Game Engines

Task 2

Design Brief

Dante Camenzuli

BEAT Stomper

Target Device

BEAT Stomper is a mobile game available on Android and iOS phones. Typical resolution for the two vary, especially for Apple phones. On average, Android phones have a screen resolution of 1440 x 2960, while Apple can't seem to decide on one as each iteration of their phones has a different resolution (mediaGenesis, 2018). I'll be using the standard 720 x 1280 resolution as it is also the most common (Piejko, P, 2016). As the game is a mobile phone app, all it requires to function is the user's finger tap on the touchscreen.

Game Objectives

The objective of the game is straightforward: Achieve the highest score each round.

Controls and Mechanics

BEAT Stomper is rather simple to play: Tap to start the game, tap to make the cube jump up and tap again to slam down onto a platform.

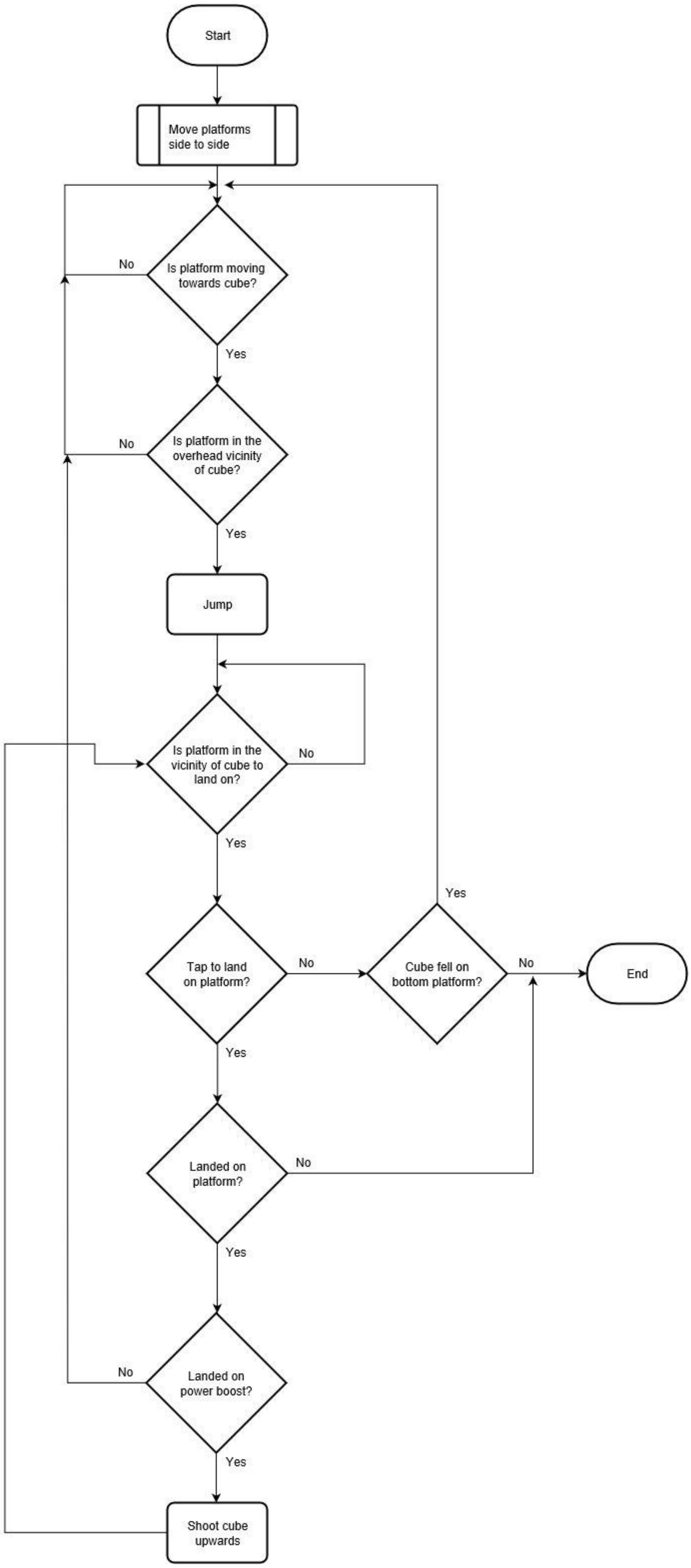
Every ten platforms a power boost icon will appear somewhere on the platform; land on it and the cube will shoot upwards around twenty-one platforms and pause just long enough for you to tap again to land back down safely.

As you advance upwards, the platforms will shrink in size down to a certain point to increase the difficulty. The platforms also slowly move downwards towards the bottom of the screen after a short period of time, compelling you to move upwards. Falling off of the platforms will end the level.

Game Screens

The game has multiple game screens but the most important two are the title screen and the level screen. The title screen shows the high score and allows you to navigate and play different levels as well as access the settings and credits screens. The level screen is where one plays the game. These differ in colour, music, and particle effects, but all play the exact same way.

Gameplay Flowchart


Start
Move platforms side to side
Is platform moving towards cube?
No
Yes
Is platform in the overhead vicinity of cube?
No
Yes
Jump
Is platform in the vicinity of cube to land on?
No
Yes
Tap to land on platform?
No
Yes
Cube fell on bottom platform?
Yes
No
End
Landed on platform?
No
Yes
Landed on power boost?
No
Yes
Shoot cube upwards

Visual assets

As a 2D game, BEAT Stomper is very visually appealing that will be hard to replicate exactly. Each level has a different soundtrack, colour scheme, particle effects, and floating artefacts in some cases. The floating artefacts would need to spawn offscreen and thus will require prefabs. The cube has a trail that also changes depending on the level being played. Each level also has a simple piece of artwork acting as a background, so I will need to find these images or similar ones.

The platforms themselves will be spawned from a prefab and move from left to right at a constant speed. A boost icon, also spawned from a prefab, appears every ten platforms somewhere along the width of the platform; this glows and has a rotating halo – I can make it glow, but not rotate. When a platform lowers itself off the screen, a fan of rectangles animate to indicate it's destruction – this can be done with either a prefab that simply shows up on screen or I can use particle effects.

Any time the cube bounces off the walls or lands, it leaves particles as if it broke a piece of the infrastructure. It also has sound effects when it jumps up, when it lands, when it shoots upwards, and of course if it falls into to void.

When the boost icon is activated, the colours are inverted, the score takes up a large part of the screen as it pings, and multiple shapes take up space around the cube to indicate the something is happening while also shooting the cube upwards as it moves towards the centre of the screen. TextMeshPro game objects would need to be used to update the score in real-time as well as to keep track of the highscore.

Development Timeline

The game will need to be developed in small sections using the spiral model and will include the following, with persistent debugging through the entire process:

- Jump and land functions
- Spawn and despawn platforms
- Score system
- Title screen and game screen activation and deactivation
- Camera movement and reset
- Artefact spawn and despawn
- Colour overlay with buttons
- Music management

Artefact spawning, colours, and music are the last to be implemented and should be done at the end of the development cycle.

References

Piejko. P. (18 Noc 2016) *720×1280 is the most common mobile screen resolution in Q3 2016 (new report).* mobiForge. [online] Available at: https://mobiforge.com/news-comment/720x1280-is-the-most-common-mobile-screen-resolution-in-q3-2016-new-report

mediaGenesis (25 Mar 2018) *Popular screen resolutions: Designing for all.* [online] Available at: https://mediag.com/blog/popular-screen-resolutions-designing-for-all/