Game Engines

Task 1

Analysis

Dante Camenzuli

BEAT Stomper

BEAT Stomper is a 2D arcade game released in March 2016, and was developed and published by the one-man company Rocky Hong. It's a very simple game that requires the user to simply tap on the screen to make a cube with a very void expression jump from one platform to the next. The game has many visual assets such as particle effects, background artefacts, changing colours as well as inverting said colours, screen transition, sound effects, and a great selection of music that play as levels.

I would require a few scripts to recreate the game as similarly as possible, at least in terms of mechanics, namely:

- Game manager
- Platform manager
- Jump control
- Camera control
- Score system

Game manager would control most of the games functions: how it starts, transitions, colours, and music.

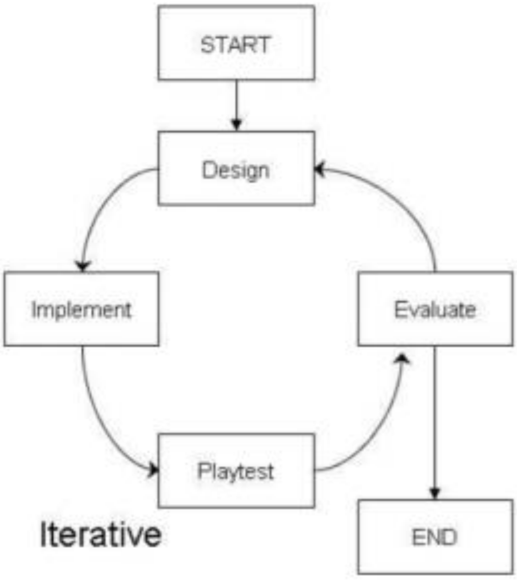
Jump control looks after the cube's jumping and landing mechanics.

Platform manager controls the spawning and despawning of platforms and their triggers.

Score manager handles the scoring system.

Camera control moves the camera upwards then resets at the end of the game.

The most common development model nowadays is the Agile model, and it is used in many large companies. It's a very efficient way to tackle a project in small increments but my main issue is that this type of model is that it is meant for large scale projects; it relies on patch updates to continue customer satisfaction (Try QA, 2019). To code the mimic efficiently, I will use the spiral development model as it makes the most sense in this situation, that is, it's a small project and I'm working solo. The spiral model works well since I'll be instantly debugging every new piece of code I implement, and going back to fix any issues that arise, then move on to the next piece of the game and repeat.



This allows the core mechanics to be built and finalised without interruption from other bits of code leaving you with a ready prototype. Once all the bugs in the core mechanics have been ironed out, other mechanics, effects, and so on can be coded in and evaluated (Mignano, M., 2019; Try QA, 2019).

References:

Mignano, M. (2019) *Spiral model for game development: Techniques to develop games.* GameDeveloperTips. [online] Available at: http://gamedevelopertips.com/spiral-model-for-game-development/

Try QA (2019) *What are the software development models.* [online] Available at: http://tryqa.com/what-are-the-software-development-models/

Try QA (2019) *What is the agile model – advantages, disadvantages and when to use it?* [online] Available at: http://tryqa.com/what-is-agile-model-advantages-disadvantages-and-when-to-use-it/