# Project Write up

## ****Problem Specification****

Some friends want to play a game which has the following features.

A 2d platformer / shooter which should have two game modes:

* Vs - two or more players play against each other in a deathmatch game
* Single Player / Coop - one or more players fight against waves of npc enemies

It should use local multiplayer and platforms should be destructible breaking into pieces when shot using an algorithm to determine the fragmentation pattern of the platform. The NPCs should use path finding algorithms to find the way over rubble from broken platforms to the player(s). As a last addition I will add a custom level designer which is easy for users to use.

## ****Analysis****

### ****Stakeholders****

#### ****Users****

The main group to benefit from the game are going to be the players. Players will be a diverse group of users. They will likely already have some experience of how games work. This means an in depth tutorial on how to use a controller is not necessary, however controls for the game should be similar to other games the users have played as this will avoid a long learning curve for the player which maximises the time they have enjoying the final product. A screen should be shown when the player first opens the game which describes what the controls are briefly. Controls should be easy to change and sensitivity should be easy to change.

#### ****Accessability for Different Users****

The game should be usable with many types of controllers including those for people with disabilities. It may be worth considering adding support for mouse and keyboard as not all players will have access to a controller however this may give some players an advantage against others. options to change colours to make use of the game for colour blind users should be implemented and differences between colours should not be the only signifier for a differences in two items e.g. different guns should have different shapes and different colours.

Users who do not speak english should be considered so easy methods of chosing language should be implemented even if different languages are not added at the time. This will make further improvements to the game easier in the future.

#### ****Other Stakeholders****

My teacher is also a stakeholder. They need the game to be completed within a time frame and to have a large amount of algorithmic complexity.

### ****Research and Identification of Solutions to the problem****

#### ****Platforms for designs****

I have considered two options for languages to develop the software:

1. C++
2. Python

The benefits of C++ are that it is compiled, fast and a standard for game development however my knowledge of C++ is not as good as in python and getting help is harder as the language is not taught in my college also the libraries. Python is slower but it is the language that my college uses which means help is easier to find. Also I have more experience in python. For these reasons I decided on using python for the game.

I did some research on [python game libraries](https://geekflare.com/python-game-development-libraries-frameworks/)

I decided on using Pygame for my game as it will make the development process far quicker since I do not need to spend time handling displays and events. Pygame is the best option as there is good documentation on the python module and lots of information on YouTube and the web. It is also a fast option compared to other python modules for game development. There are also two options with Pygame: I can use Pygame-ce or I can use the official Pygame. The official Pygame is currently not in development and the owner of the GitHub repository will not let progress continue because of political reasons. Pygame-ce is currently a much more active project. For this reason I plan to use Pygame-ce.

#### ****Looking at other solutions****

I looked at other games that had the some of the features that I wanted in my project.

Stick fight:

Pros:

* Fast paced
* Easy local multiplayer
* Some platform destruction

Cons:

* Not open source
* Costs money
* Online multiplayer can be buggy

frog smashers:

Pros:

* Fast paced
* Simple
* Open source

Cons:

* Only controllers can be used. This limits the users as some users may not have a controller or may have only one controller which means they cannot play with friends
* Very simple game mechanics which can do have variety in the ways they can be used but can get monotonous / boring
* uses unity which is first helps with lots of the programming would not show my skills in programming

One feature in Stick Fight that I liked was the [menu screen](https://drive.google.com/file/d/1VF03cVP2xV6kSCA_54Ucrns5NLPGiZRU/view?usp=drive_link). It uses a diegetic menu which is where game's interface elements exist In-Universe. In this case the players can physically hit each other and move into different rooms to choose whether to play local multiplayer or online multiplayer. The game uses a nondiegetic menu for some of the less common parts.

I also liked thought that the fact it was not too competitive was a benefit as it fit the games style as a game to play with friends. The randomness of each round means that players who are not so good still win a certain amount of rounds. This keeps the game fun for all players no matter their skill level.

**Conclusions**

My game should be open source and fast paced. It should avoid...

### ****Essential Features****

* Player versus player (PvP) combat is an essential feature. It is important as users want to play against each other.

### ****Limitations****

* The proposed system will only use local multiplayer. There will be no online multiplayer as this will be difficult to implement and is out of the scope of my game.

### ****Objectives****

* Fragmentation of platforms when hit by projectiles
* Collisions between entities in the game
* Collisions between players and objects
* The game should run with FPS over 60 for smooth game play
* NPC enemies which can path find to the player