# Component 3 (Project)

[Component 3 (Project) 1](#_Toc1)

[Analysis 3](#_Toc2)

[Project Identification 3](#_Toc3)

[Summary of my project 3](#_Toc4)

[Characters 3](#_Toc5)

[Sounds 3](#_Toc6)

[Sprites 3](#_Toc7)

[Computational Methods 4](#_Toc8)

[Thinking Abstractly 4](#_Toc9)

[Thinking Ahead 4](#_Toc10)

[Thinking Procedurally 4](#_Toc11)

[Thinking Logically 4](#_Toc12)

[Thinking Concurrently 4](#_Toc13)

## Analysis

## Project Identification

### Summary of my project

The project that I intend to complete is an infinite running game. Games like these can be played by a single player at a time. The player navigates through the game using the space key to jump navigating along the infinitely scrolling level manoeuvring themselves around a constantly changing environment. A high-score is calculated by the distance that the player completed plus the number of rewards that the player collects. This high score can then be compared with others scores to enable competition between players. The game-play gets progressively harder with a higher speeds.



### Characters

In the original game there is only a single playing character. In order to increase the difficulty of the game my client has suggested adding more characters (enemies). These objects, when hit, decrease the users score by adding a time penalty.

Although there will be multiple types of malicious objects (sharp objects, low branches, ect) they will only differ stylistically and the same time penalty will be given to all collisions.

### Sounds

I want the game to have an arcade like feel. Therefore I will implement simple sounds to create this impression. I will have a theme tune that plays continuously, speeding up as the game gets faster, this will increase the adrenaline response of the player.

### Sprites

Much like my inspiration I aim to use pixel art for my sprites. This will add to the arcade feel of the game. Although the monochrome aspect of the original game is visually interesting I believe that sprites with a wide range of vibrant colours will be more visually appealing for younger audiences.

## Computational Methods

The game is suitable for a digital solution as it has principles / physics that can be easily enforced by simple algorithms. A non-computational version of my project would lack immersion as in game objects would not be able to be individually controlled. Due to the fundamental lack of refreshing controllable graphics in non-computational situations it is imperative that my program uses a computer.

Another benefit of creating this game computationally is the potential for networked gameplay. In a future version of this game networked gameplay would allow players to play the game socially, creating an element of competition.

### Thinking Abstractly

In order to simplify the gameplay and the production of the project many unnecessary aspects of reality are discarded. The main example of this is the 2D nature of the game, although in reality the world is seen in 3D a 3D game would overcomplicate the design / ease of gameplay. Another example of abstraction in my design process is the blocky graphics used, the need to make the game photo-realistic is diminished allowing programmers to focus on creating an engaging playing experience.

### Thinking Ahead

My games gameplay will be controlled using the space bar (To Jump) / arrow keys (^:To jump).

In order to navigate through the menu screens / settings and to pause the game a mouse cursor will be used.

### Thinking Procedurally

I will make use of object oriented programming in order to maximise efficiency of my program. All moving objects will share a class with more specific sub classes being employed for each specific character.

### Thinking Logically

For collision detection the program will continuously check to ensure that the character hasn't collided with the platforms.

### Thinking Concurrently

Te program will use event driven programming meaning that when the user triggers an event code will be executed. This is a form of parallel processing.