

Treamcast



An experiment in vr by Andrew Schons



View the game manual At
xedouteyes.com/Treamcast

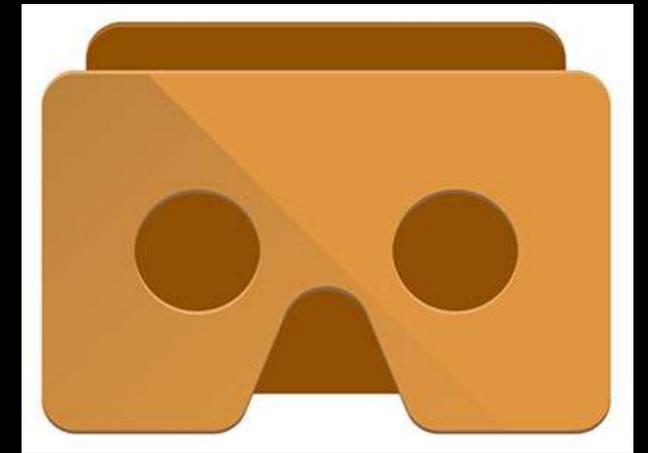


Dreamcast

MP3
PLAYER
FOR
Dreamcast

Cardboard & VR

- Google cardboard is a vr headset that was created by google as a cheap joke & ended up getting stupid popular.
- Google cardboard works by having two slightly different images on a phone screen & then that phone screen is shoved up against your face and it looks 3D.
- Through the use of a phones gyroscope & Bluetooth controllers, crude head tracking and movement in 3D spaces is possible, which prevents cardboard from being a glorified virtual boy.
- Most cardboard headsets have a single button that either taps a fixed spot on a phones screen or uses a magnet as a button trigger.
- VR, or virtual reality, is a technology that enables someone to immerse themselves in a reality not of their own.
- Vr didn't begin with computers. Devices enabling people to feel as if they are in another place have existed for a really long time



Is VR the future of our society?

The gameboy & it's obvious successor, the virtual boy.



I dunno.



The goals of this project

- I am going to community college to learn about game design. I took a class in unity, had a great time with that, and wanted to learn more about unity.
- My big learning goals:
 - Get a better grasp of unity
 - Learn more about modeling and animation in maya
 - Tell a story through a game
- I had a hot and heavy love affair with vr for a week after getting a new phone, which lead to me deciding to make this game for cardboard.
- I don't regret this choice, as it lead to me developing a deeper understanding of the engine, but creating a game for vr limits accessibility greatly.



Problems with most VR games

Perceived problem

- Forward motion is awful and nauseating.
- The google cardboard button sucks and has to do way to much.
- The ones with the anime girls are never as hot as they say.
- If combat is implemented, it's clunky.

My solution

- Eliminate forward motion, force user to remain stationary.
- Draw influence from Atari games. Make the button a simple clicker. Eliminate any unnecessary menu stuff.
- Stop looking
- Just don't have combat.

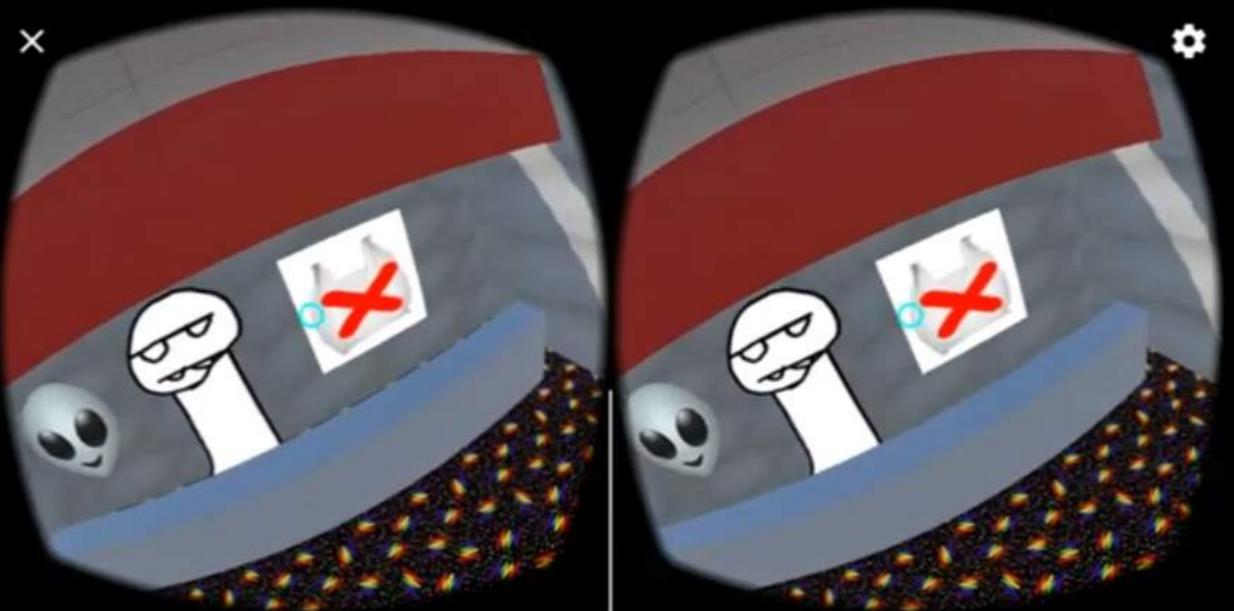
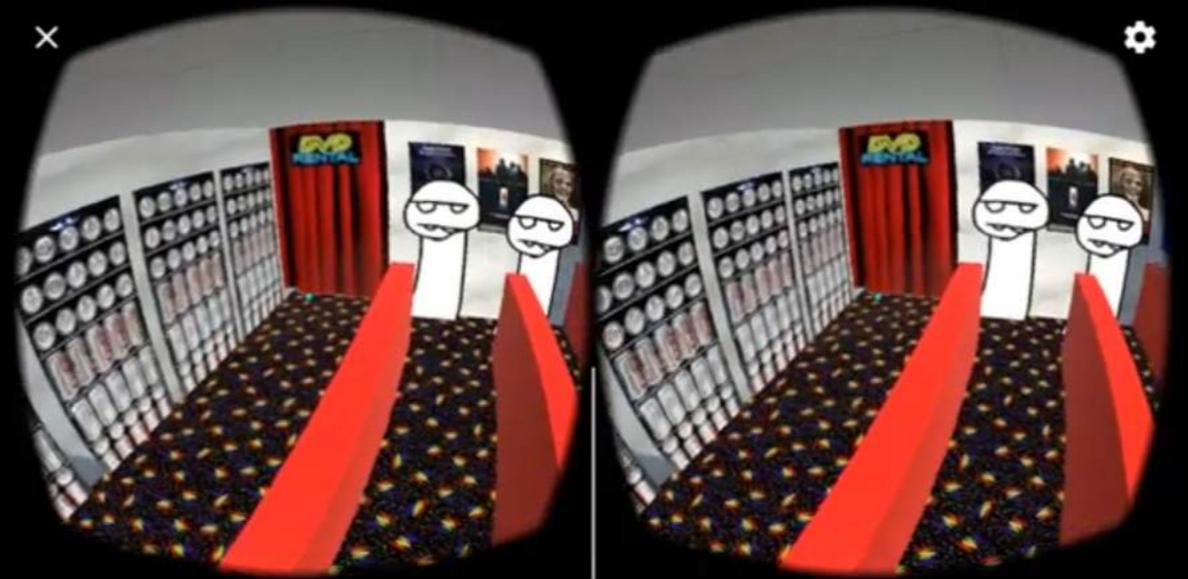
First Demo: Target Shoot

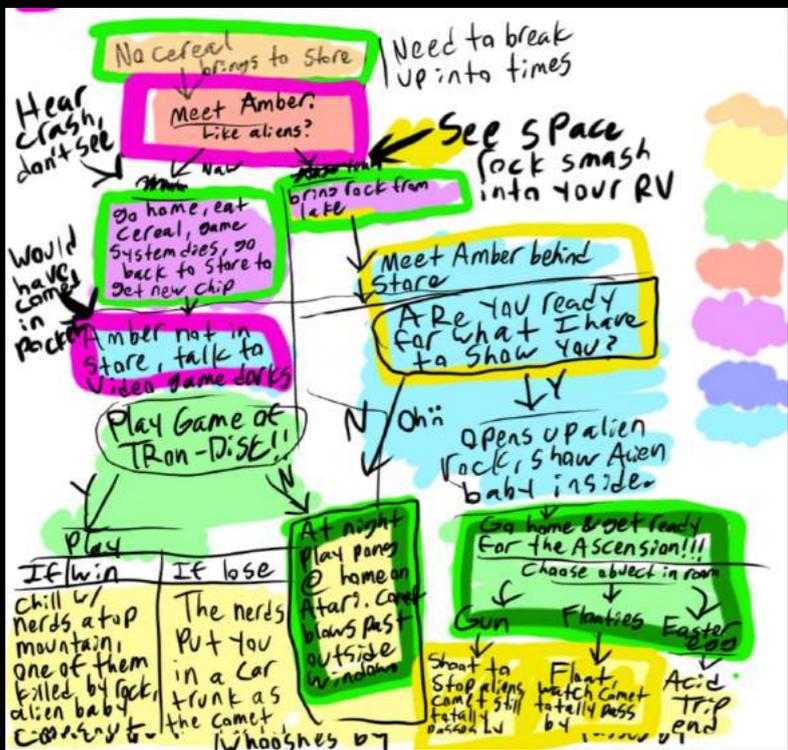
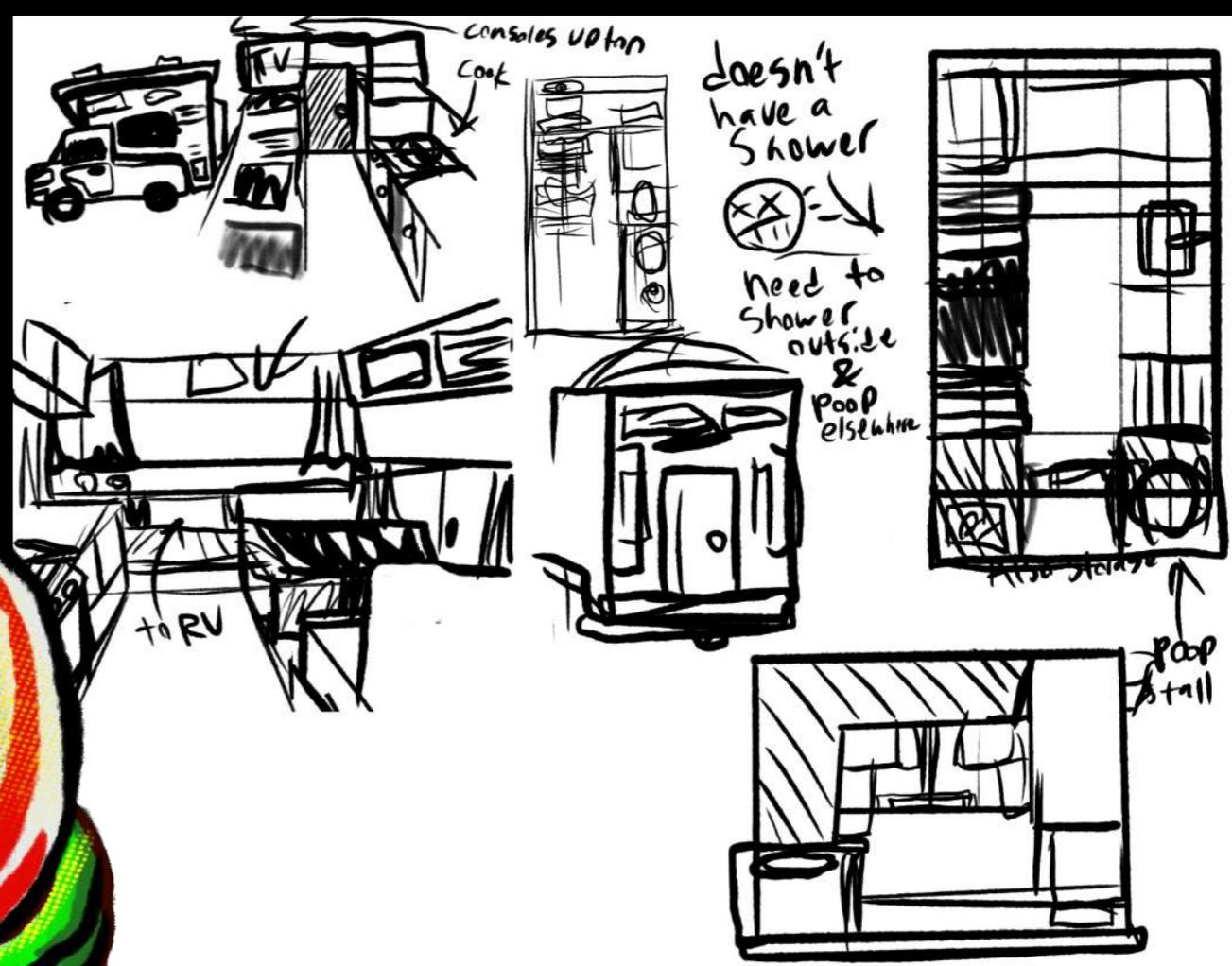
- Goal of the game was to hit targets and knock down cans using Google Cardboard as your weapon of choice for this purpose or whatever.
- I chose to do this because it had a simple end goal.
- The process of creating the target shoot demo taught me a lot about sounds in unity, lighting, animations, refamiliarizing myself with maya, and gave me an idea of how my basic method of game interaction would work.
- Setting up the cardboard sdk was several days of hard work.



Convenience store demo

- This was the start of what my final game would be
- I had a very vague idea of what the story would be at this point. Wanted to prioritize creating a functional set of game systems over a complex story.
- Used what I learned from the target shoot demo to make a simple conversation system.
- After I got this demo working & learned more about audio sources, I began drafting story ideas, making Placeholder art and sketching environments.





Then the snow happened, and all of my ambitions melted away for a few weeks.

Robodaddy Demo

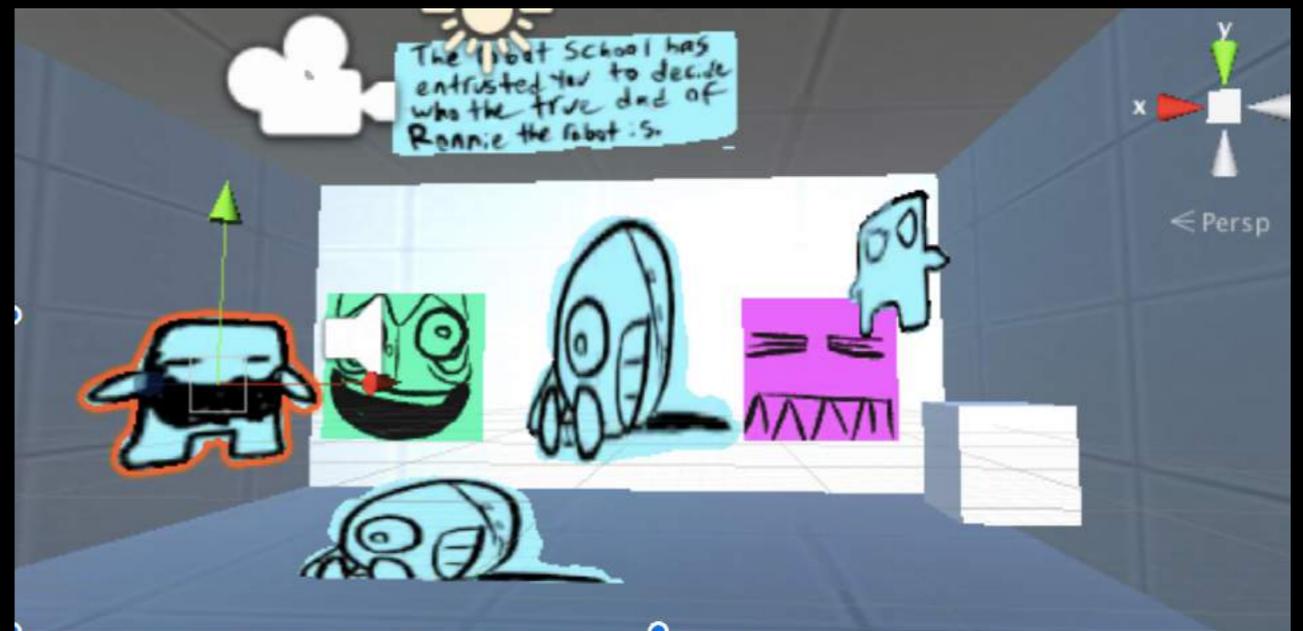
- Robodaddy was an awful game demo that taught me more about randomizing numbers in unity, figuring out how functions work in c++, instantiating prefabs, working with the built in sprite editor, and most importantly STATIC VARIABLES
- Robodaddy was a game where you have to choose which parental guardian to entrust the robobaby to. If you choose the right one, you are swapped to the next room to see the happy baby. If wrong, you see the bad guy in the next room.
- Sprites of the robodaddies and their qualities were placed in an array and randomizes upon awaking.
- Because of what I learned from this demo, I was able to figure out how events from one room of the game translated to events taking place in another, and changing the outcome of my game based upon that.



The Robot School has entrusted you to decide who the true dad of Ronnie the robot is.

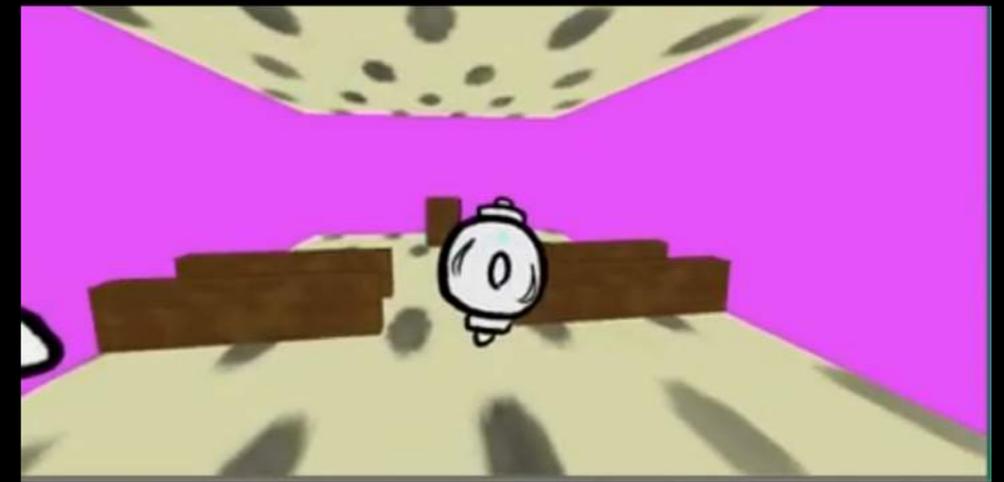
See what has come of your choice

Ronnie has Starved from your indecision



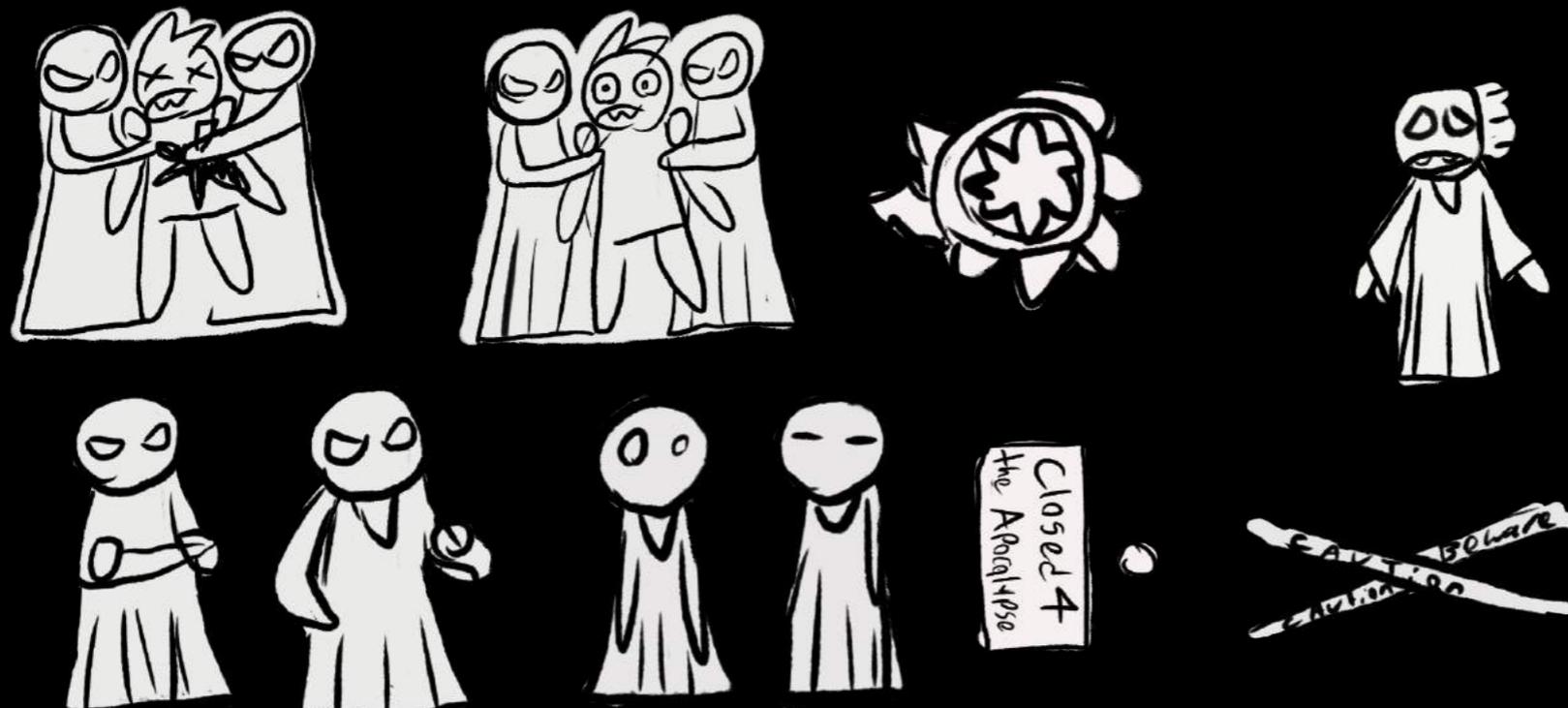
Back to the main game

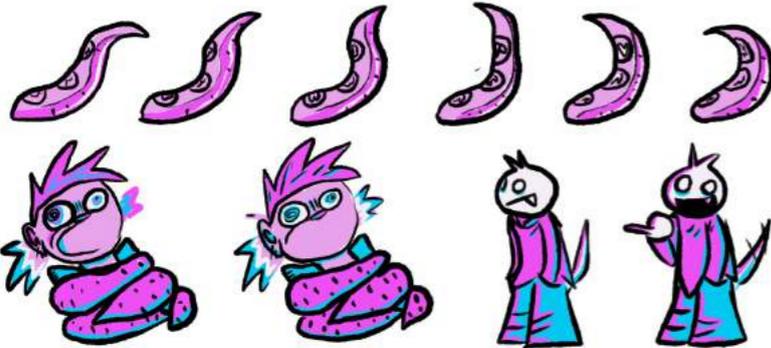
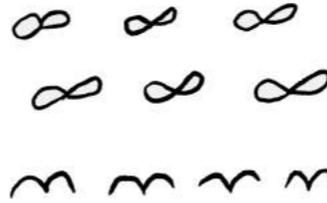
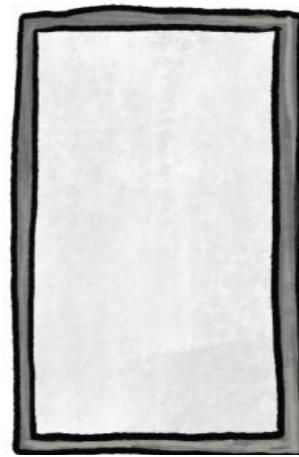
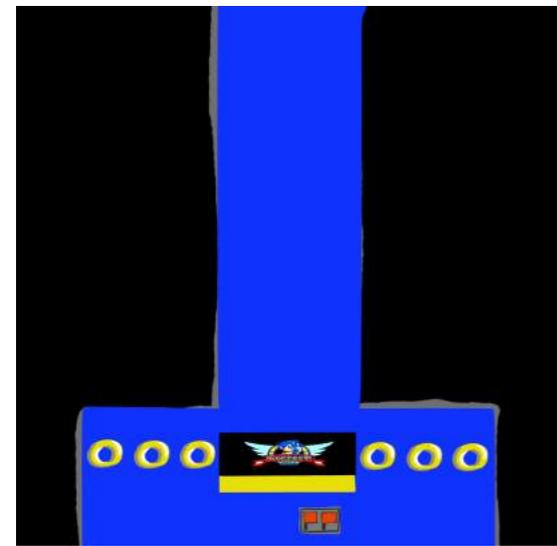
- Using the conversation demo and what I had learned from robodaddy, I began really fleshing out my game. I made several rooms where different things would spawn depending on if Booleans in a master variable sheet were marked true or false.
- I stopped regularly debugging the game on my phone because sending it to my phone to build was a real pain and I didn't want to deal with it.
- I spent lots of time procrastinating and looking at garbage on the internet and staring at crap on eBay. I did not shower for days and I became smelly like an ape.



Lots of work happened

- After making the framework and most of the art of a very basic a to b type of game, I decided my game was boring. My game really sucked.
- I got rid of most of the other storyline stuff from my outline idea, and it had become a drag.
- I added a storyline involving a cult in the town and all of this other stuff extremely last minute, and my game became much harder to finish, but more rewarding.





Make sure these sprites are all lined up to the bottom



Tech Demos! They can be used for other things as well.



Blah blah blah

- I shifted focus from 3D elements to sprite based art, I learned more about working with unity's update function and rewrote much of my original code from this. I also started learning about sprite animations and other junk.
- Blah blah blah, stuff happened. It really was a lot of work even if it doesn't seem like it 😓



Sonic demo

- I stole some sprites from sonic cd and made a demo where you would click a thing and that would make sonic run or jump or whatever. There's not much to it, it's just there.



Here I am today

- I got the game. I don't feel like it's finished, as several animations haven't been implemented and there are still several placeholder elements. However, it functions and all endings are accessible.
- I've learned what I wanted to learn from this project and don't feel like I have much left to learn. After I add a few more things to it, I'll add it to my portfolio and call it a day.
- Or a month. Or 3. Or whatever.



The final game

- There are over 8 unique settings in my game: the inside of an rv, a convenience store, a church, the outside of a town, the lakeside, a game menu, an Ending, and other stuff.
- My game has beautiful sprite art, voice acting and stolen music, and it's very cool.
- It actually works!



Reflection on project

- I feel that I learned a lot from this project and had a lot of fun doing it
- Rewriting code and redrawing placeholder art took up a lot of time, but the code rewriting contributed to my learning, even though it detracted from the timeline of the final project.
- Using unity documentation as a reference was my greatest learning resource. Due to the snow and changing project goals, I didn't follow the project schedule super closely.
- Vr is cool but it kinda sucks.
- I had fun!



End