, the sun etc. However, for objects like a star system we plan to just make objects representing many stars and enable lighting on them. Next for shading we plan to set up lighting for each part of the simulations differently e.g. for our solar system the Sun will have ambient lighting so that the planets rotating around will always having light on the side facing the sun. This app will require the use of a mouse primarily to move the camera around from a fixed point and maybe the use of the keyboard to move the camera itself to a different position allowing the user to adjust their views according to their preference. Overall, we plan to make the app as smooth and attractive as possible while making sure we can let people learn something new about the universe.

Thinking about audience

Our primary audiences are between the age of 7 and above. It is so that they can actually understand what the app is about and what they are getting from it. Moreover, since our main target is to educate and interest people about the outer space and its beauties, we believe children are the best choice to start with when older people are usually busy with their lives. On the other hand, children are always amazed to find out what lies beyond our sight. We also believe that if we can fascinate children more towards space, they will grow more conscious and curious. "Space travel" will provide an interesting piece of work for people learning or interested in learning about 3D graphics, as we are planning to make the base of our app using 3D graphics and adding texture for realisticity. We are planning to involve the audience at every small step we take. We plan to do so by adding user interaction, so they feel like they have some control over what they get out of the app. We believe that fresh sight of potential users is significantly important and can help us to open the doors that we did not realize that even exist and so we plan to make the app available to everyone by hosting the files online. It can not only allow more potential users to access it but also more developers to improve it or add to it if they want.

Tools, equipment and knowledge

Our code template will be based on 3D workshops from this course and 3D templates from Principles and Applications of Programming course from this year. On top of that, we have already found some example codes of solar system online and we will attach them in the Supplementary Materials. We mostly want to check the logic behind the movements of planets and how to organise the code well.

At least one of us next term will take part in the Graphics course of Perception and multimedia module, which will provide us knowledge about OpenGL pipeline, Frag Shaders, Vertex Shaders and many more things which we might use in out project. To fill up the gaps in our knowledge we will use C++ Primer book, watch some YouTube tutorials and we will be consulting our ideas and problems with our classmates, lab assistants and lecturers.

We already know how to use Visual Studio and Open Frameworks in general. Specifically, we know how to use the camera in 3D environment, how to make classes of objects and how to rotate, translate, add textures and simple lighting. We just want to use all of it at more advanced level. We will make a project on GitLab, so it will be very easy for us to change the code and have a backup version if one of us will mess up something. We also decided on using Trello board, which will help us to set the deadlines ourselves and later keep track of them, so the work will go smoothly.

Iteration and planning

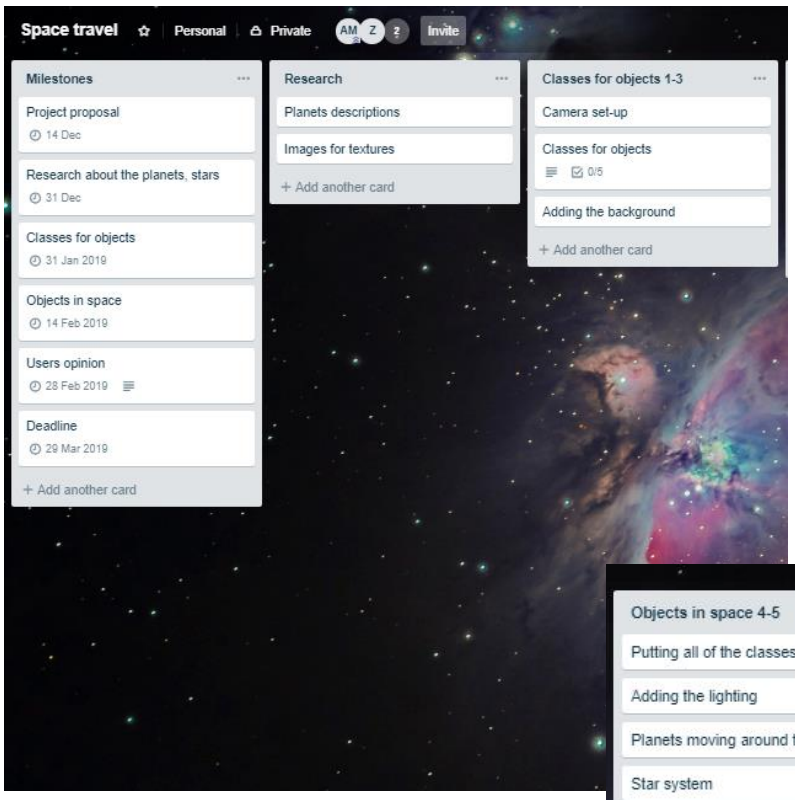
We plan to spend 10 -15 hours per week for the projects.

Project in pieces (week 1 – week 3)

- At first we plan to set accurate speed, size, distance from the sun for the planets and implement rotation in the Y axis for each individual object and for the planets add an additional rotation in the Y axis centering the Sun. Find the texture images and set them, figure out how to add the ring to some of the planets and add them. Make the simple class for the planets.
- The texture, size and the light for the sun. The texture for the sun needs to be bright enough to feel realistic since the sun will be the centre of lighting for the solar system. We plan to do that using Openframeworks ambient lighting technique so that it lights up in 360•.
- Make the general camera set up.
- Set up the background with an image of the outer space and that has enough elements to juice up the look and the feel of the simulation.
- We need some actual physics and maths to use. Scale the size of planets, and unfortunately as one of our potential users pointed to us the distance between planets will not be possible to be scale as in a real world.

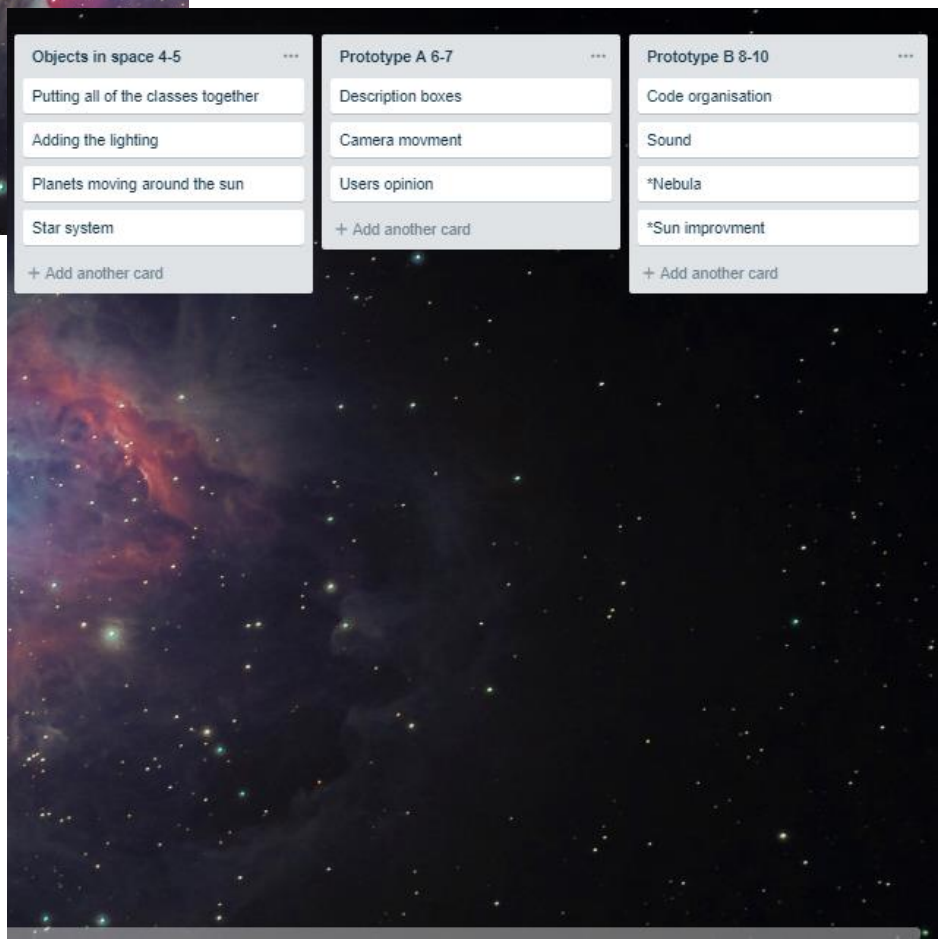
Naive in pieces (week 4 – week 5)

- Make the particle system for the stars system with size and light set up inside. Each star will have different size and colour representing old, young and dying stars. This will help us put more information to the content for display to our users.



- Ask potential users about work we have done in week 1 - 3. Our main users are children starting from the age of 7 but we potentially plan to influence people of all ages to be fascinated about space and how space travel can look like, so we plan to ask people of different ages beginning from our target audience to get feedback on out idea and progress

- Draw all the objects together, figure out how to make them work together and add the movement of planets around the sun. Additionally, setting up individual axis and distances from each other and implement movement.



Prototype A (week 6 – week 7)

- Adding the description boxes, when the mouse pressed/hovered with information of each object representing a celestial body.
- Implementation of camera movement. The camera will have user interactions, so users will be able to move the camera around and in and out of the solar system to get better or preferred perspectives.
- Users opinion.

Prototype B (week 8 – week 10)

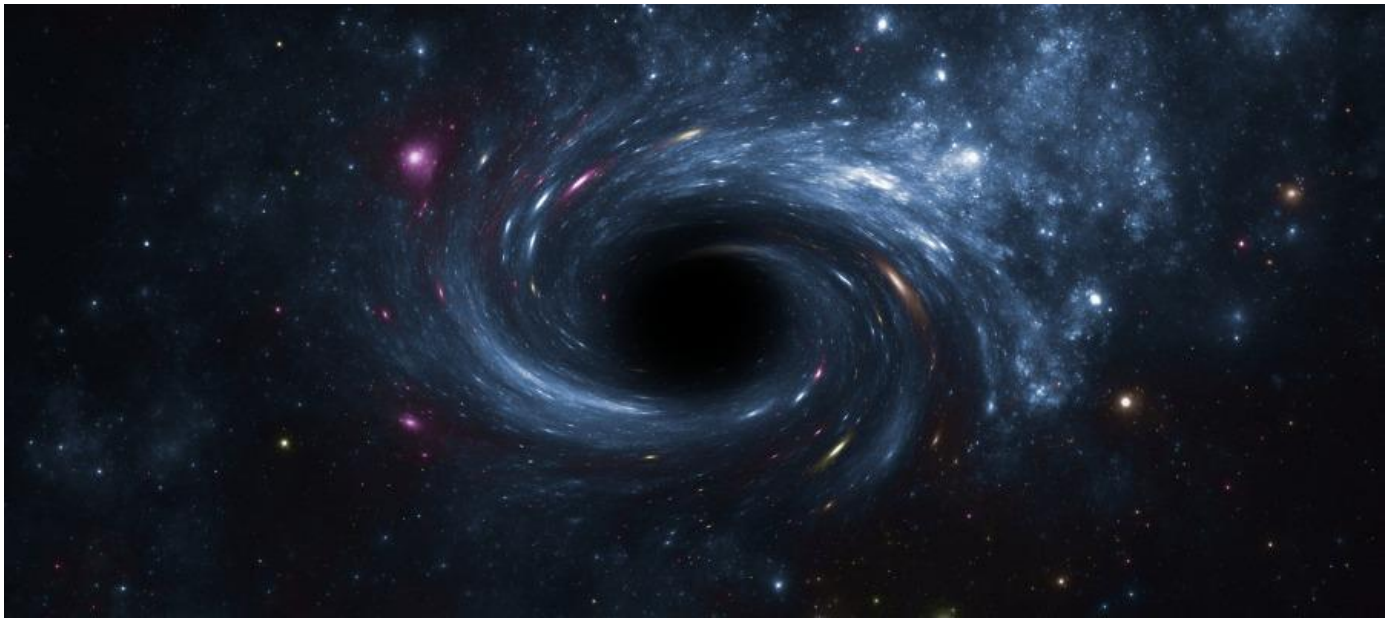
- Improving the visibility and readability of the code: adding comments.
- Adding sound. Even though there is no sound in space since there is no air, we are still planning to add some sound effects for the user interactions to make the user more occupied and excited.

Potential pitfalls and contingencies

We did our research about the penguins and because we did not find enough interesting information to make our final project, we decided to switch to completely different area. It means that we will need to make some extra research before we will start to work with code. We are planning to spend the Christmas break on that.

One of our biggest fear with our code is that our star system, which will produce so many stars might make the simulation really slow and working live on the project might be difficult. The logic of rotation might give us some troubles as well. We are also concerned about the openFrameworks documentation, which is not well documented.

Organising and planning are quite easy but sticking to the actual plan will be very challenging for us. Even though, on our long way to the last semicolon in code, we will need to adjust our plan many times. It is almost predictable that one of us will get ill at some point or other unexpected personal problem will appear. While working with code, bugs are unavoidable, and sometimes fixing them might take hours, if not days.



Supplementary Materials

- YouTube OpenGL tutorials by TheChernoProject – Yan Chernikov
- Santiagolizardo code on GitHub

  It has been implemented in C++ 11 and it uses the game engine, but this code helps us a lot to plan our logic.
- It's really similar to what we want to code
- Solar system - https://www.youtube.com/watch?v=z8aBZZnv6y8
- Star System similar to - https://www.youtube.com/watch?v=Xj5Vv0L-nLQ
- Black-hole - https://www.youtube.com/watch?v=bdHLO6-s8XE
- Solar system and lighting similar to-https://www.youtube.com/watch?v=z8aB-ZZnv6y8&t=47s

Structure of code

Classes:

1. Planet: We are not sure yet if we will make just one Planet class for all of the planets, or we will make separate class for each planet. We need to involve the maths and physics first to find out.

- ofVec3f position(x,y,z)
- ofImage texture
- float size
- float speedRotatation
- float aroundSunRotation
- void rotate (speedRotation)
- void aroundSun
- void move
- void draw (position, texture, size)

2. Sun

- ofVec3f position(x,y,z)
- ofImage texture
- ofLight light (ambient)
- void rotate
- void draw
- *add the fire clouds

3. Stars – particular system (vectors)

- Optimization idea: appearance of stars depending where the camera is right now, so it will not make the program that slow
- Position, size, own lights
- Colours

4. Black-hole - object class for the black-hole and the surroundings

- ofTexture, bind(), unbind() - texture for the event horizon
- void rotate
- void Draw

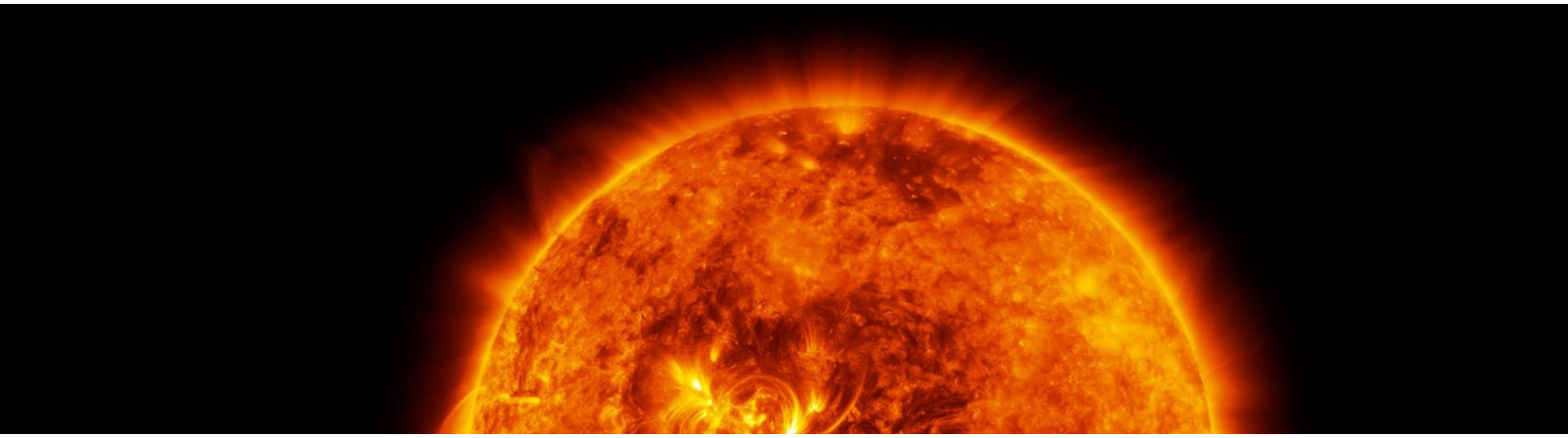
5. Asteroids

- ofVec3f position (x,y,z)
- Simple physics for asteroid speed
- ofTexture texture
- If mouse pressed asteroid shows

***6. Nebula** – is an interstellar cloud of dust, hydrogen, helium and other ionized gases

- We are planning to use shader, so we need to wait for the Graphics course next term, and we do not know how much we will cover in that course yet
- Use of Mesh for the basic structure
- Changing angles of the mesh triangles to create a tunnel

(*) Extra (if we have enough time)



References

1st image: Anon, (2018). [image] Available at: https://www.sciencefocus.com/space/do-the-planets-ever-align-with-one-another/ [Accessed 13 Dec. 2018].

2nd image: Anon, (2018). [image] Available at: https://3iom3142cnb81rlnt6w4mtlr-wpengine.netdna-ssl.com/wp-content/uploads/2016/11/fig13.jpg [Accessed 13 Dec. 2018].

3rd image: Anon, (2018). [image] Available at: https://www.worldatlas.com/articles/important-facts-about-the-asteroid-belt.html [Accessed 14 Dec. 2018].

4th image: Anon, (2018). [image] Available at: http://news.mit.edu/2012/deflecting-an-asteroid-with-paintballs-1026 [Accessed 14 Dec. 2018].

5th and 6th images: Trello board which we created for this project

7th image: (2018). [image] Available at: https://www.google.com/search?q=black-hole&source=lnms&tbm=isch&sa=X&ved=0ahUKEwidh5X5jJ_fAhWOC-wKHY2YCI4Q_AUIDigB&biw=1364&bih=648#imgrc=KXCLoJSpnQmqcM:

8th image: (2018). [image] Available at: https://www.quantamagazine.org/what-is-the-sun-made-of-and-when-will-it-die-20180705/