

Design Document Outline

1 Executive Summary

- Imagine a world where everyday objects and tools held memories, desires, and music experiences. What if you could go inside? Or the other way around? They're invisible to most, but you unwittingly take a bite into the magic muffin. Your environment becomes weird in appearance, as the magic muffin grants you the ability to become one with objects and experience the music hidden inside. When you pick up an object, you and the environment are taken over by its essence, where you are compelled to move your controllers and head to the beat of the music. As you conquer the challenges presented by these objects, the muffin room starts to get back to normal. You must complete all the challenges in order to survive and win. As an action/rhythm game with creative visuals and an element of exploration that is not concretely based in reality, the game is suited for a wide range of users, ranging from gamers to those who just want to see the visualizations. Our target audience mainly consists of users between teens and 30s, with some interest in music games and visual experiences. The game will procedurally generate the music visualization gameplay, and rely heavily on intuitive signalling that allows users to make sense of the game with little instruction. We are scheduled to develop this game in the month of April, with alpha demo in early April, beta in mid April, and the final release in early May.

2 GAME Design - CREATIVE

2.1 High Concept

An interactive music-driven game, where players experience a variety of environments and play multiple modes of interaction inspired by different genres of music!

2.2 Design Goals

2.2.1 Main Design Features

What is it like to play the game, including:

2.2.1.1 Player goals and objectives.

Main goal: To get back to a normal state without losing your mind!

Secondary goals: To complete individual challenges with high accuracy

Type of challenge(s):

The specifics of each challenge will depend on the particular 'music-room', but challenges will generally involve performing some movement in a specified order, determined by qualities of the music such as beat.

Current room challenges are:

- Smash certain bottles to the beat of the music
- Hit certain light-up targets to the beat of the music
- Pop certain bubbles to the beat of the music

Type of conflict(s):

The main conflict is between the player and the music - in each music-room, the player must perform a specified action according to the tempo and beat of the music, which will progress in difficulty as time goes on. The movements themselves are simple, focusing the conflict on performing actions accurately with high precision.

The secondary conflict is the player's struggle to understand the progression of the game. We want our game to have a slightly mysterious and explorative element in which the user feels like they are uncovering some backstory to understand what has happened to them and how to improve their situation. Initially, when the player enters the Magic Room (to be expanded upon later in this document), they should be confused about why their environment has changed and what they are supposed to do next. After some initial exploration, their situation will become more clear.

Winning condition – how does the player win the game.

Player wins by completing all tasks found in each of the music-rooms to a certain high level of accuracy

2.2.1.2 Main rules and procedures

Operational rules – Player enters music-rooms by picking up certain highlighted objects that transport them to the room. Within the room, the player performs a specified action to the beat of music. The player will be guided to the specific required action at the start of their time in the music-room, and the music will only start once the player has demonstrated an understanding of what they must do. Once the song(s) are over, the player will be returned to the Magic Room, where (if they have completed the task well) the Room will have partially returned to normalcy. The player will then select another highlighted object, continuing until they have either returned the room entirely to normalcy or until they have faded into darkness.

Main game mechanic: The user's main mechanic is their synchronized-task performance, when they are in the music room. They will use the hand controllers to interact with some highlighted objects according to a beat. This is fun because it is similar to dancing, or a game like Bop-It, where the combination of moving to the music and doing something repetitive creates a fun hype environment.

The user also gets to interact with the system in an explorative way through their interaction with the portal objects. The user will be able to walk closer to objects, reach in the object's direction, pick them up, and manipulate them slightly in their hands.

2.2.1.3 Player Resources

Player has "health" that can either be damaged or restored depending on their performance. Health is indicated to the player by their display in the Magic Room - as health goes down, their vision becomes blurred and darker at the edges, the room becomes more unsettling, as it improves their vision clears and the room improves. In each music-room game, the player also has an accuracy measure based on how many objects they successfully interact with. If their accuracy drops below a certain threshold after a certain number of objects have passed, they will be 'kicked out' of the room and taken back to the Magic Room, with their health depleted. As a stretch goal, the player can also earn bonus points / special health improvements based on performing particularly well in a game.

2.2.1.4 Boundaries and Constraints

The two main rooms (Muffin Room & Magic Room) will be designed so the user has just enough space to physically walk to anywhere in the environment. In the music-rooms, the user will not be able to walk around and instead will interact with the room through their performance of the specified action.

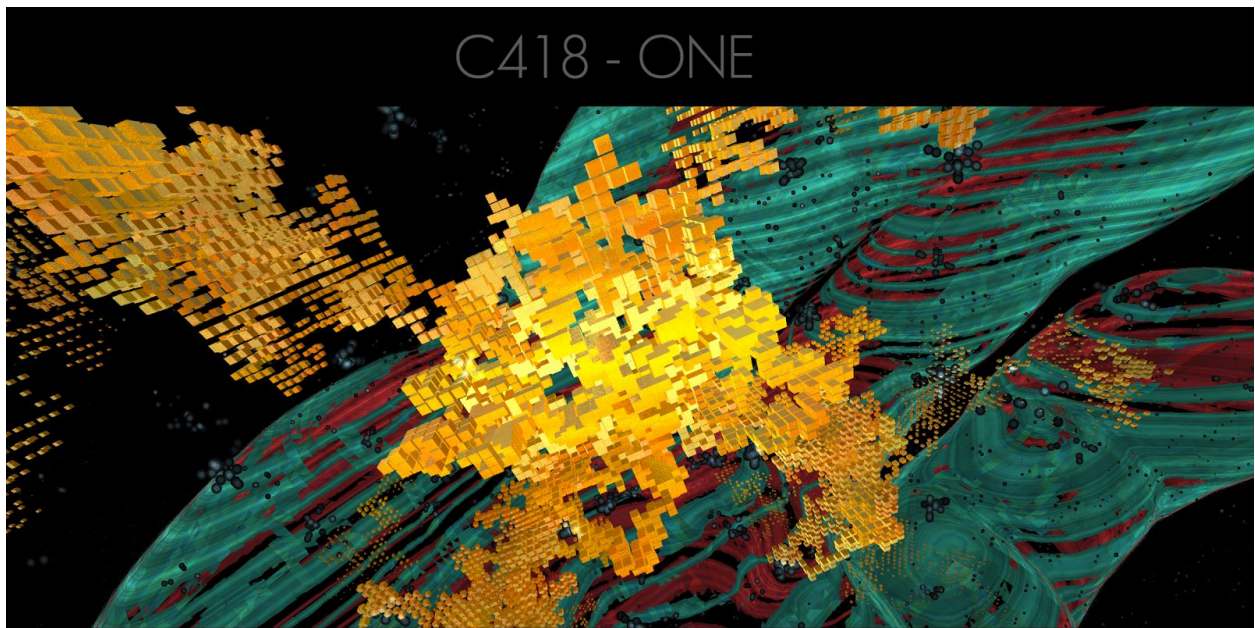
2.2.2 Appeal

Genre would be action/rhythm. We are hoping to target both gamers looking for a fun time and non-gamers looking for an artistic/engaging visual music experience. Our target age group is between 6 years and 89 years old. They need to physically be able to put on a HTC Vive and walk around. This game blends fun music-driven gameplay with creative aesthetics.

2.2.3 Look and Feel

It would be great to really abstract away from a classic game world and build an amorphous environment where interaction is engaging but unexpected.

One major influence is the artist/ director [cabbibo](http://cabbibo.com/). He's made a number of crazy procedurally generated worlds using threejs (a few even support the HTC Vive). Here's another reference link <http://toolofna.com/#!/director/cabbibo>



<http://cabbibo.com/c418/>



<http://void-ii.hi-res.net/room/urban-jungle>



<https://youtu.be/q4uZ5H0IT4Y?t=56>

2.3 Worlds, Characters and Story (if relevant)

2.3.1 Back Story

You are a normal person who enters a mysterious and simple room with a delicious-smelling muffin and a clear command: Eat me! Curious, and (to be honest) a little bored with your daily life, you decide to listen. Little did you know the adventure that would commence!

2.3.2 Spaces/Worlds

1. Classical - making whimsical bubbles and sparkles
2. Electronic - colorful lights / tiles to hit to the beat of the music
3. Heavy Metal - Smash things / people and they break

2.3.3 Characters

Player - You!. You are curious, explorative, and bored by your previous surroundings. You are delighted to explore your new worlds. Player character would be represented in-game as some kind of a floating entity (ghost, or just floating arms/orbs), in order to avoid visualizing the lower half of the body.

2.3.4 Levels of Difficulty

Speed of required movements and the complexity of songs, which may increase depending on performance.

2.4 Interaction Models

2.4.1 VR Interface - Navigation and Movement/Control

1. Super man / balloon: The user can generate a large bundle of balloons from one of the controllers, and holding it up makes the user float. The user can point the controller in the direction they want to move, and releasing the button releases the balloons, which rise for a bit and fade.
2. Dissociation: the user can separate the camera from the character, and the user is essentially a floating camera, with the character in front of it. While floating, the camera could maybe move up and down very gently and slowly to indicate to the user that they are floating. The floating camera should be given a different perspective (wider angle of view? go very high up, like flying?) so that the world appears different and movement is less apparent. The user is now free to move around, and the acceleration of the movement will be made smooth to prevent sickness. Once a user decides on a location,

they can land and return to their body, which is now at the new location. This may work without sickness because vehicles and flight seem to work, all of which have distinct kind of movements that the user can make sense of.

3. Chess: Maybe the user, once in the dissociated view mentioned in the previous idea, could choose the new location by grabbing and placing their body there—this could be interesting for other uses as well, like changing things on the character. Navigating their character from a bird's eye view could also be interesting. It could work by having a minimap-type board attached to a controller, which the user can bring up and move the character around in. While the user moves the character around, the user will be able to see a preview of the character's view, with most of the surroundings blurred out or remaining still. The transition in and out of the character first person would be a quick swoop/dive/warp or preview immersion.
4. Preview immersion: the user is able to manipulate an orb or a camera (from the perspective of the character), and is shown a live preview of what it will look like when the user teleports to that spot. It could be a floating screen above the horizon, or some other way that does not obstruct the view too much. Once the user confirms, it expands and surrounds the user, who is now at the new location.

2.4.2 Game Play Sequence and Levels

Muffin room (game lobby) that starts as ordinary, and you start the game by eating the muffin. After eating the muffin, the room begins to change (not moving to a new location) and reveal a different environment, from which the user can enter a game in one of the three worlds.

The post-muffin room intuitively guides the user to start the first game, and each of the three sides of the room correspond to a world. The user enters a game by picking up a tool, which intuitively informs the user what the user will need to do in the game. The room becomes progressively more normal as the player completes the tasks and return to the muffin room.

2.4.3 User/Environment – Obstacles and Props

The user interacts with each environment in a similar way: hitting objects to a beat. These objects can be bubbles and flowers (classical music), people or breakable objects (heavy metal), abstract shapes and colorful lights (electronic).

Stretch goal: additional gestures, i.e. following a sequence of gestures/motions or drawing shapes.

2.4.4 User/Character

N/A

2.4.5 Character/Character

N/A

2.4.6 Motion Tracking

Motion tracking will be used in gameplay, as well as in the muffin room. The player's motion will affect the appearance of objects and the environment, as if it is responding to the user. In gameplay, motion tracking will be used to check whether the user has hit the right object or completed the right gesture.

2.4.7 Multi-Player, 2.4.8 Mobile, 2.4.9 Networked Play

N/A

2.5 Performance and Scoring

2.5.1 State Variables

What are all the character, environment and gameplay variables necessary to save/restore or pause/resume the game or virtual world experience.

1. Game state
 - a. Current score
 - b. Success/fail for each of the three games

2.5.2 Feedback

What are the positive and negative feedback mechanisms you plan to employ and how do they affect the game play

Positive and negative feedback will mostly involve visual signs and haptic feedback. Red/shaking motion, score dropping, etc to indicate that something is wrong, and other types of visuals to indicate that the user did something right. One major visual sign will be the blurring of vision as the user's health falls, and the improvement of environment as they progress successfully through the game.

Stretch goal: manipulating the song, i.e. if a user does not perform the right action, the music distorts or omits certain sounds.

2.5.3 Performance and Progress Metrics

How will you monitor player progress. How do you win? How do you lose?

Win by winning more games than losing, i.e. 2 out of 3. Lose otherwise. The

3 Game Design - Implementation Details

3.1 Design Assumptions

3.1.1 Hardware

HTC Vive

3.1.2 Software

Unreal Engine 4 or WebVR.

The upside of WebVR would be the reduced complexity when it comes to procedural work and collaborating on multiple computers. The upside of unreal is that it is a little more plug and play. We'd like your opinion on this.

3.1.3 Algorithms and Techniques

1. Collision detection for hitting objects with the controller
2. Translating MIDI / MP3 / streamed music data to object spawns/movements... I'd imagine doing this by creating a short random pattern that coincides with the beats in a song and using that repeatedly, and adding other things here and there based on volume, among other factors that indicate the intensity of a song at that time.
3. Changing the visual appearance of the walls and the visibility of objects based on game state.
4. Making the environment respond (in intentionally weird ways) to and encourage user movement and exploration.

3.2 Storyboards

Show storyboard sketches of your game environment and play sequences here. This should convey the look and feel of the game as well as illustrate the game play.

3.3 Design Logic

3.3.1 FSM - State/Effect

[Enter Muffin Room] → [Pick up the Muffin] → [Room warps, transported to Magic Room version of environment] → [Certain objects highlighted, glow as you walk closer / controllers move in direction] → [Pick up object] → [Vision fades out / fades in, transported to object-specific

music-room] → [First task object highlighted, pulses with urgency as time goes on until player completes first action] → [Second task object does same] → [Music begins and game begins]

[Game start] → [Accuracy drops below threshold] → [Transported back to Magic Room, vision distorted] → [Repeat process from room-exploration]

[Game start] → [Completed with high accuracy] → [Transported back to Magic Room, room restored where object was] → [Repeat process from room-exploration]

[All tasks completed] → [Room restored] → [You Win!]

[Health drops to zero] → [You Lose!]

3.3.2 User Solution/Actions

Once the player picks up a tool, it will become apparent what the user needs to do. For example, when the user picks up a bat, the game will spawn objects with visual indication to show that they can interact with those objects, i.e. break them. The user will have the chance to complete a few actions with positive feedback before the music starts and the game actually begins.

3.4 Software Versions

3.4.1 Alpha Version Features (vertical slice through total experience)

The alpha version(s) represents the first time the game is “playable”. List the complete set of features to be included in the alpha version(s).

Picking up an object to enter a game, and proof-of-concept for the gameplay, where the user can interact with the spawned objects with the object in hand.

3.4.2 Beta Version Features

List the complete set of features to be included in the beta version

4 Work Plan

4.1 Tasks

List ALL the tasks and subtasks necessary to build the game application.

Provide separate descriptions of each task and subtask, which members of the group are assigned to it, and the expected task duration.

Muffin Room: 0.5 weeks

- Create room / assets (Davin)
- Game play actions for muffin (Luke)
 - Pick up
 - Transport

Magic Room: 1 week

- Create discrete distorted environment assets (Davin)
- Create object assets (Davin)
- Room interaction (Anosha)
 - Objects glow, etc when closer to them
 - Transport upon picking up objects

Games (3 types): ~1.5 weeks

- Audio: Import audio files and extract relevant data (Anosha)
- Environment:
 - Procedurally generate objects based on audio data (MinJae lead, others assist)
 - Add other assets, flourishes, etc to create visually pleasing environments (Davin)
- Game logic
 - Instruction at start (Anosha)
 - Take player position and calculate accuracy of movement (Luke)
 - Game difficulty changes (MinJae)
 - Win/lose conditions (Luke)
 - Exit game UI (Davin)

4.2 Milestones

4.2.1 Minor

Describe the functionality you plan to achieve (and will be able to demonstrate) at the end of each week of development

4.2.2 Major

Describe the major milestones of the project.

Alpha Version

Focus of this milestone is on having one magic-room with basic functionality complete and playable. Extracting information from audio files, game logic, getting user motions should be working. Game should be testable in VR.

Beta Versions

Describe the functionality you plan to achieve and demonstrate in the beta1 and beta2 versions.

Beta 1:

Focus of this milestone is generating procedural assets, completing secondary music-room functionality, and adding a basic transition from the Magic Room to the music-room. Game should be testable in VR.

Beta 2:

Focus of this milestone is on the Magic-Room - linking health to the room appearance, adding object assets, room navigation, and object manipulation. After this beta game functionality should be done with refining appearance + perhaps updating game-play actions.

4.3 Development Schedule

Organize your work plan tasks in some kind of readable format and attach it to this document.