**Github**

Task 1 – P1.1

***Identify client requirements by listing the features required in the above scenario***

* Having the spaceship being controlled by the user at the bottom of the screen. The spaceship will go left or right, according to user input and never leaves the screen. It will also have a 4 frame animation that will never be interrupted irrelevant to game conditions
* The score will reset with every different level, however when the player loses or wins, the game will show the total score
* Sprite imaged need to be created for all elements especially for the 4 frame animations
* Appropriate sounds need to be recorded, edited and implemented
* Having a clean and suitable GUI for the game so it will be user friendly for players and target audience
* Making sure that aliens are generated accordingly, increasing in number from level to another, and having and end of game boss
* Creating power boosters for the spaceship such as a health and speed power up
* The game will be over once the player passes all the levels or else when the health of the spaceship is 0

Task 2 – P1.2

***Identify the target group your game would interest. Explain you’re reasoning in a short paragraph***

The target group for my game would interest various audience both young as old. This is so because Space Invaders was one of the very first games when arcade games were invented thus it will interest folk of that era. It will also interest the young audience due to the modern style I’ll be giving to the game.

Task 3 – P1.3

***Clarify your creative intentions by writing a short paragraph describing the overall look and feel of the game and how this game would cater for the target group you mentioned in Task 2***

The overall look of the game will have a modern and futuristic feel to it, since its Space Invaders. However, I’ll also add a little bit of comic relief to the aliens so as it won’t be too dark or gloomy for the players. Using appropriate fonts, colours and designs, I’ll cater for the mentioned target audience.

Task 4 – P2.1

***List the areas of expertise required to implement this game***

* Character and Prop Design
* Client Understanding
* Colour Management
* Organisation
* Research
* Sound Design
* Time Management
* Unity

Task 5 – P2.2

***Rate your own expertise in each of these areas of expertise. Write a paragraph justifying your rating in the light of your experience and expertise***

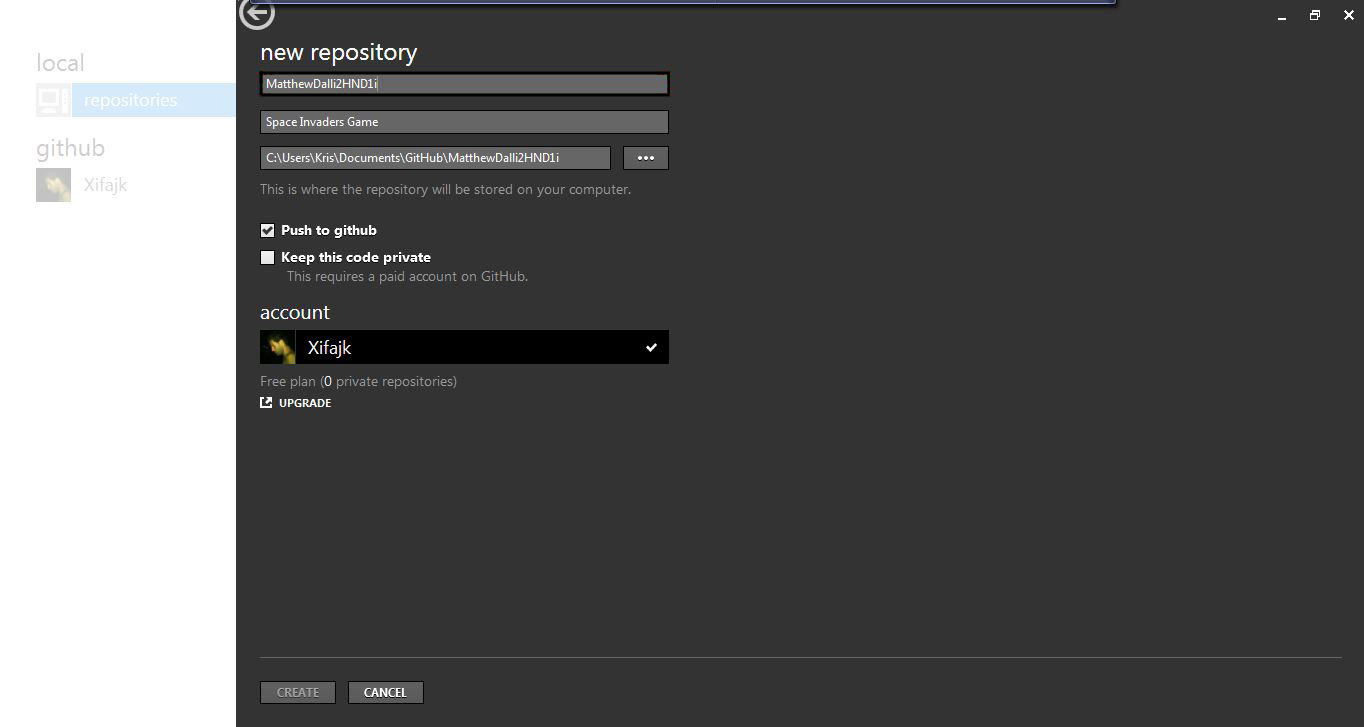
|  |  |
| --- | --- |
| Area of Expertise | Rating (Out of 10) |
|  |  |
| Character & Prop Design | 7 |
| Client Understanding | 9 |
| Colour Management | 6 |
| Organisation | 10 |
| Research | 10 |
| Sound Design | 8 |
| Time Management | 10 |
| Unity | 8 |

When it comes to organising, time management, research and client understanding I’m highly skilled as I’ve been a leader in various projects as well as always making sure that the end result will be as the client needs. With proper communication with the client and making various timetables, I always managed to keep up with appropriate deadlines and having everything neat so as the user will find and use things easily.

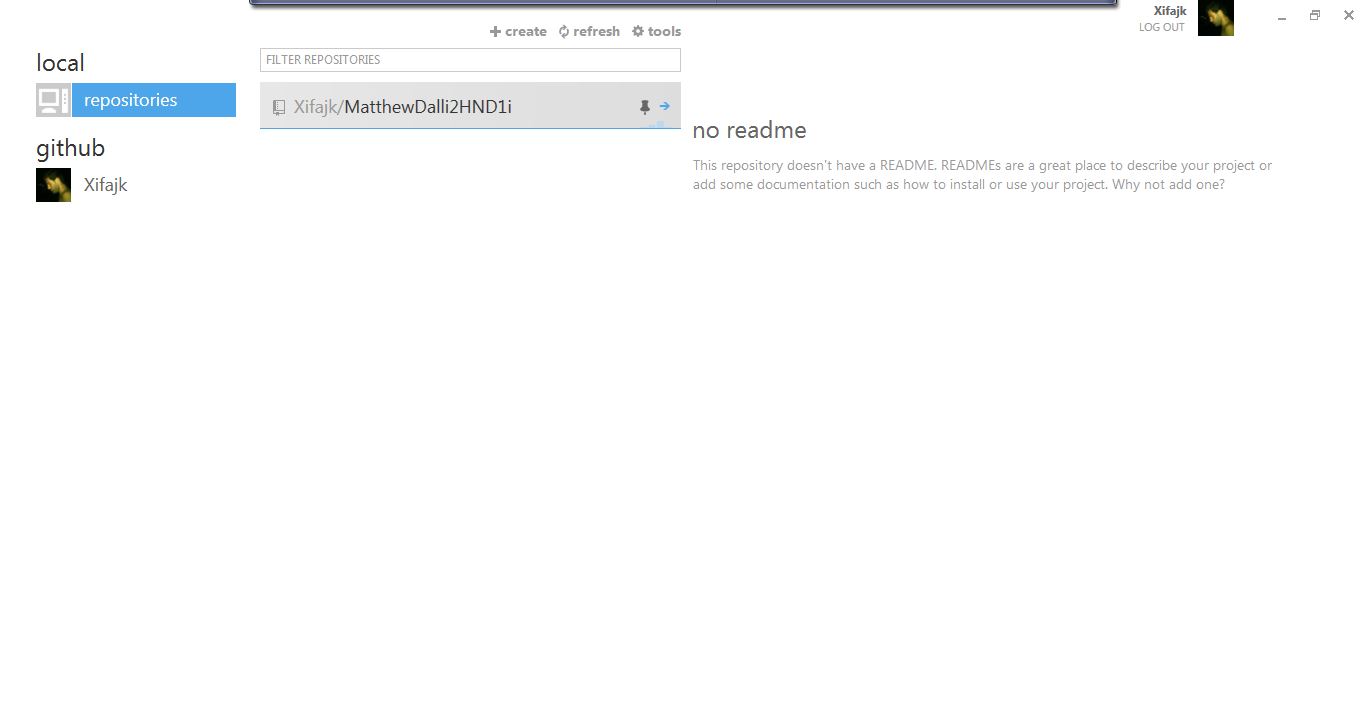
When it comes to Sound and programming in Unity, I still need a lot of practice and understanding in these areas which I’m willing to spend more extra time to improve my skills, especially in Character & Prop Design as well as Colour Management.

Task 6 – P3.1

***Produce preliminary concepts for an initial prototype by creating a new project on*** [***http://www.github.com***](http://www.github.com)***, and writing a full description of your intentions for the project. Include at least two screenshots of the project creation process and include a link to the Github project you have created***

The intentions for this project are to produce a user-friendly and fun Space Invaders game for the client’s needs. Commits will be created periodically as well as with details so the client will keep track of the project on a daily basis as well as seeing the project’s progress.

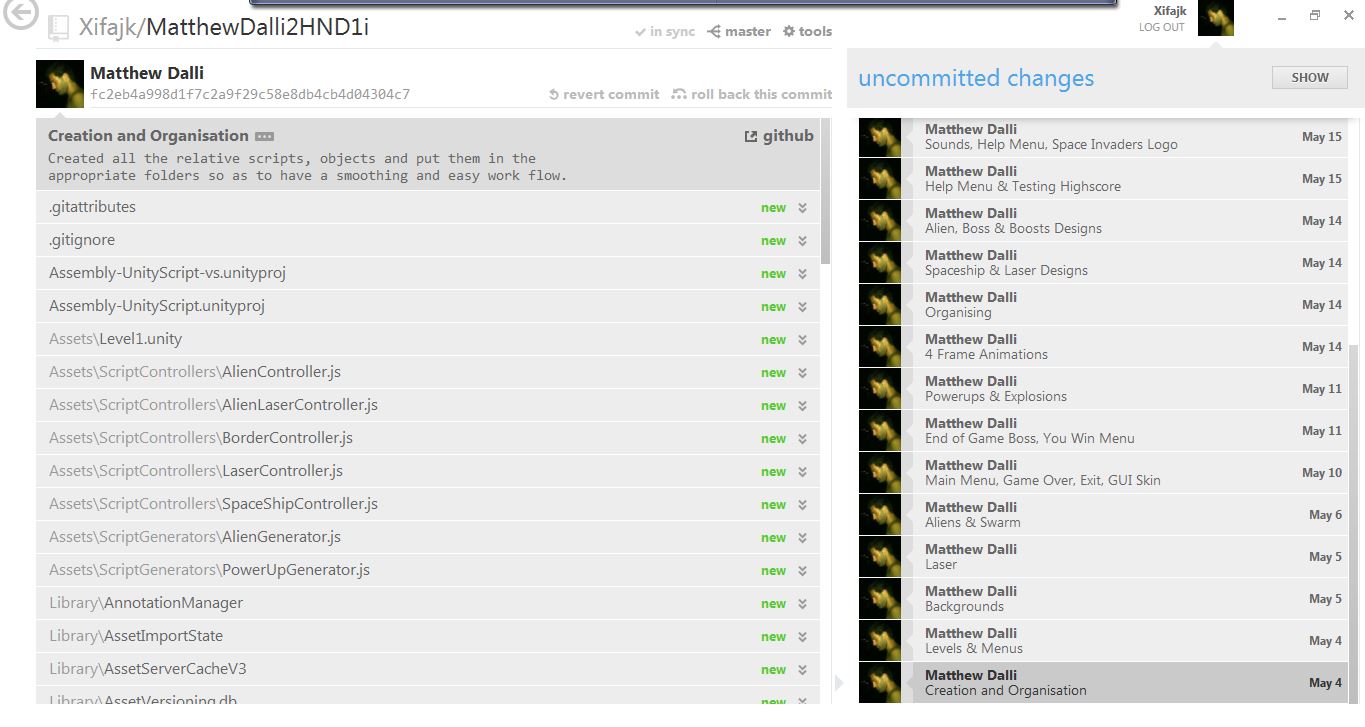
Link to the Github project: <https://github.com/Xifajk/MatthewDalli2HND1i>

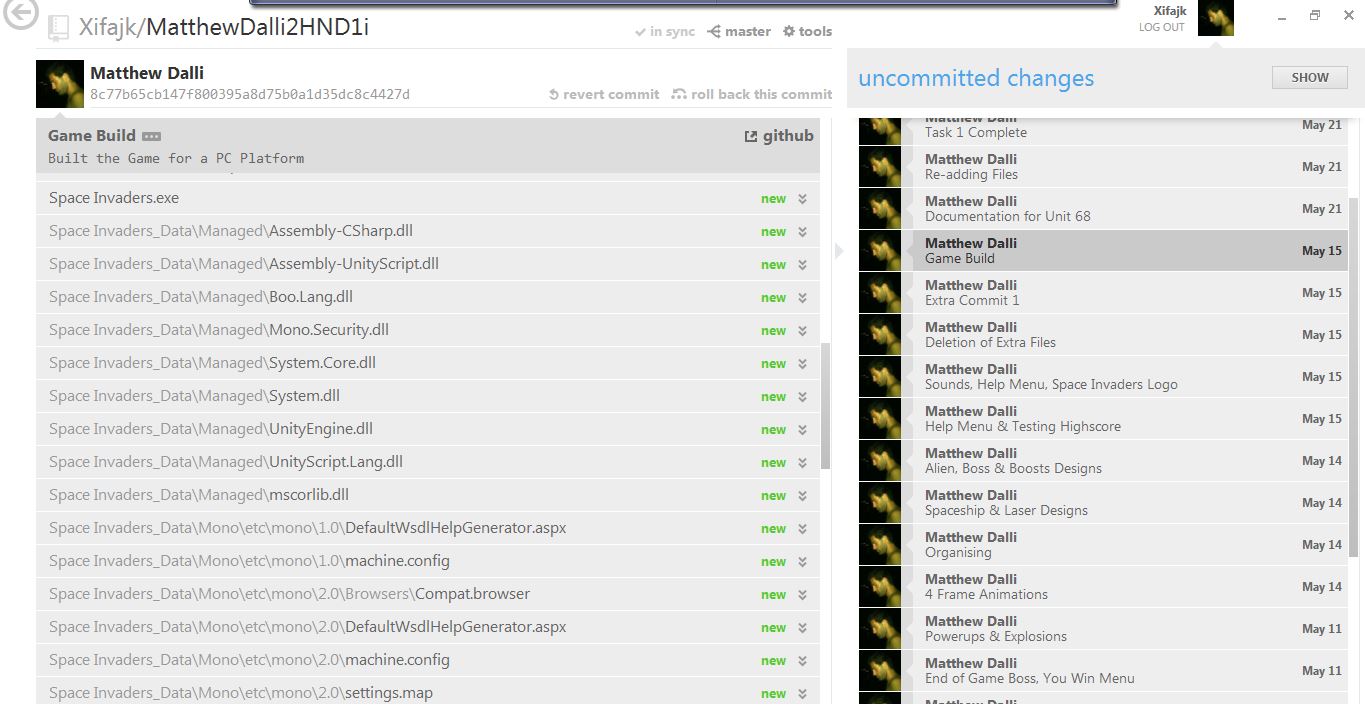


Task 7 – P3.2

***Evaluate and confirm the prototype in relation to constrains by posting the code of the Asteroids/Space Invaders game project you created in Assignment 1 of CIDP to your Github project as a first commit. Describe the first commit in full and post a screenshot of the commit description***

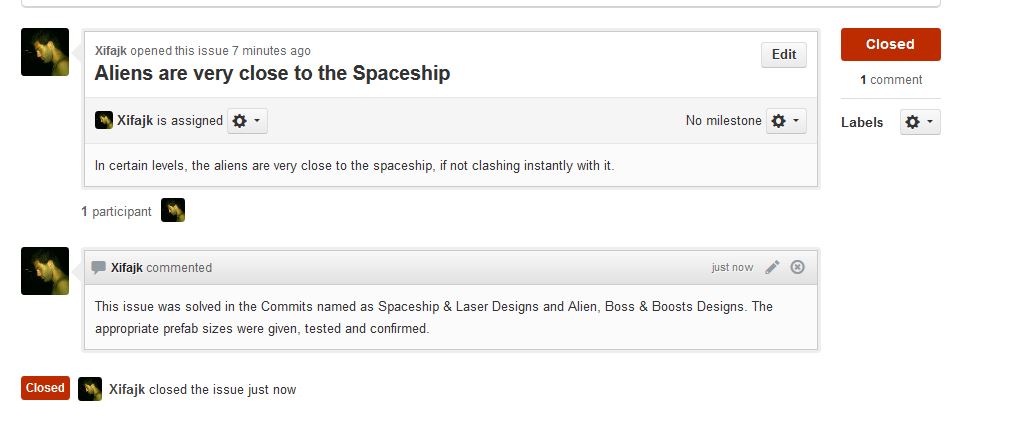
As my very first commit, I had created all the relevant JavaScripts, Materials, Prefabs & Folders in an organised way so to have a quick and easy workflow. This was highly important to have before starting the project, as well as giving the appropriate names, so I could place or add items in the correct folders accordingly.

The full code and the build of the game can be found in the 17th commit labelled *Game Build*

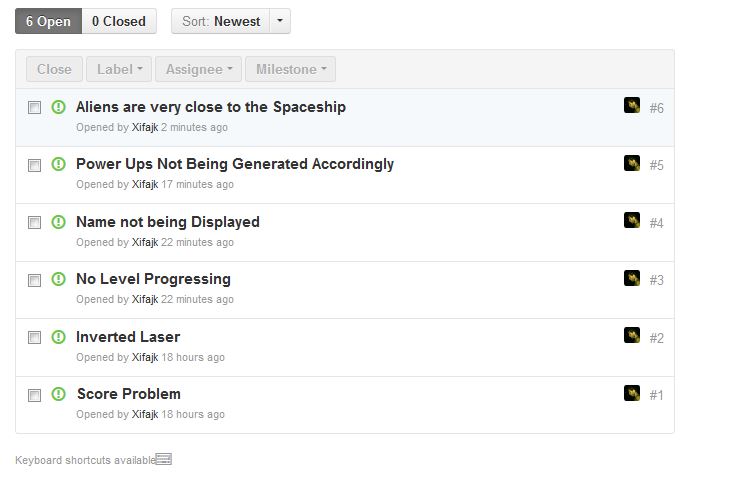
Task 8 - P3.3

***Reflect and record on feedback from prototype phases by explaining how the Github issue tracker works, with screenshots. Post five issues (bugs) concerning your game to Github and take screenshots, and then reply to each issue describing what remedial actions were taken to close the issue, before posting another commit referring to this closed issue***

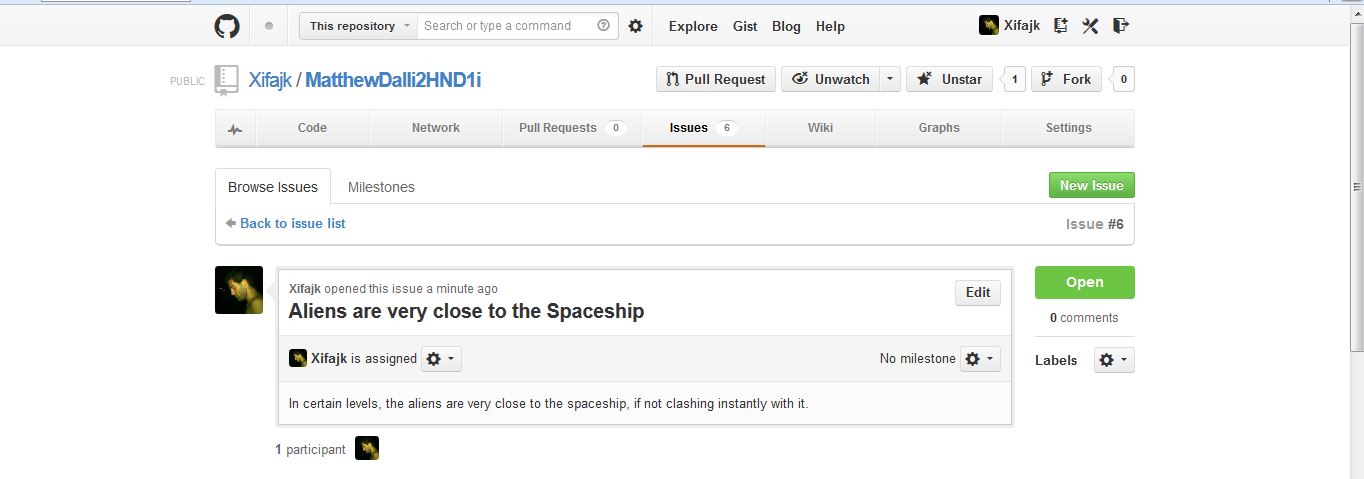
The Github issue tracker is a method where testers or the clients can post various issues about the project and the developer sees to them as well as solves them accordingly. It is a very good system in keeping the project bug free as well as constantly updated.



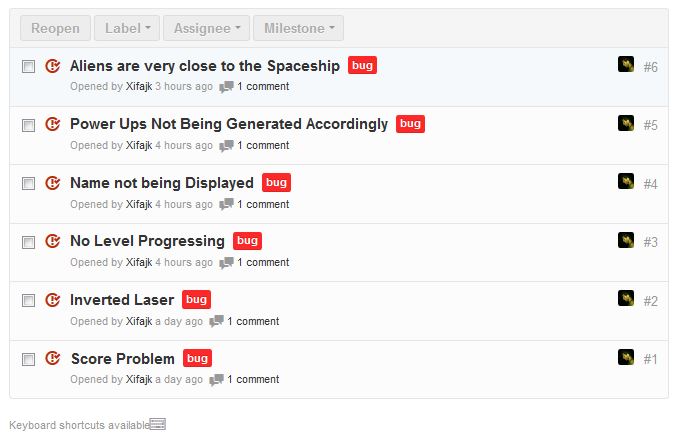
One of the issues solved and closed



Some issues that the client encountered while the prototype was being developed

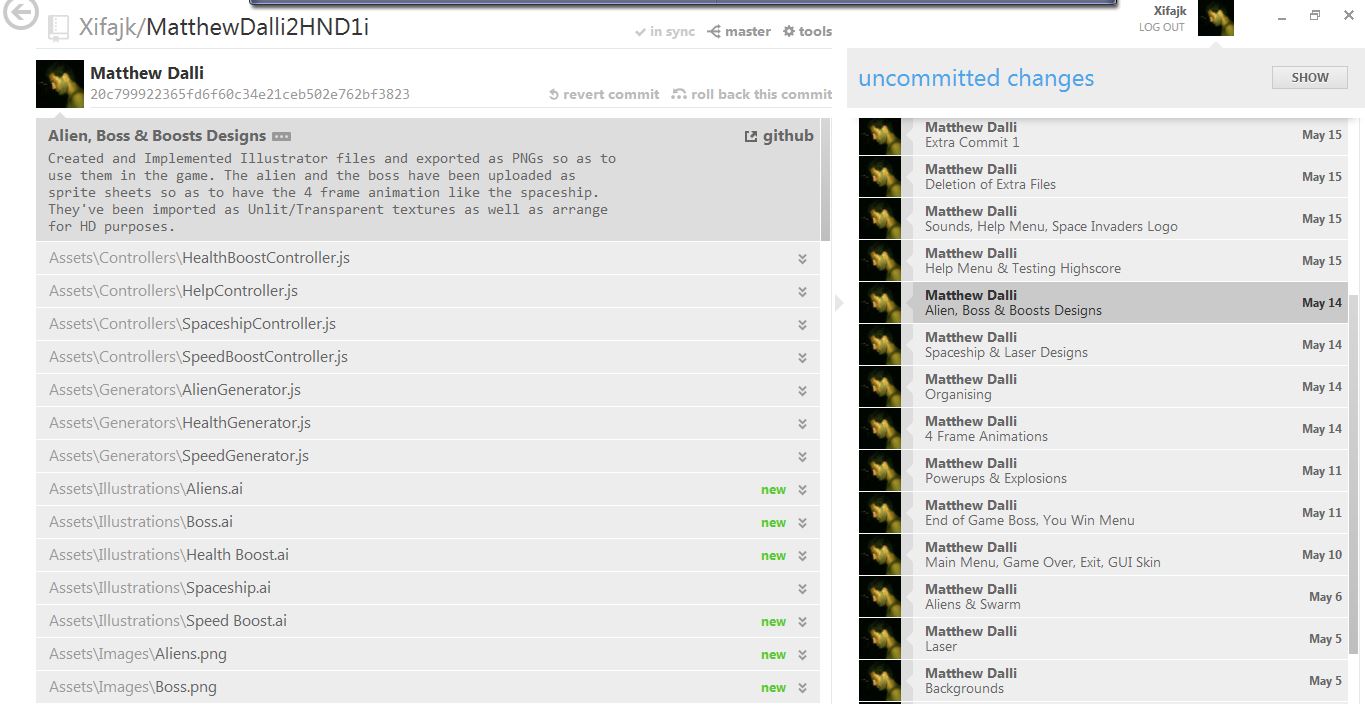


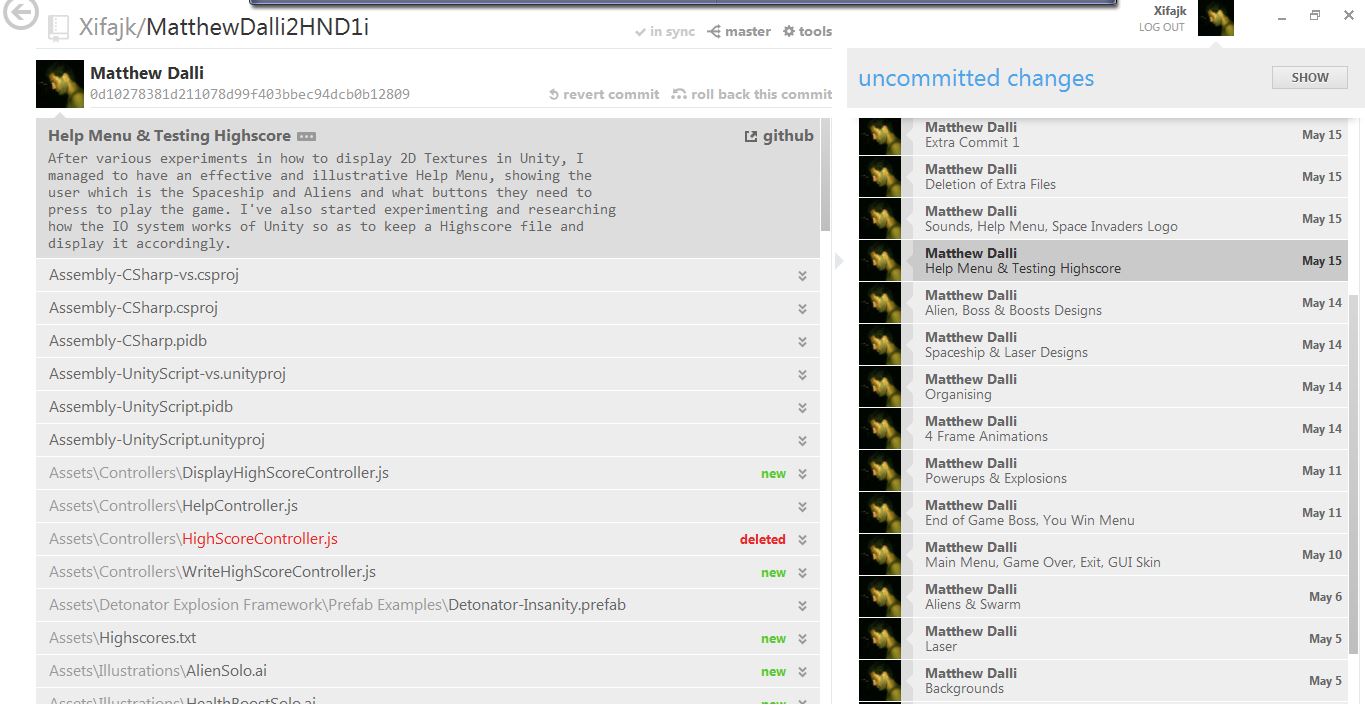
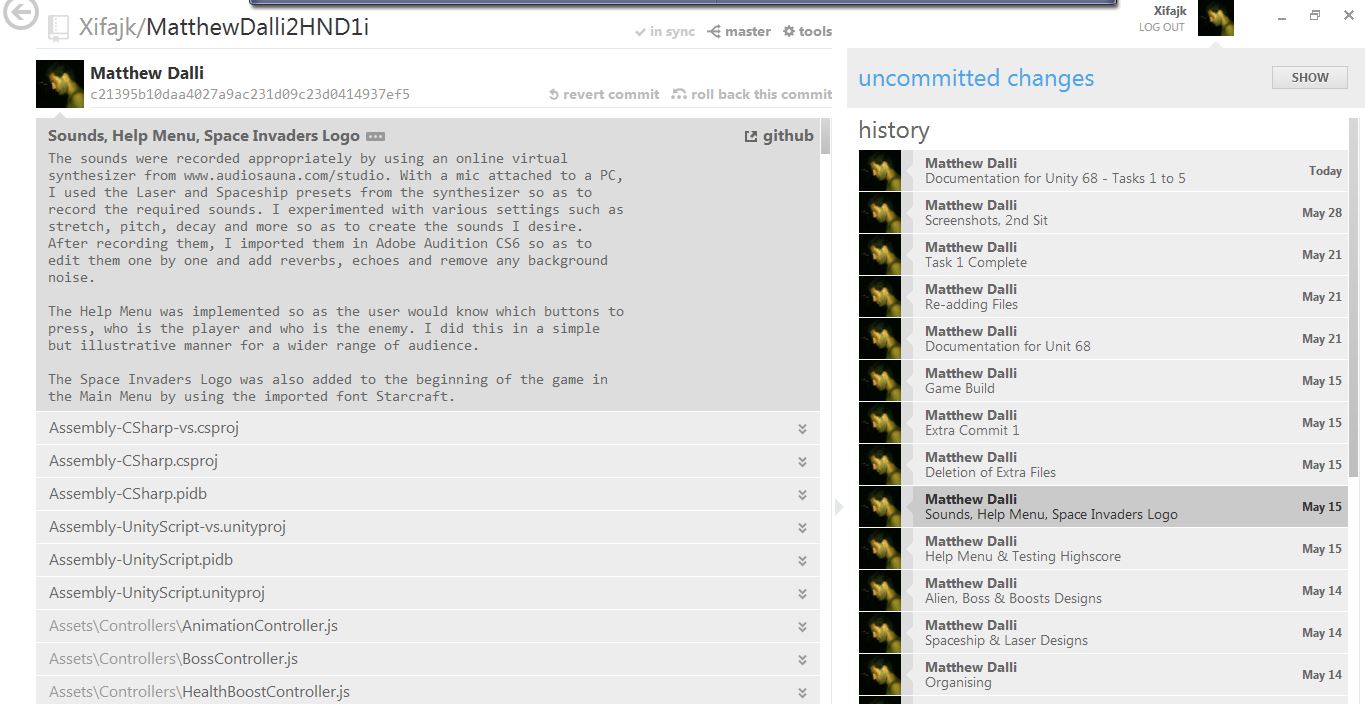
The client reporting an issue



All of the issues have been solved and closed

Task 9 – P4.1

***Develop a fully working interactive media product that meets client needs by showing three consecutive commits documenting the changes carried out to finalise the functionalities of the interactive application as defined in the case study***

**

Task 10 – P4.2

***Evaluate and record interactive media outcomes against the constrains and requirements of the brief by writing a paragraph explaining how the use of Github issue tracking makes it easier for the client and the developer to communicate and share prototypes***

The issue tracker is highly useful in such a project for a number of reasons such as:

1. Client can post all the issues that he or she encounters during the running of the program
2. The developer can constantly update the program to keep it bug free
3. It is highly effective for communication as it removes the need for emails or phone calls. Everything is on one website, in one place
4. Once the issue is solved, the developer can close that issue and continue on other tasks
5. The client of head developer can assign the relevant issue to the appropriate person

Task 11 – M1.1

***Show that effective judgements have been made by finding out about systems which are similar to Git. Write a short report explaining these systems’ similarities to Git.***

Github isn’t the only open source software which helps businesses keep track of their projects. There are a number worth mentioning such as Redmine, Sourceforge and BitBucket. However, Github is the most popular forge nowadays having various advantages such as supporting and accepting most languages. It also allows forking, where developers can edit and manipulate code, even though it isn’t ready.

***During the period Black Duck examined, Github had 1,153,059 commits, Sourceforge had 624,989, Google Code and 287,901 and CodePlex had 49,839.* (Finley, 2011)**

Task 12 – M2.1

***Show that relevant theories and techniques have been applied by explaining how a group of professionals can work together using a system such as Git***

Github comes with various features so as a group of people can work on one single project. This is highly useful, as people can work in their own time, on their own pace from anywhere they are. By giving out appropriate permissions and dividing the people into groups, this will produce a clean and fast work flow as well as one would know who did what thanks to the commit process. The head developer can give permissions of push only, pull only or total access according to the project’s needs.

With its issue tracking, analytics, code review and project management, it makes it one of the best forges in today’s modern day.

Task 13 – M3.1

***A coherent, logical development of principles/concepts for the intended audience have been carried out. Present and communicate appropriate findings by comparing Git with at least one other CVS system****.*

There are various CVS systems, as stated before, but GitHub is quite advantageous due to its functionalities as outline before as well. Mercurial is also another CVS (Concurrent Version System), worth mentioning which works like Git. A difference between these two is that Mercurial is created for larger scale projects, rather than just minor ones between one client and one developer. It is much simpler in design, faster as well as it’s scale of projects is much larger. However, one always needs to see what is best suitable for each project. In this scenario, of creating a Space Invaders game, Github is highly appropriate.

Task 14 – D1.1

***Show that conclusions have been arrived at through synthesis of ideas and have been justified by explaining the concept of branching in Git and why different branches are created***

Branching helps in Git as it is noted automatically throughout the developing of the project, whether personal or commercial. When the developer commits after he or she has done a particular task, Git creates the necessary pointers towards the root project. This is done so as Git will know when a new totally different commit is done and can be called upon for later use, and store the new commit in a different section of the root project. It is highly fast and quite easy to use since it allows one to switch between branches. The main two branches are master, where usually the source code of the project is setup, and developer which runs parallel with the master branch.

Task 15 – D3.1

***Show that effective thinking has been used in unfamiliar contexts by giving an explanation (including a diagram) of how a decentralized team would work in conjunction with a CVS system***

A decentralized team is a group of developers working on the same project however from their respective work stations, which might not be in the same office, building, or even country. To enable them to work effectively without having to transfer large files via emails, they can make use of a Git system. The Git system enables each developer to access information from the CVS repository and modify the source code, data or relative information as they need, according to permissions. They can then save their changes by committing them to the CVS repository. In this way, all developers would have access to the updated CVS repository from the comfort of their workstations.

NIKKY

KAI

IVY

BOB

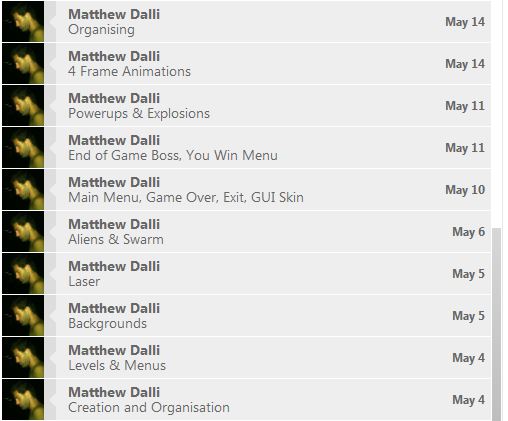
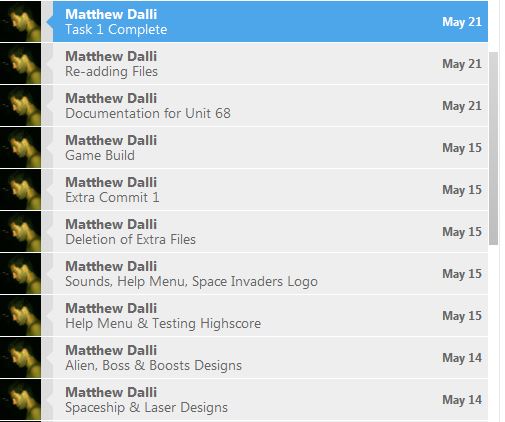
ORIGINAL MAIN REPOSITORY

UPDATED MAIN REPOSITORY AFTER COMMITS

Task 16 – D2.1

***Show that substantial activities have been planned, managed and organized by showing a chronological list of commits as well as the time it took for the work required in each of these commits in a list***

|  |  |  |  |
| --- | --- | --- | --- |
| Commit Number | Name of Commit | Work Done | Time Taken |
|  |  |  |  |
| **1** | Creation and Organisation | Created all the relevant folders, prefabs and materials in Unity | 30 minutes |
| **2** | Level & Menus | Created all the relevant scenes in Unity as well as arranged the Build Settings | 30 minutes |
| **3** | Background | Found some backgrounds online which would be suitable for my game and implemented them in each and every scene with the appropriate co-ordinates and scale | 1 hour |
| **4** | Laser | Created Laser design, imported it in unity and made it fire on keydown | 1 hour |
| **5** | Aliens & Swarm | Created the appropriate scripts for the swarm of aliens and made sure it’s in working condition | 2 hours |
| **6** | Main Menu, Game Over, Exit, GUI Skin | Applied a GUI Skin a main & game over menu as well as exit | 1 hour |
| **7** | End of Game Boss, You Win Menu | Applied an End of Game boss as well as a win menu | 30 minutes |
| **8** | Powerups & Explosions | Designed & generated powerups as well as used the Detonator from the Unity asset store | 3 hours |
| **9** | 4 Frame Animations | Created sprites for the necessary elements that needed an animation, as well as implemented the animation script | 2 hours |
| **10** | Organising | Arranged all elements in the appropriate folders with proper naming | 30 minutes |
| **11** | Spaceship & Laser Designs | Created the designs for these elements as well as imported them in Unity, checking that everything is in working order | 1 hour |
| **12** | Alien, Boss & Boosts Designs | Created the designs for these elements as well as imported them in Unity, checking that everything is in working order | 2 hours |
| **13** | Help Menu & Testing Highscore | Created the necessary information in the help menu as well as testing the highscore | 30 minutes |
| **14** | Sounds, Help Menu, Space Invaders Logo | Recorded and edited the sounds as well continued the help menu as well as creating the main screen for the game | 3 hours |
| **15** | Deletion of Extra Files | Removed any unneccesary prefabs, materials, elements, files or folders | 15 minutes |
| **16** | Extra Commit 1 | Github needed some files to be committed | 5 minutes |
| **17** | Game Build | Exporting the game from Unity to a PC platform | 10 minutes |
| **Total Time** |  |  | **19 hours** |



# Bibliography

Alternativeto, 2013. *Github.* [Online]   
Available at: http://alternativeto.net/software/github/  
[Accessed 24 May 2013].

BitNami.com, 2013. *Ruby Stack.* [Online]   
Available at: http://bitnami.com/stack/ruby  
[Accessed 23 May 2013].

Busu, S., 2013. *Team Collaboration with Github.* [Online]   
Available at: http://net.tutsplus.com/articles/general/team-collaboration-with-github/  
[Accessed 26 May 2013].

Codeape, 2009. *I want something like github, but internal for my company. Recommendations? [closed].* [Online]   
Available at: http://stackoverflow.com/questions/1509364/i-want-something-like-github-but-internal-for-my-company-recommendations  
[Accessed 23 May 2013].

Edwards, W., 2013. *How Github Slew Sourceforge and the rest.* [Online]   
Available at: http://williamedwardscoder.tumblr.com/post/24944468513/how-github-slew-sourceforge-and-the-rest  
[Accessed 24 May 2013].

Finley, K., 2011. *Github has Passed Sourceforge and Google Code in Popularity.* [Online]   
Available at: http://readwrite.com/2011/06/02/github-has-passed-sourceforge  
[Accessed 25 May 2013].

Git, n.d. *3.1 Git Branching - What a Branch Is.* [Online]   
Available at: http://git-scm.com/book/ch3-1.html  
[Accessed 31 May 2013].

Kneath, 2010. *Introducing Organizations.* [Online]   
Available at: https://github.com/blog/674-introducing-organizations  
[Accessed 24 May 2013].

Pocoo.org, n.d. *Github and BitBucket.* [Online]   
Available at: http://www.pocoo.org/~blackbird/github-vs-bitbucket/bitbucket.html  
[Accessed 24 May 2013].

Redmine, 2013. *Redmine.* [Online]   
Available at: http://www.redmine.org/  
[Accessed 23 May 2013].

Stansberry, G., 2008. *7 Version Control Systems Reviewed.* [Online]   
Available at: http://www.smashingmagazine.com/2008/09/18/the-top-7-open-source-version-control-systems/  
[Accessed 28 May 2013].