The Loft

An Interactive Qualifying Project
submitted to the Faculty of
WORCESTER POLYTECHNIC INSTITUTE
in partial fulfillment of the requirements for the
degree of Bachelor of Science

by
Joseph Acheson
Nathan Bryant
Andrew Busch
Ross Foley
Andrew Han
Sean MacEachern

Date:
30 April 2015

Report Submitted to:

V.J. Manzo
Worcester Polytechnic Institute

Contents

List of Figures

Abstract

The goal of this project was to prototype an application capable of facilitating social interactions between musicians and their fans through an immersive, interactive, computer environment. Once the prototype was developed, we surveyed a group of musicians in order to obtain feedback as to the viability of this application as an effective mechanism for artist-to-fan interaction.

Executive Summary

Introduction

For many musicians, sharing their creations with the world is a major goal. However, existing online applications for sharing music, like Spotify and SoundCloud, lack the depth of interactivity that a lot of artists want. To some artists, sharing music means more than just reaching as many listeners as possible. It's also about getting a deeper level of interaction with their listeners, being able to talk about their music, and experiencing others' music. Normally, artists spread their music through travel, performances, and word of mouth, and while there are online equivalents to purchasing music, there is an opportunity to create a digital version of the musical performance, or for interactive discussion of music. The aim of this project, named *The Loft*, was to create an avenue for artists to share and discuss their music with listeners and fans in a virtual space. We wanted *The Loft* to be easy to use for listeners to find new music and allow artists to connect with fans in a simple and satisfying manner.

Background

Before designing our solution, we decided to research games and applications that have attempted some of the ideas we wanted with *The Loft*. During this stage, we found a couple applications that attempted a similar goal to what we had in mind for our project. In addition, we analyzed features that other games implemented that could be effective in our game. These features ranged from creating and sharing user content, to how to allow communities to form in-game. From our research a clear plan of what *The Loft* would become was created.

Our research led to six major categories of features that we considered necessary for *The Loft* to succeed. These features were social aspects (such as room chats), being able to find new music, integration with existing music-sharing applications, room and character customization, acceptable

network connectivity, and instanced rooms. Studying these features gave the project a systematic design that could easily be followed.

Methodology

With the features necessary to make an interactive music sharing platform successful, we set forth on implementing them from scratch. We found the best way to create the game was by using two separate servers and the Unity game engine. One server would handle all player interactions happening live in-game, like chatting and moving, while the other would handle the 'state' of the game. The 'state' would include aspects of the game like rooms and friends lists. Unity would be where the actual game was built. Everything the player used and interacted with would be through the Unity engine.

The survey focused on the two major groups that *The Loft* was designed for: musicians looking to share and interact with listeners, as well as listeners looking to find new music and fans of similar artists. The questions focused around gaining insight into how usable *The Loft* was, and how effective it was for sharing, finding, and discussing music. Unfortunately, certain aspects of the data are muddled by the small number of participants in the survey.

Data and Analysis

The data clearly showed that listeners found it easy to find new music and artists they enjoyed. This is a huge success for our project. Allowing users to find enjoyable music is essential to making *The Loft* achieve its goal. However, users reported a difficulty with finding fans of similar music, and musicians also found it difficult to discuss their music with fans. This is most likely due to a lack of users online at any given moment. Without a decent user base, testing a user's options to interact with others will be difficult to test.

Conclusions and Recommendations

The general response we received from the survey was that we had a good start, but some bugs needed to be worked out, and some interactivity needed to be added to the game. From the survey and our own experience, we came up with what the next steps should be in the continued development of *The Loft*. Foremost is to implement room and character customization – the main feature cut due to time constraints – which would allow for a more personalized experience. One of the most popular suggestions from the survey is to overhaul the user interface, since it confused many users. Finally, more user testing should be done to assess the efficacy of our prototype.

1 Introduction

For musicians, sharing their creations with the world and interacting with their fans is a major goal, which provides an opportunity for us to create a new virtual space for musicians to interact with their fans. Current music distribution systems, like SoundCloud and Spotify, tend to focus solely on spreading the music itself. For some artists, creating music means more than just reaching as many listeners as possible. It's also about getting a deeper level of interaction with their listeners, being able to talk about their music, and experiencing others' music. In real life, this is typically accomplished by going out and performing in a club or at a concert and then being able to chat with their fans afterwards. While this is a great way to interact with listeners, it's limited by the constraints of real life. Musicians can't always find a place to perform whenever they want, they can't necessarily chat as they're performing, and there's only so much time to chat before these musicians have to pack up their gear and head home.

The aim of this project was to create an avenue for artists to share and discuss their music in a virtual space with anyone at any time. We planned to meet these goals and solve these problems with a social expansion to the game *Before Heaven*, titled *The Loft*. *Before Heaven* is an existing game project that teaches music techniques and theory to the player through a role-playing video game. We decided on an online video game for the above problem because they are valuable outlets for connecting people. Video games break down geographical barriers, so people across the world can all play together at the same time. The massive multiplayer online role playing game *World of Warcraft*, at its height of popularity, had roughly 12 million players [5]. *World of Warcraft* achieved these numbers by creating an immersive interactive fantasy world for players to adventure through and experience with others. These numbers prove that interactive online video games can be used to reach a large number of people to build a community. With *The Loft* drawing on how other successful games handled interactive worlds, musicians could have access to millions of potential fans.

The Loft is a building that houses both bands and music fans. The game allows music fans

to create a character and visit band rooms in the building with the functionality implemented to listen or chat about their favorite music. The idea with playing as a listener is to discover new bands/groups in a fun and simple way. Bands and music groups will be able to create a room in the building to play their music and attract new fans. Through our game, players are able to find new music or share their own to garner new fans in new and exciting ways.

Currently, a prototype of *The Loft* has been completed and tested with a group of students. We conducted testing of both types of players of our game—musicians and listeners—and were able to retrieve insightful comments for analyzing our game. With our results, we were able to conclude that we achieved our goals with the prototype, but there is a lot of work that can be done in the future to improve the game.

2 Background

Currently, artists have some outlets for their music online. There are many applications that allow musicians to publish their music and connect with fans. However, we believe these outlets do not get at an important component of an online music-sharing application—community. Traditionally, artists experience community by sharing their music live with their fans in ways such as playing in clubs and performing at concerts. Our goal was to bring the closeness of the live performance to a digital space.

Bringing the closeness of a concert to the Internet is helpful for both musicians and fans because the Internet allows for greater convenience in many ways. For one, it's easier to download an executable and upload songs than to set up a venue. Also, the "always on" nature of the Internet means that users can take part at any time no matter where they are. Giving artists an online experience that rivals a real performance lets them put on a show without needing a gig. Artists currently utilize social media to connect with their fans, which has proven effective [7]. However, social media lacks the more intimate community interaction that fans and artists could benefit from. We wanted to combine the experience of connecting with fans and sharing music into one cohesive and interactive platform.

Before designing *The Loft*, our group researched many games and applications to develop a firm understanding of which features would best allow us to achieve our goals (Appendix B). We constructed a list of ideas that could be relevant to allowing users to share and listen to music, and then found games that implemented these types of features. To research these games, we played the ones that we had available and found videos of the ones we did not. We also researched games that revolved around sharing music to see how games similar to our own dealt with issues we might face. These aspects are key to making *The Loft* a successful platform for sharing and discovering music.

2.1 Previous Attempts

We looked at many different music-sharing applications, but decided to focus on two games that we considered to be most similar to what we wanted to implement with *The Loft*. Analyzing these games was valuable since they demonstrated to us what has been effective, as well as mistakes that were made. Using the information gained from the failures and successes of these projects helped give our design of *The Loft* direction and ensure the success of the project.

2.1.1 Plug.dj

Plug.dj is an online community based around streaming music in the company of other users. This online service allows users to join a specific room and listen to music streamed from the web. Users have access to avatars, YouTube, and can log in with a Facebook, Google, or registered account. The system also has a built-in waitlist system for anyone wanting to contribute a song of their own choosing to the stream. A community leader creates rooms and has the authority to kick listeners or promote them to moderators in addition to skipping songs that aren't generally accepted by the room or are inappropriate. Leaders can shift certain users to the top of the waitlist. *Plug.dj* encourages community by broadcasting a plethora of streaming zones for users to go and participate in. Rooms also offer a chatroom so that users can communicate with one another. Users are also given a voting system to show favor towards a song or vote to skip the current song in favor of another. The audio streaming service provided is fairly consistent and users who are joining in can hear the song at that specific moment upon entrance rather than be forced to wait until the next song is synced.

However, this also comes with inherent flaws. By giving moderators and community leaders the power to kick and ban players from streams offers potential for power abuse. Also, given that the streaming service is derived from YouTube's API (Application Program Interface), any video, regardless of its content whether good or inappropriate, can be subjected to streaming. There is also a distinct lack of character customization as well seeing as users can only choose from a predetermined array of possible avatars rather than customize their own appearance. *Plug.dj*

implements many of the social features that we have included and considered for the future with *The Loft*, so looking at the problems it faces in terms of sharing content is important for moving forward with our project since they may need to be addressed in *The Loft*.

2.1.2 Coke Music/MyCoke

Coke Music/MyCoke was a social game launched by Coca-Cola in 2002 [2]. The focus of the game was decorating rooms and playing with friends, but what made it unique was the music mixing feature. Players could mix music in-game and share it with friends and other users. *Coke Music* was ultimately shut down in 2010, after years of decreasing number of frequent players.

Though the game was popular for several years, Coca-Cola discontinued it for several reasons. *Coke Music* was not able to stand out in a crowd of other similar social games, such as *Habbo Hotel* and *Club Penguin*. Additionally, the music mixing feature was limited; it was only able to mix songs that were in the game and users were not allowed to upload their own music. The game also had very limited character customization and too few things to do. Since this game is similar to what we have implemented, it is valuable to analyze its successes and mistakes and keep them in mind with our project.

2.2 Features

The common thread between the games that our group researched is the different social features they all include. To properly use the knowledge gained from researching these games, their features were categorized and analyzed to determine what each game did well. This feature analysis allows our group to decide how to implement these relevant parts of *The Loft* in the most informed way possible.

2.2.1 Character and Room Designers

Two other important features are the character creator and the room designer. Some inspiration was drawn from *The Sims*, *Animal Crossing* and *Coke Music/Habbo*, since all of these games

have both of these features. The most important features of character and room designers are organization and customization.

The player can access the room creator in-game with an edit button that will provide access to various options. They can open a catalog that will contain all the furniture that they can place in their room, as seen in *The Sims*, and will be able to place it anywhere in the room, unlike *Animal Crossing* which forces the player to use their avatar to manipulate objects. Each room in the game will be the same size and the player will not have the option to change the dimension or shape of their room, similar to how *Animal Crossing* implements its houses.

The interface should be organized in a hierarchical, tree-like fashion, which will allow the user to more easily locate what they want. Each of the categories in this tree will contain various options, such as different hair styles for the player to choose from if they choose the Hair option from the Head category. Although we want our character creator to be relatively simple, we still need to provide enough options so that each avatar will be unique and decrease the chances of duplicate avatars. *The Sims* character creator uses a hierarchical UI to sort the various categories and options that the player has for customizing their character. This creator provides the player plenty of options that allow for unique creations. *The Sims* is a good game to examine when considering these features because creators are part of the core gameplay of both games.

The amount of customization should allow for players to feel like individuals, but not so vast that it would be impractical to implement. Although we want our players to be able to create rooms and avatars that feel unique, we do not need creators as extensive as *The Sims* since these features are not the primary focus of our game. Since our game's focus is the player's bands or music groups, a primary focus of our character/room designer should be the band logo. The player will be able to upload their band logo, and be able to use the design for various items such as posters for their room or T-shirts for their avatar. Additionally, it would be better if the player has access to many items for their room and avatar from the start, that way they do not feel limited, unlike *Animal Crossing* and *Coke Music/Habbo Hotel* where the player must get the items in-game before they can use them.

2.2.2 Social Aspects

Socializing will be a very important part of the user experience within *The Loft*, so the game will require a robust and user-friendly way for users to communicate and maintain relationships. There are four core features to socializing: chats within rooms, private chats, friends lists, and character interactions.

Within rooms there will be a text chat box at the bottom of the screen, displaying what each person says to the room. This is a standard way of handling chat that has been used in games like *World of Warcraft*. Other announcements can be made, like what song is playing, or whose room the players are in. Voice chat is not a particularly valid option, as the game focuses on listening to and creating music. This would curtail communication, as all voices would interrupt music. A text chat allows for conversation to flow while music is playing.

Players will be able to invite one or more players to a private chat that will also be text. Messages in the private chat will not be heard by anyone but the players who have entered the private chat. Here, players will be able to invite others to listen to songs or to join rooms. There will be a "Conversations" button that will bring up all the players currently ongoing private chats.

Each player will be able to keep a friends list that lets them see contacts they have in the game. Players will be able to invite other players to be their friends. *League of Legends* uses a similar system with joining, chatting with, and watching games with friends. Players will be able to bring up their friends list by clicking a "Friends" button that will be near the "Conversations" button. The Friends List is where the player will be able to invite players to private chats, or to various rooms. They may also be able to see the status of their friends, like what room they are in or if they are at a show, and send a request to join them.

Character interactions will allow players to communicate with more than just words, and see each others' sprites as people with identities rather than just graphics. Characters should be able to wave, smile, frown, and other such playful and humanizing actions. There should also be an in-depth dance interaction that allows players to dance with each other and enjoy spending time together. The interface for dancing can be brought up when a player decides to dance. Other interactions should be made easily available on-screen or on the keyboard for players to more easily

express themselves and their feelings through the specific emotes.

2.2.3 Search

Being able to find what you want when you want to find it, or being able to stumble upon something you didn't know existed is an important part of a social game. Some games have good searching capabilities, and others do not. A flimsy search feature can prevent users from finding what they want, so implementing solid search is an important priority.

One game that has a lackluster search feature is *Second Life*. *Second Life* allows a user to create their own territory within the world, and as such, has many places a player can visit. To find new territories, a player can use *Second Life*'s web interface to view a detailed map of the *Second Life* world and receive information about each territory. A player can also search for a specific territory by name, which is good if they know exactly what to look for. Once a player has found a territory that they like, they can save it to quickly visit it again in the future. What *Second Life* isn't very good at is allowing a player to find new places when they don't know what they want to find. There is a browsing feature in-game called "Destinations" which shows a limited selection of territories organized by category, but it is far from exhaustive and only has a dozen or so categories among the hundreds of different types of territories in the game. What *Second Life* is missing is a good way of discovering new places that exist in the game.

A better example of searching is *Little Big Planet*. *Little Big Planet* has a compendium of levels on their website. Players can search for a particular level, creator, or genre, but can also browse the existing levels according to many metrics like Most Liked, Most Played, or Recently Published. Each level also has many labels that tell the player about the level, like its difficulty, time to complete, game type (Platformer, Shooter, Racing, etc.), and so on. Each level has its own page with heaps of information like reviews, comments, photos and recent activity. Additionally, each player also has their own page where other players can see the levels they've created, played, and liked. All of this information creates a dense web of connections between categories, levels and players which makes discovering new levels very easy and intuitive. From a single level, a player can find many other levels just like it with only a few clicks, and can easily save those levels for

later play.

2.2.4 Instanced vs. Non-instanced Property

When a player is given the opportunity to own property of some sort in an online environment among many other players, there is a major issue to deal with: overcrowding in a limited space. The most common solution to this issue used today is instanced housing. In short, each player or group gets their own "instance," a copy of an area created specifically for their own use. This was also the first solution we jumped to when contemplating the issue of a space that would have to accommodate a constantly increasing number of apartments. However, we decided to investigate implementations of non-instanced housing to see how they tackled the issue of overcrowding and what benefits, if any, they offered.

Games like *Star Wars Galaxies* and *ArcheAge* employ several tactics to manage the issue of overcrowding. The open worlds must be designed with plenty of free space to allow for housing. They utilize property tax systems with the in-game currency so that accounts no longer in use eventually free up the land they were occupying. The system of non-instanced housing allows players to take pride in their work, letting everyone see the resources and effort put into their property. It's also convenient to be able to situate yourself near friends and build up a community.

Being a project about music and socialization, there were definitely merits to having some form of a non-instanced system of apartment rooms. For example, a player could tell visitors to check out his or her friend right across the hall, or to put his or her rock-oriented room near other rock artists to promote discovery. However, an instanced system for the rooms could give the same benefits to users, and also reduce the complexity involved with finding a particular room. This system will allow similar opportunities to discover, share with friends, and associate with related artists through other aspects of the game such as the room searching system, the social and chat system, and the implementation of common lobbies and hang-outs for players.

2.2.5 Importing Third-Party Content

There are many approaches to allowing third-party content in video games. Each one has its own benefits and downsides, so choosing the correct approach greatly depends on the game. One approach is to offer in-game tools to create content. For players who are already familiar with how the game operates, this approach may be more familiar to them and can be more deeply integrated with other in-game features. In addition, this approach ensures that the content created can't be abused or exploited. However, these tools often limit the type of content that can be created. Furthermore, it is significantly more time-consuming to create these tools and for the user to recreate content they want inside of these tools. Some games that took this approach include *Animal Crossing: New Leaf* and *Little Big Planet*.

On the opposite end of the spectrum, some games allow users to directly import local content, such as images and music files. This comes with the benefit of being extremely flexible and faster than recreating content. In addition, this allows content to exist that may be nearly impossible to create with in-game tools. However, there are some serious downsides to this approach. The first is that you need to provide the infrastructure to upload and store all of this content. In addition, without sophisticated filtering tools, there is a huge potential for abuse, including importing illegal content. Therefore, only a few games, such as *Team Fortress 2*, opt for this approach.

One final approach is to integrate the game with third-party APIs from Facebook or YouTube, which allows the game to use the data held by these third parties directly. With this approach, you have a new option to authenticate users with existing credentials, such as with Facebook's OAuth 2 API. Furthermore, many users are already familiar with using these third-party services, so it is easier to understand how to use an integrated third-party service than an entirely new one. In addition, these services often filter the content that they provide, reducing the potential for abuse. However, this approach forces users to have many accounts to fully appreciate the game's features. In addition, the game becomes limited by what the third-party service is willing to provide, such as the type of content it provides and how much network traffic it permits. Some of the most well-known games that utilize third-party services include *Farmville* and *Angry Birds*.

3 Methodology

The goal of this project was to create a video game that connects music artists to music fans. The aim is to allow artists to not only share their music with the world, but also give them the opportunity to have meaningful discussions and interactions with people who like their music. Meanwhile, music enthusiasts get the chance to easily discover new music and chat with the minds behind it. The digital nature of this project means that people should be able to do this at any time, from the comfort of their home.

The background research provided ideas for how to make social music games, and what to avoid. With these ideas in mind, we settled on the concept of a virtual building with an endless number of rooms. Users can create rooms in this building and play music from their SoundCloud playlists to all the visitors in the room. Players can chat with each other and add people to their friends list, which allows them to jump immediately to a friend's room. We chose to create our game with two separate servers and the Unity game engine. After developing the prototype, we had fellow students test out the game and fill out a short survey about their experience. This feedback will allow us to make informed suggestions about where this project should head in the future.

3.1 Unity

Unity was chosen as the platform to build the game on for several reasons. The main reason for choosing Unity was that its engine and scripting language provided the means for constructing a working prototype in a short amount of time. Additionally, the project files we received were also developed with Unity so it was natural to build on top of them. Another strong reason that Unity fit the project well was that Unity's engine has built-in networking capabilities which would make creating our social features easy. Also, Unity has various third-party tools and external libraries which we could use to implement our ideas quickly and efficiently.

3.1.1 User Accounts

In the initial conception of the prototype, we needed a way for users to individually represent themselves that both linked their SoundCloud music profiles and their in-game visualization. When a user starts up the application, they are confronted by a login screen where they can either enter their information or click the 'Register' button to create an account. From there, they are led to a website where they can fill in the information necessary for a new account. In addition, players also have the capability to link their SoundCloud profiles to their account should they desire to create a room with their own personally selected songs.

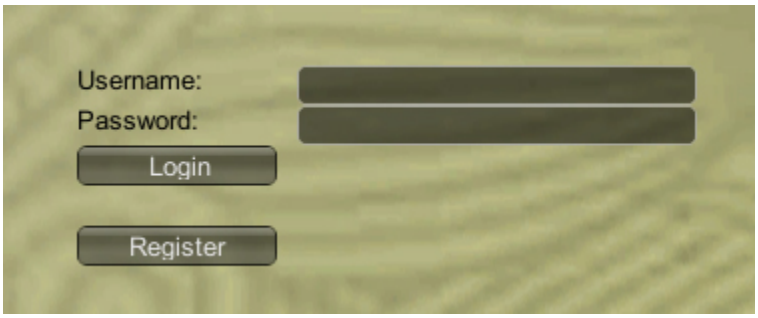


Figure 3.1: User Login Prompt

3.1.2 General User Interface

The User Interface (UI) was designed to allow the user to access all of the features while also being minimally invasive. Elements were placed around the borders of the screen and were separated into easy to understand categories. The largest elements are the chat window and the sound bar. These are placed on the bottom of the screen. The chat window can be minimized to reduce space, while the sound bar is always present so the user always knows what music is playing. Along the top are three buttons: 'Elevator', 'Create Room', and 'Friends List'. These features can be accessed with these buttons, and when clicked, they will open up in a window that overlays the main game screen.

Figure 3.2: General UI Heads-Up Display

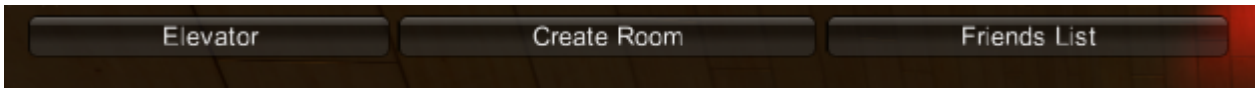


Figure 3.3: General UI Toolbar

3.1.3 Friends List and Chat

When deciding upon player communication within our game, we decided to implement a room chat and friends list. The chat is used to communicate with everyone in the current room so they can talk about the current song that is playing or chat with the creators of the room. The chat box functionality is relatively simple, since players can only send/read messages in it or choose to minimize it.

Figure 3.4: In-Game Chat Interface

One of the major social elements we wanted to have in our game was a friends list, so players can more easily connect to the people that they add. The friends list can be accessed via the top toolbar which brings up its window. In this window, the player can add friends by typing in a username of another player account. Once a friend is added, their username will appear in a list on the right side of the window. The friends appear as a list of buttons that can be clicked to join up with a friend that is in another room. If a friend's button is green, this means that they are currently online and can be joined by clicking the button. When a friend is offline, their button will appear red and clicking it will not bring the player to another room. Finally, friends can be removed by clicking the 'x' button that is to the right of a friend's name.

Figure 3.5: In-Game Friends List

3.1.4 Room Creation

The primary feature that we wanted to implement was rooms, which players would be able to create to share their music. The current implementation of rooms allows players to create, edit and delete their own rooms and to join the rooms of other players. When creating a room, a player needs to provide a name and a genre for room, in addition to a preset. The presets are 'Default Room,' 'Jazz Room,' and 'Cool Room' which are different style rooms that allow some customization. Once the player provides all the necessary information, they can click on 'Submit Room' to make the room public. One of our priorities with room creation was making it easy for the player to get their SoundCloud playlist into their room. As a result, we made it so that as long as a player linked their SoundCloud to their Loft account, any room they create automatically plays one of their playlists.

Figure 3.6: Room Creation Options

The 'Create Room' button in the general UI is also where the player can go to edit their room. When editing the room, they can change the name and genre of the room by typing in new ones, and can add other players as band members to the room. In the current implementation, adding band members gives them the same abilities as the creator of the room. On the right side of this menu, the player can change the playlist that plays in the room, as it lists all the playlists on their SoundCloud as buttons. If the player chooses one of their other playlists, it kicks all players out of the room, changes the playlist, and starts playing the new songs.

Figure 3.7: Room Editing Options

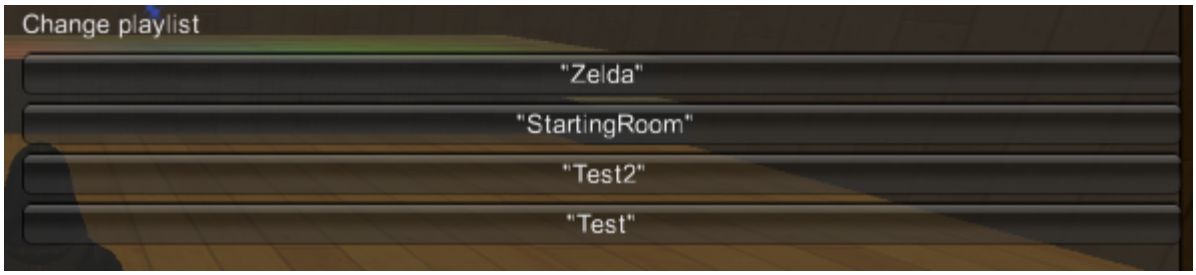


Figure 3.8: Selecting a Playlist

Figure 3.9: Room Configuration

3.1.5 Audio Bar

The audio bar is located at the bottom of the screen whenever the player is in a room. It is used to control various aspects of the audio, such as muting it or changing the volume levels. It also displays information about the current song playing in the room, with the name, genre and duration of the song on display.

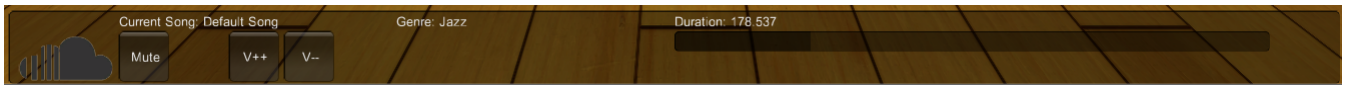


Figure 3.10: Audio Bar

3.1.6 Elevator Menu

The elevator menu is a feature that allows a user to view the rooms created by artists, either by browsing a list of all the rooms, or searching by name or genre. The elevator menu is composed of a search bar on the left and a list of rooms on the right—one for each room—that displays the room's name, genre, number of visits, and a button that allows the player to join that room. The user can use the search bar to narrow the list of rooms down to the desired keywords, and a room can be joined by selecting it. This allows the user to easily navigate between rooms to find the one he or she wants.

Figure 3.11: Elevator Menu

3.2 Server

We utilized two servers to power *The Loft*. The first one is a *Photon* server used to power all real-time events, such as syncing player locations. The second one is a *Ruby on Rails* server that handles all other player data, including SoundCloud integration, syncing music, and storing players' account data.

3.2.1 Ruby on Rails Server

The *Ruby on Rails* server is hosted on *Heroku* and utilizes a *MongoDB* database to store player data. It provides JSON (JavaScript Object Notation) API endpoints that the Unity client can use to obtain specific player data, such as a player's friend list. It also provides a web interface for creating player accounts.

3.2.2 SoundCloud Integration

Through the web interface of the Ruby on Rails server, a player can link their SoundCloud account to their *Loft* account. This linking process is done through the official SoundCloud Developer API and provides the Ruby on Rails server with an access token. This access token is used to authenticate with the SoundCloud Developer API to access information such as playlists and music tracks.

3.2.3 Syncing Music

By keeping track of the last time that a playlist was added or updated, the Ruby on Rails server can determine the current song being played in a room. This is done by computing the difference between the current time and the stored time (i.e. the time the playlist was last updated) and then comparing that length of time to the length of each song in the playlist. To ensure that the Unity client can always access the current song data, a JSON API endpoint is provided that returns the current playlist, song, and offset into the current song.

3.2.4 Storing Player Data

The Ruby on Rails server is responsible for keeping track of all player data that isn't real-time. This includes username, password, friends list, SoundCloud access token, and more. In addition, this server also keeps track of all room data, including name, genre, playlist, and theme. The data are stored in a MongoDB database. This allows for a lot of flexibility in representing data structures on the Unity client, reducing the need to convert between data formats.

Since the Ruby on Rails server stores all the usernames and passwords, it is also responsible for authenticating players from the Unity client through a JSON API endpoint. If the authentication is successful, the Unity client receives a server access token that can be used to authenticate with the server for future requests. Each time that the Unity client requests data through one of the JSON API endpoints, the server verifies the access token before returning the data. This ensures that player and room data remains private and secure.

3.2.5 Photon Server

Photon for Unity 3D was used as the networking framework for *The Loft*. *Photon* is a server-based networking tool that syncs any and all information developers need to be shared across systems in-game. As a tool, *Photon* vastly simplified real time online player actions and music syncing. All player movements and actions are linked using *Photon*, and when a room is updated, such as with a new look and playlist, *Photon* is used to remove players from the room so they may rejoin the newly designed room. *Photon* is also used to sync chat messages between players in the same room.

3.3 Survey

The point of *The Loft* was to give musicians and fans of music a better way to interact and discuss music over the Internet. In order to assess the quality of our prototype, we constructed a survey to get feedback from real users. This user base consisted of friends and fellow students that we reached out to via e-mail. We made it a point to reach out to students who were musicians as well as students who weren't, in order to get perspectives from both of our target audiences. We made two separate surveys—one for artists, and another for listeners. We created stand-alone executables of *The Loft* for Windows, Mac, and Linux to distribute to testers. The executables were hosted using Dropbox while the surveys were created using Google Forms. The e-mail invited students to test our prototype (Appendix A.1), including links to the executables, the surveys, and our website where they could create their account. We also provided step-by-step instructions for getting started with *The Loft*. After sending out the e-mail to friends and the Computer Science,

Music, and Interactive Media & Game Development departments at WPI, we received 6 responses from musicians and 12 responses from listeners. While the number of responses was not very large, the suggestions given by the users were very useful for considering what future work needs to be done.

The questions and sections of the survey were designed to allow testers to estimate how well *The Loft* achieves its goal of creating an interactive space for sharing and discussing music. It was necessary that these questions address the main functionality of *The Loft*. For example, our questions included how easy it was to find others with similar music interests. We needed to know that *The Loft* was an effective social platform with music at its heart. Meanwhile, we also wanted to know whether users thought that *The Loft* was a good platform for finding and experiencing music. This led to questions about how easy it was to find new enjoyable music, and questions about whether the sound quality of the music was good enough for listeners to appreciate what they were hearing. On top of this, we needed users to assess how usable and understandable the interface of *The Loft* is. After assessing the product quantitatively, we asked for suggestions on where to go from the prototype as a means of crowdsourcing ideas for improving *The Loft* since this could provide a solid direction for future work on the projector future work. We added a section for users to give their general feedback and feelings on the project overall in case there was something that users wanted to comment about which was not included in the rest of the survey.

4 Data and Analysis

During the survey process we received 18 total responses from fellow students about our prototype. We had them answer 6 questions about the experience to give us a better idea of where the project should proceed.

4.1 Finding New Artists/Listeners

For listeners, the ease of finding new artists was generally positive. Most users were able to find music they liked. However, for artists, the ability to connect with new users was more mixed. Many artists said they found it easy to connect with fans, but others complained about the lack of listeners in-game. The disparity between listeners and artists is likely caused by the low number of people playing the game at any point. Rooms are permanent, so listeners can find new music as long as the artist has created a room. However, an artist must be online at the same time as a listener in order to connect with them.

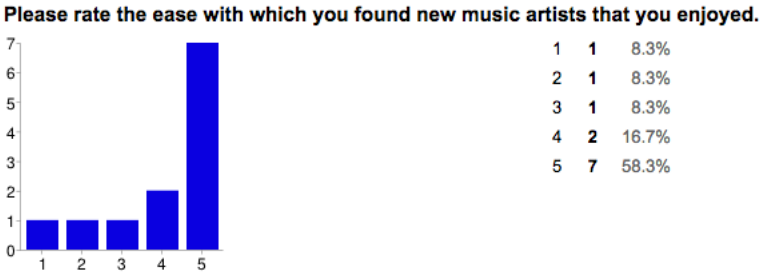


Figure 4.1: Question 1 (Listeners)

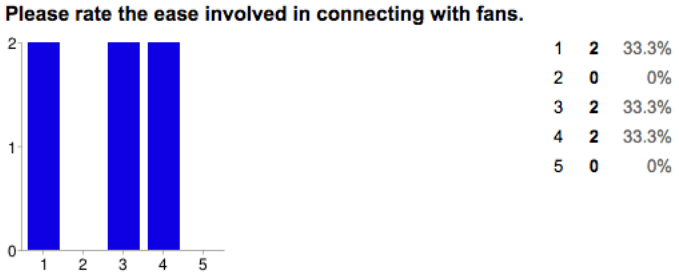


Figure 4.2: Question 1 (Musicians)

4.2 Connecting with Other Users

When we asked listeners to rate how easy it is to connect with other fans of similar music, we got fairly mixed results. The majority of responses to this question were positive or neutral, showing that at least some people were able to find music they were interested in. However, we did receive a fair number of negative responses. Similarly, asking musicians how easy it was to find new music artists that they enjoyed collected a wide range of responses. Some of the written feedback from the survey seems to suggest that, like we mentioned earlier, the people who answered negatively did so because when they tested the prototype, there weren't many other users currently online. Otherwise, those who managed to log on while others were online seemed to have a generally positive experience.

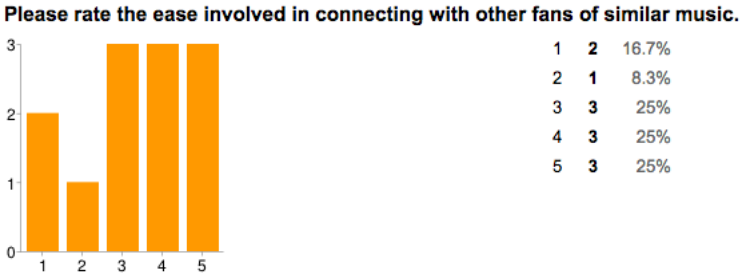


Figure 4.3: Question 2 (Listeners)

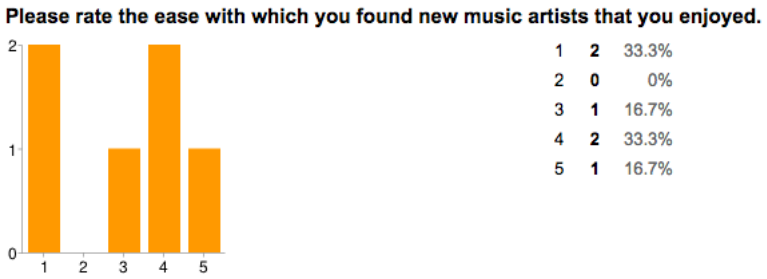


Figure 4.4: Question 2 (Musicians)

4.3 Navigating the Interface

Overall, most users found it relatively easy to navigate the interface of the game. Most users gave the interface a 4 or a 5, signifying that navigating the interface was simple. For the few users that had troubles with the interface, the most common complaint was room customization. One user thought that without the directions included in the survey, they would not have been able to figure out how to edit a room once it was created. Another user commented that "using the create room tab to edit the current room seems unintuitive," which may have been a concern for other musicians as well.

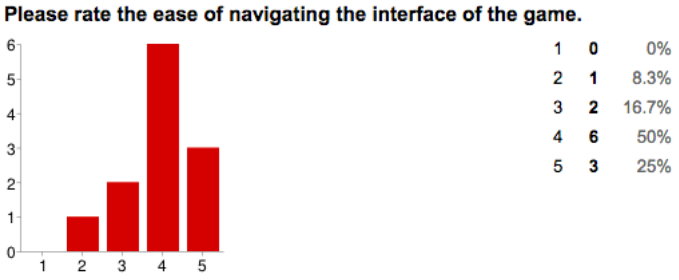


Figure 4.5: Question 3 (Listeners)

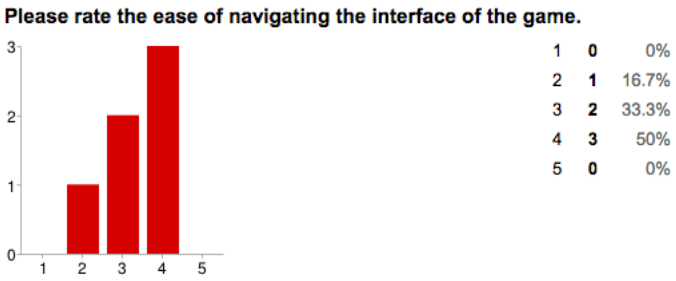


Figure 4.6: Question 3 (Musicians)

4.4 Audio Quality

In terms of audio quality, the responses were mostly positive. One tester mentioned that "the music quality and UI is good and easy to use," which tells us that, despite the flaws in the interface, the music was able to stream fluidly without too many fluctuations in the playback. While both groups generally liked the quality of the audio, listeners had a wider range of ratings than musicians. The sound of the prototype was in most cases judged as adequate; however, some listeners were unable to hear audio.

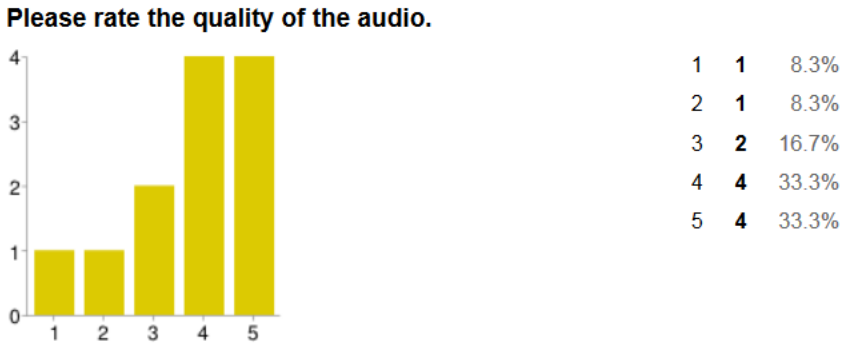


Figure 4.7: Question 4 (Listeners)

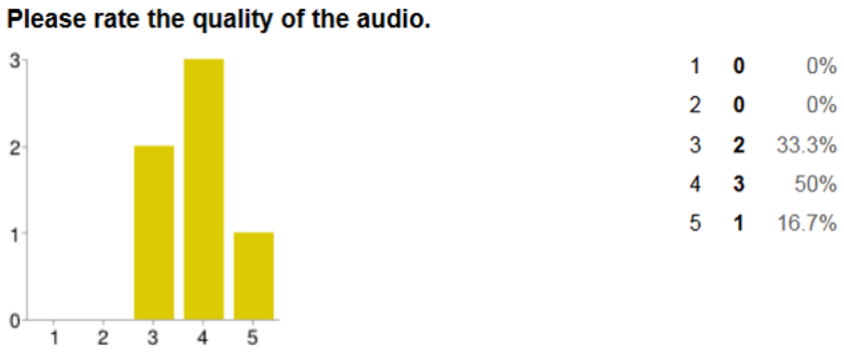


Figure 4.8: Question 4 (Musicians)

4.5 Suggested Improvements

This question asked users (both music creators and listeners) about possible features that could be added to improve the experience, which resulted in some valuable suggestions. A common response among the users was the addition of more customization options, such as furniture for rooms or ways to customize their avatar. This was something that we had wanted to include, so it is interesting to see that some users suggested it as well. Some of these suggestions mentioned rating songs, having a way to skip songs, or vote on which song would play next.

4.6 General Feedback

Most of the responses suggested bug fixes, small UI adjustments, and redesigns that would improve the user experience overall. The responses led us to conclude that the UI could use a general purpose overhaul, to ensure it is easy to understand and use without a large amount of experience dealing with the game. Many people continued to discuss issues with sound not playing. However, in the instances in where the audio did play, the sound quality was reported to be reasonable. Debugging will be needed in order to resolve this issue.

5 Conclusions and Recommendations

The goal of *The Loft* was to create an online environment in which people could find music they enjoy, meet with fans of similar music, and share their own music with others. After researching similar projects' successes and failures, and using data from individuals who used and tested *The Loft*, the following list of recommendations and features was constructed to improve the usability, effectiveness and overall success of *The Loft*. The list covers what the next steps in the project should be for groups continuing the development phase of *The Loft*.

5.1 Friends List and Player Chat

The current state of the friends list is a good start, but there are additional features that can be added to improve its quality. Currently, friend requests are not implemented: if somebody adds another player as a friend, they automatically become friends. Friend requests will improve the user experience by preventing players from having friends added without their consent.

During testing, players would not know if the creator of the room they were in was with them. In response to this, the chat window should be made to display the messages of the room owners in a different color or some other means to make them stand out in the chat.

While testing revealed that players liked the idea of being able to join their friends' current room, we found that players also wanted to know the details of the room that they would be joining. This could be included by expanding the size of the friends list window and including this information next to each friend. This information is necessary to allow players to decide whether or not they would like to join their friend before actually entering the room.

Finally, when planning out the friends list, we had envisioned a private message or chat system where players could send messages to just their friends in-game. We felt that this was not a necessary feature to demonstrate the functionality of the friends list, but would be a feature worth looking into. This would allow players to communicate without being in the same room, and allow players

to invite friends to certain events like concerts or band meetings.

5.2 Customization and Interactivity

The most popular request from the surveys was more customization and interactivity, which would greatly improve what *The Loft* offers to musicians and listeners. Overall, users said they wanted more activities and actions. A good starting point would be character customization with various costumes and faces to choose from and room customization with furniture and other objects that can be placed in the rooms by the owners. One means of increasing player interactivity would be to give users animations that would help them communicate, like dancing or playing along to music. This would give users more interaction with each other, and thus improve the social aspect that *The Loft* hopes to bring to its audience.

Another possibility for interactivity could be in the rooms with the music. One idea suggested by one of our survey-takers was that users could rate songs and all players in a room could suggest a particular song to be played by voting. This would benefit our game since it would allow users to provide feedback to the music creators. Players also wanted a way to skip a certain song to get to the next one. Suggestions mention a voting system where if enough people in the room vote, the song will be skipped. This would most likely be best implemented in moderation, in the same way other streaming services limit the number of times users can skip songs.

5.3 User Interface and Menus

Some users noted that the user interface felt clunky; that it wasn't intuitive to use. Improving the UI and adding new features to improve usability is certainly one of the things we had planned to do with more time. While the general structure of the menus and the placement of objects in these menus is appropriate, there are many improvements we can make. In fact, many of the negative responses from our survey could be addressed with some UI changes. Changes could include both fixing flaws in our existing UI as well as implementing new features to improve the user experience.

A fix we might make would be extracting the room editing tools from the create room menu

and adding a new menu for it. Users seemed to have a hard time finding where to change the properties of their room once creating it since it was in the menu labeled 'Create Room'. Also, the 'Create Room' menu is a bit cluttered, so removing the editing capabilities would improve the menu's organization. Small changes like these would lead to less confusion among players.

Meanwhile, time spent on creating new features could lead to a more positive and fulfilling experience for players. For example, some users felt that it was not easy to find new music artists they enjoyed. In the brainstorming process, we had considered a more sophisticated version of the elevator menu. A smarter elevator menu might make suggestions of rooms to visit based on the rooms a user has visited recently or frequently. Features such as this would improve both the ability to share music interests and user reception to *The Loft*.

It is a necessity that we redesign the user interface to be more usable and understandable. Creating a fluid experience is very important to having a successful and enticing game. Also we need to fix the bugs reported by users in our testing sessions. Some players being unable to hear their own music in their own rooms is unacceptable in a more finalized product.

5.4 Audio Bar

Currently, the audio bar has problems as pointed out by some of the users who took the survey. Users found it confusing that the bar lists the length of the song in seconds. There are also some graphical glitches that can occur on the bar when a song or genre name length is too long, which causes the words to overlap. There is a lot of room left to use in the audio bar, which could be used for rating or voting on songs, since it already displays all of the current information of the song.

5.5 Networking

We currently have multiple players in the same room, synchronized streaming music, and other features functional. However, user-testing has identified areas where we could make improvements. Currently, changing playlists is a bit intrusive on the experience since it kicks all users out of the room, which forces them to rejoin it if they want to listen to the new music. This can be made a

lot more fluid for the users by either having them automatically rejoin or change the transition so that users do not need to be kicked from the room.

Another problem with our current implementation is the lack of status messages. There are many situations where the player should be notified: another player joins or leaves the current room, a friend is added, a room is deleted, etc. Currently we do not tell players any of this information. Giving this information to the player gives them a better idea of what is going on in the game.

5.6 User Accounts

In the current implementation, a room's song list is set to the first available playlist on the owner's SoundCloud profile and plays all the songs from the list in sequential order. The original intent was to allow users to pick which playlists they wanted to upload rather than having to finagle with SoundCloud's interface. However, for the purposes of the prototype, simply getting users to be able to have their playlists uploaded was the primary objective. Future revisions of the project can consider developing an interface for allowing users to allocate specific playlists to the rooms that they create.

5.7 Future Testing

One of the common themes in survey responses was the lack of users online, which we believe led to some of the negative responses we received when we asked testers how easy it was to connect with fans and other users with similar tastes in music. It may help in future testing to create narrower testing dates or times to encourage people to log on at the same time. This will improve their ability to review the social features of *The Loft*.

A Survey Information

A.1 Survey E-mail

Hello everyone,

We are an IQP team developing a multiplayer music-oriented video game. In *The Loft*, musicians can create a room which they can link their music to. Other players can then visit created rooms, listen to the creators music, and discuss the music with everyone else in the room. We are currently in the process of testing our prototype and getting feedback. Please feel free to join us by doing the following:

1. Sign up for an account at http://beforeheaveniqp.herokuapp.com/
2. Download our prototype from one of the following links:
   Windows: https://www.dropbox.com/s/wd15anld1vzutqg/TheLoft-Win.zip?dl=0
   Mac: https://www.dropbox.com/s/tk7s548uq93gb6l/TheLoft-Mac.zip?dl=0
   Linux: https://www.dropbox.com/s/kanq83jx7kuar6q/TheLoft-Linux.zip?dl=0

At this point youre all set to log into the game and visit rooms that other people have created. If youd like to go one step further and create your own rooms to share music youre interested in or created yourself, follow these steps:

1. Create a SoundCloud account if you dont have one already at http://soundcloud.com
2. Create playlists of music you created or are interested in on SoundCloud.
3. Sign into http://beforeheaveniqp.herokuapp.com/ and go to the SoundCloud page. Click the Connect with SoundCloud button, then sign into your SoundCloud account and connect it to the Loft.
4. Now if you create a room, it should automatically play music from your playlists. The create room page also allows you to change which playlist the room plays.

We would appreciate it if you could take 10 minutes or more to try out our prototype. After playing, we would greatly appreciate it if you could fill one or both voluntary anonymous surveys which should take no more than 10 minutes of your time:

Musicians/Room Creators:

https://docs.google.com/forms/d/1HNj468TljtHFAaSQVi4J4-Znstexi0Dyj4npN2TSuA8/viewform

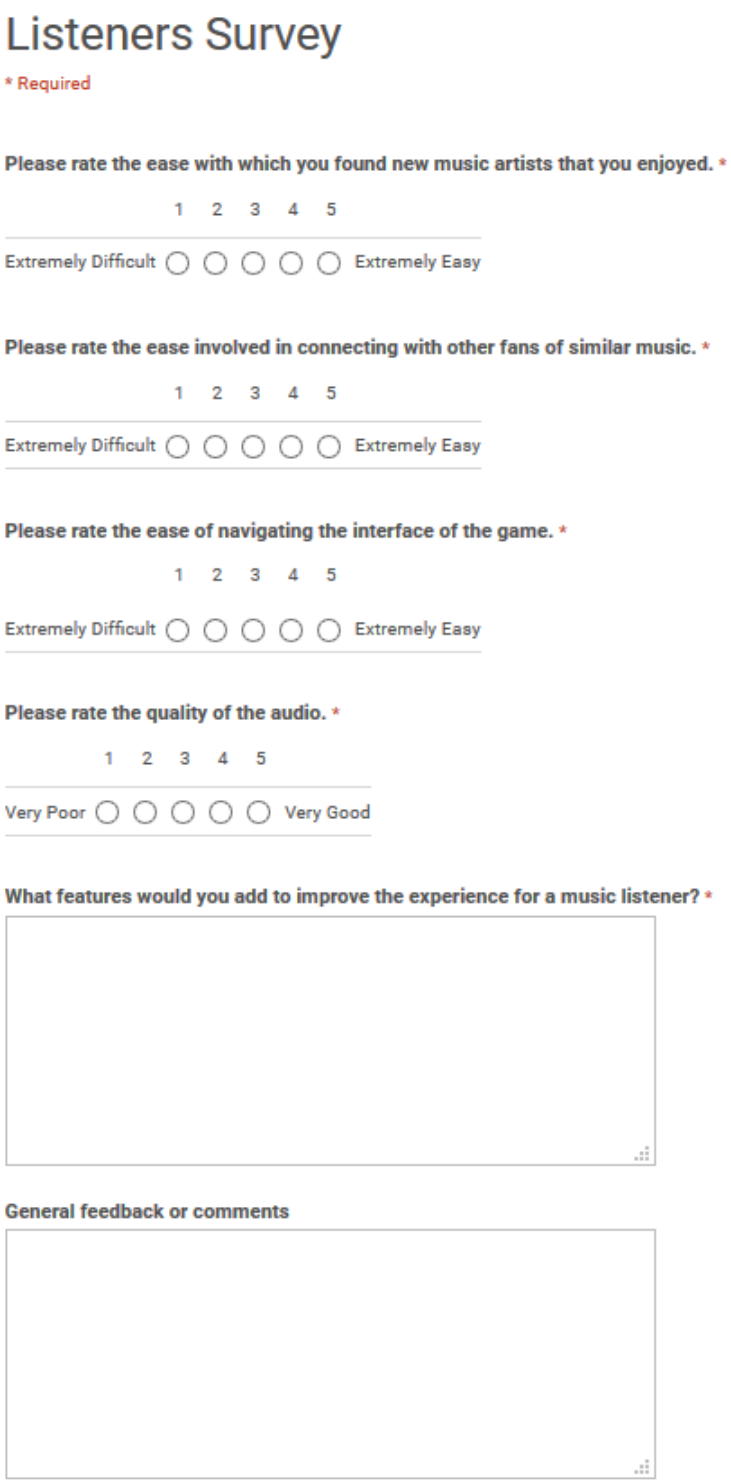
Listeners:

https://docs.google.com/forms/d/1YHIHS7iZLzNHtxSC0uV0s0gAgf1EdcjKvNXBngIJ8CI/viewform

Thanks for your participation,

The Loft Development Team

A.2 Images Of Listener and Musician Surveys



Listeners Survey

* Required

Please rate the ease with which you found new music artists that you enjoyed. *

1 2 3 4 5

Extremely Difficult ◯ ◯ ◯ ◯ ◯ Extremely Easy

Please rate the ease involved in connecting with other fans of similar music. *

1 2 3 4 5

Extremely Difficult ◯ ◯ ◯ ◯ ◯ Extremely Easy

Please rate the ease of navigating the interface of the game. *

1 2 3 4 5

Extremely Difficult ◯ ◯ ◯ ◯ ◯ Extremely Easy

Please rate the quality of the audio. *

1 2 3 4 5

Very Poor ◯ ◯ ◯ ◯ ◯ Very Good

What features would you add to improve the experience for a music listener? *

General feedback or comments

Figure A.1: Listener Survey

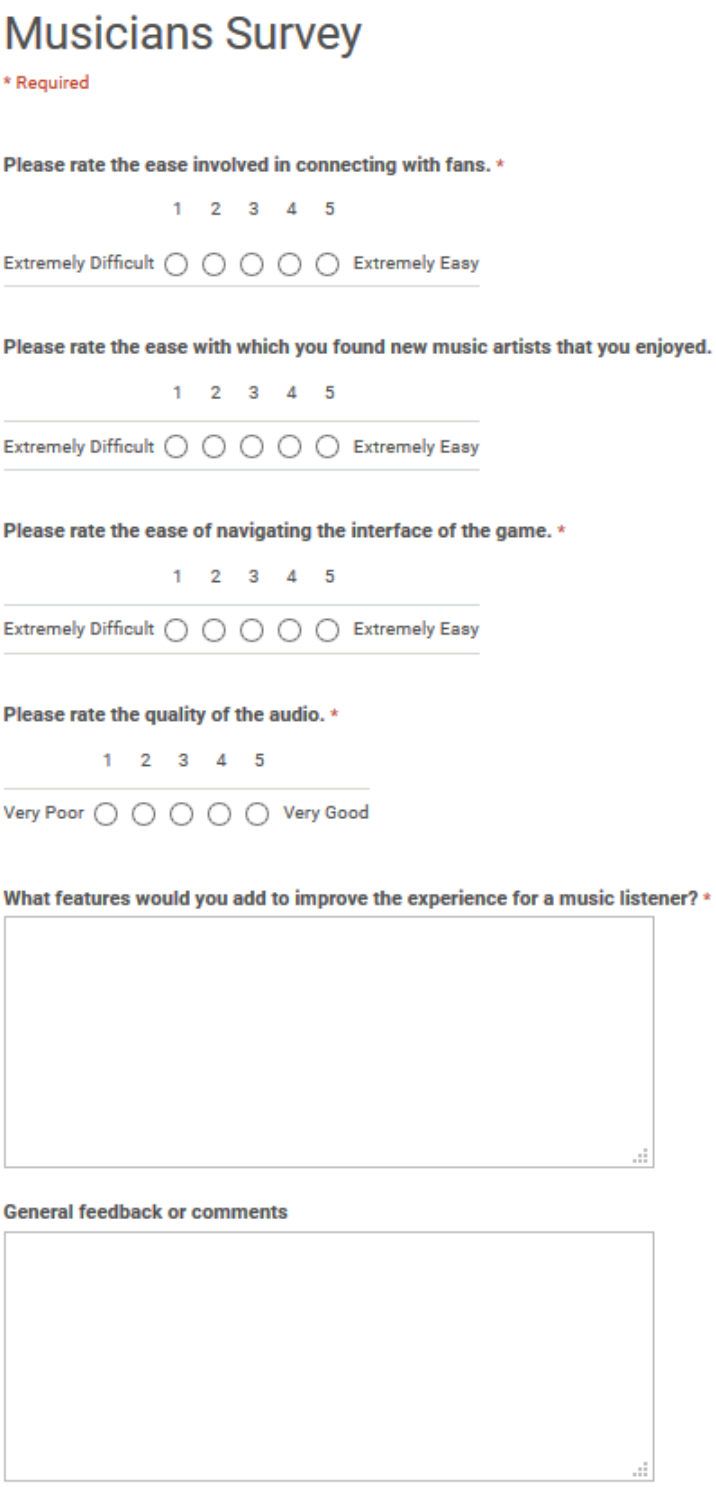


Figure A.2: Musician Survey

B Game Research Detail

B.1 The Sims: A Look at Character and Room Creators

The Sims is a life simulation video game that allows the player to control a character and satisfy its needs and wants. Character creation/customization is a core gameplay feature of *The Sims*. The character creator allows the player to customize many aspects of their character. The creator has a hierarchical model where the player can customize each feature of their character individually. The reason *The Sims* character creator is so popular and successful is because of the huge number of ways the player can customize their character. *The Sims* also lets the player change characteristics after creating a character to allow the player to change their mind at any time. Having many options allows for millions of different possible characters, so no two Sims are alike. This is especially important for online games where the avatar represents the player, since each player wants to feel that their avatar is unique to them and their personality. *The Sims* series also offers expansion packs for the player to purchase that offer additions to the game, including more options for the character creator such as clothes.

The house/room creator is also extensive in *The Sims*. The game allows the player to create the exterior of the house, which includes size, walls, windows, and so on, and then furnish the interior. The player can choose from different types and designs of furniture, as well as choose the color scheme that they want for each object. Once again, where the room editor excels is the number of options that it gives the player, which makes each room unique. The room designer is also very intuitive, which is an important factor to its success. Giving players the ability to express their creativity would allow artists to better communicate the ideas and feelings behind their music, making it a feature worth considering for this project.

B.2 Little Big Planet: Creating and Sharing Content

Little Big Planet is a platforming game where users can create and share their own levels. In the *Little Big Planet* level creator, users place objects in the world as their character. The player selects objects from a menu, places them within the world, and then can select them again to change the object's features. The changeable features available to a selected object are dependent on the specific object and the object's type. For example, certain objects are decorations, whose color you can edit, while others usable as floors and walls that can have their texture changed as well as their size and shape. These features alone allow for the basic creation of a classic decorated side scrolling platformer game.

More objects and features allow users to create objects that interact in unique ways to allow for more dynamic and customizable challenges for players to have. The player can add objects like jet packs, wires, ropes, pulleys, and fire. Combining these objects together with the basic tools and added features, like simulated physics, allows for users to create games closer to what we know as more in-depth adventure games.

The true core of the gameplay in *Little Big Planet* comes from users sharing their levels and users playing the levels that others have created [3]. The interface for finding levels is found online at lbp.me. This site offers a search function to find a specific game, and categories from which to select. Upon selection, games are then presented in order of whatever filter was chosen. Also on the homepage of lbp.me is a "Trending" list and a "Team Picks" list that contains levels that are becoming popular and levels that the development team has selected, respectively. Since the primary focus of this project is sharing and interaction, *Little Big Planet*'s prior work should be helpful in designing our interfaces.

B.3 Second Life: Understanding Community Interest

Second Life is a free-to-play online virtual world [9]. Players explore and create in a collaborative world that supports customization. There is an object builder that allows players to create objects using a simple geometric-shape based designer. The developers of *Second Life* do not create any

content; players create the world solely by themselves. *Second Life* allows players to explore and live in a world that provides endless player-to-player communication and exploration.

The main draw of *Second Life* is its community. Every player is in the same world, and the only content is provided by those players themselves. Players can create anything they want, and often come together to create large cities or environments. They are provided a simple editor and a shared space and independently create a fun environment for everyone. Chat is a big part of the experience; users can only communicate to those who are close to themselves. Players can also send location or group invites to other players, which allows players to teleport to locations that are advertised to them. *Second Life* provides the tools to bring users together and plenty of incentive to keep them there. Since we want the focus of *The Loft* to be on its community and the interaction of the players, looking at how *Second Life* achieved its success in this aspect is beneficial.

B.4 Garry's Mod: Creative Sandbox Gaming

Garry's Mod is physics-based sandbox game that allows players to manipulate assets from any Valve game or their own imported assets. The game was originally intended to be a mod to Valve's *Half-Life 2*, but due to its popularity grew into a standalone game. The core of *Garry's Mod* is the ability to manipulate assets imported from Valve's popular releases as well as custom imported ones to create custom game modes or fun environments. *Garry's Mod* freedom of creation has led users to create many viral videos called "machinima".

The game itself follows a clear-cut lobby/server structure in which any player can choose the server they wish to join. Each server, belonging to another user for a created game mode, can be customized using pre-installed assets or third-party assets. Having this user-to-user system is what makes *Garry's Mod* unique and defines the framework for creativity needed for an open-ended game.

Connor Cleary of Gamasutra states with regards to *Garry's Mod*, "When you boil it down though, video games are based on the idea of 'play', and the recent wave of creativity-games really embody that basic idea and take it to the next level. In fact some games are little more than physics playgrounds, like the popular Source-based *Garry's Mod*. But even when there are predefined goals

(generally agreed to be a defining characteristic of a game) many of these creativity-centered games have a distinctively 'playground' feel to them." [6]. The important takeaways from Garry's Mod are how they handle a sandbox style game, in addition to how content is imported into it since these are both key aspects of our project.

B.5 Animal Crossing New Leaf: Sharing Customized Rooms

Animal Crossing: New Leaf is a life simulation video game where the player acts as the mayor of a town filled with animal residents [8]. The player starts out with a fairly empty town, and from there can customize it in many ways. The player can fill their house with furniture, paintings, music, fish, and insects, and can place, move, and rotate them using a grid system. As the game progresses, the player gains resources to expand and decorate his or her house. In addition, the player can design many aspects of the town through public works projects, including where to place bridges, benches, and new stores.

The player can also visit his or her friends' towns over the internet and talk to their animal villagers or fish in their ocean. Any changes made while visiting a friend's town persist even after the player leaves. For example, if the player planted an apple tree in their town, the tree would stay there once he or she went back to single player mode. The game also uses the Nintendo *StreetPass* feature to allow copying houses with anyone that the player passes on the street. Once the game has obtained *StreetPass* data, the player can visit any house in a special part of the town. Customization of the rooms in *The Loft* is something that we have considered, so looking to *Animal Crossing*'s method of doing this may be helpful since it has proven to be effective.

B.6 Network Latency

One of the major issues confronting multiplayer games and network applications is the problem of latency. In order to obtain fluid gameplay and a non-irritating player experience, network lag must be kept to a minimum.

Cisco, a multinational corporation focused on the development of network applications and

appliances, offers some tips on how to create a faster network connection [1]. One way to reduce lag would be to lessen the distance between sender and receiver. This method requires manipulating service providers by algorithmically co-locating other service providers to find the shortest path. The method is moot as implementing a procedure as such involves service providers allowing co-location, which is a security concern. Another tip to ensure a faster connection is to reduce the size of the information, called packets, being sent across a network. Reducing overall packet size will reduce delays significantly. Smaller packets get sent quicker and have lower latency. The downside to reducing packet size is that overhead (IP and Ethernet headers) is larger when compared to fewer, larger packets. This is something to consider depending on how much data we want to relay between the server and the clients.

In the realm of multiplayer video games, Valve, a privately owned and successful game development company, offers their own take on reducing network lag within their products [4]. The Valve solution comes in the form of a system closely linking local client side input with server prediction. This process runs under the assumption that a client-server implementation is in order where the user is running a client application that consistently sends data over to a server and vice versa. This particular method starts with the user producing a stream of keyboard inputs with the application. The user's data is then passed along to the server which calculates the user's positioning and interactions. When the server has completed its iterations over the user data, it sends its "guess" of where it thinks the user should be located and asks the application to linearly interpolate to that particular position. This method reduces the number of packets that need to be sent and accounts for possible latency issues. However, a system as such would be difficult to implement.

Bibliography

[1] Design best practices for latency optimization. *Latency.fm*. URL http://www.cisco.com/application/pdf/en/us/guest/netsol/ns407/c654/ccmigration_09186a008091d542.pdf.

[2] Q&a: Coca-cola, there.com team for cc metro online world. December 2007. URL http://www.gamasutra.com/view/news/107448/QA_CocaCola_Therecom_Team_For_CC_Metro_Online_World.php.

[3] Media molecule finds success and failure alike in little big planet's sharing systems. June 2014. URL http://www.gamasutra.com/view/news/173147/Media_Molecule_finds_success_and_failure_alike_in_Little_Big_Planets_sharing_systems.php.

[4] Source multiplayer networking. June 2014. URL https://developer.valvesoftware.com/wiki/Source_Multiplayer_Networking.

[5] Number of world of warcraft subscribers from 1st quarter 2005 to 2nd quarter 2014 (in millions). 2014. URL http://www.statista.com/statistics/276601/number-of-world-of-warcraft-subscribers-by-quarter.

[6] Connor Cleary. Analysis: The era of player creativity. May 2011. URL http://gamasutra.com/view/news/124114/Analysis_The_Era_Of_Player_Creativity.php.

[7] Jason Feinberg. Five tips for musicians to engage their fans digitally. March 2009. URL http://www.pbs.org/mediashift/2009/03/five-tips-for-musicians-to-engage-their-fans-digitally068/.

[8] Christian Nutt. The quiet genius of animal crossing: New leaf. June 2013. URL http://gamasutra.com/view/news/195157/The_quiet_genius_of_Animal_Crossing_New_Leaf.php.

[9] Joe Sanchez. Second life: An interactive qualitative analysis. *C. Crawford et al*, pages 1240–1243, 2007. URL `http://research.educatorscoop.org/SL.IQA.site.final.pdf`.