Educational PTSD Game

## Table of Contents

|  |  |
| --- | --- |
| Executive Summary |  |
| Project Overview  Educational Component |  |
| User Stories |  |
| Planned Levels and Functionality Table |  |
| Obstacles  Prototype Requirements |  |
| Software |  |
|  |  |

## Executive Summary

The purpose of this game is to demonstrate to players in an interactive, safe environment what the symptoms of PTSD are and how they affect an individual. While this is an educational game, our main audience is adolescents who have experienced trauma. These users after playing and interacting with the levels are better able to understand how they are personally affected by the symptoms. We want these individuals to be able to understand what the symptoms are so they can better manage their responses after traumatic event exposure.

We plan to have physicians and doctors made aware of this interactive environment to inform through gameplay about the signs and symptoms of PTSD as a future goal. We hope to provide better patient-centered care with mutual understanding between patients and physicians.

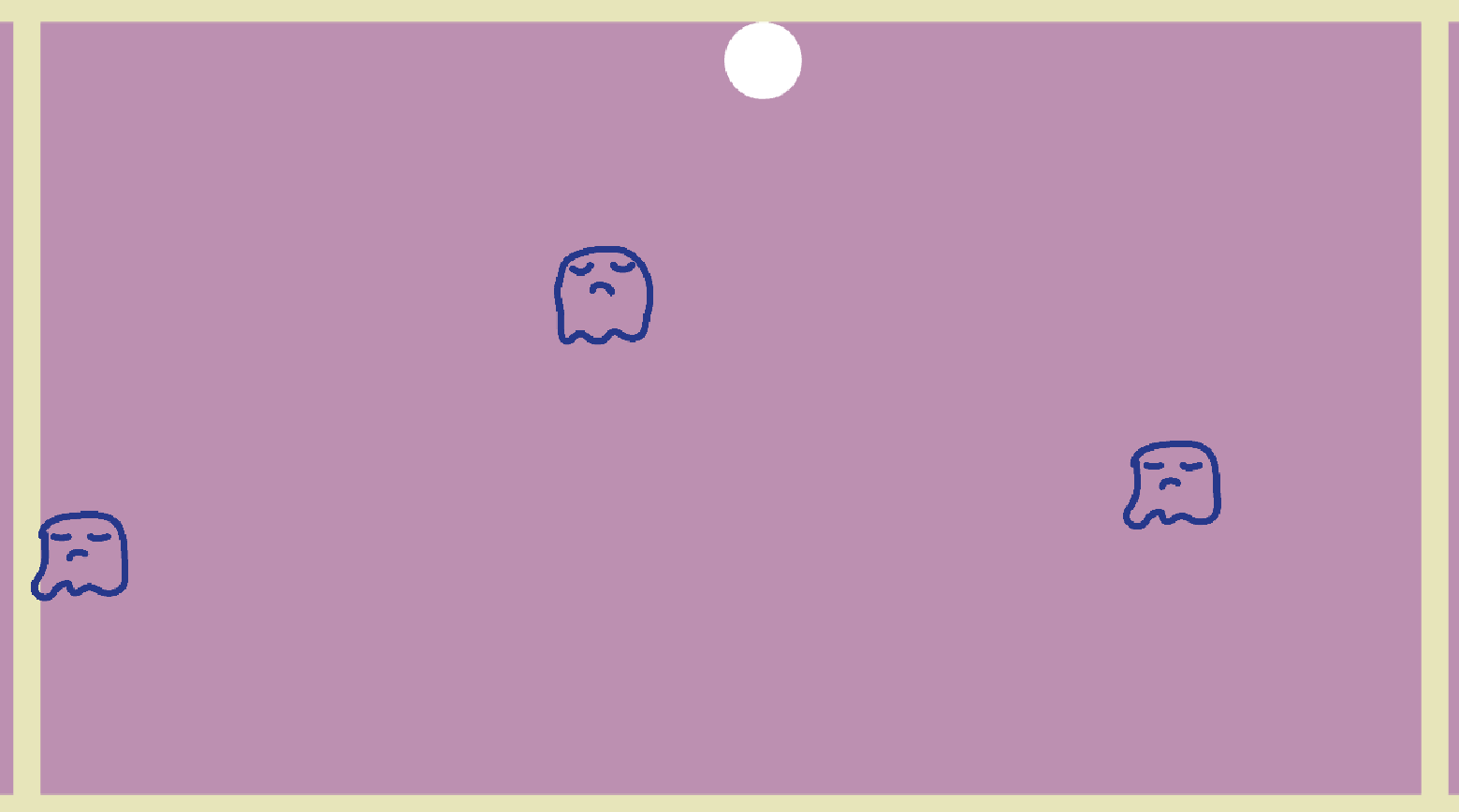
## Project Overview

The main environment this project needs to provide is one that does not produce any realistic events, figures, or images. **The project is to do more good than harm, and our team is being cautious with how we represent the player and symptoms as mechanics.** We do not want the player to feel too attached to their avatar on the screen. This is because we want the player to feel comfortable engaging in an environment that represents them abstractly. The player should have no fear of facing any visual or auditory representation that triggers discomfort.

The game will loosely follow the structure of the CPSS, or Child PTSD Symptom Scale. The game’s framework will be in a similar fashion to the 17 questions the CPSS contains. Meaning, approximately every level will be centered around 1-2 questions depending on the team’s decisions. For example, the first prompt in the CPSS is to *“Type the first word that comes to your mind when you hear the words* ***upsetting thoughts/images.****”* A level idea would be to include the word the user inputs on a monster-character and the objective is to chase that monster away.

Initial Prototype Level:

The initial prototype is an experimental piece by Katie to get used to developing for mobile. One mobile device control is the *accelerometer.* The accelerometer is a device of a phone that is able to detect orientation. When the phone is tilted left/right, forward/back, the game is able to move the player who is represented by a simple 2D circle. The objective of the prototype is to avoid the ghost enemies that appear on the sides of the game. Although no programmed end/win state, the player is able to maneuver around the map and tilt the phone to avoid the ghosts!



There is no set color scheme set for the prototype or game as a whole yet. Here are the items on the screen based on their color:

white: the player. the player is able to move all around the screen by tilting their device.

yellow: boundaries. For now, the screen boundaries are visible so when developing the prototype it was easier to see where they were.

dark blue: the ghosties/enemies. the enemies in the prototype only spawn in

Gameplay: the player tilts the screen to avoid the ghost enemies!

The player is designed to move the white circle around the screen by tilting their phone in 4 directions. The app is designed to be held horizontally, with the screen facing about 30 degrees upward to the user’s face. 

As seen in this image, the screen is slightly tilted upwards. This would be the natural ‘resting’ position of the user. The prototype jumps right in and barely moves when the user is in this resting position. When the user begins to tilt the phone *away* from them, the white ball moves to the top of the screen. All directions of the white ball move according to how the user tilts the screen.

## Educational Component

The overall gameplay will educate players on how the words associated with *“upsetting thoughts/images,” “nightmares,” “recurring events”* (to name a few questions from the Capstone document) affect an individual. For example, one mini-game idea would be to have the player chase away a monster, with the word input on the monster. Alternatively, the player could cover up the word associated with “*upsetting thoughts/images”* with clouds, or protect an image of a brain by shielding any words that come at it (as noted on page 9 of the Capstone document). The goal is to educate adolescent players on the symptoms of PTSD and help adolescents recognize how they respond after experiencing a traumatic event. We want adolescents who have had exposure to trauma to know the signs and symptoms that way they can better manage and explain their responses. Physicians as well would be able to learn about the signs and symptoms and be able to have a dialogue with their patients, thus resulting in better patient-centered care.

The game is meant to illustrate in an abstract way the possible effects of trauma with guided level questions and fun gameplay. While the game will take on a more serious role, the mechanics and actual gameplay will be made to be fun, pleasant, and offer enough distance from specific imagery, subjects, and the like so player’s are not distraught or uncomfortable. Each level/question segment will have a unique question that stems from the CPSS, a child PTSD questionnaire.

## User Stories

User stories are a way to write the requirements of software from the perspective of the person who is using it. By writing out the requirements in a statement form from a user’s perspective, it helps keep the focus on the software’s requirements. Here is a list of user stories from the perspective of its users:

1. As an adolescent, I can play this game and feel calm/at peace.
2. As an adolescent, I can input a word according to the prompt, and also have the option to skip.
3. As an adolescent, playing this game will show me how trauma impacts the mind and body.
4. As an adolescent, the game is fun to play and has replay-ability (high score based mostly)
5. As an adolescent *who has experienced trauma,* I can confidently identify my responses after seeing how trauma affects the body and mind.
6. As an adolescent *who has experienced trauma,* I can comfortably discuss with a certified physician on my experiences and responses.

Overall, regardless of if the adolescent has experienced trauma or not, the game will be educational, fun, and will not provide any specific imagery, and be a calming environment meant to safely educate.

## Planned Levels and Functionality Table

|  |  |  |
| --- | --- | --- |
| Level Name/Number | Guided CPSS Question | Mini-Game Idea and Description |
| Level 1 | Type in the first word that comes to your mind when you hear the words **upsetting thoughts/images.** | i. Chase away the monster with the word on its body  ii. OR use a happy cloud to cover up the thought bubbles with the word  iii. OR use a shield to protect a cartoon brain and chase the thought bubbles with the word that want to go toward the brain away  --------------------  Player can chase away enemies from a central target. This would combine the 1st and 3rd ideas! There could be a central ‘tower, ’ or brain the player would have to defend. The player could then chase away monsters that come too close (with the word on them). The player would be able to tilt the phone to move, then maybe tap a button on the screen to ‘spook’ or force the monsters away! Just being near the monsters isn’t good enough, they need that extra shove from the player! |
| Level 2 | Type in the first word that comes to your mind when you hear the word **nightmares** | * there may be an opportunity for a transition, to make gameplay based on waves of enemies. add layers of difficulty to the tower defense-ish first level, then because the player now associates the word with nightmares, the player has a limited field of view as the level has grown dark. |
| Level 3 | Type in the first word that comes to your mind when you hear the words **recurring events/reexperiencing** | * as a stand-alone mini-game, the player could be encountered with a never-ending ‘time loop’, meaning the section they are in will allow them to move forward but they will loop back to the start. The theme of the mini-game would have to do with repetition or deja vu, but in a way where it will not stress the player out. * One way to introduce this is to have the player loop in a room with 4 portals at the North, South, East, and West. Any portal brings them back into the same room, just through another portal. With the waves possibly still active, the player will have to push a colored circle that corresponds to a portal color inside of it to destroy it. Once all portals are destroyed the player moves onto the next level |
| Level 4 | Type in the first word that comes to your mind when you hear the words **feeling upset** | * I’d like to make a ‘feeling down’ themed level where the player has to avoid being ‘down’, so they have to go higher up in the level. This could be a jump game where you have to keep moving up the levels using the platforms and avoid enemies.   (doodle jump game)   * The player will have to avoid the word otherwise they will fall a bit lower in the stage. The player has to jump to the top as fast as they can as the camera pans up. |
| Level 5 | Type in the first word that comes to your mind when you hear the words **breaking out into a sweat and heart beating fast** | * The transition from the last stage to this level would be the player would pause on screen a second and you would hear the heartbeat, and see some sweat from the avatar. * There will be a cartoon heart that is sweating and moving really fast. The players need to collect ice cubes on the screen and place them near it/on it to slow the heart down.   1. This is to simulate a vagal maneuver   * the player would have to bounce cubes onto the heart that fall on screen and avoid the word as it falls as well. |
| Level 6 | Type in the first word that comes to your mind when you hear the words **trying not to think about, talk about, or have feelings about an event** | * once the last mini-game finishes, the heart will be content and happy! The camera pans back up to show the player in a new area. the player will be trying to relax and watch TV * in this level, the player will be trying to tune a tv (a really old school one with nobs and buttons) to a channel that does not display the word they typed in. the player will have to turn some nobs and tap the antenna of the TV to get the channels to show clearly. The word will sometimes be on the channels and the player quickly needs to change them.   Does this teach adolescents to avoid the problem instead of confronting it?? |
| Level 7 | Type in the first word that comes to your mind when you hear the words **trying to avoid activities, people, or places that remind you of an event** | * the player could be using a map to navigate around the stage and be constantly encouraged and praised for exploring/trying new things? This could be with dialogue from NPCs (non-playable characters, and they will just be shapes) |
| Level 8 | Type in the first word that comes to your mind when you hear the words **not being able to remember an important part of an event** |  |
| Level 9 | Type in the first word that comes to your mind when you hear the words **having much less interest or doing things you used to do** |  |
| Level 10 | Type in the first word that comes to your mind when you hear the words **not feeling close to people around you** |  |
| Level 11 | Type in the first word that comes to your mind when you hear the words **not being able to have strong feelings (unable to cry or feel happy)** |  |
| Level 12 | Type in the first word that comes to your mind when you hear the words **feeling as if your future plans or hopes will not come true** |  |
| Level 13 | Type in the first word that comes to your mind when you hear the words **having trouble falling or staying asleep** | i. A humanoid avatar trying to sleep on a log, but cannot fall asleep, so the player needs to roll the avatar onto a bed, turn off music, and move all the distractions (i.e. phone, computer) off the bed before the avatar can fall asleep 1. Can let them try to figure out what needs to be done instead of giving them instructions a. Give them hints after 5 seconds if they are unable to figure out what needs to be done |
| Level 14 | Type in the first word that comes to your mind when you hear the words **feeling irritable or having fits of anger** |  |
| Level 15 | Type in the first word that comes to your mind when you hear the words **having trouble concentrating** | i. The avatar on screen will be trying to make a to-do list, but keeps on stopping, so the player needs to remove distractors (such as music, phone, etc.) 1. Can let them try to figure out what needs to be removed instead of giving them instructions on which ones to remove a. Give them hints after 5 seconds if they are unable to figure out which ones need to be removed |
| Level 16 | Type in the first word that comes to your mind when you hear the words **being overly careful** |  |
| Level 17 | Type in the first word that comes to your mind when you hear the words **being jumpy or easily startled** |  |

## Obstacles

The primary obstacles to be faced are program features and ensuring the user has fun in a comfortable environment while still learning about the symptoms of PTSD. So far, the ideas waterfall off of each other meaning after each level, it transitions into the next guided question with ease, then starts a new mini-game. The art and gameplay I am trying to keep not so different from the previous level so there is a clear gameplay narrative. Also programming a system that includes all the guided questions, mini-game conditions and pause feature would require planning ahead. UML diagrams for the algorithms and data structures will ensure the game will be developed with minimal bugs, and run smoothly!

A recent obstacle involves the actual capability of my laptop computer and touchscreen keyboard. My laptop is beginning to act up and freeze when running Unity and testing the game. The touchscreen keyboard does not appear on iOS device when using Unity Remote. Other potential obstacles would be how objects in screen interact with each other and making sure unused objects in the scene are deleted, and making transitions to new mini-game levels clear.

## Prototype Requirements

The prototype refers to the end state of the programmed levels. There does not need to be an end state, the prototype will be a better version of an MVP. Mostly concentrated on the gameplay and mechanics, the prototype will not have fancy or completed visuals or art assets. The artwork will mostly be placeholder assets but have a cohesive color scheme. The game will be a calm and fun experience that educates. Here is a shortlist of overall development requirements the prototype will need to meet:

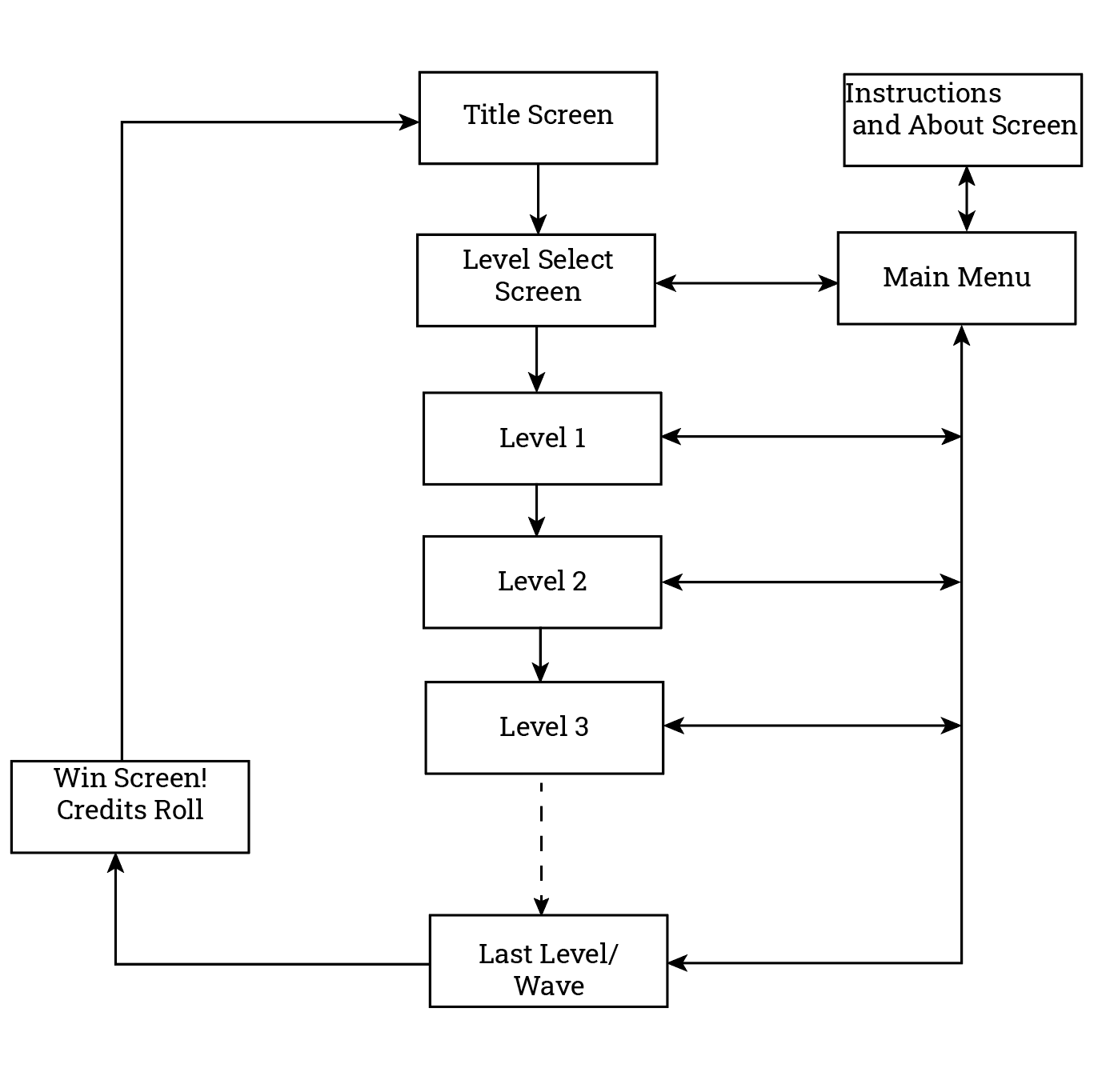
1. 5 levels finished with their own mechanics/additions to previous levels.
2. code system in place to control the player, enemies, and end-states.
3. the player will be able to type in input using the keyboard
4. levels will have a soft color scheme

## 

## Software

The main software we will be using is the Unity engine for the development of the game and Adobe Photoshop/Illustrator for any art assets. In addition, Visual Studio will be the primary method of programming with the help of some Unity assets like Lean Tween (potentially) to help add some small animation and movement details to the UI.

## User Interface Design

1. *Navigation Map for Levels*

The player will be able to open the main menu screen from any level to check controls, and read a bit about the game. Later on in development, the user would be able to save which level/wave they are on.

The Title Screen is just the splash screen with the game’s title, and you press start to begin.

Level Selection allows users to jump back to any level they have saved in. When selecting a level the player jumps back in, starting at the beginning of the level and will have to re-enter their word for the guided question from the CPSS.

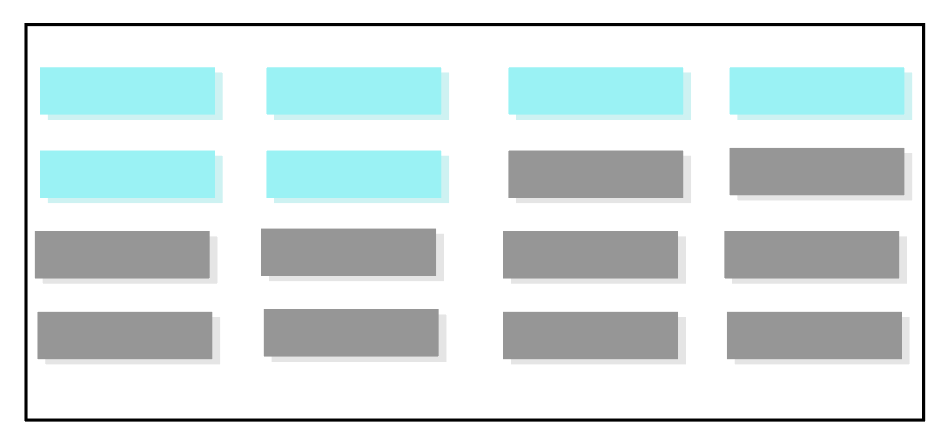
Instructions and About screen includes the instructions to play the game and the purpose of the game.

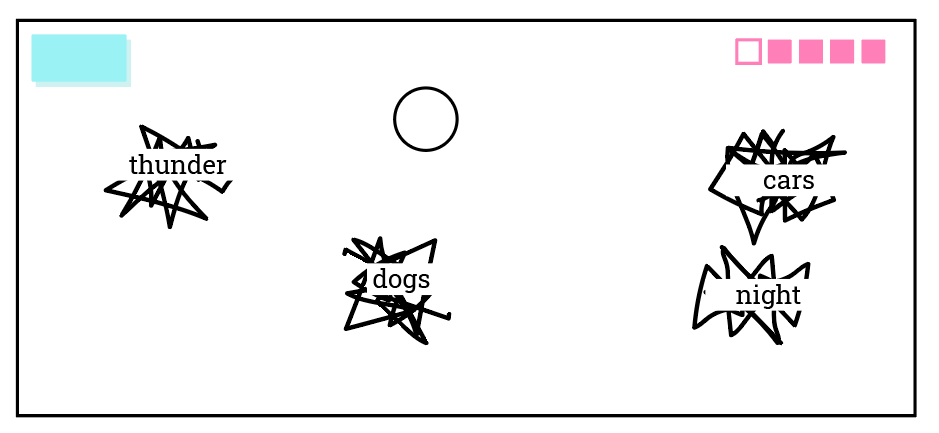
Once the player has completed all levels they will be congratulated with a win screen where the credits will roll.

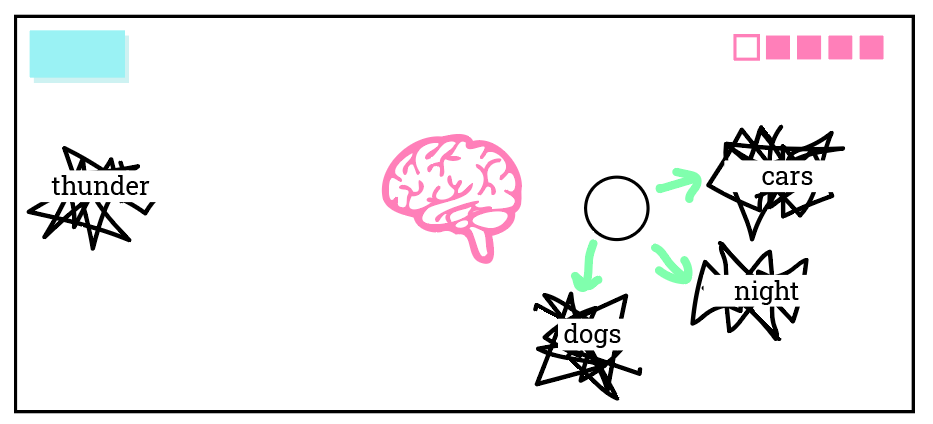
1. *Wireframes/Mock Ups*



The Title Screen here will include the title of the game and welcome text. The light blue rectangles are clickable. The player will have to click the button to start!



In the level select screen the player will only be able to access the levels they have already completed or saved in. They greyed out rectangles represent locked levels the player has yet to complete or get to. 

The levels will all contain a main menu button on the top left and health blips on the right. As the player comes into contact the words on the angry scribbles they will lose health blips. Of course the player will be in different levels that look different, so this level is the generic one to show one type of enemy that represents how the word will be in the game. 

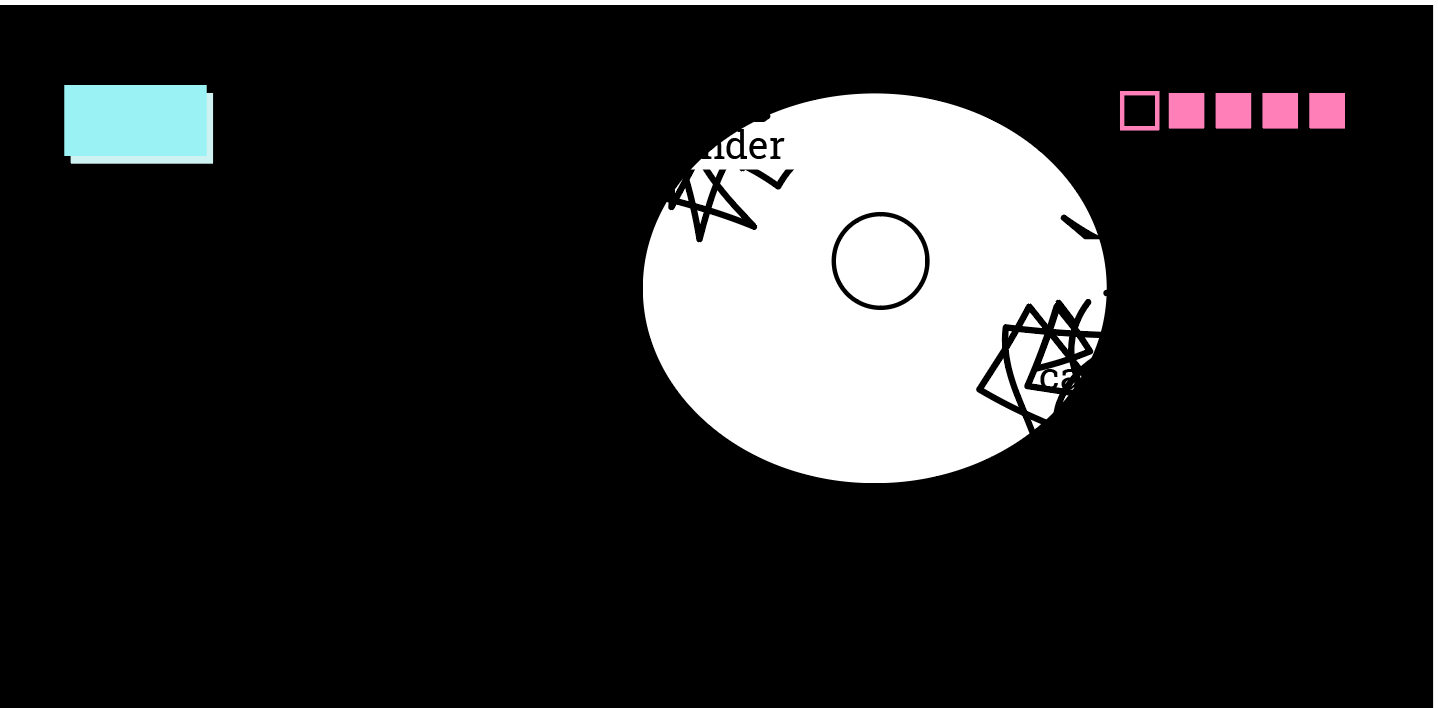
In this sample level, the player can be protecting a cartoon brain (one of the level ideas) and will have to bounce the angry scribbles with words away! The scribbles will appear all around the screen converging to the center, and the player will have to maneuver around to prevent any damage coming to the brain.

## System Design and UML

General Flowchart for Level Progression (subject to change):

1. *UML for Player*
2. *Game Manager Object*
3. *Enemies*

The enemies in the game are the word the player types in, attached to something abstract. Since we don’t want to show any imagery that would trigger a response, the less specific stimulus the better. Representing the words with angry scribbles would allow there to be an animation where the scribble would jitter and move, and let the player know that the scribbles are not friends! In some levels, the scribbles would need to be avoided, chased, bounced away, or other. This is a basic representation of how one enemy (of maybe a couple more, but that’s to be determined) would look in the game.

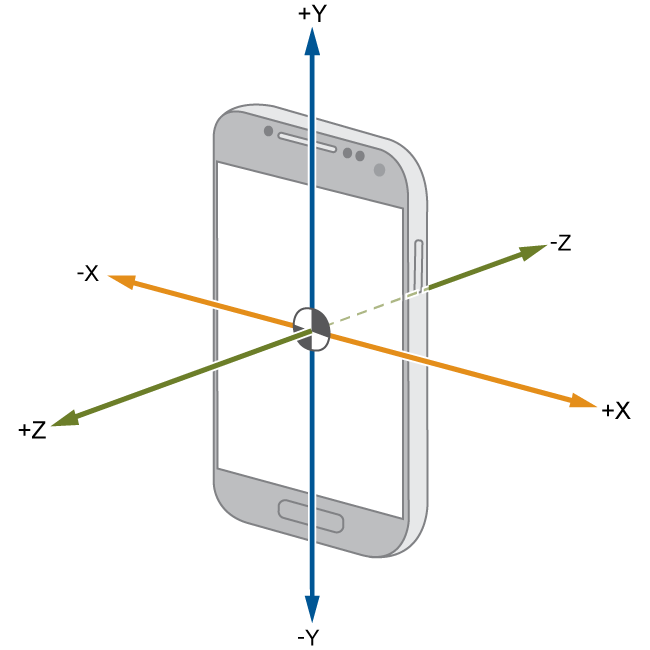
Going off the wave idea, if enemies come in waves, every so often the player would be prompted with a CPSS guided question to enter a new word. Then with each question, the enemies will get faster and the player will have new abilities and challenges each level. For instance, the first wave has the normal scribble enemies, and you have to avoid them. They only appear on one side of the screen. Then the second wave will have more enemies from 2 sides, and they get a bit faster. When I tested how many sides became difficult, having enemies appear on two sides was really easy. Then with the third wave, players will have limited visibility - everything will be in ‘black-out mode’ where the player has a small spot light on them.

It would look something like this. A spotlight on the player with enemies barely visible from the sides.

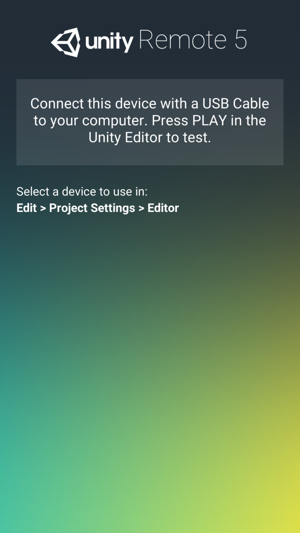
## Gameplay Mechanics

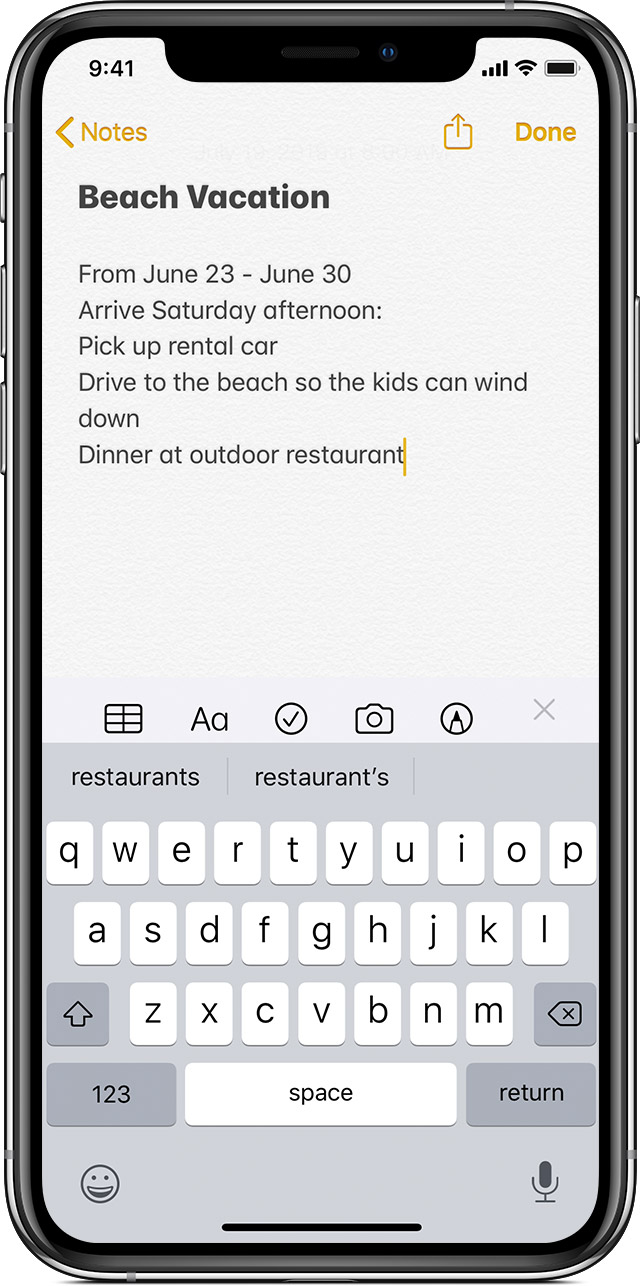
The mechanics of each level or ‘wave’ will be slightly different and rely on the capabilities of a mobile device (primarily iOS devices). There are a few key devices that are built into a mobile device that will be used throughout the game:

**Accelerometer:** This device stores where and how the phone is angled. As seen in the diagram below, there are 3 axis’ that tell the device how it is being tilted and where in 3D space it is in. Axis’ X, Y and Z are used in the game’s code to allow the player to move in 2D space.



**TouchScreenKeyboard:** the game would allow the user to input a word they associate with one of the questions in the Child PTSD Symptom Scale; ex: “Type in the first word that comes to your mind when you hear the words having trouble concentrating”.

\*\*While Unity does have documentation offering support for GUI TypeFields (where you select a box and can type your input) there are no online examples or APIs that offer guidance to how I have my Unity Remote 5 application on my iOS set up. I am still searching for documentation and tutorials for help. Unity Remote 5 allows me to use my laptop, running Windows 10, to connect to my iPhone X with a USB and emulate the game on the phone. Many of the forums I’ve seen offer solutions for Android developers but none for iOS. 

TouchscreenKeyboard Input test screen keyboard

What is supposed to happen when you click the type field that says “Type here” is the mobile keyboard of the iOS device should pop up and allow the user to type in a word. The problem is the touchscreen keyboard doesn’t actually appear on my iPhone when I’m using Unity Remote. A few Android sources have said that this is because the ‘TouchScreenKeyboard.Open("", TouchScreenKeyboardType.Default, false, false, true);’ opens the default keyboard of the device, but since the Remote app just emulates the game it isn’t quite connected to the device’s keyboard itself. The keyboard command does not work in the Unity Editor using Remote. But the application will work when building the app to the device itself. However, because building to iOS requires a Mac which I do not have, I will build to Android and test on my sister’s Android phone to see if the keyboard pops up.

**Touch Controls:** mobile devices can easily detect touch input. Touch input can be used to activate abilities, move objects, or interact with them, or interact with any UI buttons like a Main Menu button.

Resources:

1. Android TouchscreenKeyboard does not show up in Unity Remote - <https://stackoverflow.com/questions/36046136/unity-android-touchscreenkeyboard-does-not-appear>
2. Unity Manual, TouchScreenKeyboard - <https://docs.unity3d.com/2018.4/Documentation/ScriptReference/TouchScreenKeyboard.html>
3. Unity Manual, TouchScreenKeyboard OPEN - <https://docs.unity3d.com/2018.4/Documentation/ScriptReference/TouchScreenKeyboard.Open.html>
4. Unity Scripting, InputField - <https://docs.unity3d.com/2019.1/Documentation/ScriptReference/UI.InputField.html>
5. Unity Manual, InputField - <http://stalhandske.dk/UnityDocs/Manual/script-InputField.html>
6. Unity Manual, Mobile Keyboard - <https://docs.unity3d.com/Manual/MobileKeyboard.html>
7. DIY UI Keyboard - <https://www.youtube.com/watch?v=8324jUUYbKk>

I will most likely have to create a GUI keyboard like in resource #7 if the Android build does not display the mobile keyboard. If the Android build on the Android phone *does work and display the keyboard,* I can most likely assume it will work for iOS devices as well.

## Conclusion

to be filled out when the game is approaching or is completed.