Programming Project Report

Outline (of design)

For this project we were provided with a large data file filled with information about artificial space objects: Launch Date, Mass, Diameter etc. We chose to implement the idea of a browser program where the user can view information about each individual space object or groups of space objects. Our initial idea was to take a simpler approach to the user interface and make it easy for the user to navigate and explore the program and to also put into use the features we had implemented. In following this approach, we landed on the idea of having a 'Launch Screen' and a 'Main Screen', the 'Launch Screen' allowed the user to click a button to take them to the 'Main Screen' similar to clicking a link to browse on a website, the 'Main Screen' then listed out the space objects in a comprehensible and readable manner, this opened the floor to adding a 'scrolling feature' which facilitated breathless and smooth exploration of all the data available to the user. The textbox and various query buttons on this screen then allowed the user to view certain parts of the data. We also listed various sorting methods on this screen which opened a magnitude of ways to view the data available in a way that suited your particular needs at a given time. (See Image 1 at end of pdf)

On this 'Main Screen' the user could then change the appearance of the screen which displayed the unique characteristics of the space object which had just been selected or typed in the user textbox. On this screen the user could find data collected about the space object, see the data displayed in an easily comprehensible manner, such as a graph which could be seamlessly adjusted to display a variance of data over different time periods. (See Image 1 at end of pdf)

Team Management(Organisation)

When we began our project, we decided we needed to assign everyone a part of the project to get working on. Since our project needed three "screens" we split everyone up into groups by assigning people to the "Main Screen", "Info Screen" and 'Launch Screen'. The outline was as follows:

- Main Screen – Everyone
- Launch Screen – Henry
- Data Management – Faith and Meng
- Main Screen Displayed Data – Henry, Manuel and Meng
- Main Screen Graph - Manuel and Meng
- Foundation Classes - Faith

As the project moved along, we each took up other minor tasks as needed.

Individual Contributions

Faith:

On week one we met to set up a clear vision for the project. We examined other browsers for space objects and came up with a mock drawing of the layout. We decided to have one homepage where all space objects would be displayed with their essential information .You would be brought to this page by clicking a launch button on the 'Launch Screen'. Our design idea was to create a clear, visual, and accessible program to display and interact with the space object data. I created a mock-up of the original design idea. (See Image 2 at end of pdf for early prototype)

I began by loading in the data and storing it in a table object and this provided a starting point for the project. I then started working on classes that would be the foundation of the project. Once the layout was solid, I moved onto working with that data class and data points created when I loaded the initial data. I worked through null pointer exceptions, different lengths of data and space objects completely lacking data to create generic classes such as the screen, text widget, check box, radio box, slider, button and widget class. I also implemented drop down menu functionality along with search widget functionally which wasn't used in the end so was commented out. At this stage I wanted to do more comparisons on the data. To do this I wanted to have the user select a date range they were interested in and then the page would update to display the statistics for that period. I realised this was going to be very difficult to do as there was a string value in each date. To solve this I created a time class that converted the date in seconds making it easier to work with. I then went on to implement a sort by class that sorted the data based on mass, diameter and in alphabetical order. I chose the comparator interface over the comparable interface in order to allow the data to be sorted in more than one way.This worked great and looked great working with the graph straight away. I also implemented space sound to be playing which would make the program feel a lot more space related but unfortunately the program was starting to feel a bit laggy so I looked at some optimisations. I cut out the sound and tried to cut back on 'for' loops in various places'. Finally, I focused on finer details such as backgrounds, colouring, fonts and applying the coding standard to various sections..

Henry:

For this project, my first main task was creating the loading screen. To do this, I first created a "launch" button from the base Widget class and also a system to store the active screen within a reference to a Screen object. After making the base functionality work, I had to adorn the screen to make it more appealing to the user (UX). I recalled seeing a sort of "starfield" effect before, where ambient stars would slowly spin around the centre of the screen.
To implement this, I created a Star class, which would randomly initialise an x/y position for the star, and also store its rotation relative to the centre. The move() method would simply increment its rotation by a small amount modulus 2π. The display() method would draw the star onto the canvas. I used the pushMatrix(), translate() and popMatrix() functions to calculate the rotations relative to the centre, rather than the top left corner, as is default, and then restore the prior coordinate system, so as to not interfere with the rest of the program.
My second main task was to create the query system for the user. Our idea was that we would have the user input what they would like to search for via a textbox, and then we would filter the query results and display them to the user. I started by writing the logic for the actual queries. It was implemented as a function queryData() in main, which would take the ArrayList of space objects and the search field, and return a corresponding ArrayList of filtered space objects. It made much use of the .contains method in-built for Strings to check for valid results.

Next, I worked on the textbox class, which allows the user to type a query into it and press ENTER to run the search. It made use of the inbuilt key and keyCode variables to identify which key the user pressed and then append it to the current query String. After that, it was only a matter of calling the queryData() function, using the user text as a parameter.

Finally, I added a column of side buttons to finish off the query system. This would allow the user to select which field they would like to query by.

Other than the loading screen and query system, I also helped my teammates generally with some issues and queries through the project. I found the team experience overall to be very enjoyable and rewarding, as it meant I could focus primarily on doing the parts I was best at, but still having input and involvement in the whole process.

Manuel:

In the first week of the project, my goal was to print the loaded data to the screen in a table format. Initially, the function printed all of the one thousand data points to the screen, with no way to navigate between them. At the end of the first week, I made the function more efficient, by only printing to the screen the amount of data points that could be seen and by implementing a scroll down method.

The second task I took care of, together with Meng, was the implementation of a bar chart that allowed us to visualise different variables of the data points. We did this by establishing an initial x position and y position, and for each data point, we incremented x by a fixed position and subtracted a value to y that would depend on the variable's value. Again, to increase efficiency, we only printed the necessary number of bars to fill the screen.

Later on, I implemented a method to slide through the bar chart using the right and left arrow keys and a little visual treat of the bars increasing in size. I also prevented the event of having oversized bars on the screen.

Finally, I took it upon myself to resize the middle of the screen, putting the table and the bar chart on the correct places, and adding a slider to the table, using a class written by Faith and that I implemented using two boolean variables that limit the slider's movement and the mouse dragged method to move the slider.

Meng:

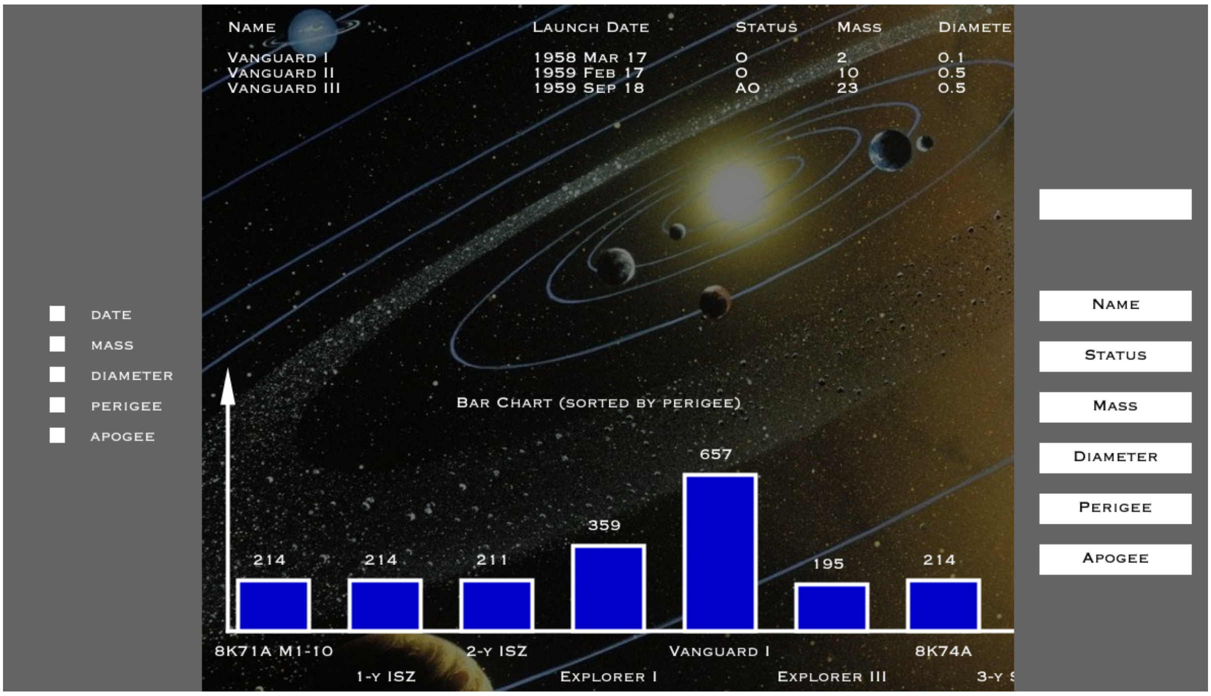
In the first week, my task was to create a class to store all the data. In order to do this, I created the "DataClass" class, added the necessary variables and the function to print the data. To check the class works properly, I used the Arraylist and "loaddata" method created by Faith, add all the data in the arraylist to check the data can be printed out properly.

Then I started the task of drawing a bar chart, which is the biggest task I worked on in this project. By collaborating with Manuel, we managed to draw out the bar chart according to the arraylist of results passed in, and display it on the screen at the end of the second week. Then, together with Henry, we checked that the bar chart can be correctly drawn out based on the search result.

Later I added some select boxes on the left of the screen by creating some widgets, so the users can choose what they want to sort the bar chart by. When any of the select boxes are selected, the bar chart will change with the corresponding choice. I also notice that the bar chart gives an error when it is drawn based on "diameter", because the diameter is a double while all other categories are integers. So I fixed that error by adding a condition when drawing the bar chart.

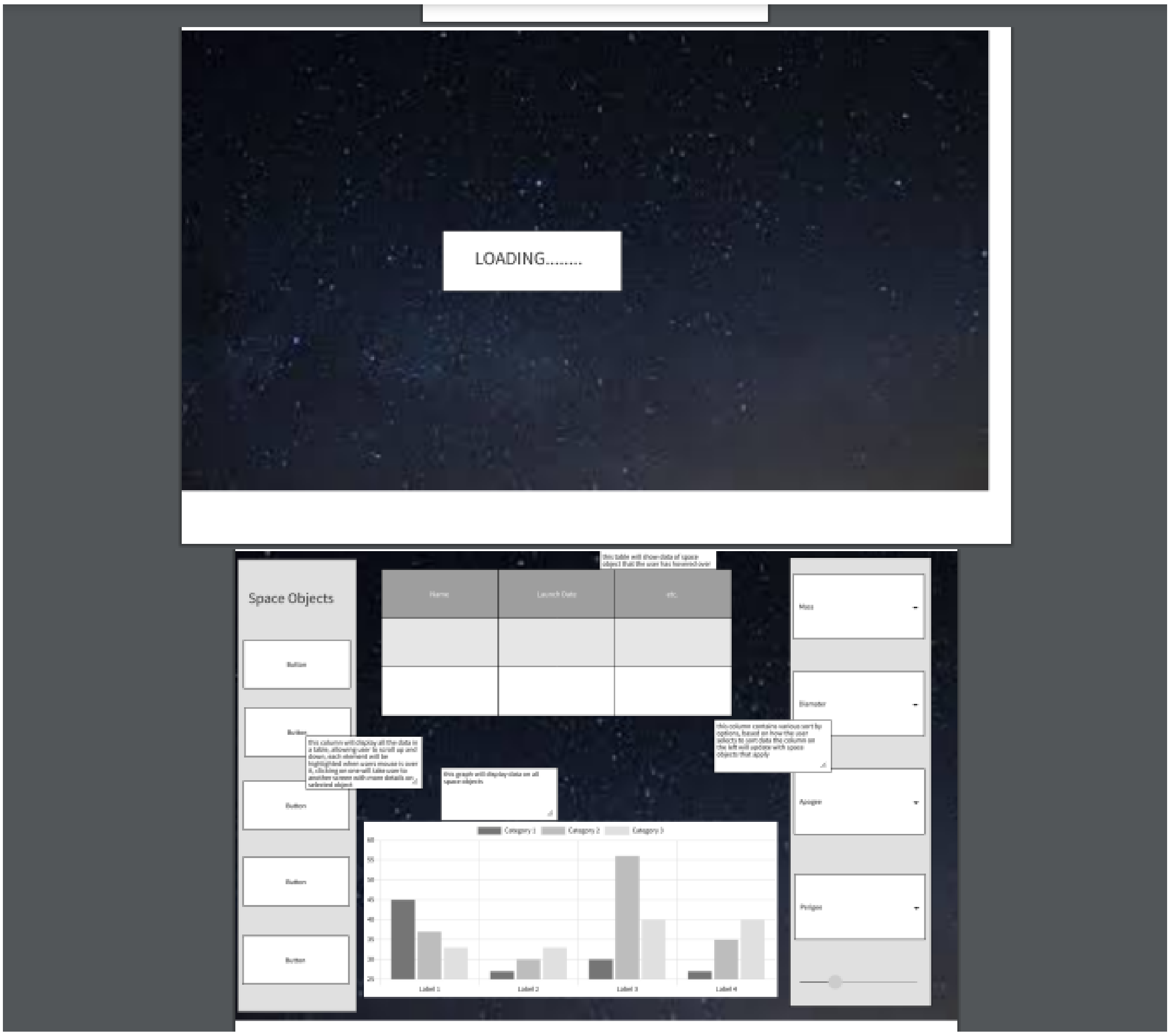
In the final week, I fixed a small error of the bar chart so the bars would not go beyond the x-axis. I also slightly adjusted the position of the select boxes to make it more clear and fit with the overall format.

In general, I find the overall program challenging and interesting, it allows me to practice many coding skills. It is also an enjoyable experience to work collaboratively with my teammates.



Name	Launch Date	Status	Mass	Diamete
Vanguard I	1958 Mar 17	O	2	0.1
Vanguard II	1959 Feb 17	O	10	0.5
Vanguard III	1959 Sep 18	AO	23	0.5

(Image 1)



this table will show data of space object that the user has hovered over

this column will display all the data in a table, allowing user to scroll up and down, each element will be highlighted when users mouse is over it, clicking on one will take user to another screen with more details on selected object

this graph will display data on all space objects

this column contains various sort by options, based on how the user selects to sort data the column on the left will update with space objects that apply

(Image 2)