TEST PLAN

**Project Name:** Megaphone Jump

**Product Release Version:** 1.0

**Prepared By:** Stoyan Rizov

**Introduction:** To test if It is possible to create a game using microphone input to control the force of the player based on the loudness of the microphone.

I carried out tests to try and make it work.

**References:**

<https://stackoverflow.com/questions/2853413/getting-data-from-a-microphone-in-c-sharp.com>

**Features to be Tested:**

Player character jumping from platform to platform.

**Features Not to Be Tested:**

Accelerometer is not be tested because it has nothing to do with the jumping force of the character as it is only used to control the player from going side to side not jumping.

**Approach:**

The approach that I will be having is to add an if statement in the code and find the right amount of microphone volume when a player is saying jump in my case, I came up with from 10 to 50, 50 being for the reason so that the player cannot jump too high from a single jump.

**Item Pass/Fail Criteria:**

The item failed in this situation as the way to prove it was by saying to the microphone jump twice let’s say which means you would have to say the word jump and then by the time you say jump again the volume would decrease but if you say the word jump twice and quick the play will continue on jumping therefore it was a fail.

**Test Environment:**

The current test environment was android (debugging, USB connected) and unity game development environment. The test was carried out on my personal phone which was OnePlus 3T and family phone Google pixel 2 XL. Results were not any different

**Estimate:**

Each test takes different time depending on the complexity of the code etc in this test case with the jumping it took roughly 2 – 3 hours researching if possible, solutions and try and error.

**Risks:**

The risk of course of running this test or the whole game is the fact of something being impossible just like in this situation, microphone being an input is very difficult thing to implement into a unity game having into mind that you cannot also use any sounds to do with your game development which limits you a lot.

**Conclusion:**

Even though the whole testing and development was not 100% successful I still got to learn a lot about microphone input , different functions , how everything works and general knowledge about testing , debugging which is a great experience.