SPACE INVADERS

P1.1 : TASK 1

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

- A Space Invaders game with 5 levels and end of game boss
- The starting screen contains a game logo and a main menu
- The main menu contains a start game button and an exit button
- The start button loads the main scene of the game
- The exit button exits the game
- A prompt screen that requires the user to enter his/her name with every game play
- The name entered will be shown at the top of the screen during the game play
- The user controls the spaceship using the left and right arrow keys
- The spaceships shoots the laser upwards using the space bar
- The spaceship moves across the bottom of the screen and does not leave the screen
- When each alien is hit with the laser, the score is increased by 1
- The aliens shoot laser downwards
- When the alien hits the spaceship, the health is reduced by 1
- The health of the spaceship is 100
- The score resets with every level
- When the player kills all the aliens on screen, the next screen of aliens is displayed
- With each kill a small explosion should be displayed
- Level 1 has a grid of two rows and five columns of space invaders
- Level 2 has a grid of three rows and five columns of space invaders
- Level 3 has a grid of four rows and five columns of space invaders
- Level 4 has a grid of five rows and five columns of space invaders
- Level 5 has a grid of six rows and five columns of space invaders
- The last level contains an end of game boss that requires 50 hits to be destroyed
- When the health of the spaceship ends, the player dies and the game ends. Then the game over screen is displayed containing the main menu button and the exit button
- If the user wins the game, a screen is displayed congratulating the user and also displays the main menu button and the exit button
- During the game play two different powerups will be popping up, the health powerup and the speed powerup.
- The health powerup increases the player's health by 5 and the speed powerup boosts the spaceship for a few second and then returns to its original speed
- The spaceship and aliens will contain image based sprites

P1.2 : TASK 2

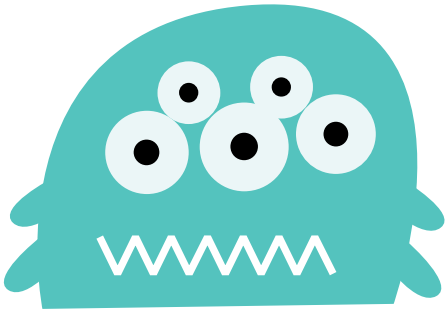
Identify the target group your game would interest. Explain your reasoning in a short paragraph.

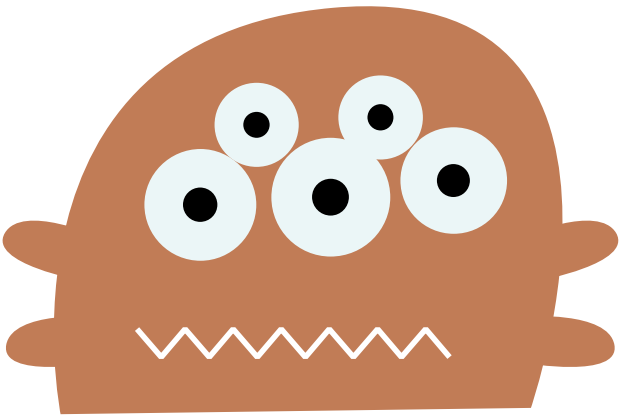
Basically this game is targeted to everyone since it is not a complex game. However it is more targeted to the younger generation of teens but there isn't a reason why adults shouldn't play this game either. It is a fun and simple game for everyone. The mature audience will enjoy playing it because it might bring back childhood memories to them.

P1.3 : TASK 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.

I tried to make the game appealing by including modern graphics. For instance I drew the aliens looking like monsters but not scary monsters so they would still be appealing for all audiences. I applied bright colours to the monsters to attract children and not scare them away. Adults playing the game would notice the difference from the traditional and could enjoy playing something different.





P2.1 : TASK 4

List the areas of expertise required to implement this game.

Unity - The developer must have at least basic knowledge of GUI of the program, how it works and how to function the necessary objects with each other to create a game. The more knowledge the developer knows, the quicker he/she can carry on with it whilst also adding enhancements to the game.

Coding - The developer must have basic knowledge about how JavaScript works because this type of code is needed to make the game function. With basic knowledge it is easy for the developer to understand code of a higher level.

Graphic Design - The developer must have basic knowledge and skills to create the aliens and spaceship. Obviously the more the developer knows about Adobe Photoshop or Adobe Illustrator the more attractive the game can be thus increasing popularity.

Sound - Good sound knowledge is necessary to edit the sounds. It is easy to record a sound through a recorder but for the sounds to be unique and sharp the developer must edit them. Therefore one must know what some of the sound editing tools do. One such useful sound editing tool is the Adobe Audition.

P2.2 : TASK 5

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.

Unity - 7

I think that throughout this semester I have gained enough knowledge about this software to make a game. However I am not that confident in the software to create something bigger than what I have did. On the other hand I believe that with some research and instructions it's not that hard to build an improved version of the game.

Coding - 7

Since I had previously studied JavaScript I understood the code quickly. However there were some new functions that I didn't know about. Coding the game is not as hard because when a problem arises and the developer gets stuck, he/she could easily refer to online forums to solve the problem.

Graphic Design - 9

As I am studying graphic design in my course I found this part of the assignment very easy to accomplish which in fact completed in a very short time.

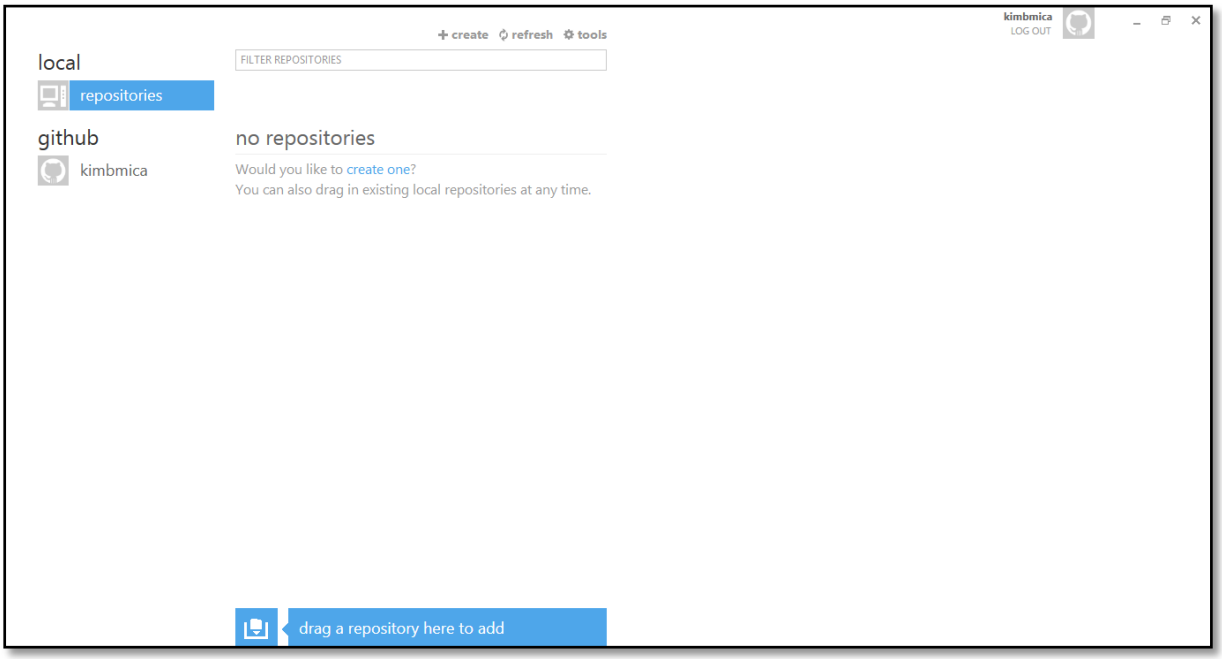
Sound - 8

Given that last year I studied sound in my course, I instantly knew what I had to do to complete this part of the assignment. I used almost the same technique last year to record the sounds and edit them. This also helped me in the knowledge of the editing tools because I knew which tools I had to make use of to edit the sound how I wanted it to be.

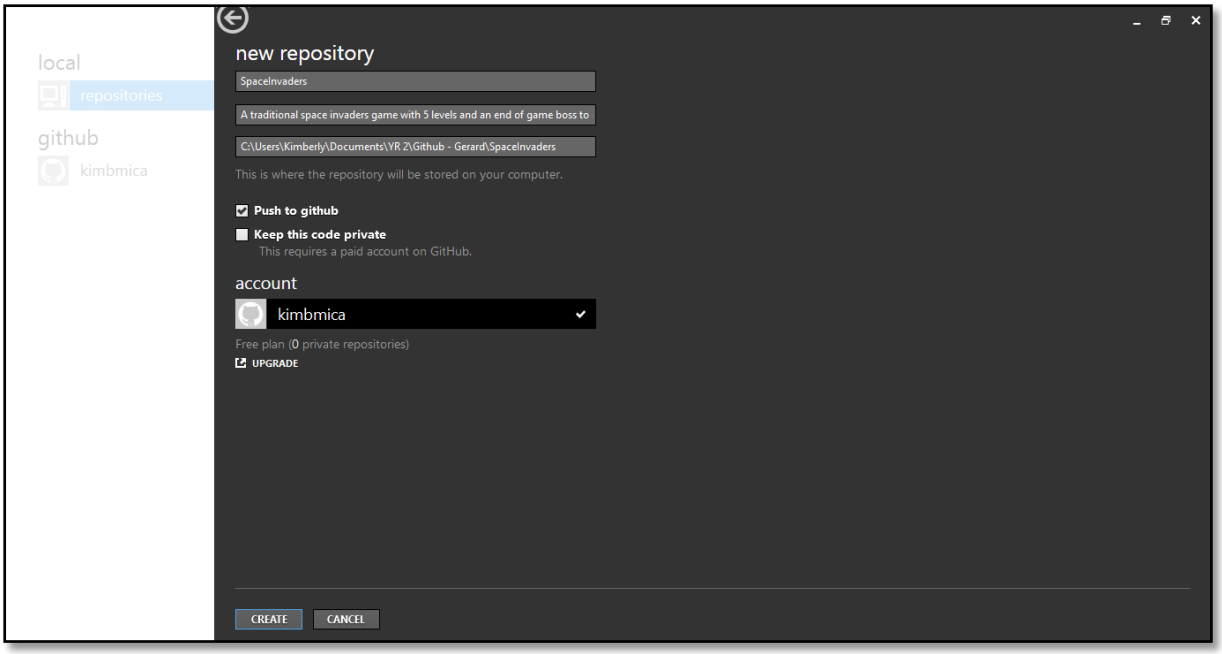
P3.1 : TASK 6

Produce preliminary concepts for an initial prototype by creating a new project on 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.

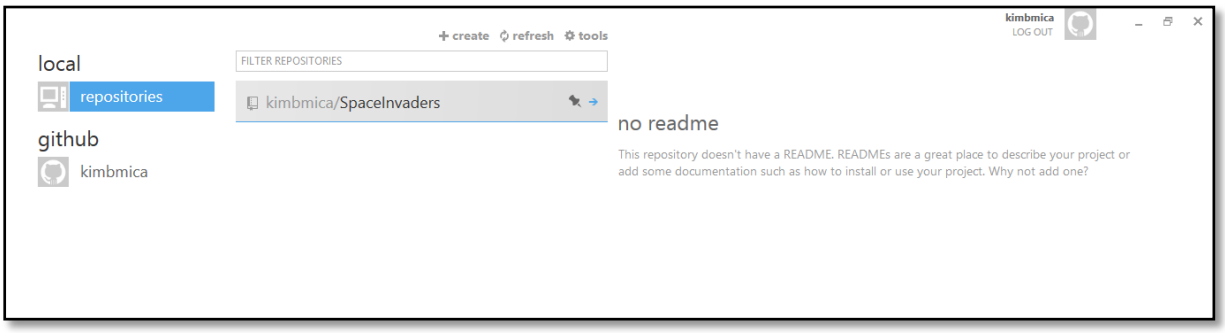
1. Open the GitHub and drag the Space Invaders game folder containing the Unity files.



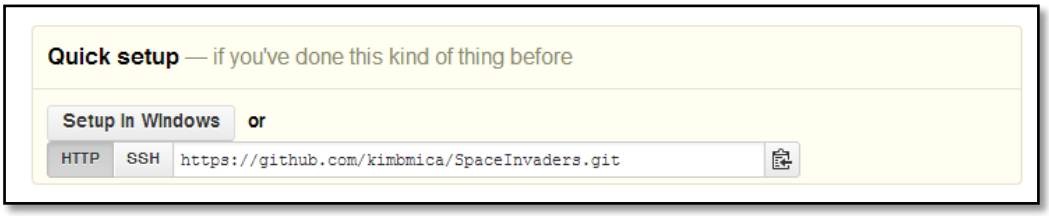
2. The new repository window pops up. The necessary details are already filled in automatically. Enter the description of the repository. Press **Create.**



3. The newly created repository Space Invaders has been added to the local repositories.



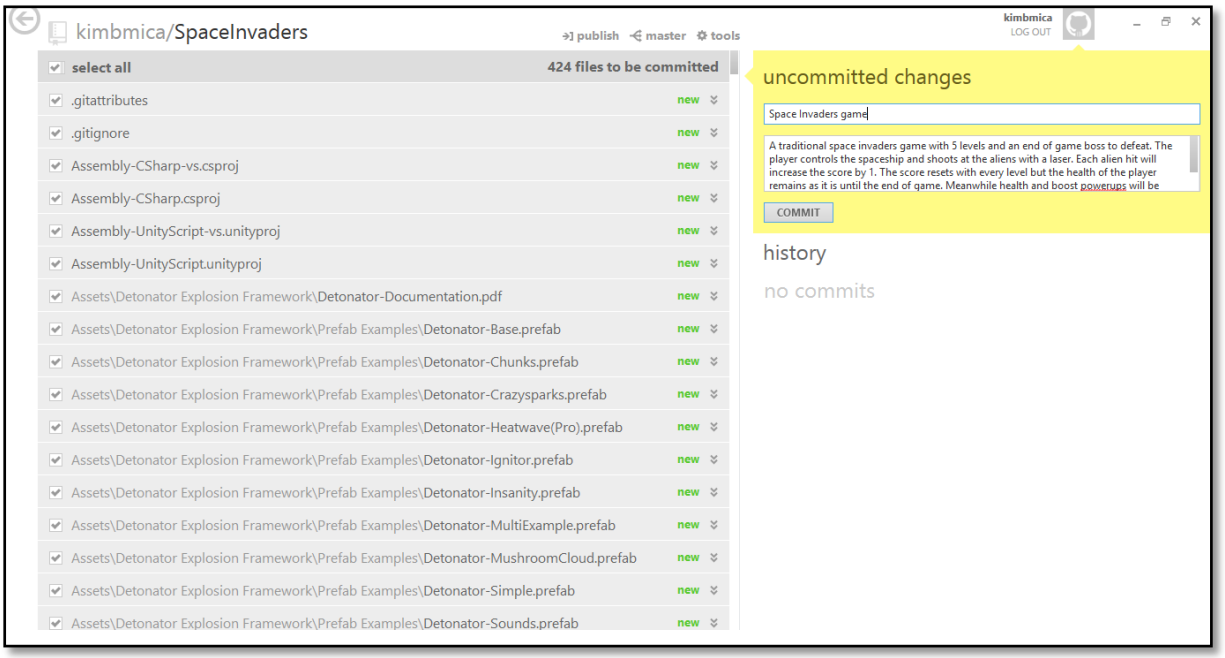
4. A link is created for your newly created repository which you can find in the website.



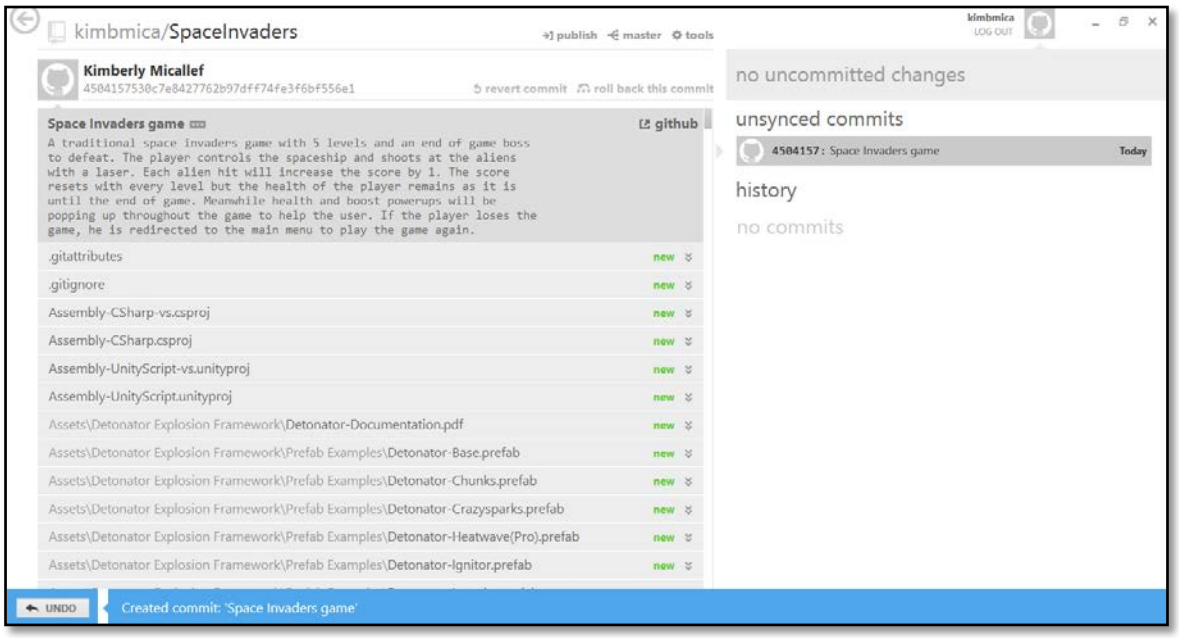
P3.2 : TASK 7

Evaluate and confirm the prototype in relation to constraints 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.

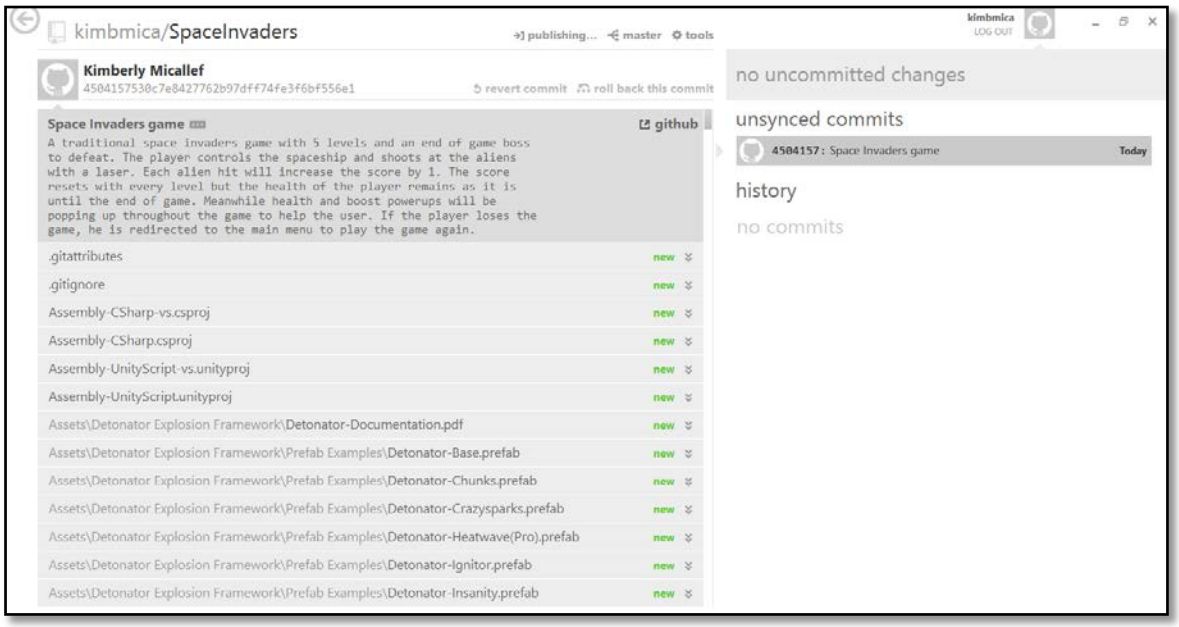
5. Enter the message and description regarding the commit. Press **Commit**.



6. After the commit is created the screen will look like this. Then press **Publish** at the top.



7. You can now see that from Publish it changed to **Publishing**.

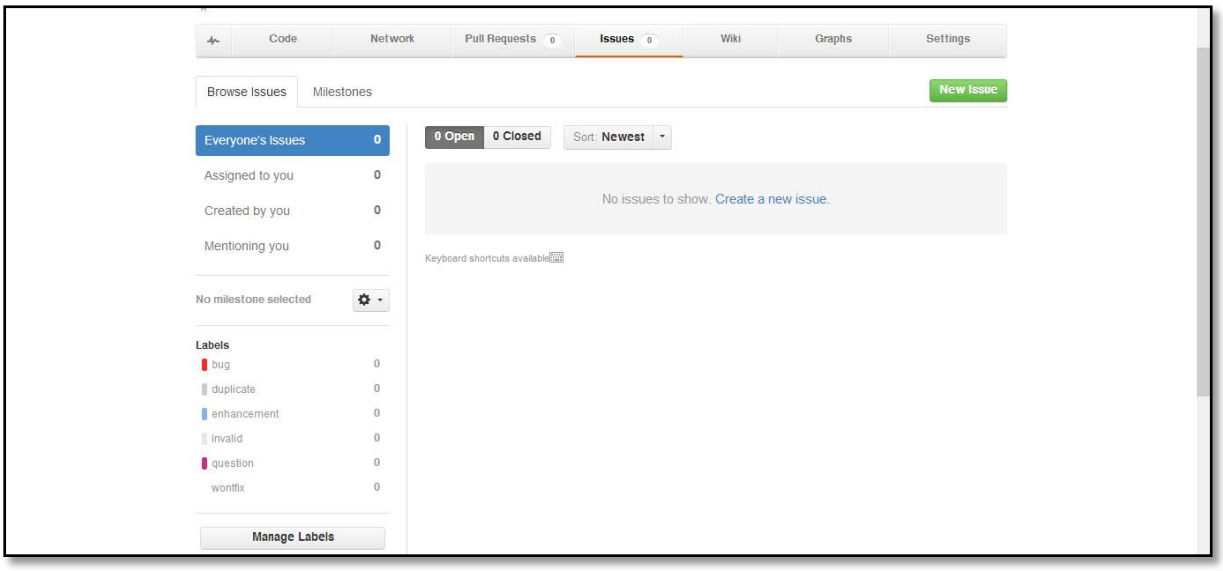


8. After publishing the commit is now synced and the screen will look like this.

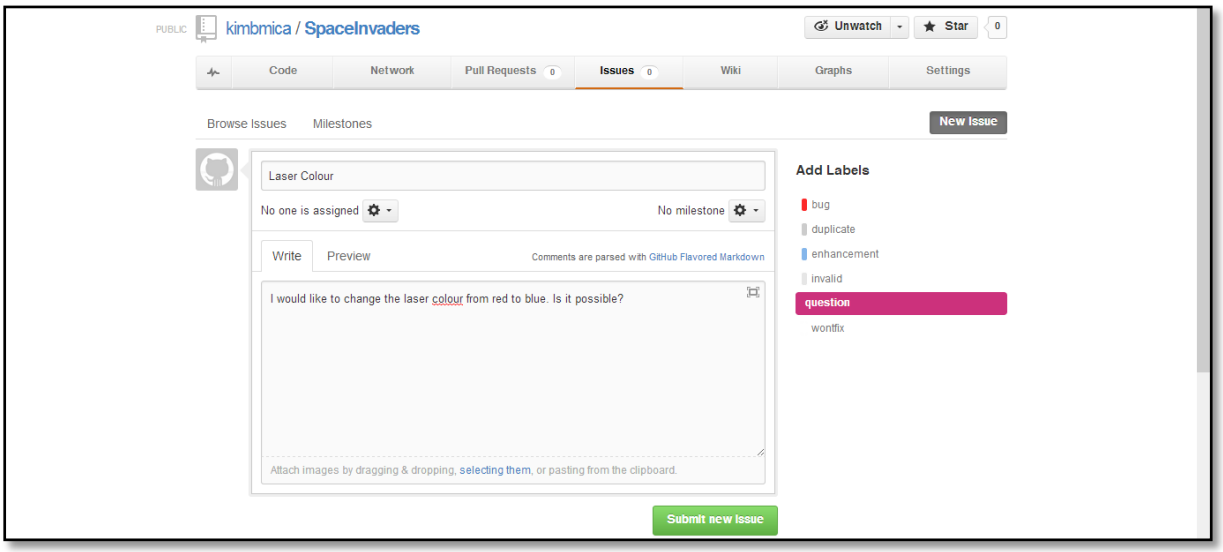
P3.3 : TASK 8

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, then reply to each issue describing what remedial actions were taken to close the issue, before posting another commit referring to this closed issue.

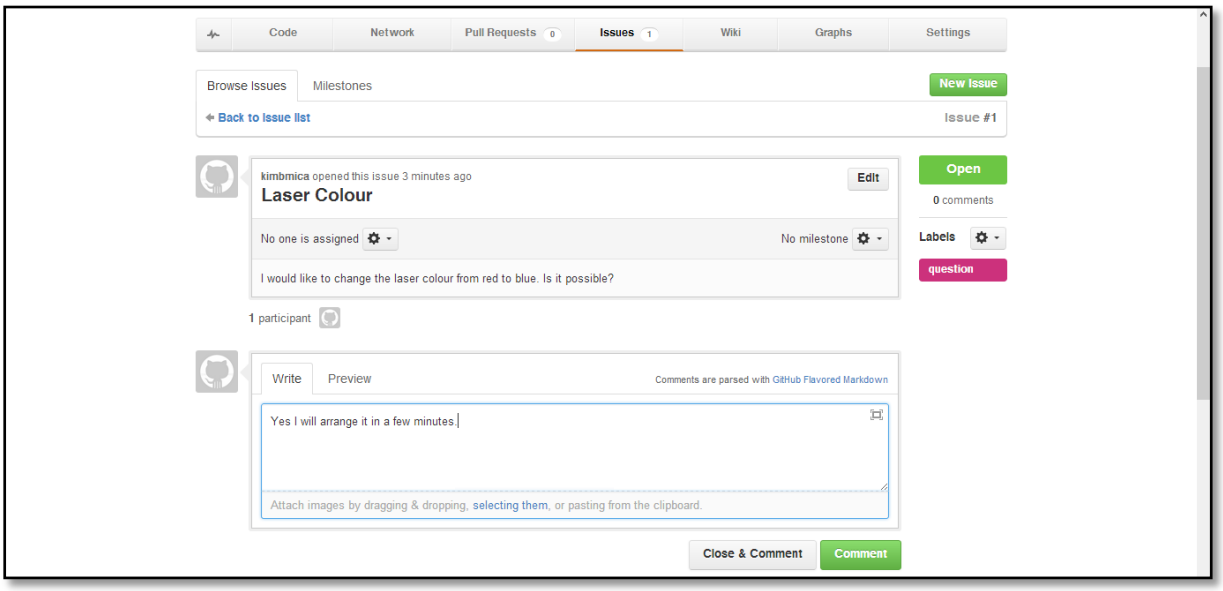
1. Enter www.github.com and press on **Issues** at the top. Then press on the **New Issue** green button.



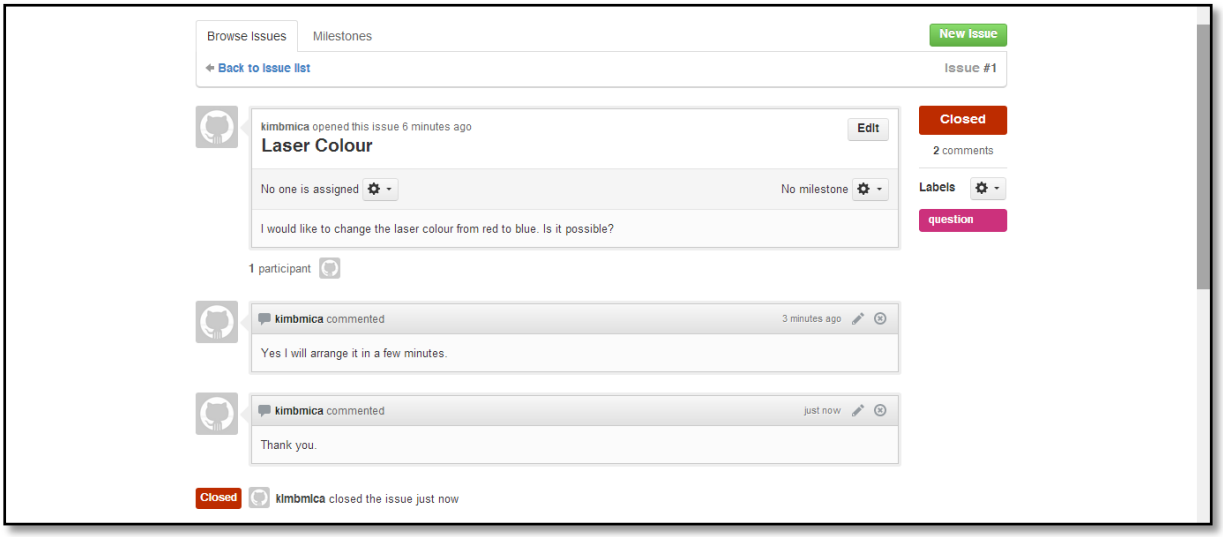
2. The client posted an issue. He wrote the title of the issue and the description of his issue. He marked the issue a question using the Labels option at the right. Then he pressed **Submit new issue**.



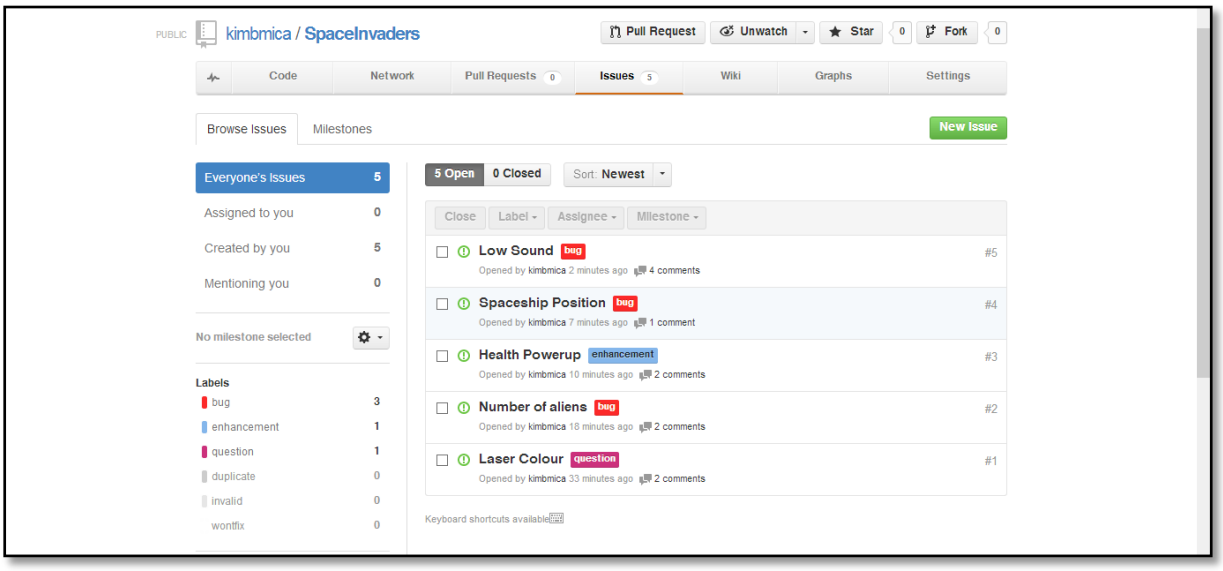
3. The developer replied to his question and pressed **Comment**.



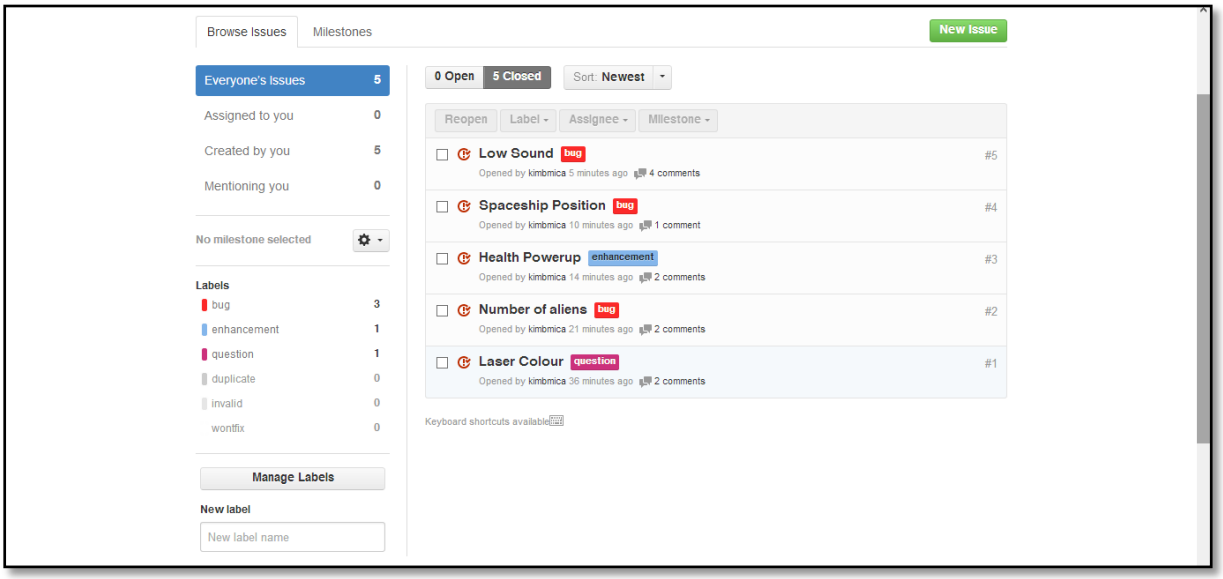
4. The client replied back and pressed **Close & Comment** close the issue. A red indicator turns on showing that the issue is now closed.



5. This is a screenshot showing all the bugs the client reported. These are still unresolved.



6. This is a screenshot showing all the bugs solved.

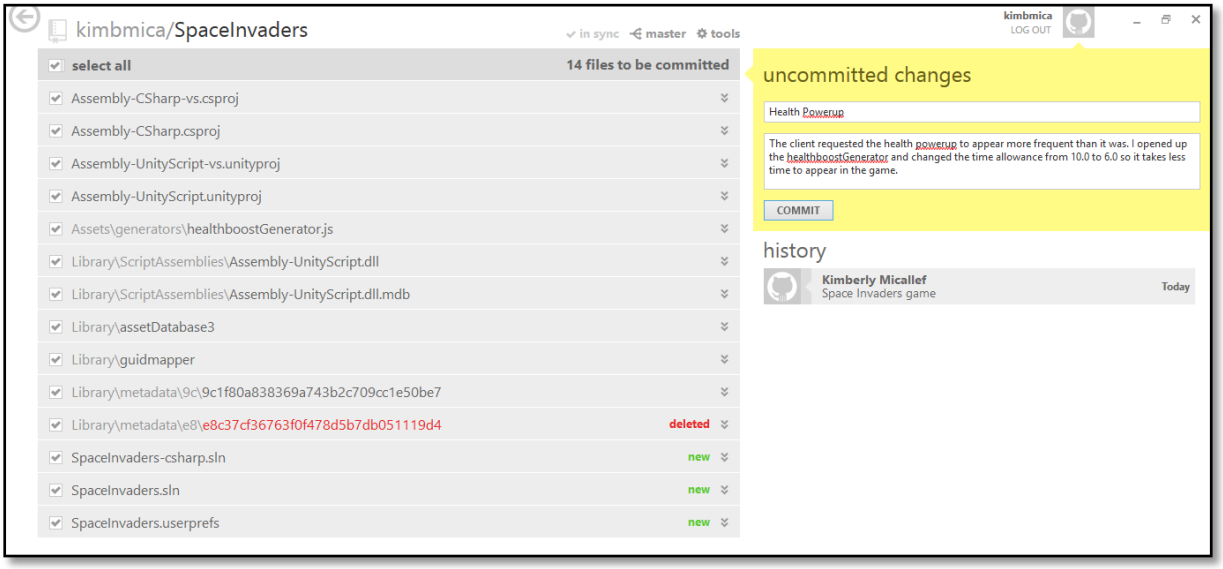


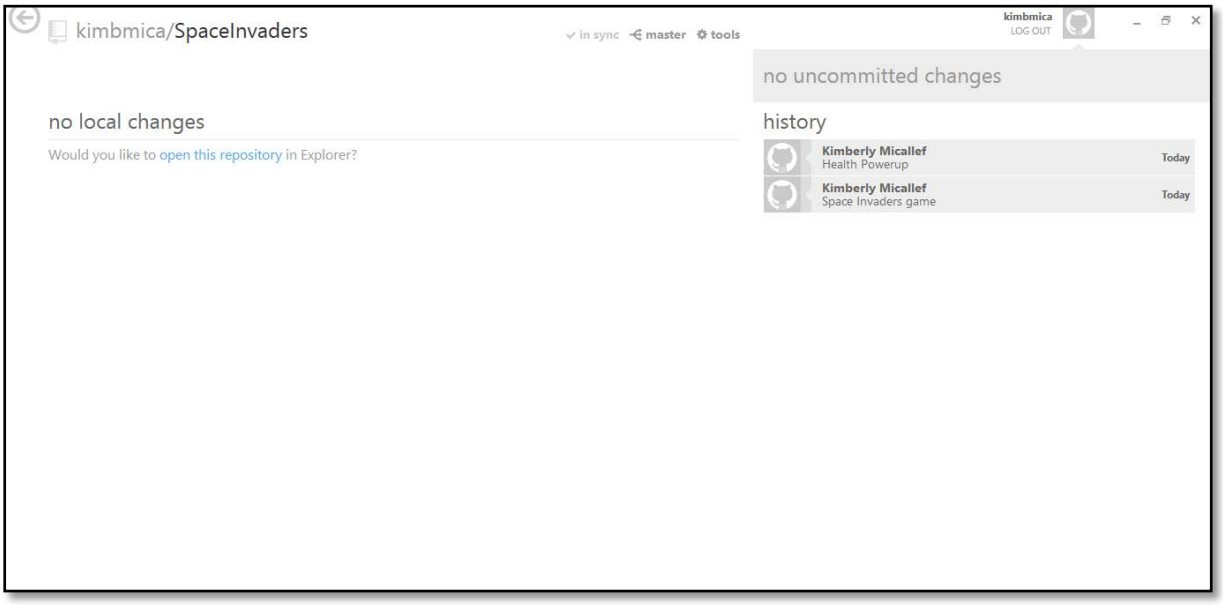
P4.1 : TASK 9

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

Problem 1: Health Powerup

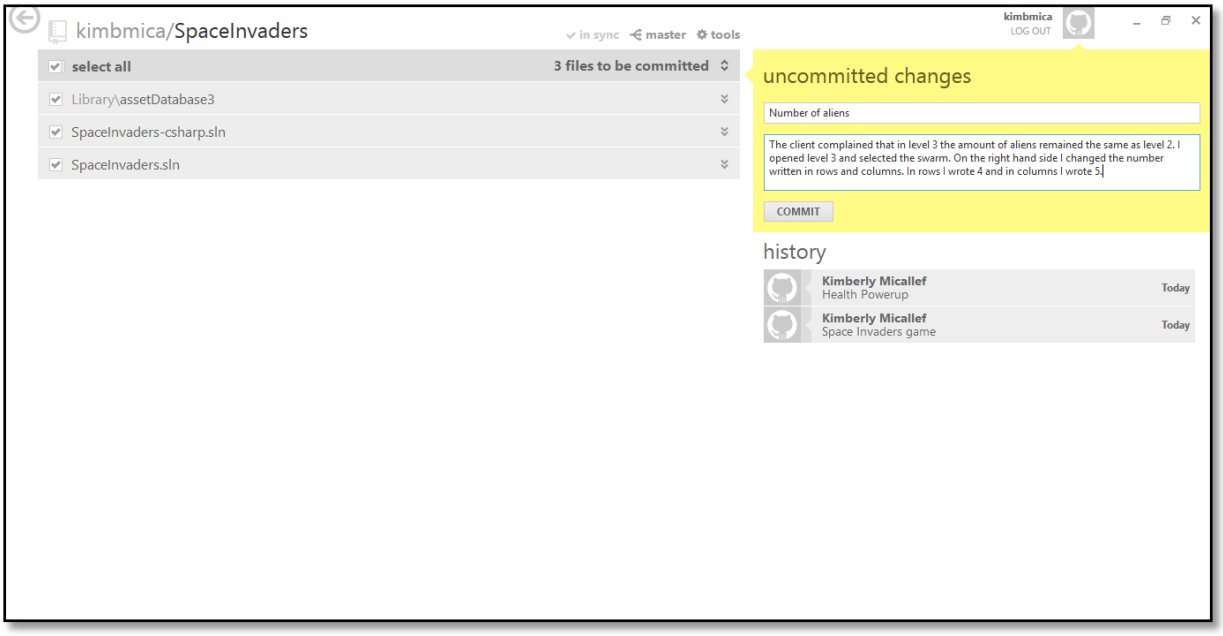
Solution: The client requested the health powerup to appear more frequent than it was. I opened up the healthboostGenerator and changed the time allowance from 10.0 to 6.0 so it takes less time to appear in the game.





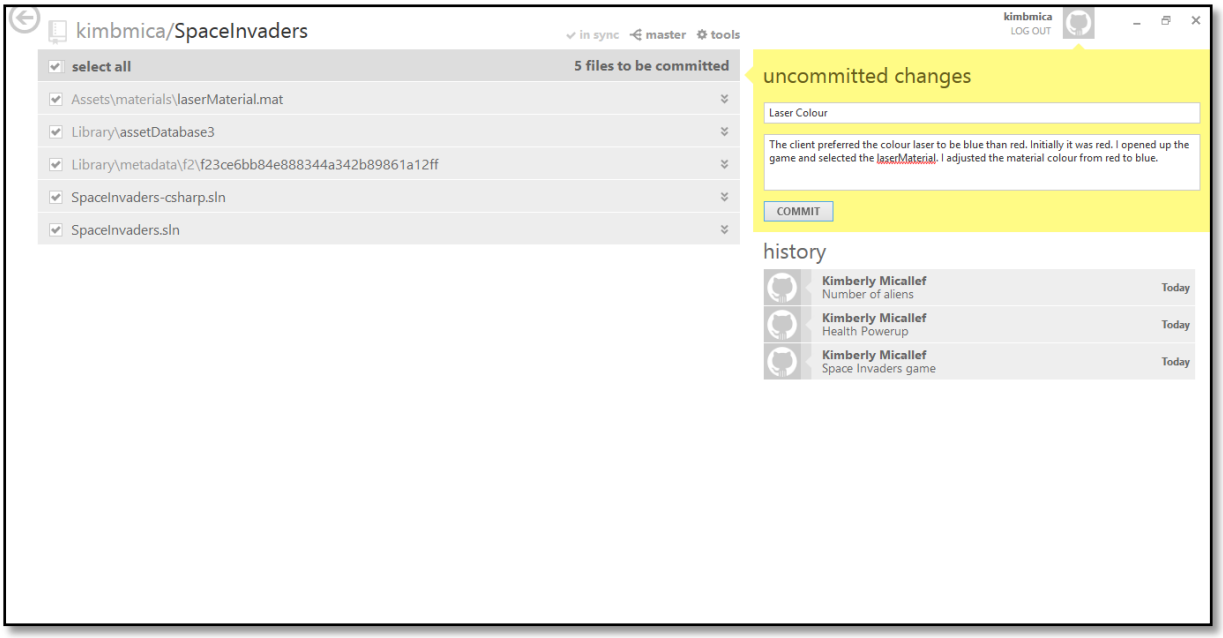
Problem 2: Number of aliens

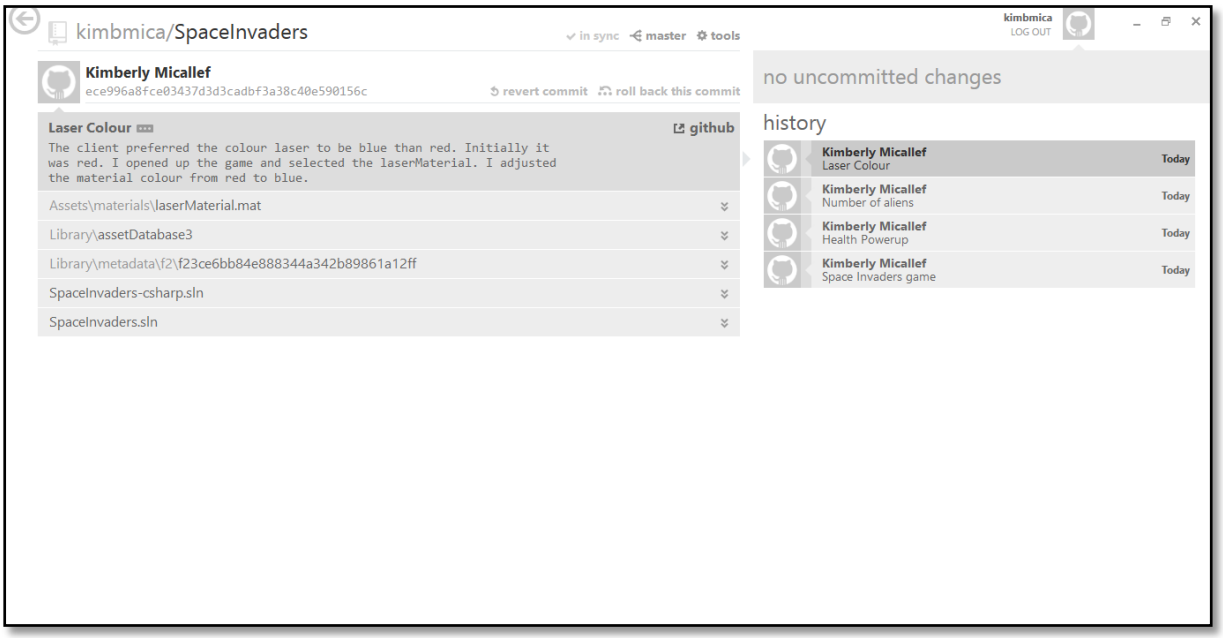
Solution: The client complained that in level 3 the amount of aliens remained the same as level 2. I opened level 3 and selected the swarm. On the right hand side I changed the number written in rows and columns. In rows I wrote 4 and in columns I wrote 5.



Problem 3: Laser Colour

Solution: The client preferred the colour laser to be blue than red. Initially it was red. I opened up the game and selected the laserMaterial. I adjusted the material colour from red to blue.





P4.2 : TASK 10

Evaluate and record interactive media outcomes against the constraints 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 provided by GitHub makes it easier to establish communication between the client and the developer. This is because the developer can deal with issues fast. When the client experiences a bug, he can report it in the issue tracker and instantly the developer gets informed about it. Here the developer can see the issue to resolve it and then reply instantly to the client. When done, the developer or client can close the issue.

The client can prioritize the bugs and apply labels. This is easier for developer since he can see which are the important ones and solves them first.

The bugs can also be searched, sorted and filtered for future reference if the same bugs keep on recurring.

They can both view the prototypes which in this case helps the developer since the client will be testing the application throughout the way.

M1.1 : TASK 11

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 is a web-based repository hosting service where software developers can control and manage open source software development. Similar to this there is SourceForge and Bitbucket which support CVS, Git or Mercurial repositories. Just like Github, Google Code offers code review, Wiki, teaming up, web hosting and issue tracker. CloudForge offers also mention the previously mentioned as well as branching, both personal and private repositories.

M2.1 : TASK 12

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 has Organizations that allows multiple developers to collaborate together on one project. An Organization allows a developer to team up with other developers, have a public organization profile, setup special permissions and keep track of what goes on in the organization.

Organizations are ideal when a repository is owned by several people thus allows multiple Github accounts. It also agrees to have read-only collaborators within the repositories. Github Organizations are also open-source but special permissions can be set up so as to control who develops what. This service is free.

This is ideal for a company employing several developers. It helps the workflow of the business and large open source projects. In other words Github Organizations are the best way to manage open source and business teams.

M3.1 : TASK 13

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.

Github and Bitbucket are two similar Concurrent Versions Systems. Github is ideal for open-source projects whereas Bitbucket is ideal for non open-source projects. Github offers private-repositories but one has to pay a pretty high price for this whereas Bitbucket offers unlimited private repositories for free.

Github has also a limited space for public repositories. Bitbucket charges money for the amount of collaborators of each project. Each project is allowed up to 5 collaborators which is a very good free service.

Bitbucket supports Git and Material repositories where as Github does not. But Mercurial repositories can be pushed via Hg-Git which is a Mercurial Plugin that has the ability to push and pull repositories from Mercurial to Github. Hg stands for Mercurial. Ultimately the advantages of these systems, is that an existing repository can easily be moved to another one since both are git-based.

Github offers inline editing whilst Bitbucket does not. If the developer needs to do a quick fix and edit a file he can if using Github but Bitbucket does not offer this option. An alternative solution to this in Bitbucket is to make a clone in a local repository and then push it.

D1.1 : TASK 14

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 in Git is very useful. Branching is when a new branch is created before continuing to work on the development. Thus if a bug comes up and the developer needs to fix it urgently, he still can without losing the new data. He just goes back to the master branch and fixes the bug from there.

As this point the project is exactly the way it was and was not messed up by the new branch. This is the necessary code to create a new branch with testBranch being the name of the new branch:

```
$ git branch testBranch
$ git checkout testBranch
```

`$ git branch testBranch` creates the branch whilst `$ git checkout testBranch` switches to this newly created branch.

D3.1 : TASK 15

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 CVS system is very helpful in a decentralized team where a group of developers located in different areas of the company or country can still work together on the same project. This is because the Git systems allows for each developer to access information from the CVS repository (such as Github) and modify it from there. The changes can then be saved using commits which are added to the CVS repository. This process can be made by all the developers allowing each of them to have the latest updated version of the repository from their own particular location.

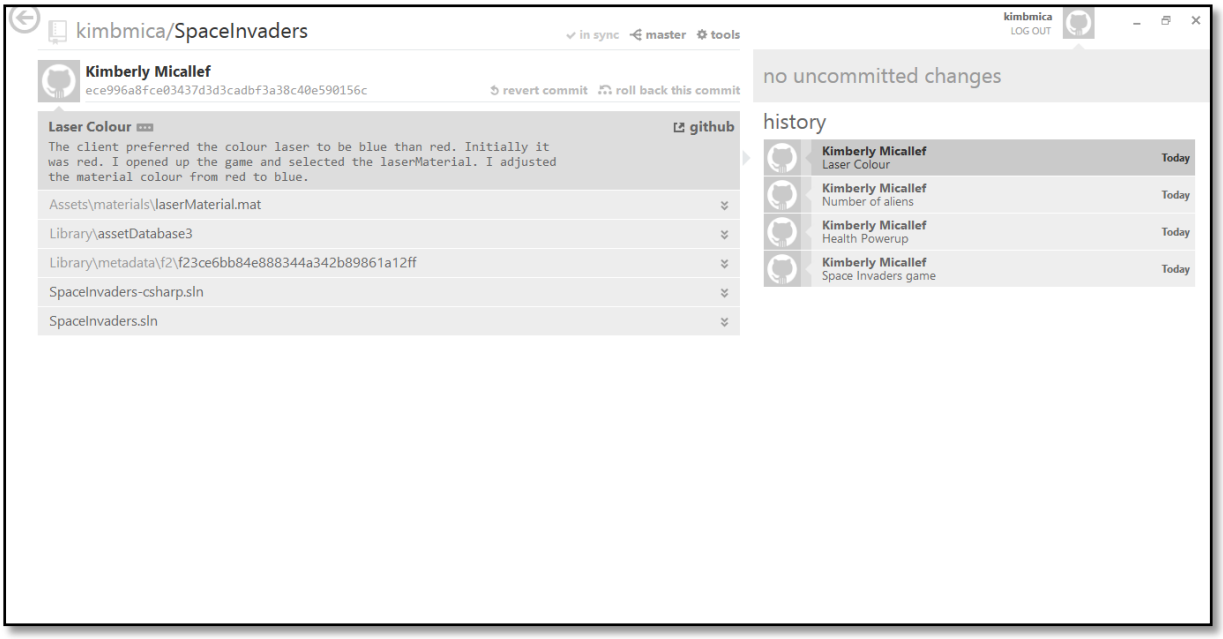
The CVS is continuously being updated by the developers by committing changes and uploading the latest repository to the Git.

D2.1 : TASK 16

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.

	Name of Commit	Description	Time Taken
1	Space Invaders game	This commit includes all the features of the game. The game is fully functional. It took around 15 minutes to commit and publish. `	12 days
2	Health Powerup	I arranged the amount of times the health powerup was appearing and made it appear more frequent. It only took a couple of seconds to commit and publish.	2 minutes
3	Number of aliens	The client noticed that the aliens did not increase in a certain level. I fixed this bug in a couple of seconds since it was a minor change. It took a few small minutes to commit and publish.	1 minute
4	Laser Colour	The client requested that the laser colour changes from red to blue. It only took a few seconds to publish.	1 minute

This screenshot shows the commits done so far.



REFERENCES

http://tilomitra.com/bitbucket-vs-github/

https://github.com/blog/674-introducing-organizations

http://en.wikipedia.org/wiki/SourceForge

http://git-scm.com/book/en/Git-Branching-Basic-Branching-and-Merging

http://www.astro.princeton.edu/~rhl/cvs/cvs.html#SEC2