FACE IT : A 3D VIRTUAL PLATFORM FOR EXPOSURE THERAPY TO MITIGATE SOCIAL ANXIETY

A Capstone Project Presented to the Faculty of the

School of Computer Studies

University of San Jose - Recoletos

Cebu City, Philippines

In Partial Fulfillment

Of the Requirements for the Degrees of

Bachelor of Science in Information Technology

by

Abella, Lieudwig A.

Fetiluna, Marvi G.

Gallaron, Jasmine L.

CHAPTER I

INTRODUCTION

Rationale

The ability to connect with others is a fundamental part of human nature. As social beings, interacting and associating with others is an excellent method to expand our social circles while improving our mental health. Others, on the other hand, find it easier said than done.

Exposure Therapy in Psychology, especially in psychological treatment, has already been a reality for some people. It has been used in the treatment of returnees from the war due to their experiences which caused PTSD in place of imaginal prolonged exposure therapy. By creating a low-threat atmosphere with controlled events by the clinician/psychologist, the users could process their feelings and experiences in a safer environment (Difede et al., 2009). Repeated exposures ensure that these patients can practice being non-anxious in these situations. The researchers created this research to incorporate this into everyday technology and make it available to professionals and their clients.

Statement of the Problem

According to a study conducted by the National Comorbidity Survey Replication and published by the National Institute of Mental Health, an estimated 7.1 percent of the adult population in the United States has a social anxiety disorder. This study aims to use video games to reduce social anxiety and develop social skills by mimicking real human interaction scenarios in a secure and controlled setting to ease the user into exposure treatment.

Traditional procedures have a high success rate, with 60-90 percent of patients reporting significantly reduced symptoms after their sessions (Legg, 2020). However, some of these sessions can be resource-intensive in real life. It takes time and money to recreate these situations or even to accompany these participants to real-life scenarios. These circumstances are why some, like our team, have taken to combining treatments with technology. Of course, exposure therapy is best captured by experiencing and devoting time to socializing with others face-to-face. The game does not aim to replace existing solutions. In these times of the pandemic, patients with social anxiety can use this application to continue treatments online and as part of the initial stages of treatment before introducing them to actual stimuli.

The use of mobile applications to help in the mitigating of anxiety caused by social activities has been around (Miloff et al., 2015). VRET, or Virtual Reality Exposure Therapy, has gained much traction as a treatment for various anxiety disorders (Gahm, G.A, Reger, G.M, 2008). In a randomized controlled trial involving VRET for social anxiety disorder, they discovered that participants improved their ability to deal with social situations in a year. "*Virtual reality exposure therapy is equally effective as exposure group therapy...*" they emphasized (Anderson et al., 2013). Multiple studies praise the flexibility of VRET. If we were to integrate these realistic simulations on a smaller scale where more people could afford and access them, the research could provide insight and help to a larger audience.

Significance of the Study

With the help of this study, researchers will be able to grasp better how video games can assist psychology, particularly in psychotherapy. We want to understand better how games can aid in the sessions people undergo by making them more engaging.

REVIEW OF RELATED WORKS

Video Game Industry and Social Anxiety

Social anxiety prevails as one of the most common anxiety disorders (NCBI, n.d.). Although the current society has opened its eyes to its significance, there still is a lack of cost-effective methods to aid treatment. The lack of cost-effective solutions is why some have turned to collaborate with the video game industry in search of alternatives.

There have been numerous positive studies where even casual gaming has benefitted patients with anxiety. For example, a clinical study researching the use of electronic games in therapy has determined results where commercial video games showed benefits in anxiety management in patients (Horne-Moyer et al., 2014). Another controlled study showed significantly reduced anxiety symptoms after playing casual video games (Fish et al., 2014). However, one study found that adolescent girls who reported the most hours spent playing games had higher anxiety levels than boys, implying that playing games is beneficial in reducing anxiety levels in boys but not in girls (Ohannessian, C.M., 2018). To further verify this, the research will also bear this factor to help determine how gender plays into the study.

Aside from casual games, several companies have recognized the value of focusing their game designs on mental health, particularly social anxiety. A similar study was accomplished that aims to treat social anxiety through a mobile app called “The Challenger App,” which also relays that with the help of gamification, even the mobile game industry shows the potential to reduce the burden of Social Anxiety Disorder. However, to successfully do so, there were still challenges, such as personal data privacy. The application focused on players completing challenging interactions with their environments while setting self-goals and reflections as motivation (Carlbring et al., 2015). On the other hand, a third of the other existing commercial applications for mental health reports focus on psychoeducation, symptom management, and supportive resources, with 23 out of 28 applications exclusively focused on social anxiety (Alyami H. et al., 2017).

A systematic review by Campbell et al. (2021) highlighted how the need for more accessible alternative treatment methods had increased, especially in the current worldwide COVID-19. The future of commercial video games has been spotlighted by how "*researchers and video game companies are beginning to find new solutions to address mental health support through VR gaming...Given the immersive nature of VR technology and the controllability of the virtual environment, it could be particularly well suited for exposure therapy*".

It is worth looking out for the rise of VRET, or Virtual Reality Exposure Therapy. Tailored for clinical use, VRET takes up the treatment for anxiety up a notch. A study took this approach to exposure to virtual social interactions to treat social anxiety. Results show that the exposure effectively reduced the levels of Social Anxiety Disorder in the 60 patients they had (Brinkman et al., 2016). Walkom (2016) supports these results who shared details of where his participants became anxious within the VRET sessions but managed to lower their anxiety levels after repeated exposure sessions. Participants with slight stutter also showed improvement in speech. Several other studies show other promising results; however, many researchers share the same sentiment that it is still not stable or efficient enough to be a stand-alone clinical treatment for various disorders (Emmelkamp et al., 2020).

Exposure Therapy and Simulating Stress Related to Social Anxiety

Exposure therapy is a type of psychotherapy that assists people in confronting their fears. This method is accomplished by repeatedly exposing the person to the stimulus that causes fear. This type of psychotherapy is rooted in assisting the individual in avoiding the sources of their fears, whether they are phobias of specific animals, people, or activities. In this case, the person would be exposed to the stimulus in a safe environment to reduce fear and avoid avoidance in the long run.

However, the question of how to mimic the distress people experience in real-life social situations remains. Luckily, some studies have delved into discussing several points of view regarding evoking similar emotions from individuals through virtual environments.

In 2016, Barros-Neto and his team found that exposure to computer-generated 3D images was enough to reduce social anxiety. They repeatedly showed the participants images of social situations within 12 sessions. According to their findings, there was a significant increase in treatment adherence despite just showing realistic imagery.

Conversations with virtual avatars were also fear-inducing, although less realistic than conversations with a Vivo assistant (Briceno et al., 2013). This study could mean that presence and being "in" the simulation with the avatars also plays a significant role in immersing an individual in a virtual environment. However, there were positive and challenging points in a study that sought to determine whether virtual reality audiences elicit distress similar to an authentic audience. The average score for subjective distress in participants with Social Anxiety Disorder indicates moderate distress. Therefore, the results show some ability of a virtual environment to evoke anxiety and distress in individuals. However, the study notes that the participants judged that the environment was less immersive. One participant stated that "*...you know that the virtual people are not thinking negatively about you*" when tasked to perform a speech in front of the virtual characters as their audience (Beidel & Owens, 2015). As the simulation utilized audiences with neutral facial expressions and static behavior, this may have been a significant factor in its lesser realism.

Another study by Nomani and Sekhavat (2017) looks into a different angle and tries to determine whether there is a difference in eliciting anxiety if participants face passive or active scenarios. True to their hypothesis, participants did experience a considerable level of stress on both, but active scenarios elicited more distress from the participants. Active scenarios consist of interactive scenarios where the content of the environment changes in response to the participant's actions in real-time.

Current issues

Expanding and creating a virtual environment that simulates a nearly realistic environment involving socializing as a tool for exposure therapy experts have been

studied extensively and has yielded positive results thus far. VRET is an excellent avenue, but we have to understand that equipment can be scarce, and only some have access to VR headsets. On the other hand, a heavier percentage of the population has desktops at their disposal.

Although there may be considerable differences between VR and PCs, desktop games are more common and easily accessible to the masses. Existing desktop games like Solitude, for example, have explored the depths of social anxiety through engaging narratives. The theme of social anxiety and mental health, in general, is not new in video games. However, other than that, there appears to be a gap in pure real-life simulations of various social situations.

Another area in this exposure therapy not yet explored is AI integration, especially a form that discerns the user's emotions through facial recognition. With advancements and accessibility to open-source dynamic analysis technology based on facial landmarks, this may be an excellent opportunity for further studies in emotion recognition and improving desktop simulations for social anxiety and handling complex emotions. By integrating this, we can better track the user's state and understand one's behavior when experiencing social anxiety in social situations. The ease of progress tracking makes for a suitable environment for adaptive gameplay that makes the game dynamic—altering the game elements based on the user's actions and emotions. Of course, it needs professional input as it entails the details of the adjustments to the game elements during listed common scenarios. There have also been conflicting studies that emphasize there may or not be different results according to gender. As long as we can effectively find a way to introduce a safe amount of stressors effectively and take into mind these other factors, this research hopes to broaden the scope and address other psychological disorders by incorporating these integrations.

Project Objectives

To create a safe, controlled 3D virtual platform for exposure therapy that is a tool for experts to mitigate social anxiety through:

- Immersive environments like having 3D models for the user to interact with,
- Realistic social scenarios,
- And adaptive gameplay, such as intervening in the game's scenario based on facial recognition.

Project Scope and Limitation

The researchers conducted this study to create a game that can assist in the early and middle sessions of social anxiety psychotherapy. The game aims first to assist college students at the University of San Jose-Recoletos to see the game's efficacy and help students ease into the idea of psychotherapy and exposure therapy for social anxiety.

CHAPTER II

GAME DESCRIPTION

Face It is a first-person 3D RPG simulation desktop game with realistic environments that simulate real-life scenarios. It employs a webcam to capture the user’s facial expressions and emotions and the user’s microphone for voice input when interacting with the manually trained chatbots in the simulation. A standardized pretest and posttest also determine their level of anxiety per scenario. The players can traverse the scenarios in order of the brief storyline or as their own.

Design Goals

The game aims to achieve the following goals:

- Establish a safe, controlled environment as a tool for exposure therapy.
- Create common anxiety-inducing scenarios with real-life influences.
- Determine which social scenarios the player has the most difficulty adapting to,

- Develop gameplay that immerses players through
- Realistic models and environment
- Dynamic gameplay that adapts to the player's emotions

Influences and Sources

The team gets "Face It "'s primary outline inspiration from training games like AimLab. The game borrows the idea of analyzing the player's progress per scenario and letting them undergo tests according to their need for help. Research on similar games that tackle social anxiety was also looked into when deciding "Face It "'s overall gameplay. Information about general social anxiety, their traditional treatments, and the typical scenarios and tasks that trigger social anxiety are both extensively researched and asked by a licensed psychotherapist.

Target Market

Our ideal user has the following characteristics:

1. Male & Female
2. 18+ years old
3. Has a PC with decent specs
4. Has difficulty in handling social situations / diagnosed with social anxiety
5. Recommended to play by their psychotherapist
6. Is in the initial or middle of total planned sessions by their psychotherapist
7. Familiar with PC games

As standardized by the criteria mentioned earlier, the expected market is the primary target. However, it is worth noting that the game is also suitable for all ages above 15, as recommended by their therapists.

FUNCTIONAL SPECIFICATIONS

Game Mechanics

Core Game Play

The PC (Playable Character) moves in a 3D (3-dimensional) plane mirroring realistic social anxiety-triggering situations. The player will undergo a calibration phase where Marbee, “Face It “’s virtual assistant, introduces the world of simulated reality. The player can choose to play in Story Mode or Session Mode.

Story Mode lets the player go through a story narrative where they can experience all possible scenarios. On the other hand, Session Mode is for players who want to choose what scenario to simulate and experience in particular sessions.

Marbee monitors as the player go along these scenarios and finishes the main objective together. Each objective completed will be rewarded accordingly with social points. These points can show the player’s progress and indicate their subsequent sessions’ levels.

The player is also given a standardized social anxiety scale test before and after a session. During these simulations, facial recognition technology detects players’ emotions in real-time. This data will allow the system to make changes to the scenario dynamically.

Game Flow

Actions that the PC can perform are

1. Move around the area.
2. Interact with NPCs.
3. Interact with objects.
4. Talk.
5. Complete objectives.

Characters/Units

1. *Player*: The player is an individual who experiences difficulty handling social situations.
2. *Marbee*: The game’s virtual assistant. Marbee floats around and guides the player within the scenarios.

3. *Humans*: NPCs patterned from human behavior that serves different purposes under the simulated situation. It may come in groups or individuals.

Game Play Elements

Below are the listed elements present in the game level that the player can interact with:

- Objectives: This lists the main quest and the player's sub-tasks to complete to finish the session. These tasks encourage the player to interact with the NPCs according to the specific context per scenario.

There also exist elements that determine the PC's stats and progress:

- Social Points: Social points are points taken from completing the various objectives per simulation. Accumulating social points gives the player a view of their progress per session.
- Anxiety Level: During simulation, whenever the system recognizes a strong negative reaction, it adds up to an account that displays the current level of anxiety the user is experiencing through a special GUI. This determines whether the user needs an intervention from Marbee or not.
- ***FER Points***: Much like the anxiety level, this keeps track of the user's emotions only during conversations with chatbots and is registered as part of the point system.

Game Physics and Statics

Physics in the game work as such:

- Presence of gravity.
- PC moves in a 3D plane from a player's perspective.
- PC can touch or pick up objects.
- PC can interact with random Human NPCs.
- PC can interact with Marbee.

Artificial Intelligence

Human NPCs

- It may appear in crowds or by individuals depending on the scenario simulated,
- May approach and interact with the PC if one of the level tasks requires the PC to do so,
- Pre-programmed "Idle" behaviors.

Marbee

- Guides PC through the simulation.
- If Marbee senses that the PC shows a higher surge of change in emotions, Marbee calms down the PC with reassuring words.

System

- If there is a continuous spike of emotions, the system deploys Marbee as an intervention to calm the user.

TECHNICAL

Architectural Design

The game's structure follows the graph above, where the game continuously receives and processes the user's camera and microphone input. The user's current emotion uses Camera input during the simulation to count when the user is in distress and needs intervention. IBM Cloud processes the mic input with SpeechToText, and NPC chatbots dynamically respond to the user's input with the Watson Chatbot Assistant API with generated voices from the TextToSpeech API. The main game logic then collects the information above. This game logic comprises the different quests, types of scenarios,

and chatbots appropriate to the scenario and presented in a 3D Virtual Environment as the user's point of interaction. Meanwhile, the Local DB then stores the data needed in progress tracking.

Database Design

The database design is straightforward and only holds the necessary data for tracking user progress per scenario, weeks, and posttests. It contains the user's complete mood logs throughout the simulation, percentages of negative, positive, or neutral mood, lists, types of scenarios, user logs, and weekly test results and interpretation.

USER INTERFACE

Flowchart

Functional Requirements

Splash: Loads the initial screen and introduces the Face It logo.

Calibration: Presents a step-by-step guide for the user to set up and familiarize themself with the game.

Main Menu: Contains all possible choices that introduce the user to the game's atmosphere that simulates a list of scenarios taking up the majority of the space.

Settings: Allows the user to modify and control the game's settings or attributes.

Profile: Displays the user's history of simulations and their accumulated progress.

PostTest: Weekly post-test the user takes to record their progress. It only appears after finishing the needed amount of simulations for the week.

Simulation: *It is wh*ere the bulk of the game happens. The simulation includes displays of the objectives for that particular session.

GUI Objects

The actual gameplay of the simulation presents a clean and barely there GUI to focus the player's attention on the environment as much as possible. The following GUIs present during a simulation are as follows:

- *Overhead Stats Bar:* This bar displays the current stats and details of the session.
    - *Timer:* Holds the total amount of time since the start of the scenario.
    - *Social Points:* Displays the total amount of points acquired during the simulation.
- *Quest Objectives:* This section displays the current scenario objective, how much time has passed since it started, and the points acquired if completed.
- *Anxiety Level Display*: Visualizes the anxiety level as determined by the system depending on the user's continuous negative emotions.
- *E to Interact*: Text appears when within the interaction area of the NPC.
- *Indication to speak:* This lets the user know when to answer or communicate with the NPCs or Chatbots.

The other GUI objects are on the main menu, where most navigation between the screens happens.

ART AND VIDEO

Overall Goals

To increase the player's game experience and immersion, the developers emphasized creating semi-realistic 3d models. The environment, the human NPCs, the objects, and additional assets to be seen and interacted with in the simulation mirrored that of real life. Until now, we have spent ample time practicing and researching methods to achieve that point of realism.

The packaging art shown is used as initial splash screens and loading screens. The icons are used for the build of the desktop game.

3D Art and Animation

The team created all of the assets for the game. As for the animation, unless the 3D model is incompatible, Mixamo may be used for common types of behavior and actions. Marbee, for example, is manually animated as its model isn't compatible with the application. Some human animations were adjusted manually to fit the model used with the action or to derive a different animation to better fit the selected scene. Shown below are some of the finished 3d assets of the game.

Character Design

Gameplay Elements

1st Person Player model;

- Idle;
- Walk;
- Talk;

Human NPC 3d models;

- Idle;
- Stroll;
- Talk;
- Laugh;

Marbee;

- Idle;
- Talk;
- Laugh;

SOUND AND MUSIC

Overall Goals

The team made all the assets. As for the animation, Mixamo may be used for standard behavior and actions unless the model is incompatible. Marbee, for example, is manually animated as its model isn't compatible with the application. Some human

animations had to be manually adjusted to fit the model used with the animation and to create another animation that fit more in a selected scene. Shown below are some of the finished 3d assets of the game below.

Sound FX

The game used simple foley sound effects to recreate a more realistic and immersive game, e.g., pencil scratches, ambient noises in classes, etc. NPCs would also have voices tailored to a general type of character they're presenting per scenario.

Music and Sound FX Assets

Soundtracks:

- Ambient opening music for the intro.
- Upbeat music for when the player is on the interfaces outside the simulation.

Ambient / Foley Sounds:

- White noise
- Muffled background music
- Plastic rustling
- Footsteps
- Glass breaking
- Mic setting up
- Barcode scanner beeping
- School bell ringing

PC:

- Breathing
- Sigh
- Thud

NPCs:

- Hushed whispers
- Audience applause
- Audience sighs
- Children laughter

- Light Chatter

Marbee:

- Sigh
- Consoling
- Giggles

STORY

Step into the world of the game and experience living in it first-hand. Take on different roles, interact with people, and watch yourself grow.

Player Character

"Nameless" MC – Often called shy and quiet by the people around them. After deciding he wants things to change, he is guided by Marbee as they take on the world to become a better version of themselves.

Secondary Characters

Marbee – Assists the player in navigating the world in the game. They guide and encourage the player with each step, letting the player know they aren't in it alone.

Story theme

One against himself, the player takes on his weaknesses head-on to overcome the monsters that govern his thoughts.

Visual theme

The story takes place in the modern world—the setting changes in every scenario where the player decides to participate. A variety of people inhabit the area. While some may be good-natured, others may interact to add spice to your day.

Story Outline

Chapter 1: The Classroom

Scene 1-1: Classroom

Quest 1: Looking for Groupmates

- When the main character arrives at the classroom, Marbee runs down the primary goal of the situation: to approach different people to ask to be groupmates.
- After explaining to the player, Marbee retreats and leaves the player to themselves.
- After greeting the NPCs, the user then asks the person if they already have a group. If they do not, then the user may ask if they would like to join the user's group.
- After asking the NPCs, the NPC will either:
- Accept the offer and move to the area designated for the group.
- Decline the user's offer.
- The user will continue to approach NPCs until two accept their offer.

Quest 2: Self Introduction

- User groups with 2 NPCs that accepted their offer to be groupmates with them.
- The user will first ask the group to introduce themselves formally.
- After the NPCs have presented themselves, the user officially introduces themselves to them.

Quest 3: Looking for a Leader

- The group then has to appoint a leader. The user will first ask the NPCs if they would like to volunteer to take up the role of the leader.

- When the NPCs both decline, the user will ask for suggestions to where the NPCs suggest the user take up the role.
- After the user accepts, the NPCs show support to the user.

Quest 4: Relay Information to the Teacher

- After agreeing to the role of leader, the user will approach the teacher to inform her of their progress.

Scene 1-2: Group Discussion

Quest 1: Project Product

- The group is responsible for developing an innovative product for a project for the class. The user asks the group their thoughts on an “innovative product.”
- After asking them for their ideas, the user will ask the group for problems they can try solving.
- Once the group has presented problems they can solve, the user will ask the NPCs for solutions that solve the problems.

Quest 2: Suggest Roles/ Division of Tasks

- After discussing their product, the user will ask the group how to divide the tasks amongst the group.

Quest 3: Check on Group / Dismiss Group

- After the discussion amongst the group, the user will dismiss the group and say goodbye to the NPCs.

Scene 1-3: Project Defense

Quest 1: Defense Preparation

- The user meets with their group and asks them to review for the defense panel.
- The user asks the NPCs a series of questions relating to the following:
    - Product;
    - Target Audience;
    - Main Problem;
    - Existing Alternatives;
    - What makes the Product Unique;

Quest 2: Answering Panel's Questions

- The panel will ask the player questions about the product, to which the user will respond accordingly.

Chapter 2: The Party

Scene 2-1: Home

Quest 1: Telling mom about Project proposal Acceptance

- The user comes home to greet Mom. The user asks how she's doing.
- The user then informs Mom that they have important news, the acceptance of the project proposal.
- After hearing the news, Mom reacts joyfully tells the user they should celebrate. The user then asks if it would be okay to invite their friends, and Mom agrees.

Quest 2: Telling dad about Project proposal Acceptance

- The user comes home to greet Mom. The user asks how she's doing.
- The user then informs Mom that they have important news, the acceptance of the project proposal.
- After hearing the news, Mom joyfully tells the user they should celebrate. The user then asks if it would be okay to invite their friends, and Mom agrees.

Quest 3: Inviting Charlie and Alex

- The player calls Charlie and Alex and asks how they're doing. The user inquires if both of them are busy.
- After the user invites both to the upcoming party, they then reply by either:
    - Accepting the invite;
    - Declining the invite;
- After speaking with the two groupmates, they invite other possible friends.

Quest 4: Inviting other friends

- The player calls three other friends, greets them, and asks how they're doing. The user asks if they are busy.
- After the user invites each to the upcoming party, the NPC will then either:
    - Accept the invite;
    - Decline the invite;
- After speaking with one NPC, the user will approach the second NPC and perform the same task.

Scene 2-2 : Talking to friends at Party

Quest 1: Talking with Mom

- The player approaches Mom and greets her. The player thanks her for throwing the party, and Mom suggests the user go around and talk to the guests.

Quest 2: Talking with Guests

- After talking with Mom, the user will go around the party to entertain guests by conversing with them. They will approach the present NPCs.

Scene 2-3: Talking to Aunt's friend/co-worker at the party.

Quest 1: Talking with Aunt Susan

- After conversing with the NPCs at the party, Aunt Susan arrives and is looking for the user. The user then greets Aunt Susan. She explains to the user that she knows some coworkers interested in helping the team pitch the idea to a company.

Quest 2: Talking to Aunt's Coworker

- Aunt Susan invites the user to speak with her coworkers. The user then walks and converses with the coworker to ask for their opinion and suggestions on the game itself.

Chapter 3: The Office

Scene 3-1: Pitching to Office Heads

Quest 1: Pitch to the Office Heads

- The player, alongside Charlie and Alex, will be pitching the product idea to a group of panelists in an office.
- After the pitch, the panelists will ask questions that the user will answer.

Scene 3-2: Office Party Celebration

Quest 1: Speaking with CoWorkers

- The player walks around the party and talks to office workers congratulating them.
- The player then asks the coworkers for their opinions and suggestions for the app.

Scene 3-3: Pitchfest / Public Pitch

Quest 1: Public pitch

- The player stands in front of a large crowd to perform the pitch in front of a large audience.

LEVEL REQUIREMENTS

Level Design Seeds

The game first gives an overview of the actions the user should take in each scenario. The player must engage in conversation and interaction with specific tasks and NPCs to move through the scene. The user progresses through the quests while remaining positive. The NPC will end the conversation and tell the player to return when they feel better if the player ever feels upset while speaking to them. Marbee

congratulates the player for a job well done as the scene comes to a close, where Marbee will also provide feedback on the player's performance.

CHAPTER III

SOFTWARE DEVELOPMENT AND TESTING

Development Software Platforms, Environments, and Tools

The game application used Anaconda, MySQL, Unity3D, Blender, IBM Cloud Services, and Mixamo in terms of the development tools and platforms used to create the web application. Visual Studio Code was the primary integrated development environment (IDE) used to develop FaceIt. In contrast, the primary programming languages employed were Python to integrate the FER API system with the application, C# for programming in Unity, and PHP for handling the game's database.

Third-Party Softwares

Anaconda

Anaconda provides a conda environment that links the Python API flask used for Facial Emotion Recognition. It transfers the data it gets from Unity, processes it from the web, and sends it into the application.

MySQL

MySQL is a database software for all the data used, saved, and processed in the application.

Blender

Blender is the software used to create the game's 3D assets. It covers all depictions of the environment and humanoids at various levels.

Medibang Paint Pro

Medibang Paint Pro is a drawing software where the team created most, if not all, initial concept art and sketches.

Adobe Photoshop

Adobe Photoshop is a graphics editor used to manually create all graphics seen in the game, from the logo down to the small GUI details.

Mixamo

Mixamo is an Adobe third-party website that provides a wide range of free rigged humanoid animations. The team combined this with human models created in Blender to give the game's NPCs human-like behavior.

Unity3D

Unity is the main game engine that makes all the game mechanics, physics, and interaction possible. All the environments and models made from Blender were imported into it and used together with the coded logic.

IBM Cloud Services

IBM Cloud Services provides the game with three different but essential functionalities, Watson Chatbot Assistant, Text-to-Speech, and Speech-to-Text. All NPCs' voices and dialogue use cloud services to converse with the user.

Development Process and Usability Testing

Pre-production

The game's goal is to go through the stages of different social interactions the user chooses to undergo. The game begins with a social anxiety pre-test to assess the user's anxiety scale. The user gains points for completing quests, conversing with the NPCs, and maintaining positive emotions(i.e., happy, neutral). Each scenario is a unique social interaction representing a different type of social tag the user wants to improve. Anytime the facial emotion recognition notices a prolonged period of negative emotions, Marbee, the virtual assistant, will intervene to ask the user whether they want to continue the scenario or leave. Marbee will also provide feedback and improvements the user had in the scenarios they performed. After every session, a posttest will check the user's emotional state.

Game Concept

Face It is a simulation game that aims to help users suffering from social anxiety perform scenarios of common social interactions of their choosing that they wish to improve. Each scenario has different tasks that the user must complete while conversing with the NPCs and maintaining positive emotions to gain points.

Character Design

Regarding character design, the developers decided to use references to existing uniforms and clothing. Although we desired a realistic model to simulate real people, time and technological constraints limited the models' details. However, the creation of detailed and realistic models was still possible.

The creation and sculpting of the characters are all made in Blender. A high poly model would first be baked into a lower poly version to save on performance and memory. The models would be duplicated and sculpted into different body types to create variants of the same gender, thus creating an assortment of NPCs to populate a scene.

The clothes are created by copying and using parts of the base model, and then the details are sculpted in Blender using the Multires Modifier. After finalizing the design, similar to the base, the higher resolution mesh is baked into a normal map and used on the low poly version of the clothes.

On Marbee both rigging and animation are done via Blender. The other rigs and animations with humanoid avatars are created in Mixamo.

All the texturing is done entirely in Blender. First, for the clothes, the basic cloth textures are downloaded from free websites such as Textures.com, then combined with the normals generated from the high poly models using Blender nodes. Custom textures with patterns, such as the skin with its procedurally created bumps, are done with Blender nodes and baked into each model. For the painted textures, such as make-up and specific highlights on each of the models, the designer utilized Blender Texture Paint to paint in details of each character.

In creating the hair, the designers generated hair cards with Blender's Particle System. Each hair card would vary in thickness and size used on different parts of the hair. These cards would be overlapped and structured to resemble real-life hair with as few vertices made as possible.

Lastly it is imported into unity with other textures like emission and glow where needed.

Structure Design

Scene structures were designed with realism in mind. Though constrained, the developers decided to add details to the surroundings to add realism to the scenarios. And thus, details like leftover chalk on chalkboards, books scattered and writing on classroom tables add realism to the game.

Similarly the character models, all structures are made and arranged in Blender, with the textures either taken from free platforms, procedurally made using nodes, or manually painted in texture.

To lessen performance costs, we baked the shadows of the static objects into each scene while Unity applied real-time lighting to dynamic objects such as the NPCs.

Level Design

The team built the levels upon the countless research on what types of scenarios commonly induce social anxiety. Some of these were approaching strangers and asking questions, meeting new people, writing and reading in front of others, and making small talk. Using this information, the developers decided to make the scenarios apply these small triggering events and use it to make one scenario playable individually but still connect if played whole. The player follows the narrative and starts fresh as a student who has to find a group in a new classroom setting and, in future chapters, propose Face It as their project. This way, the player can learn about the application they're playing.

As for the efforts in replicating social anxiety-inducing scenarios, the aforementioned triggering events are used in the scenarios as the conversation quests appropriate per the setting and scene.

Another design implemented is the intervention system, where it's triggered whenever the player gets in too much distress and needs some time to calm down. Other than exiting the level to try another time, the player can try a mini-game as an alternative. The mini-game consists of a simple level where the player drags and drops colored boxes to their appropriate places as a form of casual game. This game was because of the recommendation of the people in the field of psychology the researchers interviewed. The reasoning behind this game was that it's supposed to calm troubled people down, especially if they feel out of control. Here, the player can focus on menial tasks and gain control in focusing on what they're doing, no matter how casual or mundane, and eventually stabilize themselves.

Icon Design

The icons used in the game's UI/UX design are either manually created or edited from images to fit the overall theme and atmosphere of the game. Since we try to keep the in-game GUIs relatively simple to increase focus on the simulation itself, a lot of the edits are more on colors, opacity, and glow of the images.

UI/Icons outside the simulations, on the other hand, are more intricate and sci-fi themed with most being created manually within Photoshop.

Production

Scenario Recommendation Queue

This section shows how the user's input from the calibration affects the recommended scenario and weekly tasks needed to be completed before the user can take a weekly post-test for social anxiety scale progress tracking.

Recommendation Manager Script

This script matches the scenarios fitting the user's preferred scenario type and lists them out in a hierarchical queue that still follows the game's overarching narrative where recommended scenarios from Chapter 1 will always go first as the top recommended scene as opposed to those in the later Chapters.

In the script above, the first function gets all user chosen scenario tags and the whole scenario list from the database. If the player has past sessions, it's put into a separate container. The second function matches the full list of scenarios to the user's chosen tag list. If there are past scenarios that haven't been completed, it's also added to the queue. Once the queue is filled, a simple sort is done to make sure it's in a chronological order. The third function takes the first scene inside the recommended queue and displays it as the top recommended scene right when the user arrives at the main menu.

Facial Emotion Recognition Script

Facial emotion recognition is a process of interpreting facial expressions and non-verbal cues to determine the emotional state of someone. It can monitor people's reactions to different products or services, help with research and mental health treatment, and, generally, gain better insights into people's feelings and opinions. The script extracts the facial features from the image or video and then analyzes them to determine the dominant emotions. The emotions include facial expressions, such as

frowning, smiling, surprise, fear, and sadness. Once the emotion vector is established, the script can accurately determine the person's emotional state.

FER Controller Script

This script starts and stops the FER system and communicates with the third-party flask application to send image input and receive processed data to determine the player's mood in-game.

The code snippets above show how we connect to a separate local flask server and send over a captured image. In the second part, the calculated data of the top mood and its value is parsed from JSON into an array and sent over to a container to where the game has access.

In this snippet, the function to process mood data from before is called when triggered by the main game logic. If no face is detected, the game resets the values of the container.

Game Manager Script

The script handles all in-game processes and evaluations, keeping track of the player's progress in the scenario. One of these is receiving the data from the FER script to record and use.

The image above is the general flow of how the mood data is filtered and processed into the game. If the current top mood calculated is received, data consisting of the mood description, value, and duration it's felt within the scenario is created and stored in a new class. This class is sent to another huge data container only given to the web server and stored in the database once the scenario is done.

During a quest, mood data is stored in a container to be used as a point criterion. This process is done every 7 seconds from the start of the simulated scenario.

The code snippet above deals more with filtering the recent top mood stored and giving it a particular value. If the received top mood's strength goes beyond 80%, it's given 2. If it's around 50%, it's a value of 1, and a 0 if it goes below 50%. These values are to increase accuracy and only record heightened emotions with more than 50% of mood

strength. If the overall value goes beyond a set border value, the system calls for an intervention where Marbee appears.

IBM Scripts

This section shows the main flow and structure of the application of the different IBM services, together with the Watson Assistant Chatbot. The IBM Chatbot was chosen over other alternatives for its popularity in the community, with tutorials and guides to using the project and simplicity in integrating a chatbot with built-in Text-to-Speech and Speech-to-Text functionalities.

The creation of dialogues for the chatbot website was straightforward. Its intent recognition was strict, where the bot focused on capturing the intent of what the user was saying as opposed to accepting keywords to allow conversations and not allowing bypasses in the simulations through saying keywords of the conversation.

BotControl Script

This controller script handles the cycle of input, processing, and output of response from the user to the chatbot and out again.

This controller script handles the input, processing, and output cycle of the user response to the chatbot and out again. The code snippet above shows how three different IBM Services: SpeechToText, Chatbot, and TextToSpeech, work together in a conversation with a working chatbot in-game. Additional functionality is added where the chatbot is forcibly stopped and exhibits behavior where it wishes to end the conversation with the user.

Chatbot Script

Initially, this was a sample API script that the developers restructured to fit into the desired chatbot behavior for the game. Now, it sifts through the results it receives online and determines how to deal with each possible response.

Shown above is the initial part of processing the result. We first determine if the result received over the API is a valid response. Then, if certain values are present, we deal with the result and hand over the final text response to be outputted to the user. In this particular part, we sift through if the current quest uses probability responses, where we then use a randomized index and split from the array of possible lines.

To continue, we go to the next stage of filtering, where we identify if the response is equal to any of the set failed lines. If not, it undergoes another step and compares it to the appropriate responses for the specific conversation and verifies which dialogue nodes the player has completed with the chatbot, triggering the successful ending of the conversation if it hits the specified last node.

Here, we reach the last stage of filtering and confirm if the response is valid but not found within the specified dialogue nodes for the conversation. The result is then treated as one of the failed responses but with specified behavior to be exhibited. On the other hand, if the player has accumulated more than five(5) failed responses, the chatbot is forcibly stopped and cuts the conversation short.

Watson Assistant API

The section shows how the developers use the Watson Assistant Chatbot to manually create the flow of dialogue and possible responses to different inputs. Due to the limitations of IBM Cloud free accounts, the account used was limited to 5 Assistants per account. The developers worked around these limitations using Intents and Entities to create virtual NPCs and grouped the whole assistant by scenes. However, this solution would mean NPCs' chances of replying with the same outcomes are high as they share identical intents in the assistant API.

The image above shows how intents and entities are created and trained to use the chatbot dialogue system. Intents are created when a dialogue requires the user to speak defined phrases to trigger the chatbot to return the necessary response to simulate

a conversation for the quest. Intents are trained with multiple examples to allow the user to speak different possible sentences or phrases that still carry the same intention.

The image above shows the structure to which the chatbot dialogue is made to simulate possible quest conversations within the game. It uses a mix of Intent and Entities to determine the user's intention and respond accordingly. Quest NPCs within the game utilize the dialogue system to understand and converse with the user.

Quest Scripts

This section shows the main flow and structure of the application of the different IBM services, together with the Watson Assistant Chatbot.

Quest Manager Script

This controller script handles the cycle of input, processing, and output of response from the user to the chatbot and out again.

The code above shows the flow of a talking quest used in unison with the chatbot. It feeds the chatbot the correct set of valid lines for that quest. It constantly checks if the specific chatbot is still active or if the conversation’s finished, where the quest is immediately completed, and post-quest processes are triggered.

Another type of quest shown in the snippet is where the player reads a specified script. Unlike the talking quest, it only uses the SpeechToText IBM service and the translated user input to compare with the set script.

Animation

This section shows the process by which the NPC character models and Marbee were animated. NPC chatbots follow default behaviors and transitions set in the controllers. The figure above is a sample controller template created for quest NPCs.

This is created with a set of trigger values used in common with each other. Customization between each NPC is done by dragging and dropping the appropriate animations to be used specifically for their scenario.

The image above deals with animating Marbee. Unlike the NPCs, Marbee already has triggers connected to every manually created animation and only transitions between those.

Post-Production

Once the game has finished initial testing by the developers, the game is set up in a separate unit for testing. All users are informed about the project and what it entails, along with feedback sheets for those who accept. All participating testers are given scheduled use of a borrowed unit to beta test the game individually in a quiet, isolated space.

Usability Testing

The team conducted the testing with one round of the recommended session in the game and a Google Form survey with randomly chosen respondents ranging from 19 - 23 years old. Their mood data and initial social anxiety pretest results are stored in the database, while the post-survey form is composed of a 6-point Likert Scale on the general game and its appeal to the user. The developers were able to get ten(10) random testers. The team reduced the limited number of participants due to time constraints.

The data above shows that ten(10) respondents were the only ones to play test the game due to limited time. 50% of the respondents fall within the range of 21-23 years old, with 30% 20 & below, and the remaining above 23 years old. A more significant percentage of the respondents were male. However, because random testing was used, any student the researchers met that had the time to test was picked. There is no assurance whether the lack of female participants has any connection with the game and its appeal to the possible audience.

The social anxiety scale self-evaluation survey was used in the game as the team borrowed a progress test from the APA (American Psychological Association) called the “Severity Measure for Social Anxiety Disorder (Social Phobia) - Adult.” The test consisted of 10 questions numerically determining the level of social anxiety in different situations and is encouraged to be used as a tool for research and evaluation to enhance decision-making for the diagnosis.

Since the developers couldn't secure proper testing with students clinically diagnosed with Social Anxiety Disorder, these were initial data on random students. Luckily, one female participant admitted to being clinically diagnosed with severe anxiety while the results had reflected in the self-evaluation. From the table, only three(3) students had none to moderate social anxiety, while the rest showed severe to extreme social anxiety. This might indicate that social anxiety could be more common in young adults but has remained undetected or traded off as normal behavior.

In the Calibration phase of the game, the testers were asked to pick which type of scenarios they preferred or could use help with the most. The top scenarios and their corresponding scene type are shown above, with most testers picking Casual Talk and Public Speaking as the skills they'd like to improve. Because of that, Scenes 11 and 13 mainly involved talking quests with strangers and reading quests in front of people, which were the top repeated scenarios.

Data during simulations where the users' moods are logged and averaged throughout the ten(10) different sessions per the varying scenes. Certain emotions are then classified into three(3) categories: positive, neutral, and negative.

The data results show that in Scene 11 involved a casual group setting in a classroom, users' top moods in the three(3) categories were Happy at 31.71% of the time, Neutral at 27.63% and Angry at 25.17%. Scene 12 simulations averaged 46.72% Happy, 21.9% Angry, and Neutral with 10.22%, while Scene 13, the only scenario with public speaking quests, gave 36.92% Neutral as its highest, 28.59% Sad, and 20.09% Happy. Lastly, Scene 21 had users experiencing 42.88% Happy, 26.29% Neutral, and 24.54% Sad.

Bringing all scenarios together, the top mood experienced by the users were, on average, 40.46% positive emotions, 34.3% negative, and 25.27% neutral, the top emotions being Happy, Neutral, and Sad, respectively. Though this could also include external factors where the testers were exposed to outside elements such as the environment and the

people around the testing area, the results show users experiencing the simulations with positive emotions ruling at least a 6% difference over negative emotions.

After analyzing in-game data, the post-survey game questionnaire was done with the Game User Experience Satisfaction Scale (GUESS), specifically the GUESS-18 Scale. The surveying of users as it is a psychometrically authenticated and inclusive scale that measures user satisfaction through the implemented sub-scales. GUESS usually contains nine subscales, but only seven subscales were applied to this survey. Those subscales are Audio Aesthetics, Enjoyment, Narratives, Personal Gratification, Play Engrossment, Usability/Playability, and Visual Aesthetics. It also uses the Likert Scale from 1 as Totally Disagree to 7 as Totally Agree.

Overall, 23 statements were applied to the game post-survey and categorized in the following table. More information on organizing the 23 statements into their respective subscales is found in Chapter 5: Appendices D.

Based on the survey results and their comments or suggestions, the respondents have decent feedback after running through the scenarios. However, some have mentioned the slow progression of the game due to the slow internet connection and the limitations of the chatbot aspect of the game to reset through the creation of new Text-to-Speech and Speech-to-Text resources on IBM's cloud services website. Some also mentioned enlarging some of the GUI during simulations.

The table above statistically uses the Top 2 box tool to simplify the user's survey results regarding the game's applicability to each subscale. Overall, it shows that most respondents agree that the game motivates them to overcome the challenges in the game. The game's easy mechanics with proper narratives and scenarios also was said to be suitable for patients at the early stage of therapy, although some parts still need some improvement.

Personal Gratification received the highest rating, with users feeling successful whenever they completed the conversation quests with the trained chatbots. Even one

conversation quest was said to be challenging and thus motivated them to try harder to complete the next one.

Usability/Playability and Narratives received a pretty good rating with the straightforward controls and menus coupled with immersive simulations placed in different settings following a storyline. Story contexts are given at the start of every simulation with the needed information to complete the quests per scenario given through hints and tasks in the GUI.

Visual Aesthetics, Play Engrossment, and Enjoyment nearly tie with each other on an averagely positive scale. This is with the help of Face It's 3d models and assets that make up the semi-realistic scenes presented to the users during simulations.

Lastly, Audio Aesthetics nearly average out on neutral agreement that the sound effects and audio cues give a sense of cohesivity with the simulations the users go through. However, some still want an improvement, especially on the NPC's voices and more varied background sounds and audio per scene.

CHAPTER IV

SUMMARY, CONCLUSION, AND RECOMMENDATIONS

Summary

The game was conceptualized to be made with Virtual Reality in mind. The decision was to allow a more immersive experience for social scenario simulations. However, due to the time and financial constraints and the developers' inexperience with the technology, it became a desktop game instead. Although this may have removed the greater immersive experience, this does not hinder meeting the developers' goals.

Before the game's development, the developers reached out to a professional psychotherapist to have an insight into the processes, ideologies, and treatment process

of exposure therapy and how the players can utilize the game within a therapist's session for a client's exposure therapy plans. With the given insights, the developers were able to conceptualize initial plans for how they would create the game's flow and the different possible scenarios to cater to different situations that a user might want to undergo in their exposure therapy sessions for social anxiety.

The initial concepts of the game started with a basic game that ran a Unity package that allowed Facial Emotion Recognition in-game. However, there wasn't an existing market for this feature in Unity that was open source, and other options required a fee. Thus, a Python flask server that ran the Facial Emotion Recognition (FER) script was created to allow real-time checks over a certain time of the user's current mood while going through the simulation. The game also started with a simple linear conversation check where Windows Speech Recognizer allowed the user to converse with NPCs in-game. This feature limited the user's ability to speak their way or ask questions, mainly focusing on receiving a "keyword."

The next iterations of the game focused mainly on creating high-quality NPC models and environments while maintaining a low poly count and detailed textures that would still allow lower-specced system units to render. Given that the game was initially conceptualized as a tool for psychotherapists to aid clients, it is unlikely that they would have a highly specced computer system in their office. Newer iterations also used IBM's cloud services for their chatbot, text-to-speech, and speech-to-text services made through an open-source Unity project to allow NPCs to recognize speech, understand dialogue or conversations, and be able to converse back with the user. However, with the integration of the open-source capabilities of the chatbot, it carried with it different limitations that take away from the user experience. Namely, its dependency on a stable internet connection as the feature is hosted on IBM's cloud services. Another is the limitations of the text-to-speech (TTS) and speech-to-text (STT) services' word limits, which would deactivate the service until the service is deleted and recreated or its monthly reset of the word count allowance.

After finalizing the integration of the much-needed features of the game, the developers reached the testing phase, where the game's limits were seen. The developers could see how there were still some minor bugs and functionalities that could be done better with more time. One of these bugs was the inconsistency of the usage of cameras within Unity. The Developers have found that built-in cameras, such as those found in laptops, generally work best. At the same time, any "plug-and-play" webcams would result in cameras shutting off and needing to restart the Editor altogether.

Another problem was the incompatibility of WebGL with the game. Due to incompatibility with additional devices like the webcam and mic input in the official WebGL, other scripts and third-party interventions are needed to deliver mic and cam input into the Web application and back again, lengthening the process of talking with the chatbots within the simulation. Nevertheless, according to the developers' requirements, the game passes as a minimum viable product.

Conclusion

The game was conceptualized to provide a tool that was not only less resource intensive than face-to-face exposure therapy but also an engaging one through the means of immersive gameplay. Creating a game with environments, models, and UI that are high in quality proved to be an excellent practice for developers to learn how to optimize assets while still delivering as close as possible to the original perceived models.

Though the developers were limited to the constraints of the project, they were able to find solutions to be still able to create the ambitious game. The final product still has a few components that need more fixing and modularizing. Still, integrating different solutions gave the developers a deeper understanding of the game engine and the technologies used. The developers also learned a lot of patience, skill, and knowledge on the topic and methods for developing future applications and games through the considerable research and trial & error done throughout the long duration of the project.

The project's unit testing has provided insight into how the game and other factors have affected the user's stress level when playing the game. The factors in and out of the game have induced varying stress levels for the different users to stimulate exposure therapy. Though the developers would prefer to have gathered more data, especially in the post-test aspect, the constraints mentioned hinders the capabilities to do so in a timely and proper manner. However, with the given data and different observations from the testing, the game does show promise with the initial project goals to be a tool in aiding in the process of exposure therapy for social anxiety. The results from the in-game survey also revealed how common social anxiety is for students, with 7 out of 10 respondents getting severe to extreme social anxiety. If the application is further developed, it can be useful for detecting social anxiety, especially in the early stages.

In regards to the viability of the application's use towards achieving the initial goals, adjustments to the different features and alternatives are a big benefit to creating a more optimized and smoother experience for the application to run well and focus on inducing stress within the simulations. With the given constraints and circumstances that the developers experienced, much more can be done to definitively say the effectiveness of the application in meeting the goals. However, with the data and feedback from the unit testing respondents, the application shows promise in achieving its initial goals.

Recommendations & Future Works

The developers recommend future works with similar goals to integrate the Python Flask FER(Facial Emotion Recognition system) server as a unity feature. Another option is to utilize FER Unity packages, as the flask server solution creates spikes in CPU usage and causes lags and bugs within the Unity Game. It also involves external files that the Flask FER system requires that the game builds need to locate within the system files, which results in errors or bugs in the final build.

The developers also recommend utilizing any available alternatives for the API chatbot feature. This recommendation is due to the system's complexity of manually

creating scenarios and conversations for the NPCs. This issue of the chatbot means the creation of levels/scenarios is laborious and time-consuming, and the capability of learning new conversations, phrases, and cues is to be encoded manually through the API's website. The conversations in-game are delayed due to unstable, which also prevents any offline potential of the game. Additionally, adding the chatbot behavior, like prompting the user for any responses when it reaches past a few minutes of inactivity from the user side when in conversation, is also recommended as part of the immersive experience.

Another recommendation is to utilize more realistic yet optimized models to add to the realism of the scenarios. Lip sync and eye movement are highly beneficial to create more realistic simulation models. Creating proper animations that prevent the models from clipping through themselves is also a benefit to maintaining the realism of the simulation.

Optimizing the code is highly recommended as the simulation is more CPU dependent, focusing on optimizing the game with the multiple features, especially the processing of FER data and the timely responses of the chatbots needed for the game. Spikes and lags take away the user's immersion and experience within the simulations.

Lastly, due to the financial and time constraints of the developers, a psychotherapist and psychometrician advisor is needed in the aiding of creating scenarios and planning out events, as well as having a defined scoring system that can aid in the data collection. The psychotherapist advisor aids in creating realistic scenarios for the users, and psychometrician advisors aid in quantifying and scoring the user performance based on the user's interactions, events, responses, and other factors that show their results.