Updated Design Concept

# Game Design

The team have been tasked with designing and creating a resource management game with the aim of monitoring and investigating how territorial ownership affects human behaviour.

We have proposed to make the game in a 2d space setting, with players starting off on a “home world” with an administrator defined number of colonist ships of which to gather resources. The players will use the ships to travel and colonise planets in the generated universe, each planet will produce a different amounts of different resources.

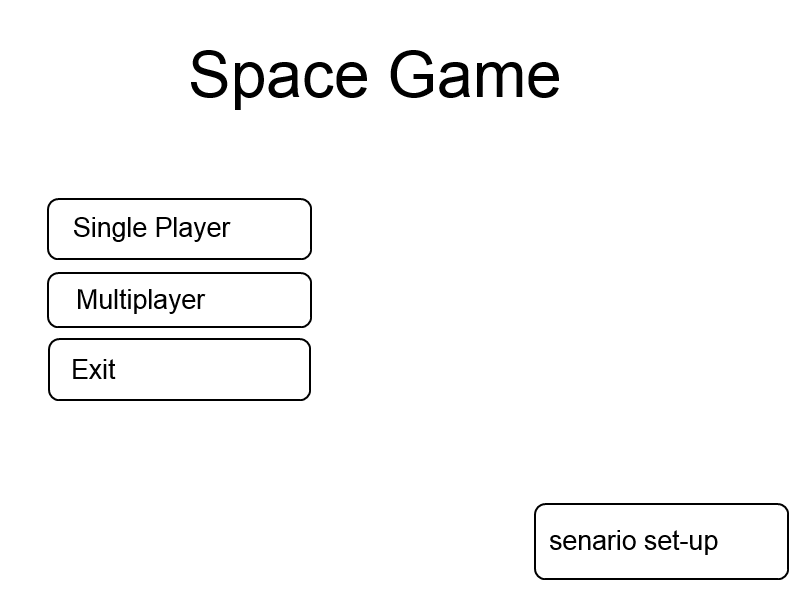


Figure 1: Game Main Menu

Players will start in a corner of the map and gradually move out from their home planet colonising planets as their ships fan out occupying planets as they go. Only when a ship is on a planet will it collect the resources there eventually moving on to the next when the resources are used up or if they choose to move for tactical reasons. The team proposed a variation of units which will have different characteristics such as collector ships, colony ships and soldier ships.

The three main differences between the ships will be defined as their ability to collection resources and also how effective they are in battle with any other ships. The collector ships will collect resources much faster but will also find battles tough, dealing little damage and being easy to kill. The colony ships will be the ship that has equal attack and collection power but does not excel in either. The soldier ship will have a high attack and defence value and will be difficult to destroy, however it will collect resources a lot slower than the others. There are two different ways that these ships could be assigned to a user depending on the administrator’s preference, they could either be given set units that they have to use as decided by the administrator or they have the freedom to choose their own.

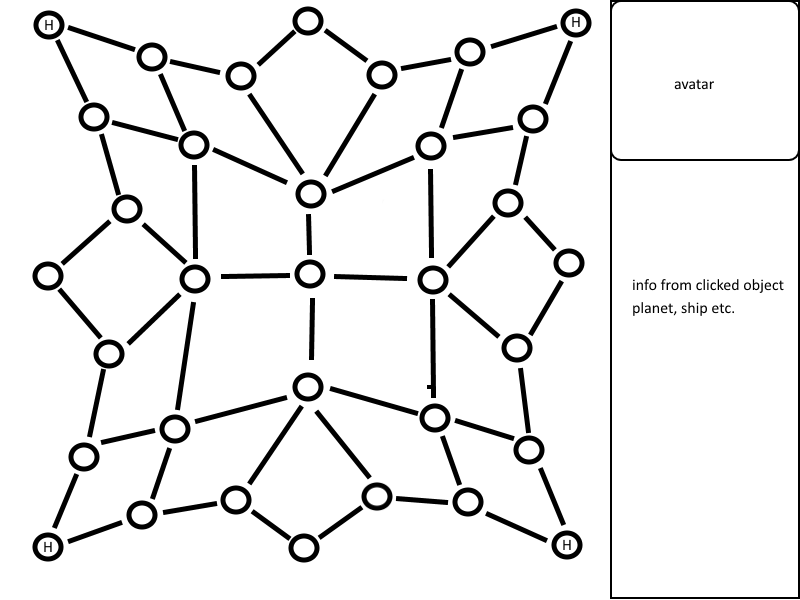


Figure 2: Example of Game Map

Above is a mock-up of the in-game map. Each node is a separate planet and the H is a possible home planet. The maps could be sized up or sized down depending on players or preference.

The proposal is to use 3 different resources like minerals, fuel and anti-matter. The rarity and amount of said resource would be up to the experiment runner to set within the customizable maps, however each map would have a standard set up with anti-matter being rarer than the others.

Each resource that is available to the user will have a different function which will give them an incentive to go to collect them and also encourage user interaction. The first of these resources, fuel, allows the user to progress to further planets to gather more resources, and also provides them with a single point towards their total. If the user runs out of fuel then the ship that they are operating will be destroyed, and fuel will become lower each time they move planets so collecting fuel is important. Minerals and Anti-matter both provide the user with points that will go towards winning the game when the time limit becomes zero. Minerals are common and will provide the user with a small point bonus on five points towards their total but Anti-matter is worth 20 points for the user. After a period of time each user will be made aware that anti-matter can be found in a certain area. This will encourage user interaction as each user will actively move towards this planet for the bonus points the anti-matter provides.

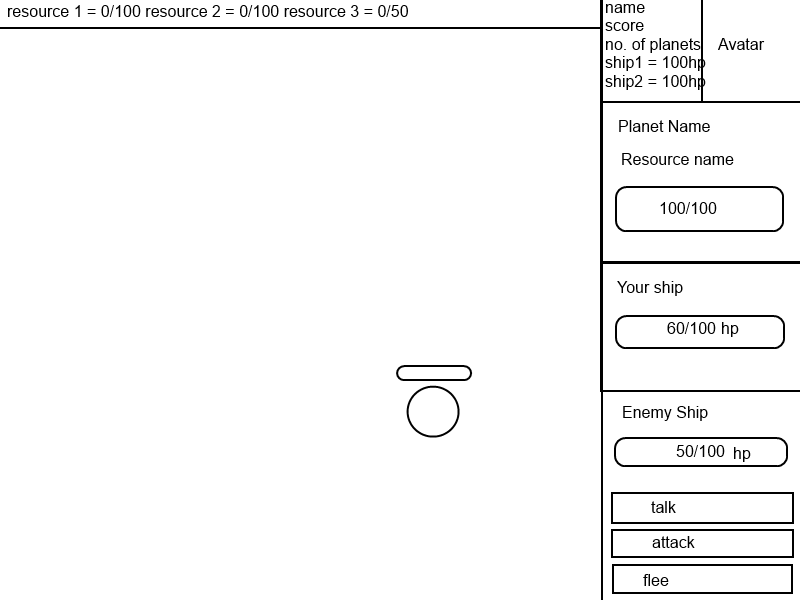


Figure 3: Player Game Window

If 2 player’s ships meet on a planet then a dialog between them will be produced e.g. Talk, attack or run. Depending on players actions a number of different scenarios can occur for example:

If both players talk then they can exchange set dialog options allowing them to express their intent.

If both players attack then they will each take damage.

If both players flee then they will each leave the planet.

If one player talks and the other attacks the player who talked will take damage.

If one player attacks and the other flees then the attack will miss and the fleeing player will leave the planet with the other player occupying it.

If one player talks and the other flees then the talking player will remain to occupy the planet.

Discussions of a trading option could also be implemented so that players can trade with each other, trading unneeded resources for ones they do need.

For a player to win, they must either destroy all other players’ ships or have the highest points total when the games time limit comes to an end. Resources will become more and more contested as players will have spread out and going for the rarer or more abundant planets, meaning later in the game player interaction will be more common as the players have spread out making it harder to gather the resources you want.

As put across in the brief and meeting, there are 2 different versions of multiplayer intended. On one hand is a one on one type gameplay where it is player vs player or player vs AI and on the other is a version with multiple players and AI against each other, each with a separate home world and ships or possibly a version where only players have a home world and the AI are roaming ships who move around the universe randomly and interact with other ships randomly.

As a team we have decided on some customisation in not only each players look but in aspects of the game that effect the players behaviour, e.g. Changing the amount of resources a planet produces or the starting variables of the players like number of ships they start with. As such this would be made easier by having a set map for each scenario, meaning you could change everything in the maps rather easy as this would be done either in game when setting up a match, or could be done using a text file outside of the program that is associated with the given map. The same process could be used for editing a lot of other things as well such as the ships health or resource collection speed using a text file to be loaded when the game is started. If we go with a single ship version then the customisation could also be used to personalise the ship as well if multiple then perhaps only the colour.

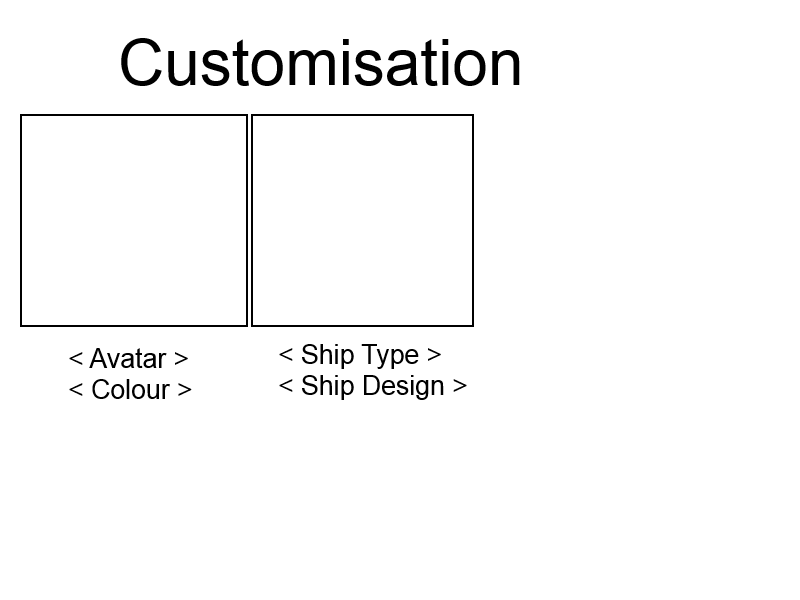


Figure 4: Customisation Window

The specification has a number of needs and functions that the program has to do and the team feel that the design is beginning to shape these needs. The need for territory and resource management is fulfilled by the planets and ships, as players will only be able to harvest from a set number of places meaning they will have to manage where they put their ships depending on what resources they need. The interaction between players, consequences and territorial invasion is fulfilled as that players can still interact with each other if they both have a ship on the same planet with different ships providing different benefits. The customisation options will be met by allowing players to customise their avatar before they start e.g. Look, gender or clothes but also allowing the person running and monitoring the experiment will be able to change the environments variables using a text file.

# Asset Consideration

We decided to go down the open source route for our assets for the game, we have come up with several different assets that we will use within our game, the model for the spaceship that we have will remain consistent but change colour when selected by the player or administrator, and there will also be several different characters that can be selected to create variety and also a wider range of results for the user.



Figure 5: Potential Assets





Figure 6::Above are the asset designs we have chosen with colour schema added

The assets that we have chosen are all space themed as the game is in a space setting. The game will encompass 4 different types of coloured units and 4 corresponding ship colours. The above assets that have been designed with custom colours are the refined assets that will appear in-game.

The character avatars were created by:

Justin Nichol <http://opengameart.org/users/justin-nichol>

The ship avatars were created by:

Skorpio

<http://opengameart.org/content/space-ship-construction-kit>

<http://opengameart.org/content/spaceship-tutorial-0>

kanadaj

<http://opengameart.org/content/titan-battlecruiser> which uses Skorpio’s ship construction kit

# Roles and tasks among group members

To ensure that the creation of our game runs as smoothly as possible we have dictated roles and specific tasks to each member. This section of the document will discuss the roles of each group member and the various tasks that they should be expected to complete.

Jack Hoyle – Group leader/Programmer/Designer  
Jack’s role within the team will be that of the lead Group leader/Programmer/Designer. He will have a leader type role within the group helping people keep on track and make final decisions based on input and debate within the group.

Jack will also assist with the programming of the game itself, ensuring that any features that need to be implemented will be within the scheduled time plan. Jack completed his role in the group as well as aid with team organisation and developing the designs for the UI.

Daniel Masterson – Lead Programmer   
Daniels role within the team will be that of a programmer, he will be assisting with the writing of the game and also ensuring that features are implemented whilst following the time plan. Daniel will also have the job of creating the base engine of the game that we will be expanding on to ensure that we have a good base to work from. Daniels roles where all achieved as the game was completed.

Adam Kadow – Programmer/Artist  
Adam’s role within the team will be that of a programmer and artist. Like the previous two members of the group he will be writing the game itself and ensuring that all of the features are implemented on time and within the constraints of the time plan.

Furthermore, Adam will be in charge of the various different art resources that will be available within the game, he will be in charge of finding relevant open source assets that can be used and also creating his own when needed. Adams roles where all achieved as he implemented the art assets and aided in the character select screen.

Josh Gibson/Philip Devine – Producers/Project Managers/Testing  
Josh and Philips’ role within the team will be to handle to producing side of the project. It is essential for the project to be run smoothly that it is managed and it will be their job to ensure that the project is on track and also that each member of the group is made fully aware about things such as meeting time and locations, it will also be the job of these two members of the group to check the current progress of the task against the created time plan to assign tasks accordingly. When checking against the plan it is also important to update the plan to reflect any changes that may have occurred during that week. To continue to ensure that the project will stay on track.

It will also be up to these members of the team to produce and quality control any documentation that will be attached to the project. This will include the maintenance and completion of the design document, as well as any user documentation that will be produced. They will also be in charge of ensuring that all meetings are documented, including timesheets and minutes documents.

When the project is completed it will also need to be tested, as these two members of the group will not have had as much of as hands-on input when it comes to programming the game itself they will be the best candidates when it comes to testing the game as they complete tasks as a user rather than a developer. This will help to provide a more in-depth testing document and help minimise errors when the project is completed. Jack and Philip both completed their roles in creating and updating documentation as well as ensuring time-keeping was well managed and that all timesheets where up-to-date, they also tested the game in its final stages of development.

# Project Evaluation

The project overall has been smooth with all the hand-ins and the game has had steady development throughout the project, as well as the documentation. The group as a whole worked alongside the time plan as well as having good communication throughout the project. A major reason that the project succeeded was due to the engine framework developed by Daniel as this set ground for us to branch of and develop different parts simultaneously.

In evaluation to the project it has gone smoothly and to be expected with the goals and timesheet used. Everyone has been active within the group when it comes to communication and meetings as well as everyone pulling their weight and completing assigned tasks when required.

In hindsight some improvements we could have made would have been to spend more time working on the project to create a networked version of the game, created different versions of ships and assigned programming tasks to more people and been more specific with assigned tasks, as that is the main time consumption within the project.

# Appendix:

## Market Research

Looking at market scope for territorial and resource management there are a few games that would fit this genre however they do it a lot of different ways, as such the territorial and resource management genre is extremely large with many different niches across numerous games.

One of the biggest examples that is relatively similar to our brief is the Civilisation series. We will be concentrating on comparing with not the most recent version but CIV 5, arguably the most well received iteration. Players start with a settler and found a city wherever on the map they like, from there the players work to build up their cities by claiming resources within their expanding territory which gradually grows as their city does. There is a number of ways to win a game but it usually comes down to who has the most resources to build better defences, a better army etc. CIV 5 also has extensive player to player/AI interactions as you can trade, make treaties and spy on each other. As such you would say that if you stripped down CIV 5 to its bare bones it would be very similar to our plan, however as it is a AAA game it can be expected to be more expansive and refined than ours will be but at the lowest level they could be seen as similar, CIV 5 has just been expanded profusely. If we look outside the resource/territory management in terms of movement and map ideas, you could compare the design to a game called FTL: Faster Than Light, in this game you are navigating through a galaxy one planet at a time while running from and ever approaching enemy. As you go you battle ships and complete random encounters all with random loot, it is more of a rogue-like strategy game then a management game. However, the movement system through the galaxies is very similar to the kind of thing I would like to use in mine, moving from planet to planet one jump at a time.

However, at a lower level of game there are very few games that are similar to ours when not looking at the research objectives, this means there is a gap in the market for a lower level version of this kind of game, as such if the game was put to the market it could either do relatively well as an accepted lower level version or could be harshly compared to the higher level versions like CIV 5, because of this the game would need to be extremely refined and bug free.

Comparing to the social experiment provided to the group for research purposes the game consisted of 20 periods, each lasting 1 minute each. The game proposed by the team has been estimated at anywhere from 15 minutes upwards, which is entirely dependent on the number of players per session. A typical player of CIV5 played for around 8-13 hours per session, as our style of game is a mixture between a RTS and collection and resource management this would make the game appealing to our target audience.

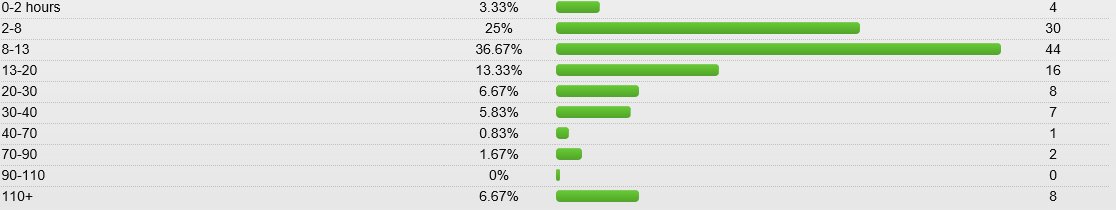


Figure 7: http://www.gamefaqs.com/boards/938528-sid-meiers-civilization-v/66782899

As the game has ongoing data collection per game this allows it to be used for a variety of research projects specifically patterns in human territorial behaviour which would be useful towards the psychology department as some modules encompass the aspect of human territorial behaviour, allowing them to run this game as an experiment and receive active data and feedback.

## Solar Dominion – User Guide

The following document is a guide for you! The user of Solar Dominion, it will discuss how to use the game and show you how to use all of the features going forward. We will start by discussing how to start a new game.

### Starting the Game

Before you begin playing Solar Dominion and ruling the galaxy you will first have to get the game in a position to begin. You have been provided with all of the resources and files to make Solar Dominion work, so all that you need to do from here is place it within a directory of your choice and run the SolarDominion.exe file, which will bring you to the start screen, and gets you ready to begin the game option of your choice.



*Figure 1 – Start Screen*

As can be seen in the screenshot above when beginning the game you will be greeted with this screen, which gives you the option to take part in various different game modes that will be available to you. The first of which is a 1v1 match that will see you take part in combat against 1 AI opponent. The 4 player match will see you competing against 3 AI opponents and the 8 player match will see you compete against 7 different opponents, depending on your preference at the time. The final option ‘Eject’ will see you leave the game and the window will close. (But why would you wanna do that?!)

### Customisation

After you have chosen to begin your game you will be greeted with the ability to customise your ships pilot from a wide variety of different models available, and you will then also be able to assign a colour to each of those models, making them your own.



*Figure 2 – Customisation Window*

When you are greeted with the customisation screen if you left click on the model it will change to a new character. Once you have decided on which character you want, pressing the change colour button will then allow you to customise the colour of your characters armour and ship, customising your ship and pilot further.

### Gameplay

When you begin the game you will be welcomed by the gameplay screen, which will show all of the planets that you can capture and use to complete your ultimate goal. Win the game.

When looking at the gameplay screen there a various different pieces of information that you will need to keep into account to ensure that your quest for domination is going to be as successful as possible. The first thing to keep a note of at the top of the screen is your current resources, showing you how many of each resource that you currently have in your stockpile



*Figure 3 – Stockpiled Resources*

To be a successful captain you also need to take care of each of your ships and ensure that you are not over-farming planets. This can be done by checking your status window which shows all of the details of the current ship.

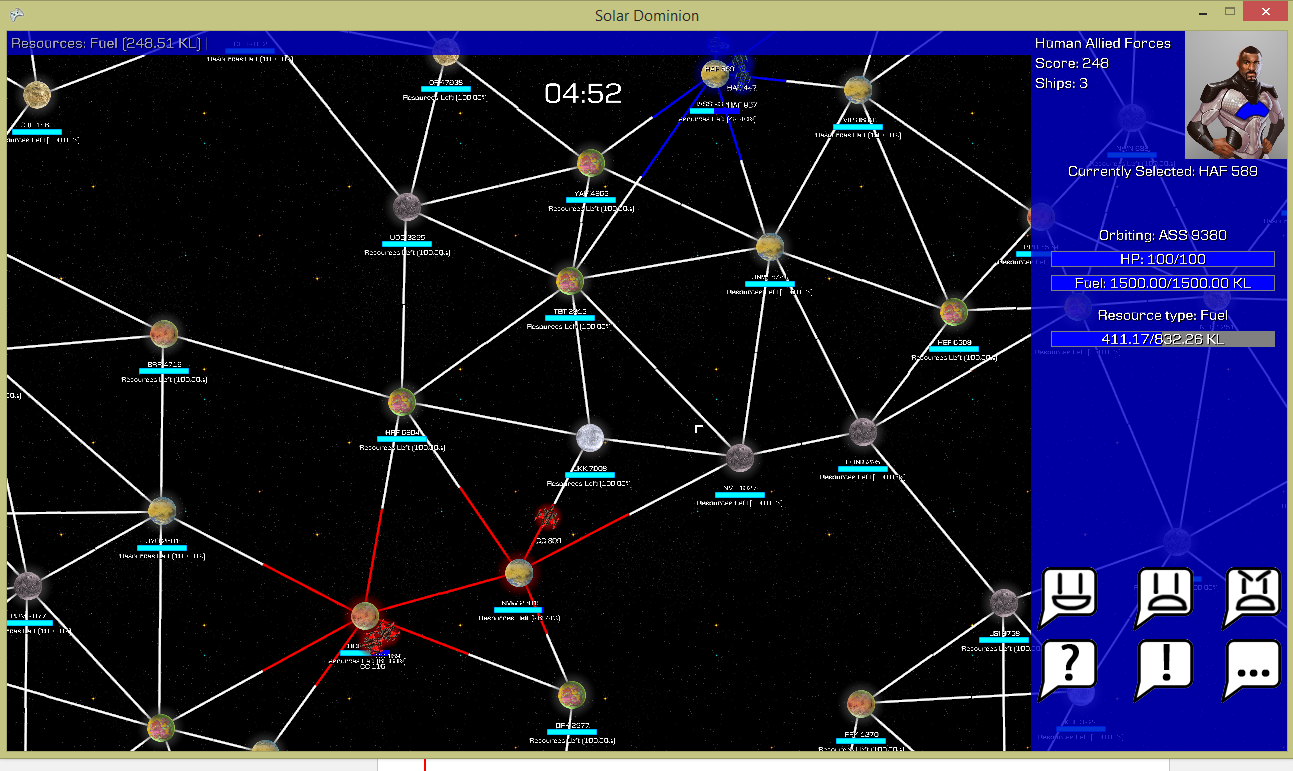


*Figure 4 - Ship Information Window*

When viewing the ship information window there are several different pieces of information that it can provide for you to help you on your journey. The first of these is the HP of the current ship that you are operating, you need to try and keep this as high as possible, and if it is falling low ensure that that ship stays away from any battles as it could be destroyed.

The second piece of information provided for you is the amount of fuel that your ship currently has remaining, this is also very important as when your fuel runs out you will not be able to advance to any further planets, so if you see that this is running low ensure to get some fuel stockpiled first!

The final bar available to you is information about the current planet that you are on – it tells you about the planet and the resources that it currently produces, over time this will also begin to lower and you will need to keep an eye on it to ensure that you aren’t trying to farm on a planet that has no resources left



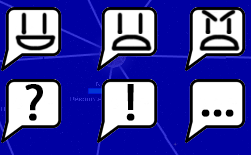
*Figure 5 – Game Window*

When you begin the game you (and any AI players) all start on your own planet, with your objective being to capture as many planets as you can before the time runs out. To take control of one of your ships all you need to do is left click on it, once you have clicked on the ship you then need to click on another planet to move your ship towards it, capturing that planet will then allow your ship to begin collecting resources that will help you reach your aim of winning the game. When collecting resources on a planet the amount of resources assigned to that single planet will begin to lower until it reaches zero. When a planets resources hit zero it will produce less resources, making it important that you move on and let the planet regenerate its resources.

There are three different kinds of resources that you will encounter on your travels throughout the game world; Fuel, minerals and anti-matter. The first of these, fuel is very important to you as it is what allows you to move from planet to planet, but also keeps your ships from being destroyed. If you keep your fuel level high it will allow you to take over more planets and help ensure that you have the best chance possible to win the game. Furthermore for each fuel resource that you gather you will gain one point towards your total which will help to dictate if you win the game. Other ways to gather these points are by collecting the other resources that are available to you on other planets. The first of these resources, minerals, are very common and can be found on most of the planets that you encounter throughout the universe, and each normal mineral will add one point to your total. The final resource that you will encounter within the game world will be Anti-matter, this will is a very rare mineral that will provide you with a huge advantage to your score, it can only be found on certain planets and once it is gone its gone although you are provided with a huge 20 point bonus to your score for each anti-matter mineral that you find, helping you gain a large advantage towards winning the game.

As you float around the world adding planets to your collection you will eventually begin to run into planets that are owned by other colonies throughout the universe, if you land on one of these planets you will begin to colonise it and make it one of your own, eventually turning it into your side. You may however find that the other colonies send one of their ships to meet you there, initiating an interaction between the two of you within which you will have to make your own decisions, there are two buttons that will appear in your status screen, the first of these two options will be the ability to capture the enemies planet while you are circling it, allowing you to take the enemies planet whilst you are on it. The other option that will be available to you while you are on there will be to attack the other ship. Attacking the ship will see your HP(and the enemies HP) fall as you do damage to each other. Be wary though however because if your HP hits zero then your ship will be destroyed, and if all of your ships are destroyed it will be game over. The final option that is available would be to retreat from the planet back to one of your own to prevent them from you receiving any more damage to yourself.

Another way that you can interact with other ships within the environment is by using your “emotion buttons” these will display an image above your ship showing your current mood at the time, the AI ships will do similar to show how they are reacting to your actions. So beware, if the AI has an angry face towards you, you could be about to be attacked.



*Figure 6 – Emotion Buttons*

There are two main ways available to you to win the game and control the galaxy, the first of these is to eliminate all of your opponent’s ships, if they have no ships left and no other ways to collect resources you then rule the galaxy. The other way to win is by having the highest amount of points at the end of the designated time limit. This would make you the strongest colony in the galaxy and everyone else would have to bow to your greatness, making you the winner.



*Figure 7 – GAME OVER screen*

Although you can win these ways you also have to be wary of the ways that you could lose the game, the first of these is by losing all of your own ships, either by your opponents destroying them or them being destroyed by your ship running out of fuel, the other way that you could find yourself losing this battle is by not gaining enough points from collecting your resources, this means that you need to ensure to keep collecting resources as much as you possibly can to give yourself the biggest advantage you possibly can.

## Solar Dominion – Administrator Guide

This section of the user guide is for any potential administrators that may be using this game for their research purposes. The first section of the administrate use that this will cover will be the data logging what will be important for your research.

### Data Logging

Within the ‘DATAOUTPUT’ folder that is available in the root directory folder that will hold all of the data that will be recorded from both the users and the AI, the decisions that they made and the actions that they decided to partake in. It will also note what model was being used by the user and the AI at that time, which will help to tailor the results and provide a wider range of information, which can all be easily accessed.

### Rulseset Alteration

There will also be, alongside the ability to view the logged data, another file in the root directory called ruleset, this will allow you, the administrator to alter the settings that will be used within the game, such as how many ships available to each user, or how much each resource will be worth. This can also help when it comes to tailoring the results.

## Consideration issues

### Legal

With consideration to the legal issues of the game there could be a potential case that we had used an FTL style node map but in hindsight a large portion of these style of games also use the node map. The games assets are all open source with some editing done which requires no legal action.

### Ethical

In consideration of the customers’ requirements against our game specification the game meets all ethical requirement s as the game does not contain any form of violence or gore as well as this the game doesn’t include any discrimination towards race or religion. An ethical consideration would be to let the player know that their data is being recorded this could be in the form of a small on screen message on the main menu.

### Social

The game is built on having player interaction from a research perspective and due to the nature of the specification given this it is therefore not seen as an issue.

### Professional

With consideration to the legal framework all assets from the game are open-source and the game engine was built in XNA due to the scope of the project the project wouldn’t require any form of patent or copyright law. The project would require a visual studio license and not an educational licence if the game where to be published.

## Time Plans

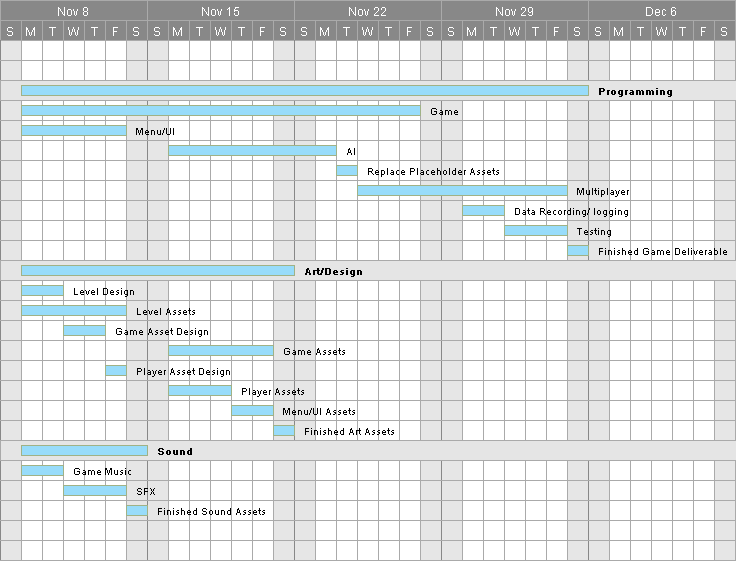


Figure 1: Initial Time Plan Developed

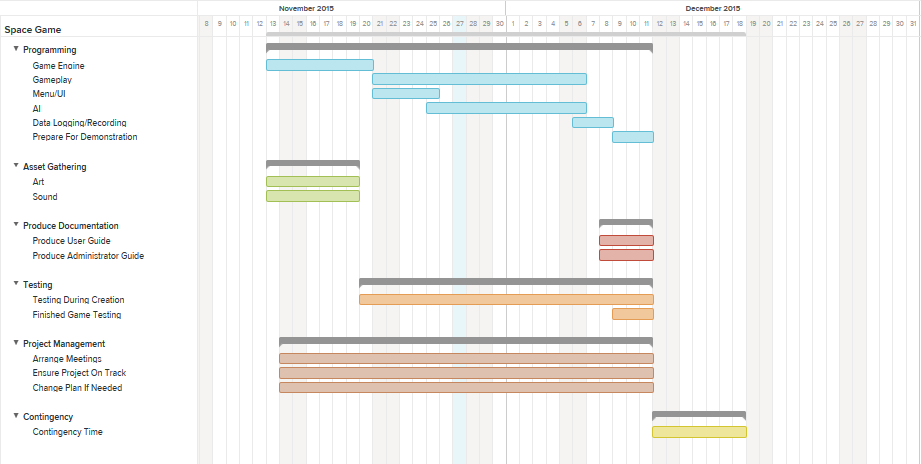


Figure 2: Final time plan

## Timesheets

### WEEK 12: 19/11/2015 -

|  |  |  |
| --- | --- | --- |
| Name | Task | Time |
| Jack | Helped in task management and revision of engine architecture, as well as helped gather assets. | 6 hours |
| Dan | Developed sample game engine during TTG , gathered assets for game development | 6 hours |
| Adam | Revised the game engine and helped gather art assets for the games development. | 6 hours |
| Josh | Aiding in report writing specifically in the area of team job allocation. | 6 hours |
| Philip | Aided in writing an initial design concept based of off the game concept given by Jack. | 6 hours |

### WEEK 13: 23/11/2015

|  |  |  |
| --- | --- | --- |
| Name | Task | Time |
| Jack | Storyboarded screen concepts such as main menu and play screen as well as acquiring sprite assets. | 5 hours |
| Dan | Updated and added some functionality to the game engine. Added ship movement to the game. | 5 hours |
| Adam | Generated and finalised the UML diagram describing all the classes. | 5 hours |
| Josh | Expanded and updated the design documentation with the refined game concepts and description of the units. Updated the time plan. | 5 hours |
| Philip | Expanded and updated the design documentation to include the business perspective and target audience. | 5 hours |

### WEEK 14: 04/12/2015

|  |  |  |
| --- | --- | --- |
| Name | Task | Time |
| Jack | Refined game assets to fit the game designs space theme | 5 hours |
| Dan | Updated the game engine to utilise the new art assets. | 5 hours |
| Adam | Refined the UML of the games overall architecture | 5 hours |
| Josh | Expanded and updated the design documentation with the refined business plan | 5 hours |
| Philip | Expanded and updated the design documentation to include ethical , legal , social and professional issues | 5 hours |

### WEEK 15: 10/12/2015

|  |  |  |
| --- | --- | --- |
| Name | Task | Time |
| Jack | Worked on implementing data logging for the player. As well as created a draft for the user guide. Also implemented a scenario rule set. | 6 hours |
| Dan | Worked on general player interaction, gameplay and bug fixes. | 6 hours |
| Adam | Put the assets into the game to allow the player to choose their ship and their character. | 6 hours |
| Josh | Finished off various parts of the documentation including in writing the user manual. | 6 hours |
| Philip | Aided in helping finish off the various parts of documentation including the project evaluation. | 6 hours |

Overall Development Time: 22 Hours

Overall Development Cost: £ 12.50 Per hour Per person

= £ 62.50 Per hour

= £1375 Overall

= £275 Overall Per person

## Individual Contributions

### Philip Devine

My contribution to the project as one of the two producers was to oversee the time management and create documentation of the project as well as keeping track of timesheets. My base contribution on week one started as giving a description of the game design and client requirements, in the coming week I then went on to explain about the refined game concept that the team had chosen I also began tracking the timesheets from this week onwards, during this week I analysed the games target audience through researching games of a similar genre.

On the fourth week I explained our aesthetic choices as well as develop a marketing plan with the other producer developed a document to explain the legal, ethical, social and professional issues. At the final week of project development I sorted out the design document and updated anything that may have been left out as well as create an evaluation of the project. Over the course of the project I ensured that the timesheets and minutes had all been kept up-to-date.

### Josh Gibson

During the creation of this project my contribution has been largely documentation related. I have created and updated some of the various different documents that need to be delivered to ensure the project runs as smoothly as possible and that all of the data needed for each member of the group is readily available in one place

I started by creating a document that helped to describe each team members role within the project. It describes their individual role and also what they need to complete to ensure that the project is completed to the highest standard possible, I also helped with the implementation of the initial design document, adding some new ideas and also tuning some of the existing ones, ensuring we were making the best game we possibly could that matched the clients specification.

During the second week I then refined the design document that we already had, adding some of the other ideas that we had come up with over the meetings during the week, I also created a new time-plan for us to follow that was more tailored to the progress we had already made and also implemented any new ideas that we may have had.

I then went on to create a marketing plan for the project, which looks at the best possible options for marketing the game if we were looking to place it in a commercial environment. Looking at best places to market the product and also what audiences to aim the game towards.

Finally I looked at creating a user manual for any potential users of the game both for possible gamers and also anyone within the psychology field who would have admin control to alter the game how they wanted

### Jack Hoyle

I have acted as the group leader and have had a global presence within the group helping in all aspects of the project where needed, making final decisions and organising the group in terms of meetings and assigning tasks.

I have helped with programming and documentation. In the programming sense I have been working with both Dan and Adam aiding them where needed as well fixing bugs and optimisation as such as helping to implement the logging feature which uses a data manager class that outputted formatted data to a file each time the game is started and scenario setup/rule set with includes reading from a file which can be edited to change variables within the game. In documentation I have helped both Josh and Philip with various documents providing a person to proof read and propose additions and changes as well as creating drafts for certain documents eg. User manual. My main roll has been to keep people up to date and working with the timeframe on the tasks required at that point as well as organising group meetings and the such but overall I have aided everyone within the team when required.

### Adam Kadow

When the project began, I was tasked with finding / creating art assets as well as programming alongside Dan Masterson and Jack Hoyle. I decided that creating our own art, while it would ensure a consistent art style, would be too time consuming, so instead I looked for online art sources that were free to use. These included planet sprites, player avatars and ship textures, which were then edited later in the project.

I also created and updated the project’s class diagram, making sure the correct authors were listed and that the attributes and methods listed were up to date. I also ensured that the layout was correct and that the inheritance was clear and easy to see at a glance.

As for programming, I worked on top of the framework created by Dan Masterson. This included editing and building on some of his components and managers in order to increase their functionality, such as being able to alter an avatar’s colour. My main programming task however was to include player customisation. This involved created multiple classes and managing textures and player data within a clear and easy to use interface. This information is then passed on to the main game class which allows for the player’s colour, avatar and ship to be aesthetically altered. I designed and implemented the interface and created the classes and logic for handling the user’s choices and then passing them onto the main game class so their choices can be seen.

### Daniel Masterson

Throughout the development of the project, I have primarily focussed on engine development while also working on some of the more complex gameplay systems that would involve engine work. Using this engine, the other programmers then easily implemented the game and various menus to match the given specification.

I started the engine by creating a prototype based off of my original concept before the first team meeting. Coincidentally, the concept we finally settled on used a node capturing system that – while different to the prototype – was similar enough that we could easily repurpose existing engine code, while swapping out a majority of the gameplay code. In doing so, we had a rough prototype of the final concept within a week.

The engine uses a pseudo component architecture, with entities being made out of simpler components like images, buttons and text labels. Components are actually a subclass of entity, meaning that components can contain other components, to make a component chain (This can be seen in the button entity, which creates its own label entity for drawing the button text). The entities themselves modify the components as appropriate, while the components themselves handles things like drawing, handling mouse events and positioning (in the case of labels). While this did make it easy to add otherwise complex code, the need for resolution independence (particularly with UI elements) meant that there was a significant amount of boilerplate code when creating the components. If I were to make this engine again, I would have a separate UI subsystem based around HTML or another design markup language to help reduce the amount of boilerplate and to help make designing the UI significantly simpler.

As time went on, I moved onto handling and helping with the more complex gameplay programming tasks. This including rewriting the scene system - A method of containing large numbers of entities separate from each other - and improving key parts of the AI that proved challenging to solve. I also assisted the other programmers in their tasks at times, and helped modify assets so that they were in a game ready state.

Nearing the end of the project, my work shifted towards correcting bugs and completing any half-implemented features, as well as continuing to help the other programmers with their remaining work.

