

MAND: INTERACTIVE THRILLER

By: Jasmine Fan

FICTIONAL EXPLORATION



MAND



MAND



01

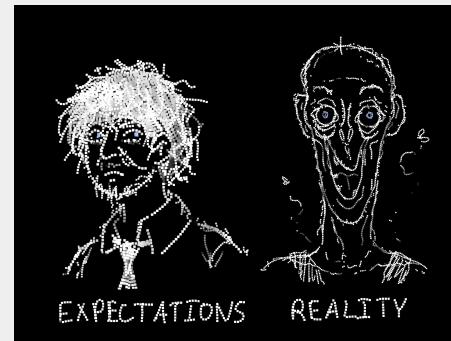
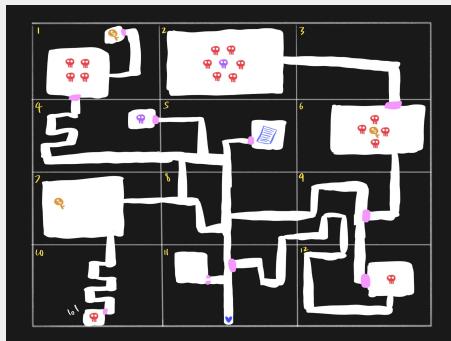
Revisions

What I have changed from the
prototype



What I've changed

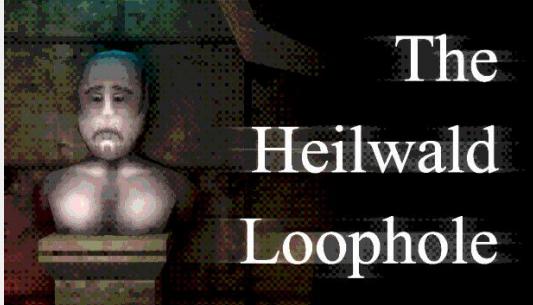
- Completely scrapped p5js, switch to Unity
- Scrapped original story
- Scrapped Original Character and Monster designs
- More "escape room" esque (more interesting graphics, places to look around rather than a blank black and white map)



02

Inspirations and Goals





The Heilwald Loophole
by Jan Malitschek on
Steam



Inspirations

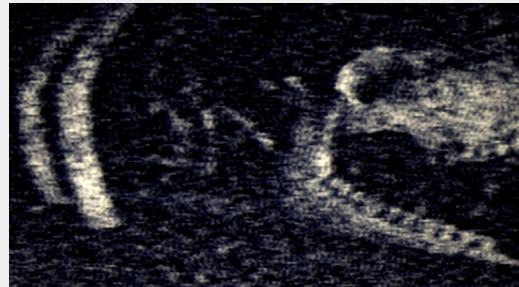


ENA by Joel G. on
Youtube (video series)

(and other PS1 style
niche horror games)



Iron Lung by David Szymanski on Steam



Goals

- Learn Unity
- Create own sound
- Character Design
- Story Development
- Unity to Arduino to Human Interaction

I quickly found out that I was pulling myself in too many directions, and ended up neglecting some aspects of my goals, you'll learn why later

Original Goal:

Challenge the classic gamepad interface and the companies who use it to bring on the discussion of making new controllers for games

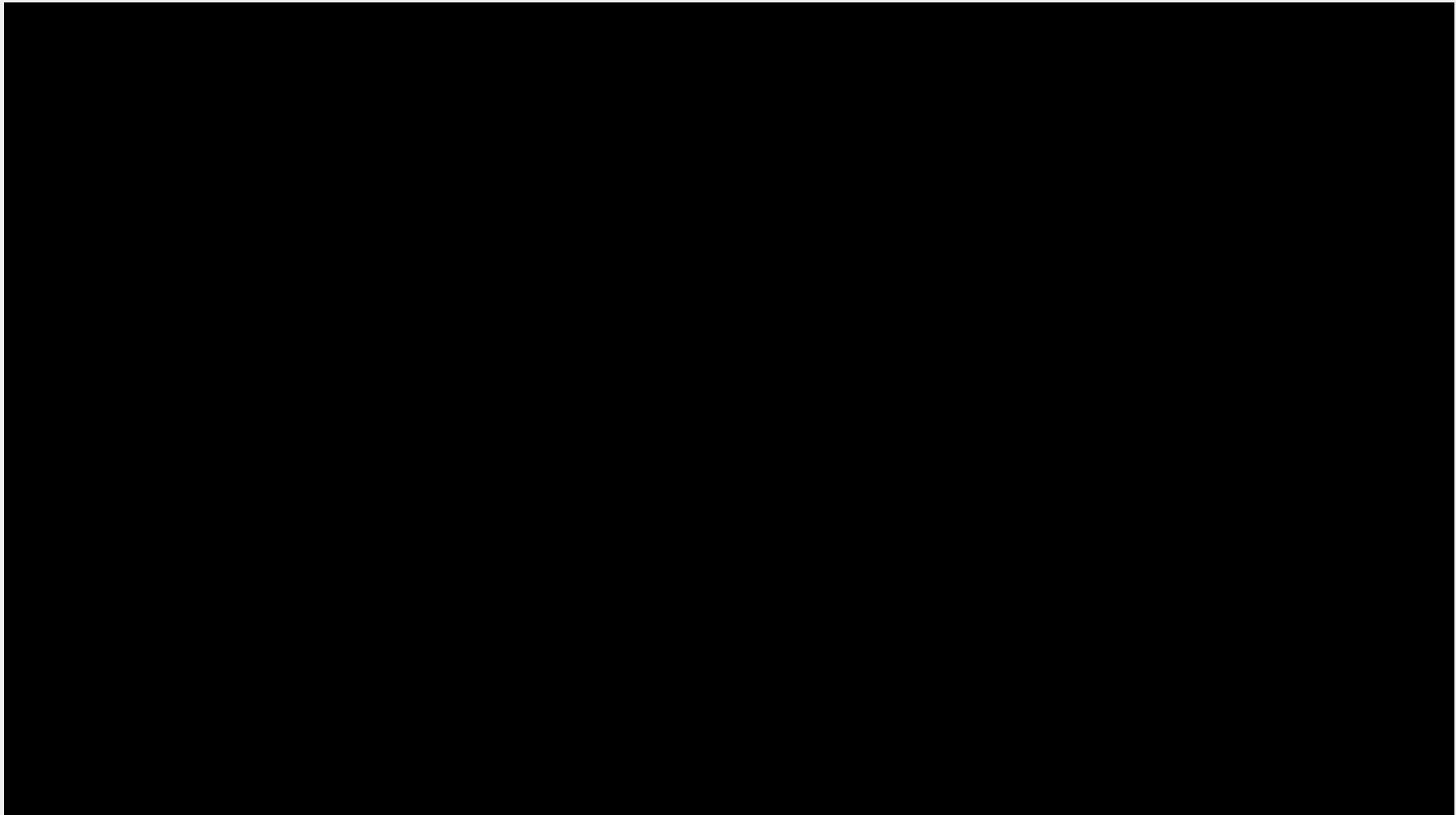


03

Trailer

'Fake' promotional video





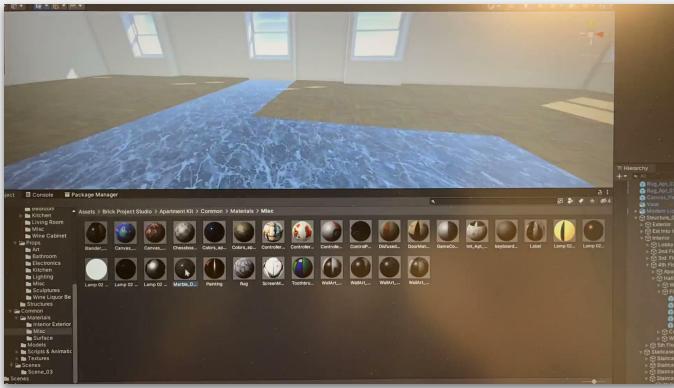
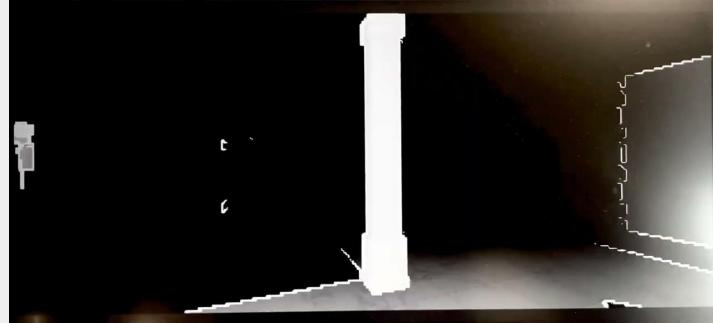
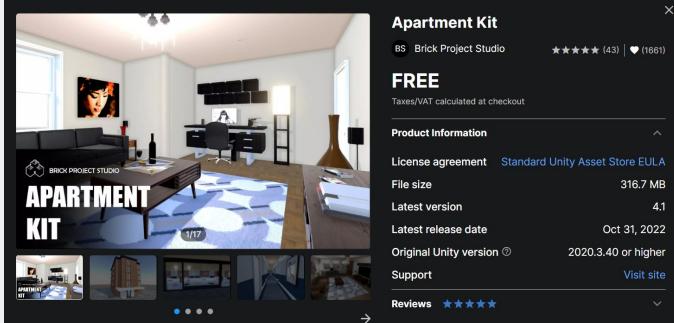
04

Code, Documentation, and Hiccups

(My process)



Environment



Environment

Post Processing



Pixelation

DT Dmitry Timofeev

★★★★★ (59) | ❤ (616)

FREE

Taxes/VAT calculated at checkout

Product Information

License agreement Standard Unity Asset Store EULA

File size 96.5 KB

Latest version 1.0

Latest release date Jun 30, 2016

Original Unity version 5.3.5 or higher

Support Visit site

Reviews

★★★★★



Outlines

TD The Developer

★★★★★ (9) | ❤ (111)

FREE

Taxes/VAT calculated at checkout

Product Information

License agreement Standard Unity Asset Store EULA

File size 365.2 KB

Latest version 1.0

Latest release date Oct 15, 2020

Original Unity version 2019.4.11 or higher

Support Visit site

Reviews

★★★★★

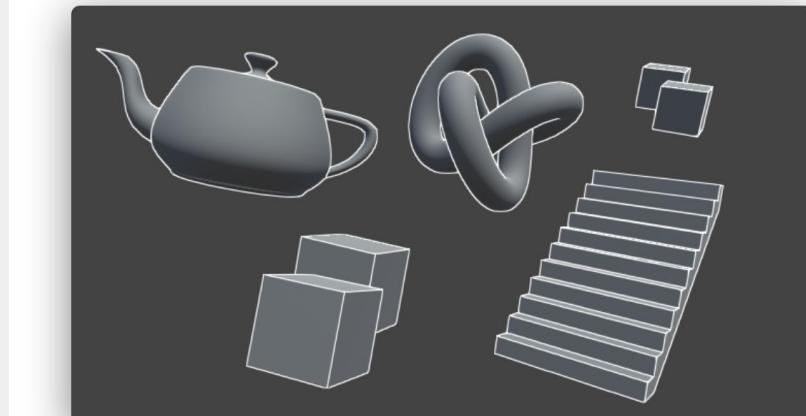
Built in Rendering

Outline Shader

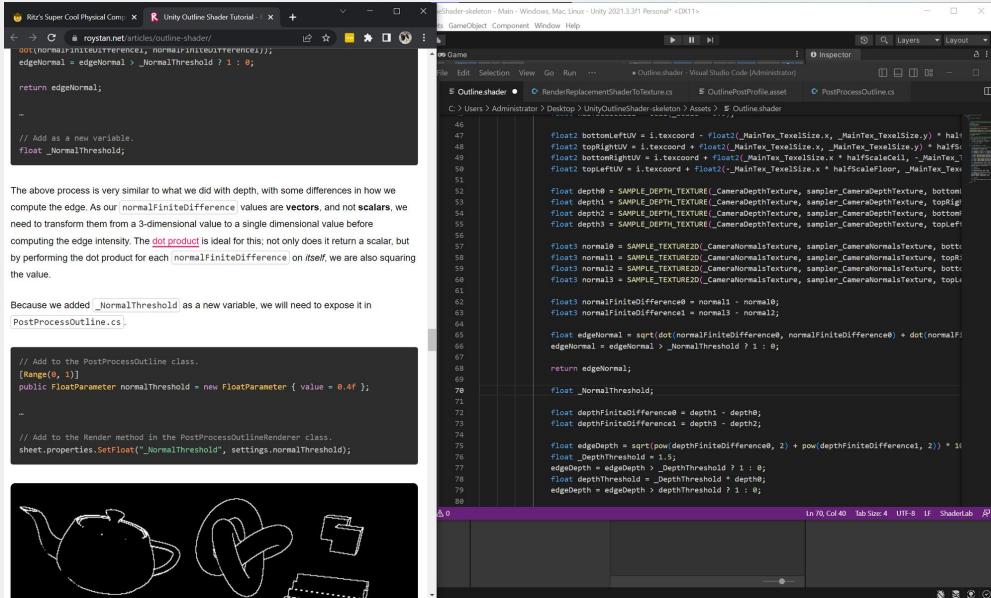
>> using Unity engine 2018.3

<https://roystan.net/articles/outline-shader/>

⌚ 50 minutes to complete



Environment



The above process is very similar to what we did with depth, with some differences in how we compute the edge. As our `normalFiniteDifference` values are **vectors**, and not **scalars**, we need to transform them from a 3-dimensional value to a single dimensional value before computing the edge intensity. The `dot product` is ideal for this; not only does it return a scalar, but by performing the dot product for each `normalFiniteDifference` on itself, we are also squaring the value.

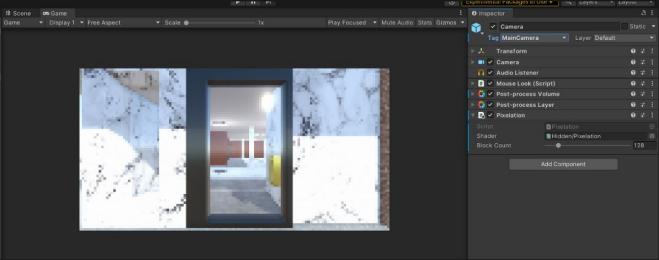
Because we added `_NormalThreshold` as a new variable, we will need to expose it in `PostProcessOutline.cs`.

```
// Add to the PostProcessOutline class.  
[Range(0, 1)]  
public FloatParameter normalThreshold = new FloatParameter { value = 0.4f };  
  
// Add to the Render method in the PostProcessOutlineRenderer class.  
sheet.properties.SetFloat("_NormalThreshold", settings.normalThreshold);
```

Because we added `_NormalThreshold` as a new variable, we will need to expose it in `PostProcessOutline.cs`.

```
// Add to the PostProcessOutline class.  
[Range(0, 1)]  
public FloatParameter normalThreshold = new FloatParameter { value = 0.4f };  
  
// Add to the Render method in the PostProcessOutlineRenderer class.  
sheet.properties.SetFloat("_NormalThreshold", settings.normalThreshold);
```

- I had problems importing the custom outline shader into my own project, so I decided to import the pixel shader into the apartment kit first
- I also had an issue with this, as I imported the pixelization script onto the camera, but it did not work
- I realized by changing the tag of the camera, the pixelization shader worked



- as you can see here, I changed the Tag for the camera into MainCamera, pixelizing the screen
 - this only works in the game, not in the scene
 - I'm suspecting that to get the outline shader to work, I had to change the camera tag to MainCamera as well
- For now, I've found an outline shader for free on the unity asset store

Outlines

Add depth to your project with Outlines asset From The Developer. Find this & more VFX options on the Unity Asset Store.

<https://assetstore.unity.com/packages/vfx/shaders/fullscreen-camera-effect...>



Environment

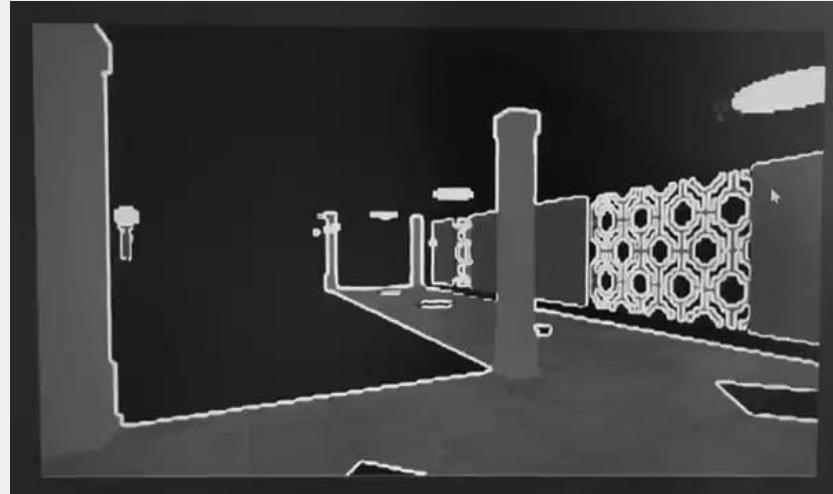
- Used asset outliner, built-in rendering

Built-in Rendering:

- Rendering of 3D objects into 2D so it can be displayed
- Geometry
- Shading
- Lighting
- Limited use of URP (Universal Rendering Pipeline) and HDRP (High Definition Rendering Pipeline)

Post Processing:

- Apply effects to rendered image (think of the filters you apply on your images when you edit them)
- Gaussian blur
- Color grading
- Ambient Occlusion
- Etc.
- Visual Effects



- Because the outlines were a part of built-in rendering (not post processing), I was unable to use culling masks and two cameras to remove the outline from the enemy
- Got rid of outlines



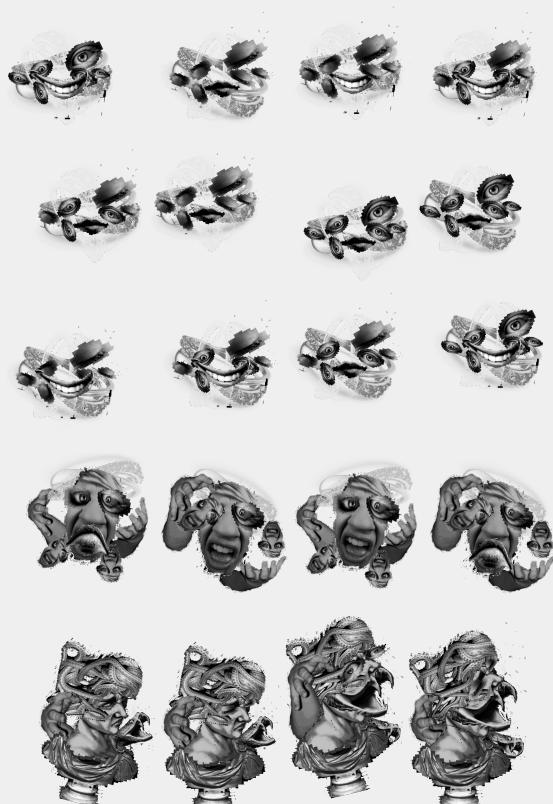
Enemies





Enemies

All
Things
Good
and
Before



Enemies



Followed tutorial from
DanCS on Youtube



Enemies

```
private void Chasing()
{
    // The enemy is chasing the player
    m_PlayerNear = false;           // Set false that hte player is near beacause the enemy already sees the player
    playerLastPosition = Vector3.zero; // Reset the player near position

    if (!m_CaughtPlayer)
    {
        Move(speedRun);
        navMeshAgent.SetDestination(m_PlayerPosition);          // set the destination of the enemy to the player location
    }

    if (navMeshAgent.remainingDistance <= navMeshAgent.stoppingDistance) // Control if the enemy arrive to the player location
    {
        if (m_WaitTime <= 0 && !m_CaughtPlayer && Vector3.Distance(transform.position, GameObject.FindGameObjectWithTag("Player").transform.position) >= 6f)
        {
            // Check if the enemy is not near to the player, returns to patrol after the wait time delay
            m_IsPatrol = true;
            m_PlayerNear = false;
            Move(speedWalk);
            m_TimeToRotate = timeToRotate;
            m_WaitTime = startWaitTime;
            navMeshAgent.SetDestination(waypoints[m_CurrentWaypointIndex].position);
        }
        else
        {
            if (Vector3.Distance(transform.position, GameObject.FindGameObjectWithTag("Player").transform.position) >= 2.5f)
                // Wait if the current position is not the player position
                Stop();
            m_WaitTime -= Time.deltaTime;
        }
    }
}
```



Enemies

```
private void Patrolling()
{
    if (_PlayerNear)
    {
        // Check if the enemy detect near the player, so the enemy will move to that position
        if (_TimeToRotate <= 0)
        {
            Move(speedWalk);
            LookingPlayer(playerLastPosition);
        }
        else
        {
            // The enemy wait for a moment and then go to the last player position
            Stop();
            _TimeToRotate -= Time.deltaTime;
        }
    }
    else
    {
        _PlayerNear = false;           // The player is no near when the enemy is platroling
        playerLastPosition = Vector3.zero;
        navMeshAgent.SetDestination(waypoints[_CurrentWaypointIndex].position);    // Set the enemy destination to the next waypoint
        if (navMeshAgent.remainingDistance <= navMeshAgent.stoppingDistance)
        {
            // If the enemy arrives to the waypoint position then wait for a moment and go to the next
            if (_WaitTime <= 0)
            {
                NextPoint();
                Move(speedWalk);
                _WaitTime = startWaitTime;
            }
            else
            {
                Stop();
                _WaitTime -= Time.deltaTime;
            }
        }
    }
}
```



Enemies

```
void EnviromentView()
{
    Collider[] playerInRange = Physics.OverlapSphere(transform.position, viewRadius, playerMask); // Make an overlap sphere around the enemy to detect the playermask in the view radius

    for (int i = 0; i < playerInRange.Length; i++)
    {
        Transform player = playerInRange[i].transform;
        Vector3 dirToPlayer = (player.position - transform.position).normalized;
        if (Vector3.Angle(transform.forward, dirToPlayer) < viewAngle / 2)
        {
            float dstToPlayer = Vector3.Distance(transform.position, player.position); // Distance of the enemy and the player
            if (!Physics.Raycast(transform.position, dirToPlayer, dstToPlayer, obstacleMask))
            {
                m_playerInRange = true; // The player has been seen by the enemy and then the enemy starts to chasing the player
                m_IsPatrol = false; // Change the state to chasing the player
            }
            else
            {
                /*
                * If the player is behind a obstacle the player position will not be registered
                */
                m_playerInRange = false;
            }
        }
        if (Vector3.Distance(transform.position, player.position) > viewRadius)
        {
            /*
            * If the player is further than the view radius, then the enemy will no longer keep the player's current position.
            * Or the enemy is a safe zone, the enemy will no chase
            */
            m_playerInRange = false; // Change the state of chasing
        }
        if (m_playerInRange)
        {
            /*
            * If the enemy no longer sees the player, then the enemy will go to the last position that has been registered
            */
            m_PlayerPosition = player.transform.position; // Save the player's current position if the player is in range of vision
        }
    }
}
```



Enemies

```
public class AIController : MonoBehaviour
{
    public NavMeshAgent navMeshAgent; // Nav mesh agent component
    public float startWaitTime = 4; // Wait time of every action
    public float timeToRotate = 2; // Wait time when the enemy detect near the player without seeing
    public float speedWalk = 6; // Walking speed, speed in the nav mesh agent
    public float speedRun = 9; // Running speed

    public float viewRadius = 15; // Radius of the enemy's view
    public float viewAngle = 90; // Angle of the enemy's view
    public LayerMask playerLayer; // To detect the player with the raycast
    public LayerMask obstacleLayer; // To detect the obstacles with the raycast
    public float meshResolution = 1.0f; // How many rays will cast per degree
    public int edgeIterations = 4; // Number of iterations to get a better performance of the mesh filter when the raycast hit an obstacle
    public float edgeDistance = 0.5f; // Max distance to calculate the minimum and a maximum raycast when hits something

    public Transform[] waypoints; // All the waypoints where the enemy patrols
    int m_CurrentWaypointIndex; // Current waypoint where the enemy is going to

    Vector3 playerLastPosition = Vector3.zero; // Last position of the player when was near the enemy
    Vector3 m_PlayerPosition; // Last position of the player when the player is seen by the enemy

    float m_WaitTime; // Variable of the wait time that makes the delay
    float m_TimeToRotate; // Variable of the wait time to rotate when the player is near that makes the delay
    bool m_PlayerInRange; // If the player is in range of vision, state of chasing
    bool m_PlayerNear; // If the player is near, state of hearing
    bool m_IsPatrol; // If the enemy is patrol, state of patrolling
    bool m_CaughtPlayer; // If the enemy has caught the player

    public GameObject uiObject;
    bool m_PlayerHidden;

    public float attackRange = 0.8f; // Range threshold
    public float healthPoints = 200;

    void Start()
    {
        m_PlayerPosition = Vector3.zero;
        m_IsPatrol = true;
        m_CaughtPlayer = false;
        m_PlayerInRange = false;
        m_PlayerNear = false;
        m_WaitTime = startWaitTime; // Set the wait time variable that will change
        m_TimeToRotate = timeToRotate;

        m_CurrentWaypointIndex = 0; // Set the initial waypoint
        navMeshAgent = GetComponent<NavMeshAgent>();

        navMeshAgent.isStopped = false;
        navMeshAgent.speed = speedWalk; // Set the navmesh speed with the normal speed of the enemy
        navMeshAgent.SetDestination(waypoints[m_CurrentWaypointIndex].position); // Set the destination to the first waypoint

        m_PlayerHidden = false;
    }
}
```

```
private void Update()
{
    CanvasGroup canvasGroup = uiObject.GetComponent<CanvasGroup>();
    Image uiImage = uiObject.GetComponent<Image>();

    if (!m_PlayerHidden){ // Check whether or not the player is in the enemy's field of vision
        EnviromentView();
    }

    float alphaValue = canvasGroup.alpha;
    Debug.Log("Alpha value: " + alphaValue);

    if (alphaValue > 0.6){
        m_PlayerHidden = true;
    } else {
        m_PlayerHidden = false;
    }

    float distanceToPlayer_Attack = Vector3.Distance(transform.position, GameObject.FindGameObjectWithTag("Player").transform.position);

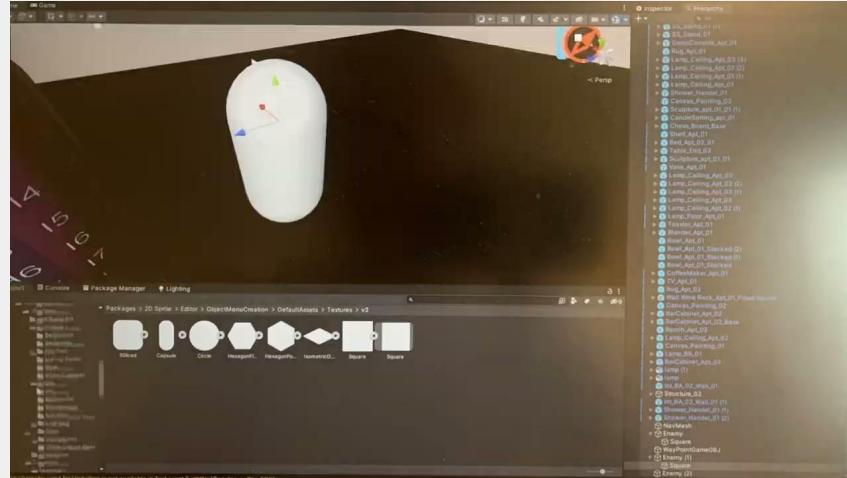
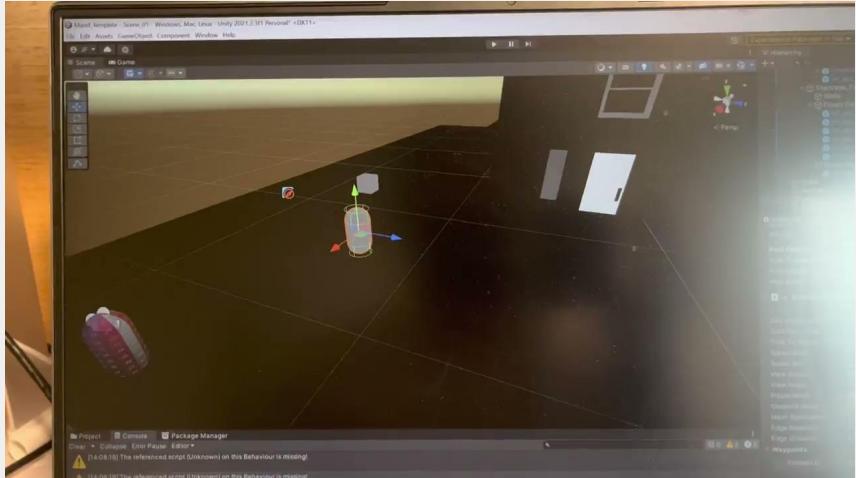
    if ((distanceToPlayer_Attack <= attackRange) && !m_PlayerHidden)
    {
        healthPoints -= 10;
        canvasGroup.alpha = (float)0.6;
        uiImage.color = Color.red;
    } else {
        uiImage.color = Color.black;
    }

    if (healthPoints < 0){
        SceneManager.LoadScene(2);
    }

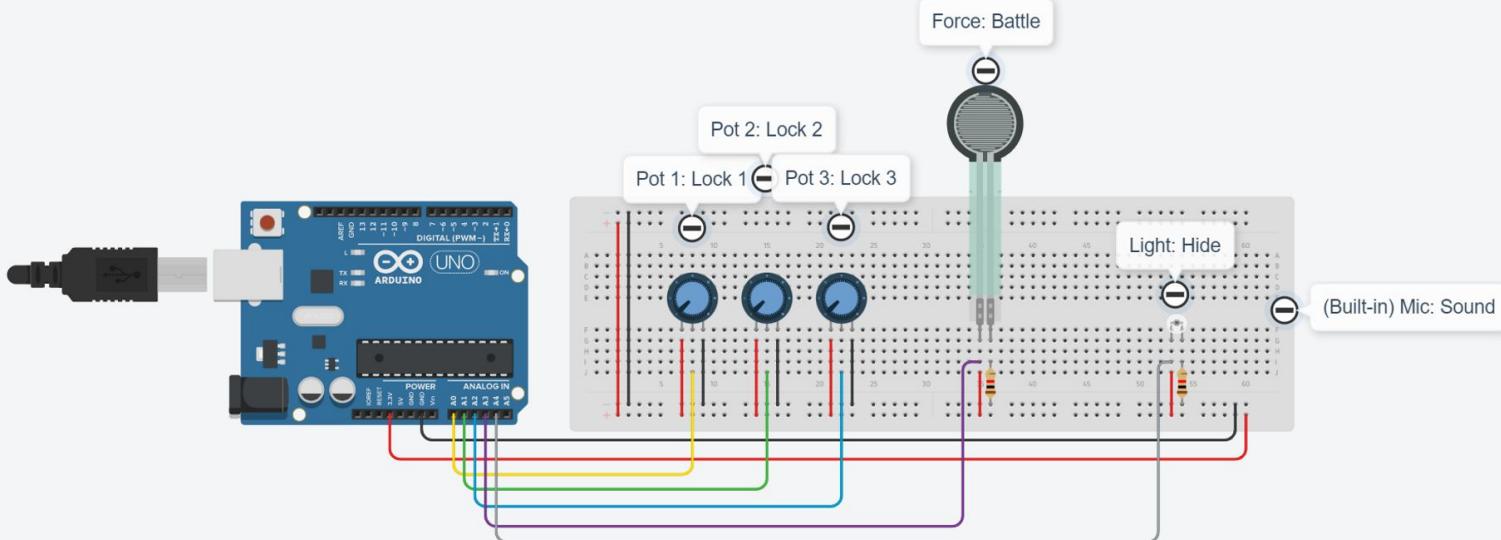
    if (!(m_IsPatrol && !m_PlayerHidden))
    {
        Chasing();
    } else
    {
        Patrolling();
    }
}
```



Enemies



Arduino



Arduino



Ardity: Arduino + Unity communic...

DW Daniel Wilches

★★★★☆ (21) | ❤ (225)

FREE

Taxes/VAT calculated at checkout

Product Information

License agreement [Standard Unity Asset Store EULA](#)

File size 718.5 KB

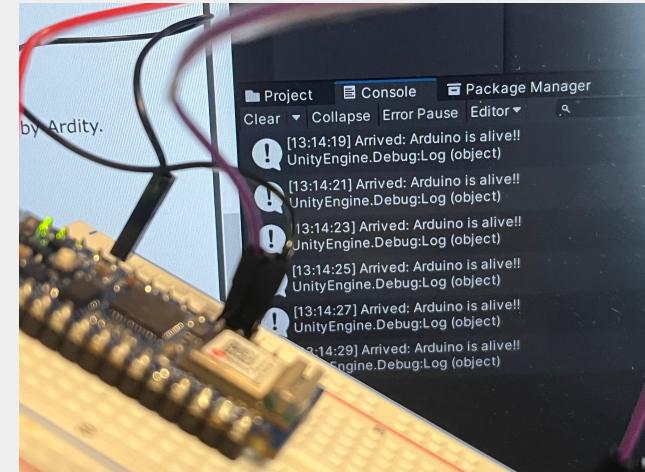
Latest version 1.1.0

Latest release date Aug 20, 2018

Original Unity version ② 2018.2.0 or higher

Support [Visit site](#)

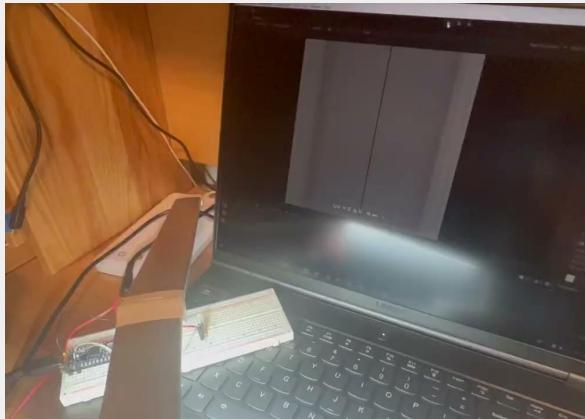
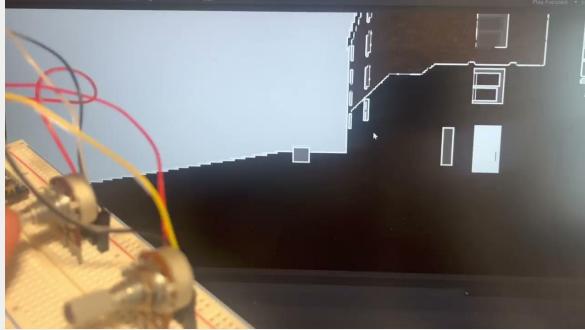
Reviews ★★★★☆



Arduino

```
Mand_Arduino_Code\Arduino IDE 2.0.4
File Edit Sketch Tools Help
Mand_Arduino_Code.ino
5 using std::endl; using std::string;
6
7 // the setup routine runs once when you press reset:
8 void setup() {
9   // initialize serial communication at 9600 bits per second:
10  Serial.begin(9600);
11 }
12
13 // the loop routine runs over and over again forever:
14 void loop() {
15   // read the input on analog pin 0:
16   int potValue0 = analogRead(A0); // Potentiometer 1
17   int potValue1 = analogRead(A1); // Potentiometer 2
18   int potValue2 = analogRead(A2); // Potentiometer 3
19   int轻光Value = analogRead(A3); // Light Sensor
20   int ForceValue = analogRead(A4); // Force Sensor
21
22
23   // Convert the analog reading (which goes from 0 - 1023) to a voltage (0 - 5V):
24   float voltage = potValue0 * (5.0 / 1023.0);
25   float voltage1 = potValue1 * (5.0 / 1023.0);
26   float voltage2 = potValue2 * (5.0 / 1023.0);
27   float voltage3 = lightValue * (5.0 / 1023.0);
28   float voltage4 = ForceValue * (5.0 / 1023.0);
29
30   // Print to Serial float to String conversion
31   String msg = "";
32   msg.concat(voltage);
33   msg.concat(",");
34   msg.concat(voltage1);
35   msg.concat(",");
36   msg.concat(voltage2);
37   msg.concat(",");
38   msg.concat(voltage3);
39   msg.concat(",");
40   msg.concat(voltage4);
41
42   // print out the value you read:
43   Serial.println(msg);
44   delay(100);
45 }
```

```
File Edit Selection View Go ...
MyMessageListener.cs - Visual Studio Code [Administrator]
C:\Users\Administrator> Mand_Template> Assets > MyMessageListener.cs
13 {
14     GameObject cubeModifier;
15
16     void Start() // Start is called before the first frame update
17     {
18         cubeModifier = GameObject.Find("Cube");
19     }
20
21     void Update() // Update is called once per frame
22     {
23     }
24
25     void OnMessageArrived(string msg)
26     {
27         string[] msgList = msg.Split(',');
28         Debug.Log("moving at speed: " + msg);
29
30         // .....msgList KEY.....
31         // msgList[0] = Potentiometer 1
32         // msgList[1] = Potentiometer 2
33         // msgList[2] = Potentiometer 3
34         // msgList[3] = Light Sensor
35         // msgList[4] = Force Sensor
36
37         float speedForward = float.Parse(msgList[0]) / 10;
38         float speedUpward = float.Parse(msgList[1]) / 10;
39         cubeModifier.gameObject.transform.Translate(Vector3.forward * Time.deltaTime * speedForward);
40         cubeModifier.gameObject.transform.Translate(Vector3.up * Time.deltaTime * speedUpward);
41
42         // Invoked when a connect/disconnect event occurs. The parameter "success"
43         // will be "true" upon connection, and "false" upon disconnection or
44         // failure to connect.
45         void OnConnectionEvent(bool success)
46         {
47             Debug.Log(success ? "Device connected" : "Device disconnected");
48         }
49 }
```



```

Mand_Arduino.Code.ino

1 #include <iostream>
2 #include <string>
3
4 using std::cout; using std::cin;
5 using std::endl; using std::string;
6
7 const int buffer = 20;
8 int sensorBuffer[buffer];
9 int bufferIndex = 0;
10
11 // the setup routine runs once when you press reset:
12 void setup() {
13     // initialize serial communication at 9600 bits per second:
14     Serial.begin(9600);
15
16     for (int i = 0; i < buffer; i++) {
17         sensorBuffer[i] = 0;
18     }
19 }
20
21 // the loop routine runs over and over again forever:
22 void loop() {
23     int lightValue = analogRead(A0); // Light Sensor
24
25     sensorBuffer[bufferIndex] = lightValue;
26     bufferIndex = (bufferIndex + 1) % buffer;
27
28     int movingAverage = 0;
29     for (int i = 0; i < buffer; i++) {
30         movingAverage += sensorBuffer[i];
31     }
32     movingAverage /= buffer;
33
34     // Use stabilized value
35     float stabilizedValue = (float)movingAverage;
36
37     // Print to Serial float to String conversion
38     String msg = "";
39     msg.concat(stabilizedValue);
40
41     // print out the value you read:
42     Serial.println(msg);
43     delay(100);
44 }

```

Arduino

C:\Users\Administrator\mand 2> Assets> MyMessageListener.cs

```

1 /**
2 * Ardity (Serial Communication for Arduino + Unity)
3 * Author: Daniel Wilches <dwilches@gmail.com>
4 *
5 * This work is released under the Creative Commons Attribution license.
6 * https://creativecommons.org/licenses/by/2.0/
7 */
8 using System.Collections;
9 using System.Collections.Generic;
10 using UnityEngine;
11
12 public class MyMessageListener : MonoBehaviour
13 {
14     public float aValue = 1;
15     private CanvasGroup trans;
16
17     void Start() // Start is called before the first frame update
18     {
19         trans = GetComponent<CanvasGroup>();
20     }
21     void Update() // Update is called once per frame
22     {
23     }
24     void OnMessageArrived(string msg)
25     {
26         Debug.Log("Fading: " + msg);
27
28         float fadeValue = 1 - (float.Parse(msg)/10);
29         trans.alpha = fadeValue;
30     }
31     // Invoked when a connect/disconnect event occurs. The parameter 'success'
32     // will be 'true' upon connection, and 'false' upon disconnection or
33     // failure to connect.
34     void OnConnectionEvent(bool success)
35     {
36         Debug.Log(success ? "Device connected" : "Device disconnected");
37     }
38 }
39

```



Sound

Name ↑

Attack Monster.m4a 

Boof.m4a 

Daisy Bell.m4a 

Monster Attacks.m4a 

Monster in Range 2.m4a 

Monster in Range.m4a 

Name ↑

160898_ldnsoundproduction_distorted-piano-2.wav 

329866_deleted_user_4772965_creepy-vocal-ambience.wav 

402025_cybermad_creepy_ambient_2.wav 

486081_nox_sound_object_drawer_wood_open.wav 

519065_angelkunnev_door-unlock.wav 

530588_nox_sound_footsteps_boots_tile_mono.wav 

531947_straget_the-rain-falls-against-the-parasol.wav 

536076_eminyildirim_door-squeak.wav 

560773_podcapocalypse_siseo-hiss.mp3 

Ergo Phizmiz - Sad Story.mp3 

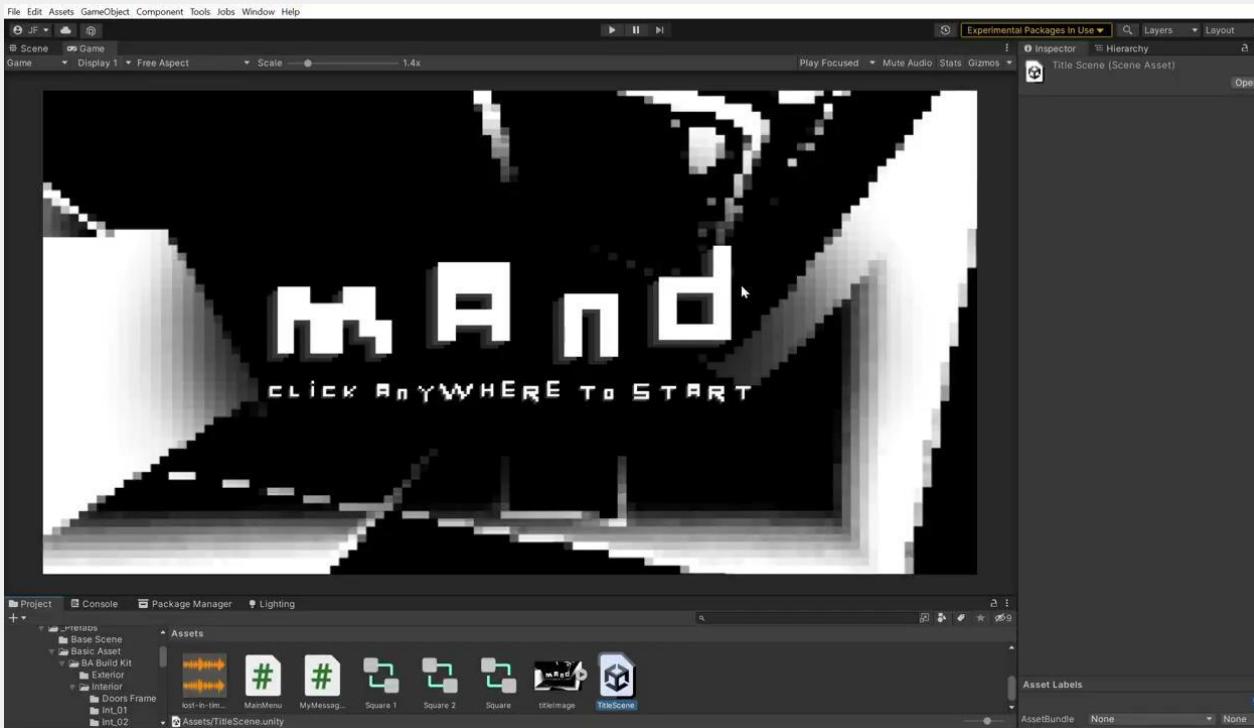
friendly-ghost-danijel-zambo-main-version-01-23-1369.mp3 

HoliznaCCO - Death.mp3 

lost-in-time-aylex-main-version-21983-02-05.mp3 



Sound



User Feedback

Yuris

- Higher visibility
- Need control instructions
- Wants to jump out of window
- Greyscale instead of only black and white

Chloe

- Controls instruction page
 - Mouse sensitivity
 - Change color of dark objects
- Trigger warning

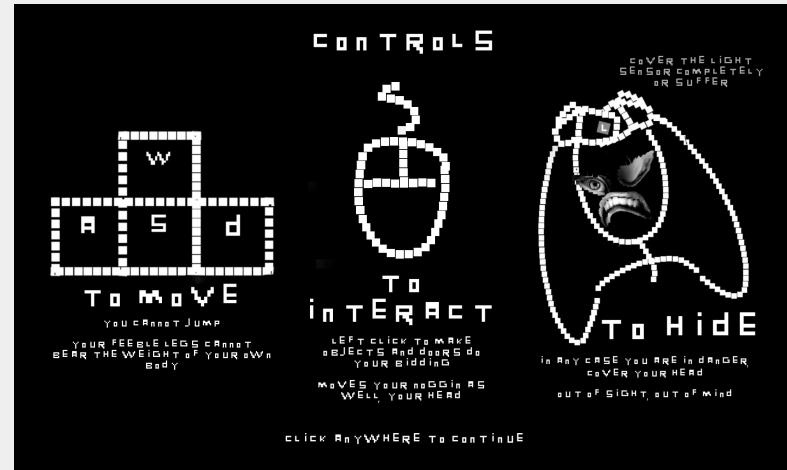
Spencer

- Higher visibility
- Need story
- Need object feedback (if can open door, then need something to pop up so user knows that they can open it)

Edits: Changed colors for some darker objects, higher fog visibility, controls instructions, story



Implementing User Feedback



Trigger Warning for next slide

Mention of child abuse, blood, drug abuse



Implementing User Feedback



- Take this story with a grain of salt, I personally do not like it and kept changing the storyline until last minute



Implementing User Feedback



```
○ TestDisappear.cs  ○ PageCount.cs
C:\Users\J\Administrator>nuke\2\Assets>○ PageCount.cs
1  using System;
2  using System.Collections;
3  using System.Collections.Generic;
4  using UnityEngine;
5  using TMPro;
6
7  public class PageCount : MonoBehaviour
8  {
9      private TextMeshProUGUI pageCountText;
10     private TextMeshProUGUI storyTextPage;
11     public GameObject storyTextObj;
12     public GameObject storyTextObj1;
13     public int pageCount;
14     public float dist = 5f;
15
16     public void OnMouseDown()
17     {
18
19         GameObject player = GameObject.FindGameObjectsWithTag("Player");
20
21         // If (player != null && Vector3.Distance(player.transform.position, transform.position) <= dist)
22         //{
23             pageCount++;
24
25             if (pageCount > 3){
26                 pageCount = 3;
27             }
28
29         //}
30
31         pageCountText = gameObject.Find("PageCounterUI").GetComponent<TextMeshProUGUI>();
32         if (pageCount >= 3){
33             pageCountText.text = "PAGE COUNT: " + pageCount + " !HDOOR UNLOCKED!";
34         } else {
35             pageCountText.text = "PAGE COUNT: " + pageCount;
36         }
37
38
39         if ((pageCount == 1) && (pageNumber == "page1")){
40             storyTextObj.SetActive(true);
41
42             storyTextPage = gameObject.Find("StoryText2").GetComponent<TextMeshProUGUI>();
43
44             storyTextPage1.text = "I DONT DRAW WITH CRAYONS ANYMORE";
45
46         }
47
48         if ((pageCount == 2) && (pageNumber == "page2")){
49             storyTextObj.SetActive(true);
50
51             storyTextPage1 = gameObject.Find("StoryText3").GetComponent<TextMeshProUGUI>();
52
53             storyTextPage1.text = ".....";
54
55         }
56
57         if ((pageCount == 3) && (pageNumber == "page3")){
58             storyTextObj.SetActive(true);
59
60             storyTextPage1 = gameObject.Find("StoryText4").GetComponent<TextMeshProUGUI>();
61
62             storyTextPage1.text = "I NEED TO GET OUT OF HERE";
63
64         }
65
66         gameObject.SetActive(false);
67     }
68 }
```



1 on 1 Feedback

Fifi Zhang (sound)

- How to make sound sound “pixel-like”?
 - compression
 - eq, filter
 - telephone filter
 - zoom h4 or h6
- Basics on creating Sound Effects/Music:
 - get samples of sound and splice them together to make sound effects first then focus on music
- Adobe Audition/ GarageBand to create sound effects
- Other tips/ suggestions:
 - Touch Designer (visualize sound, probably won't need though)
 - Programs: Ableton Live, Logicpro (for audio)
 - Splice

Anna Kim (marketing)

- Want to create “promotional” type video to show as a part of my documentation
- What are some tips you would give to help market towards a specific audience? What are some things to consider?
 - research on horror game players
 - responses of games (look at reviews)
- What are some questions I should ask myself while developing the trailer to help convince the viewer to ‘download’ my game?
 - how much should I reveal/ keep secret?
 - clips and pieces of game
 - incorporate own opinion
 - everyone see vibe of game
 - who game is for?
 - what aspects do they like?
- Any other recommendations:
 - have fun, download or finish or try?
 - Download: trailer
 - Finish: story
 - Try: both

Grace Kim (misc)

- testing
- build what's good/ not, become aware of it
- take it to groups of people (audience) and get their feedback
- user could be anybody, important to do testing
 - Testing types (search online):
 - AB Testing
 - research based
 - what kinds of questions to ask users
 - test biases
 - intuitive
 - meaningful
 - be self aware
 - What do people want to see from test?

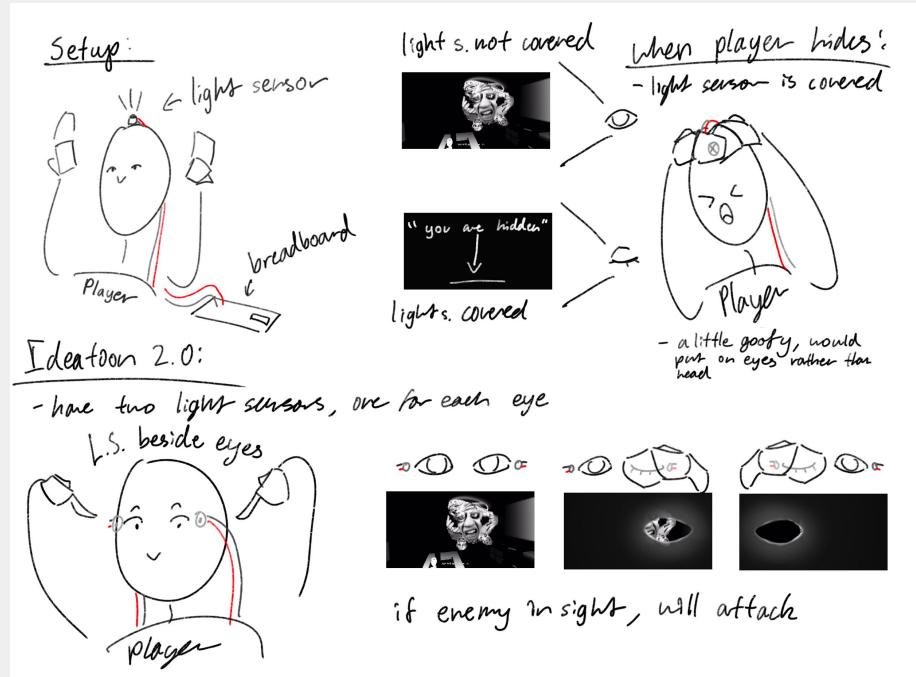
Prof. Cody

- Focus on interaction
- Better to have one interaction work really well than multiple not working
- Other recommendations:
 - Hiding make intentional move hand (light sensor on head?)

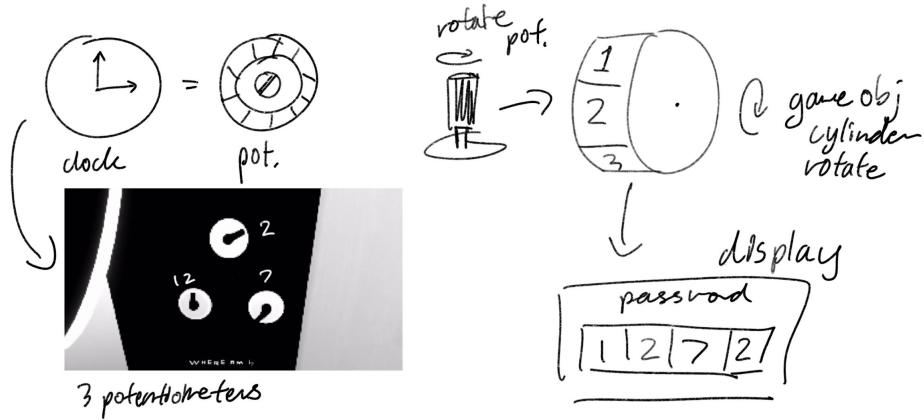


Implementing 1 on 1 Feedback

- Focus on implementing light sensor interaction
 - Got rid of fighting/ force sensor (can only run from enemies)
 - Rethought potentiometer use
- Needed to add story and win/ lose state
- Needed to flesh out who my audience is
 - Beyond myself and the class
- Audience: people like myself, who are passionate about PS1 style horror games and love playing them



Implementing 1 on 1 Feedback



- forced interaction w/ outside source
- unnatural (as Prof. Cody says)

What would make this
- more natural?



05

Future Development

If I had money and a team
because it'll literally be
impossible for me to do alone,
no matter how powerful I am :(



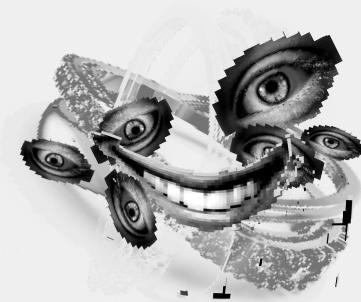
Future Development/ What I've learned

What I've learned:

- Make backup files for your Unity project, there's a specific way to do it
- Pay attention to your tags and layers
- C# (very similar syntax to Java, small differences)
- Pay attention to your Unity version (mine's 2021.3.f31)
- Unity Documentation manual is your friend
- Challenge myself more (Unity is lowkey easier than p5js)
- Next Step: Learn how to 3D model

Small Scale:

- Change map so that stairs are in different places
- Respawn!
- Notes placed in random locations every time
- Jump out of window easter egg
- Implement all sound interactions (doors, cabinets, light on off, etc)
- STORY REVISION!!!! (need to go back to the drawing board for this one)

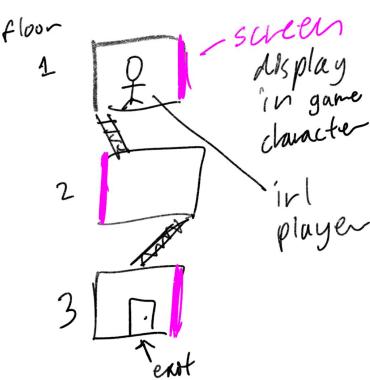


Future Development/ What I've learned

Big Scale:

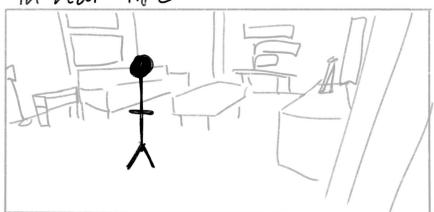
- IRL escape room
- In-game room mimics/ mirrors irl room
- Wherever the player moves/ interacts with, the in game version of the player interacts with the game objects
 - The player's entire body is the controller
- In-game version of player is made through capturing player movements through multiple cameras or a body suit
- In-game player can control irl objects and vice versa

Brainstorm



- game objective:

- get both your in game and irl self out of the rooms



- increase difficulty , have exits on different floors and switches to change the mirroring of the in game character



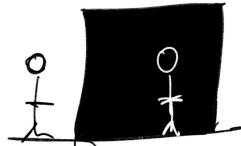
Future Development/ What I've learned

Petscop by Tony Domenico



Brainstorm

- whole TV screen on wall
- player inl is player in game ↴ game screen
- duplicate reality
- need to consider player eye level & height
- motion camera, space/room camera, get position of inl player



top view:



- clues for in game inl to help unlock both exits
- escape room
- petscop (inspo)



THANK YOU FOR LISTENING !

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon** and infographics & images by **Freepik**

Please keep this slide for attribution

