GAME DESIGN DOCUMENT

Game one-liner

Donut Pack is a 2D Baking Simulator where players go through a series of tasks to create the ultimate donut and pack them up for delivery.

Game theme

The theme of the game is based on a Donut Shop, the experience the player will have whilst playing Donut Pack will be a fun experience where they can bake their own Donut from scratch by playing a series of mini-games and then customize it later by adding an icing of their choice. The player will also have a challenging experience once they play the packing mini-game where they will test their aim with a slingshot and shoot the Donut they created into boxes.

Once the player has completed all the mini-games to create the Donut they will feel a sense of accomplishment, this feeling will be enhanced by a lot more when they see that they get a final rating after finishing each level. Not only does this enhance their feeling of accomplishment but it will make them want to replay the levels that they didn't get full ratings on.

Context and backstory

The game is set in a Donut Shop where customers can watch their Donuts be created from scratch. The customers can get a first-hand experience on the process of making a Donut but they don't only have to watch, they are able to participate by adding in their own icing and sprinkles onto the Donut.

Game Aim

The aim of the game is to complete each mini-game so that a Donut can be created, these mini-games consist of:

Ingredients - The task of this level is to correctly add each ingredient to their respective bowls (cups as well) and fill them up to the top. Any bad ingredients that come through must be thrown into the garbage bin.

Roll-A-Dough - The task of roll-a-dough is to roll your dough to a very big size whilst avoiding objects that have been misplaced through the entire area whilst under a time limit. The player must swipe their way to victory whilst maintaining their size.

Frosting and Sprinkles - The goal of this level is to create your very own donut that you want to make. Choose from 4 special flavors and then shake your phone to get lots of sprinkles onto your donut and show it off to the world.

Packing Up - The goal of this level is to pack up the donut that you have just created into the company packets. Slingshot your donuts into the packets that are falling from the sky after a malfunction in the system.

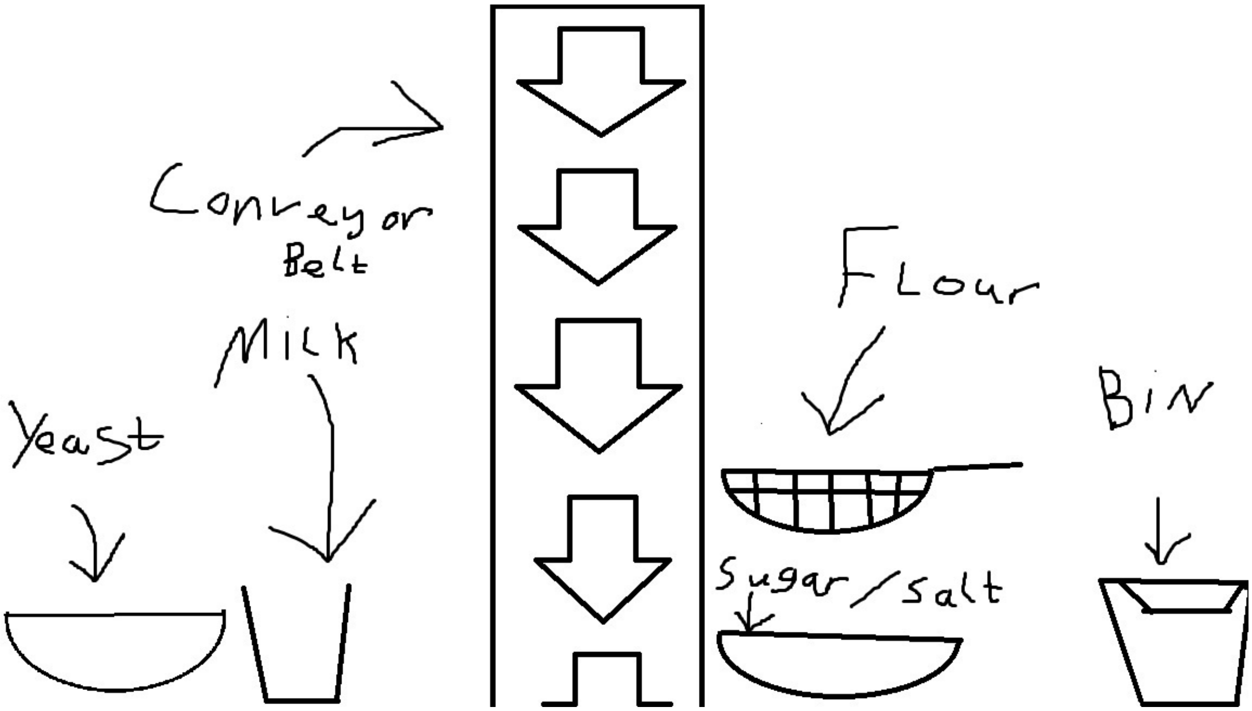
Level Design, Game World and Aesthetics

The design for this game was completely made from scratch and was developed to follow the concept ideas we had thought of whilst creating our game. The environment of the game was to follow designs that could be found in a kitchen. From Bowls to Donut cutters and rolling pins. Most of the level assets follow this exact theme.

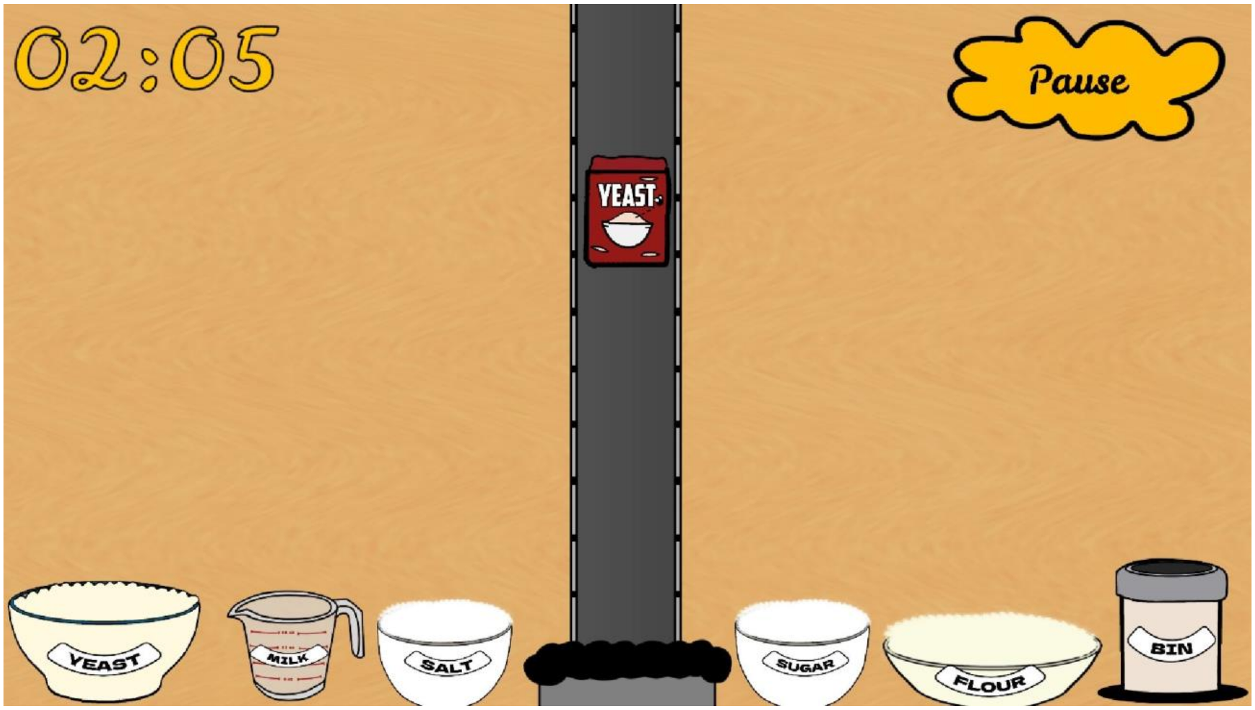
The art style was made from scratch in photoshop. The assets were created to have a cartoonish style to them that would look enjoyable and suitable to a young audience and donut lovers.

The game world takes place in a donut shop, where the player is tasked to go through the procedural order to make a donut. We start from the basics of creating donuts which is gathering ingredients all the way to packing up.

Design For Ingredients Mini-Game:
Concept Level Design:

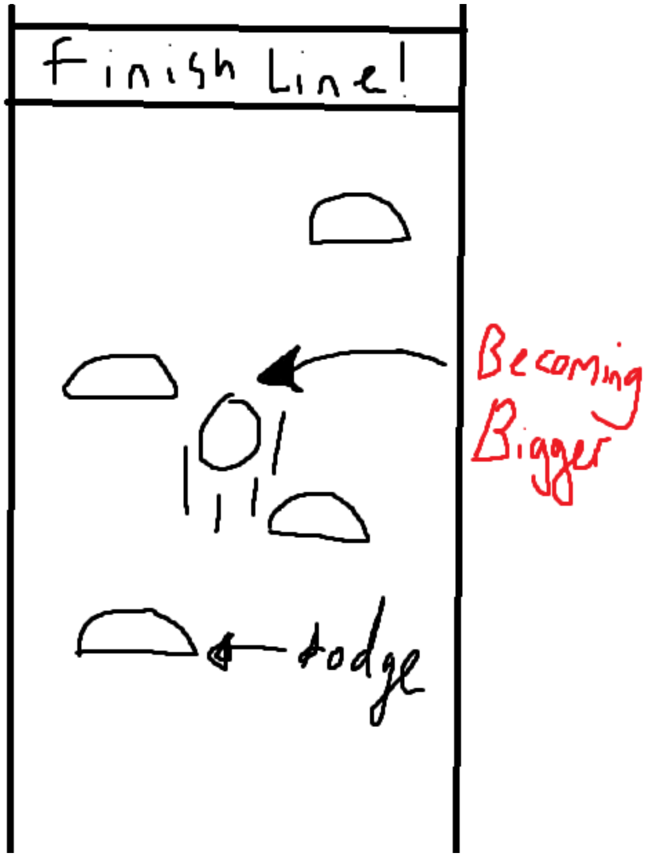


Final Level Design Layout:

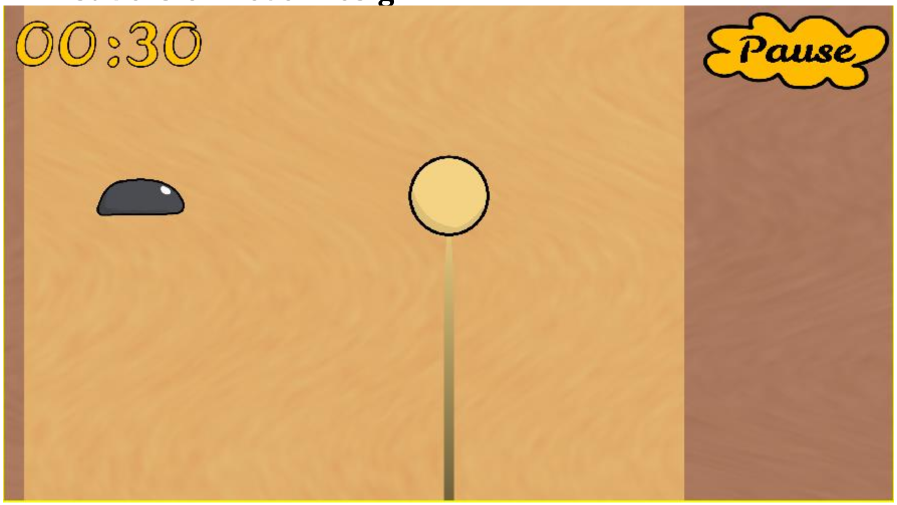


This is the Level design for the Ingredients level, The point of this level design is to grab the item off the conveyor belt and place it in its correct placement. As you can see the concept art and final level design hasn't had many changes but to make them fit in one whole line rather than all over the place as well as separating salt/sugar and making them two separate items.
Although it isn't seen in the concept level design, we also decided on having false items which need to be placed in the bin if seen coming down the conveyor rather than placing them in the ingredient.

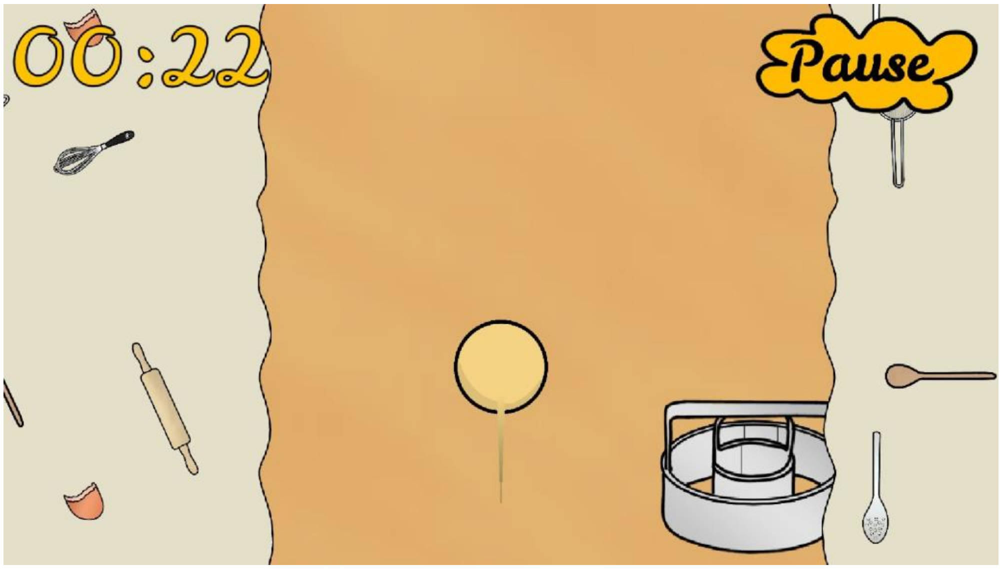
Map For Roll A Ball Mini-Game:
Concept Art Level:



First Version Level Design:



Final Map Design:



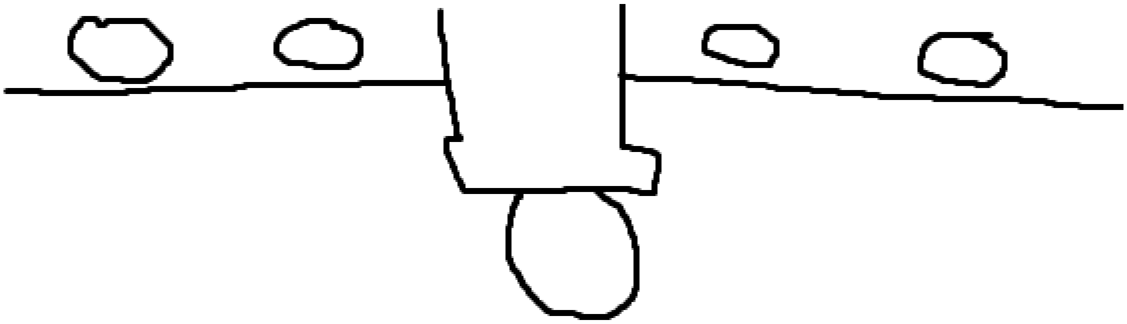
Originally creating the roll a ball game we didn’t have much going for the level design of the game, The Level idea was always remaining the same but after much feedback, we went on to make many assets to improve on the look of the game. We also added a new object, the donut cutter, which forces the player to lose as they have molded a donut very early and must start the level again.

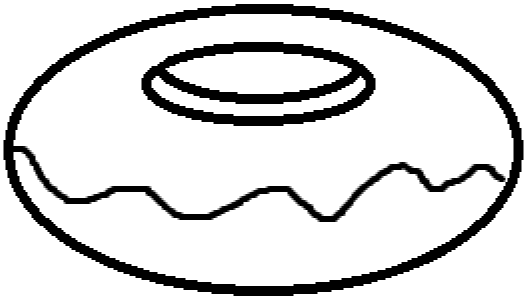
We created many assets to have nice designs on the side to make the game look as if it was being made in a messy kitchen with loads of equipment lying around the kitchen board as well as unused dough to create the walls of the level.

We also increase the sizes so that the player can utilize the swiping mechanic more rather than only swiping in one direction the entire game.

Design For Sprinkles Mini-Game:

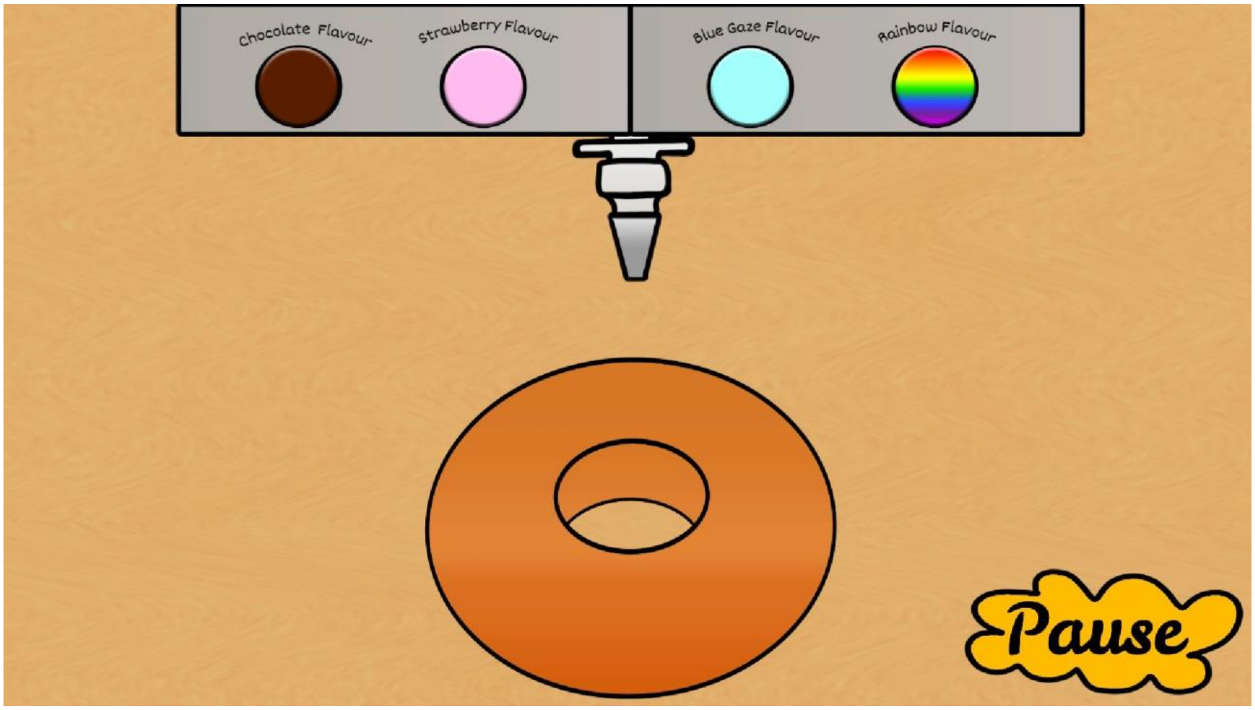
Concept Art Design:





Create donut!

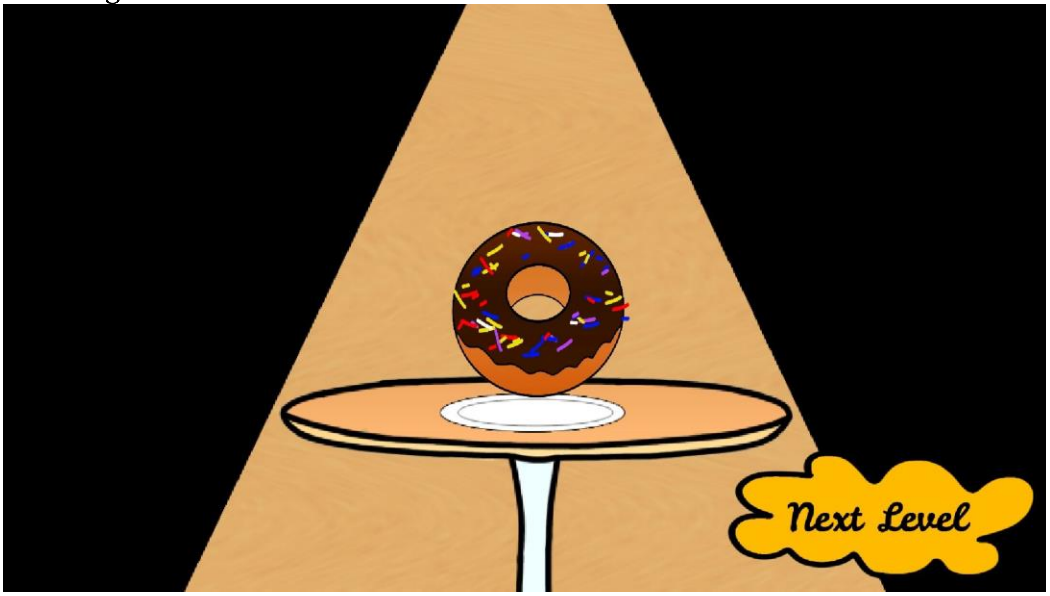
Final Level Design:



The design for the sprinkles mini-game had a very simple level design, the player just chooses from multiple flavors then they had a time limit to shake their phone as hard as possible for as many sprinkles as possible. We always intended on making the scene work this way.

No changes were needed when creating this level.

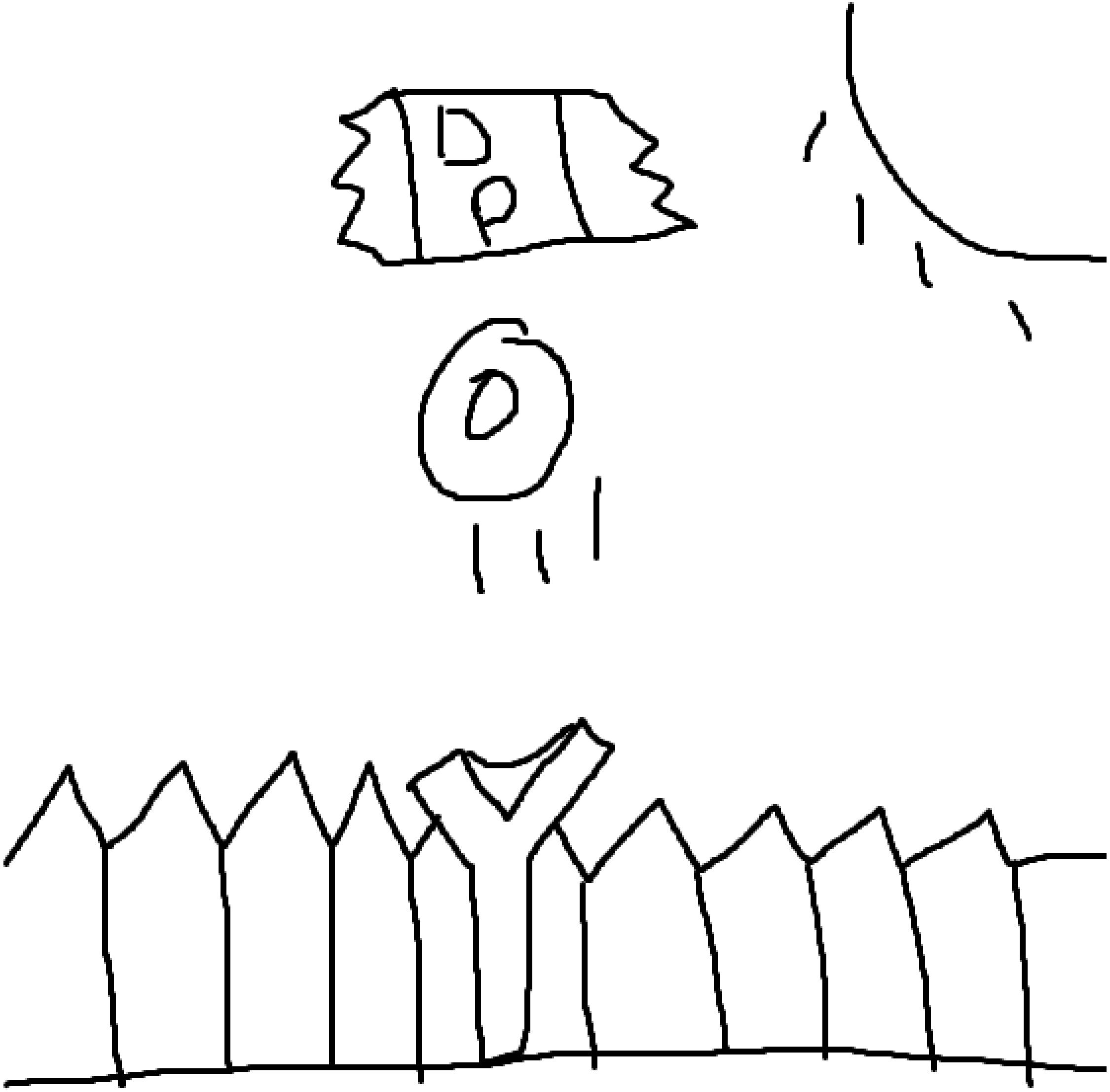
Design For Donut Showcase:



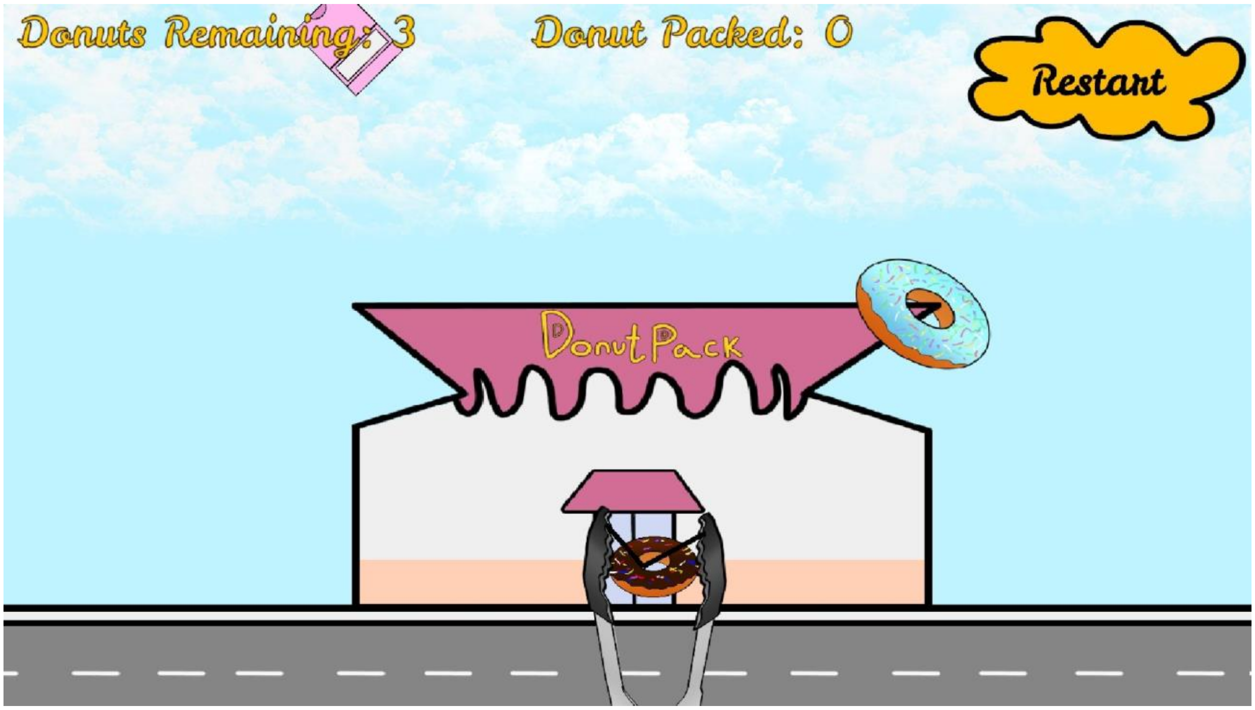
The Level design was thought of during the last week of making the game to just show off what the player has accomplished before moving onto the next scene. This level was added to make the player feel as if they have accomplished the goal of creating a donut after going through the steps of making one.

Design For Packing Mini Game:

Concept Art Design:



Final Level Design:

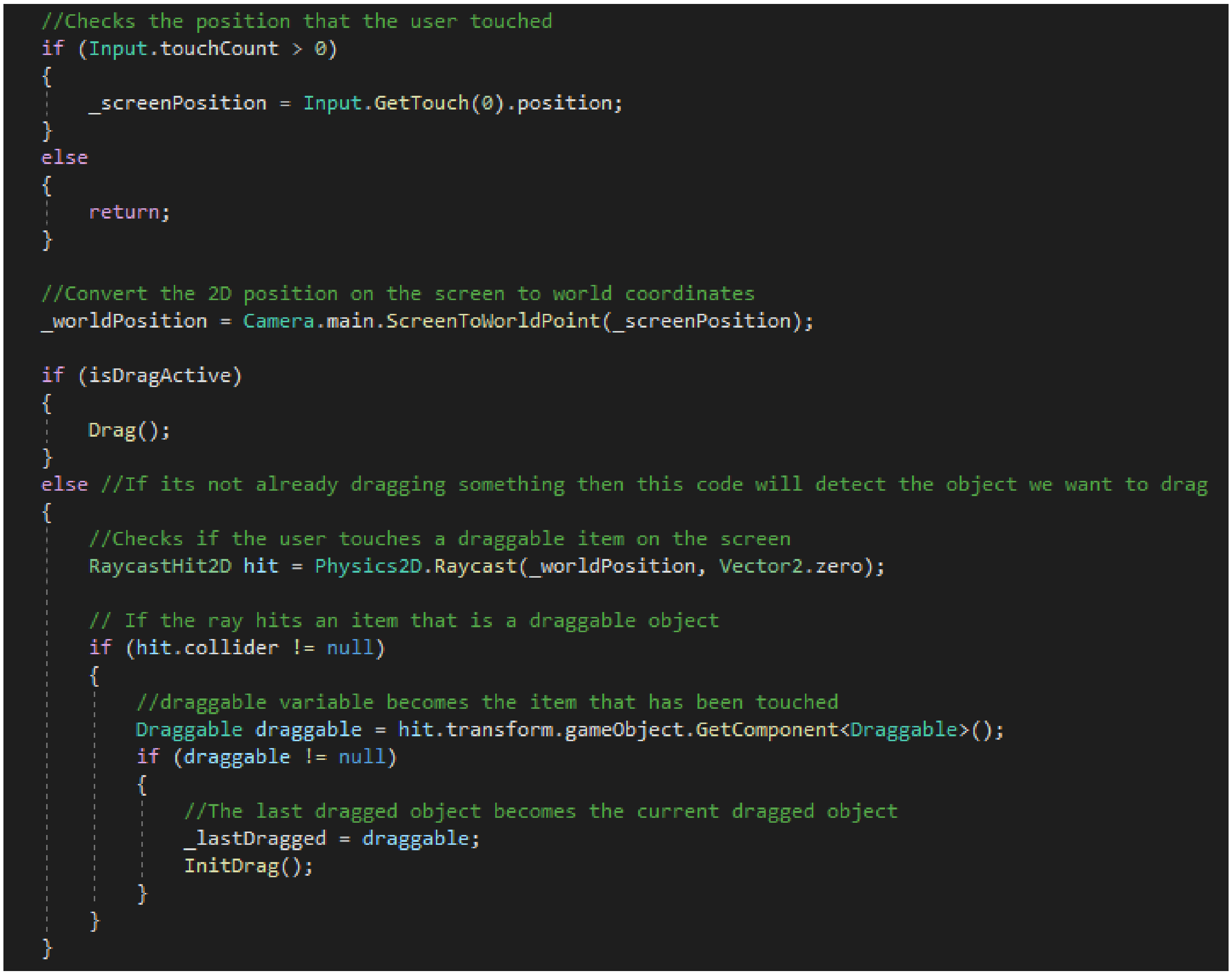


The original plan for the packing scene was to have it fall from a faulty manufacturer however, we decided that we wanted to stay within the donut store as all the other scenes follow the exact same procedure. So the final level design was made by saying that the packing up were blown into the air and that the player must now fling the donut using tongs into the donut packages.

The level idea had not changed even after changing the look of the environment.

Game Mechanics and Source Code

Dragging And Dropping

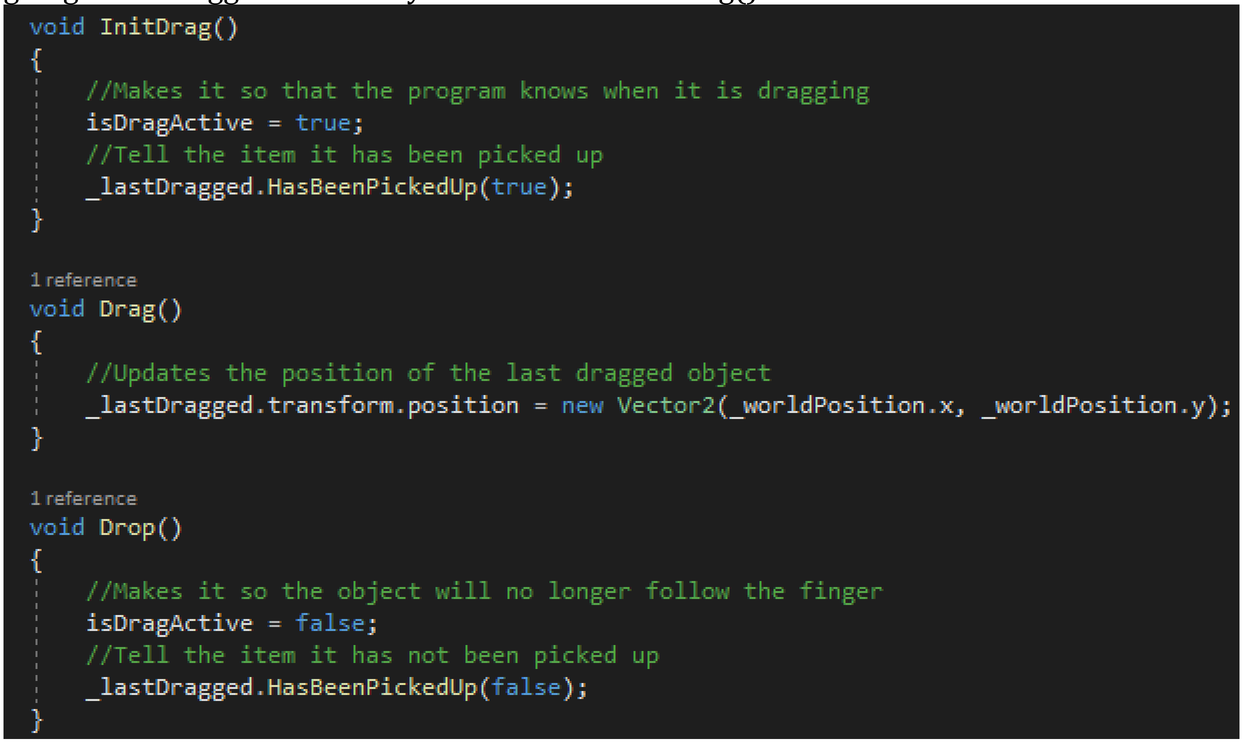


```
//Checks the position that the user touched
if (Input.touchCount > 0)
{
    _screenPosition = Input.GetTouch(0).position;
}
else
{
    return;
}

//Convert the 2D position on the screen to world coordinates
_worldPosition = Camera.main.ScreenToWorldPoint(_screenPosition);

if (isDragActive)
{
    Drag();
}
else //If its not already dragging something then this code will detect the object we want to drag
{
    //Checks if the user touches a draggable item on the screen
    RaycastHit2D hit = Physics2D.Raycast(_worldPosition, Vector2.zero);

    // If the ray hits an item that is a draggable object
    if (hit.collider != null)
    {
        //draggable variable becomes the item that has been touched
        Draggable draggable = hit.transform.gameObject.GetComponent<Draggable>();
        if (draggable != null)
        {
            //The last dragged object becomes the current dragged object
            _lastDragged = draggable;
            InitDrag();
        }
    }
}
```

The way the dragging and dropping on mobile works is that when the player puts their finger on the screen then the position of the finger will be registered inside a variable called "_screenPosition" once the position has been registered then that will be converted to world coordinates. What converting to World coordinates basically means is that it will convert the position of the finger on the screen to be the position in the game.

Now that the game knows the position of the player's finger it is now time to see if the player is actually touching any objects. The game will first check if it's already dragging anything which is checked if the "isDragActive" variable is set to true and if it is true then it will call the Drag() function which we will get onto later. If the player is currently not dragging anything then a Raycast will be created which will detect if the player is touching something on a screen.

If the Raycast finds anything we will then check if it has a component called “Draggable” (Which is just an empty Script that separates what objects can be dragged or not) and then if it does have the component then the “lastDragged” variable will be set to the item that is going to be dragged and finally it will call the InitDrag() function.



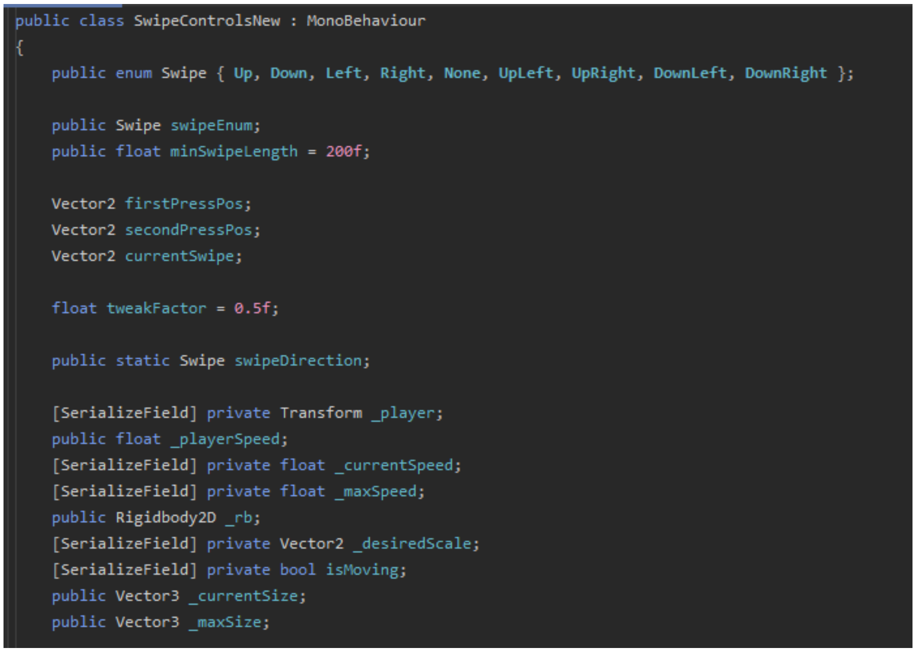
```
void InitDrag()
{
    //Makes it so that the program knows when it is dragging
    isDragActive = true;
    //Tell the item it has been picked up
    _lastDragged.HasBeenPickedUp(true);
}

1 reference
void Drag()
{
    //Updates the position of the last dragged object
    _lastDragged.transform.position = new Vector2(_worldPosition.x, _worldPosition.y);
}

1 reference
void Drop()
{
    //Makes it so the object will no longer follow the finger
    isDragActive = false;
    //Tell the item it has not been picked up
    _lastDragged.HasBeenPickedUp(false);
}
```

There are 3 functions that are needed the InitDrag() function, which will be called whenever an item needs to be picked up and will tell the item and the game that an item is being dragged, the Drag() function which will be called when the position of the item needs to be in the same position as the player’s fingers.Last but not least we have the Drop() function that will be called when the player stops touching the screen and it will tell the item and the game that an item is no longer being dragged.

All the code for the Dragging And Dropping mechanic is put inside of the Update() function, this is so that we can keep checking throughout the game whether or not the player is dragging an item or not and if they are dragging an item then it will keep updating the position of the item to match wherever the finger of the player touches.

Swiping Controls

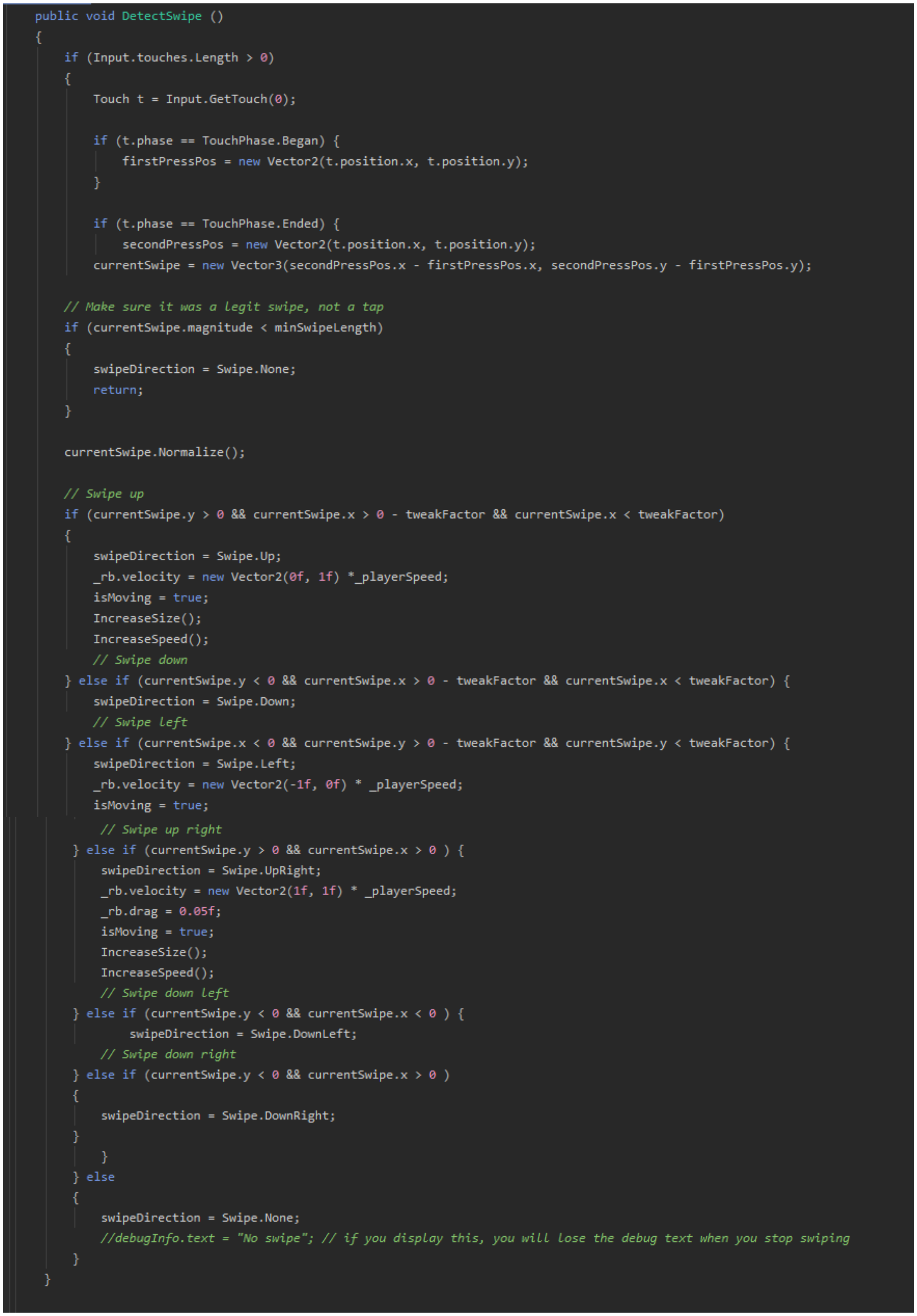
With the swiping controls, the way it works on mobile is that there are different states known as enums for when the player does an action and there are different variables such as “firstPressPos”, “secondPressPos”, “currentSwipe” which hold the players position of the first swipe to when they finished swiping. The current swipe holds what type of swipe the player has done and it will detect if the swipe was long enough to be a swipe and if it wasn't then it would be a tap.



```
public class SwipeControlsNew : MonoBehaviour
{
    public enum Swipe { Up, Down, Left, Right, None, UpLeft, UpRight, DownLeft, DownRight };

    public Swipe swipeEnum;
    public float minSwipeLength = 200f;

    Vector2 firstPressPos;
    Vector2 secondPressPos;
    Vector2 currentSwipe;

    float tweakFactor = 0.5f;

    public static Swipe swipeDirection;

    [SerializeField] private Transform _player;
    public float _playerSpeed;
    [SerializeField] private float _currentSpeed;
    [SerializeField] private float _maxSpeed;
    public Rigidbody2D _rb;
    [SerializeField] private Vector2 _desiredScale;
    [SerializeField] private bool isMoving;
    public Vector3 _currentSize;
    public Vector3 _maxSize;
```

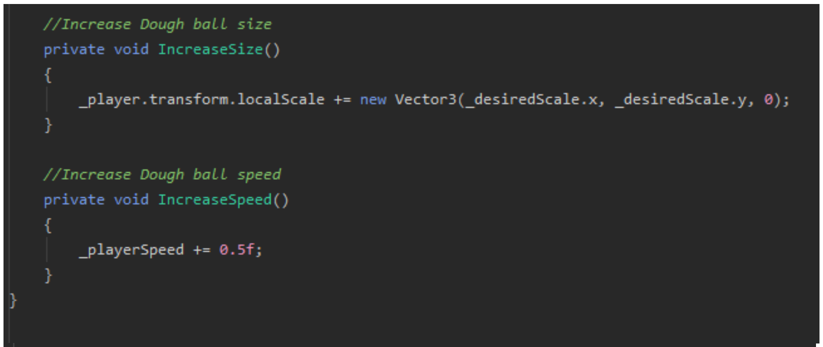
There is also an important variable which makes the above variables work and that is the “minSwipeLength'', and this is so that the player will have to swipe a minimum amount before the input can be registered. There are also other variables which are needed specifically for the dough ball and this was the transform of the player (_player) and the size(_currentSize), desired size(_desiredSize), max size(maxSize), speed, current speed and max speed. The size and speed variables were important so we could increase the size when a specific input was being done. The transform variable was important so that it could handle the scale of the doughball.

From here, there are 3 Functions in order to make the entire swipe controls work correctly and this is the DetectSwipe(), IncreaseSpeed() and IncreaseSize(). The DetectSwipe() function holds all important movement information. If the player swipes from left to right, upwards or diagonally up left to right then they will move and it will change the enum state that the player will be in and if the player state will change to none as they haven't done a correct swipe also if they swipe upwards or diagonally then they will increase in size and their speed will also increase until the maximum speed and size has been reached.

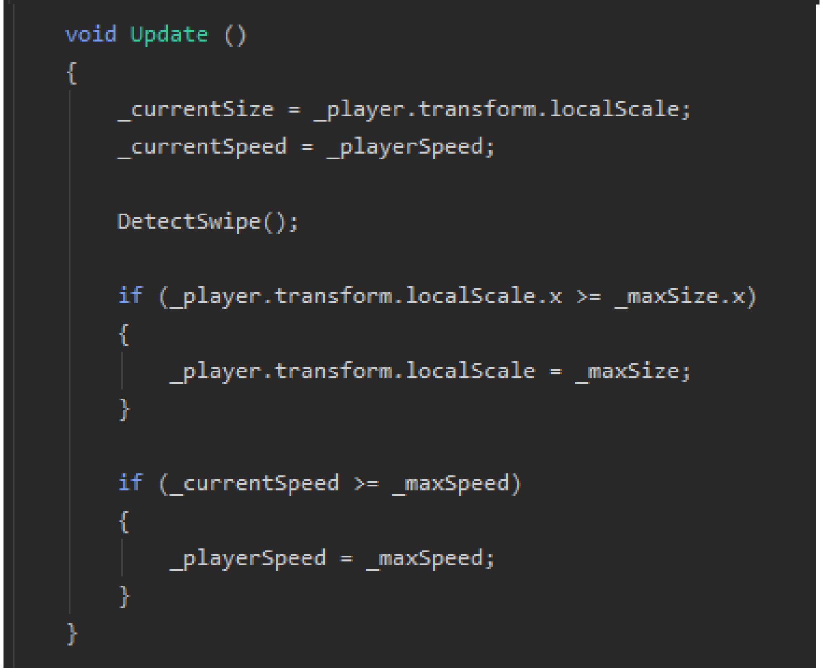


```
public void DetectSwipe ()
{
    if (Input.touches.Length > 0)
    {
        Touch t = Input.GetTouch(0);

        if (t.phase == TouchPhase.Began) {
            firstPressPos = new Vector2(t.position.x, t.position.y);
        }

        if (t.phase == TouchPhase.Ended) {
            secondPressPos = new Vector2(t.position.x, t.position.y);
        currentSwipe = new Vector3(secondPressPos.x - firstPressPos.x, secondPressPos.y - firstPressPos.y);

    // Make sure it was a legit swipe, not a tap
    if (currentSwipe.magnitude < minSwipeLength)
    {
        swipeDirection = Swipe.None;
        return;
    }

    currentSwipe.Normalize();

    // Swipe up
    if (currentSwipe.y > 0 && currentSwipe.x > 0 - tweakFactor && currentSwipe.x < tweakFactor)
    {
        swipeDirection = Swipe.Up;
        _rb.velocity = new Vector2(0f, 1f) *_playerSpeed;
        isMoving = true;
        IncreaseSize();
        IncreaseSpeed();
        // Swipe down
    } else if (currentSwipe.y < 0 && currentSwipe.x > 0 - tweakFactor && currentSwipe.x < tweakFactor) {
        swipeDirection = Swipe.Down;
        // Swipe left
    } else if (currentSwipe.x < 0 && currentSwipe.y > 0 - tweakFactor && currentSwipe.y < tweakFactor) {
        swipeDirection = Swipe.Left;
        _rb.velocity = new Vector2(-1f, 0f) * _playerSpeed;
        isMoving = true;
         // Swipe up right
     } else if (currentSwipe.y > 0 && currentSwipe.x > 0 ) {
         swipeDirection = Swipe.UpRight;
         _rb.velocity = new Vector2(1f, 1f) * _playerSpeed;
         _rb.drag = 0.05f;
         isMoving = true;
         IncreaseSize();
         IncreaseSpeed();
         // Swipe down left
     } else if (currentSwipe.y < 0 && currentSwipe.x < 0 ) {
             swipeDirection = Swipe.DownLeft;
         // Swipe down right
     } else if (currentSwipe.y < 0 && currentSwipe.x > 0 )
     {
         swipeDirection = Swipe.DownRight;
     }
         }
     } else
     {
         swipeDirection = Swipe.None;
         //debugInfo.text = "No swipe"; // if you display this, you will lose the debug text when you stop swiping
     }
 }
```

This is where the IncreaseSpeed() and IncreaseSize() function come in and the main purpose is to increase the speed and size of the ball which is called into the DetectSwipe() function. The Detect Swipe is then called into the Update() function so that it can be called every frame for precise input. With the size and speed of the ball if the ball has reached its max size and speed then it will no longer increase.

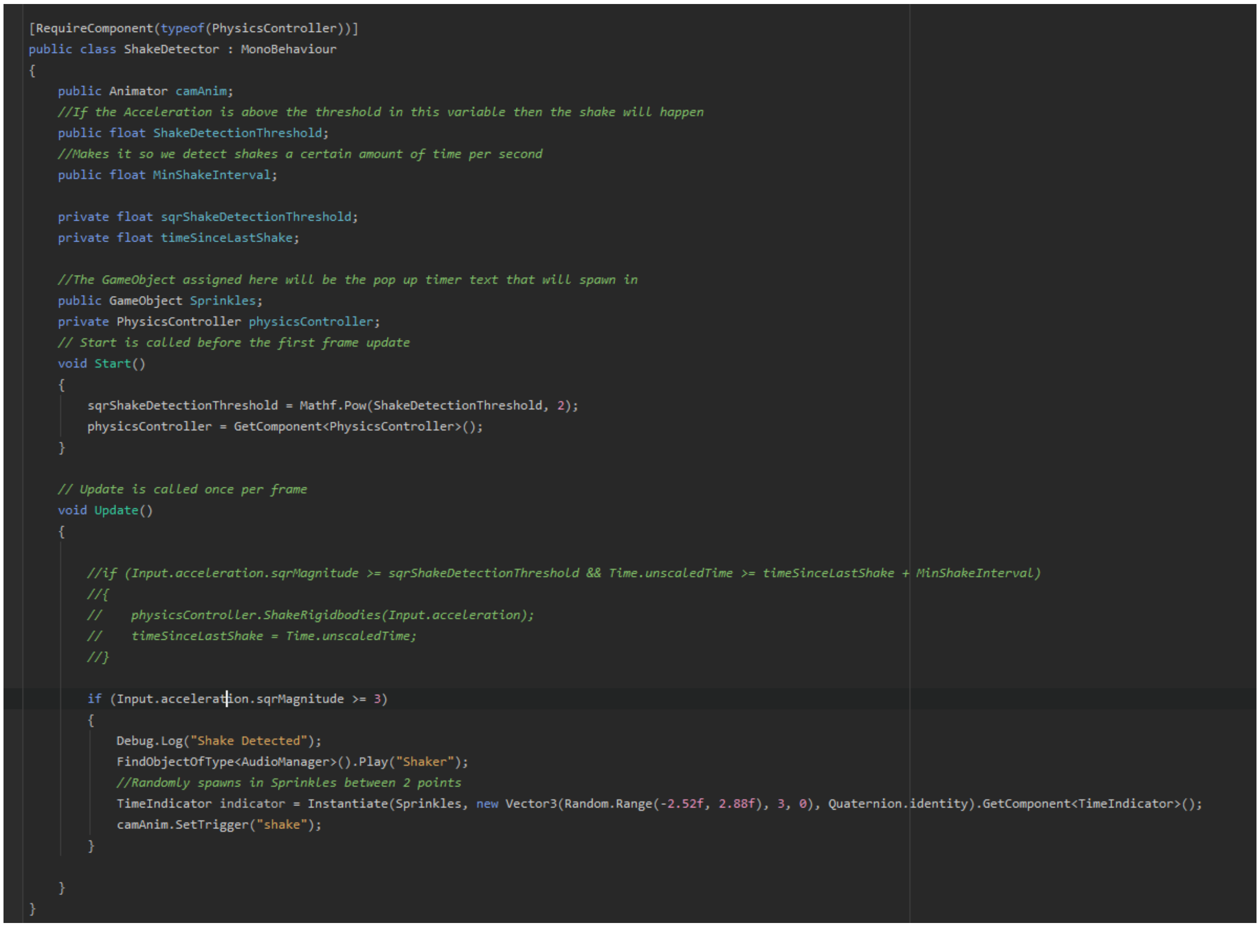


```
    //Increase Dough ball size
    private void IncreaseSize()
    {
        _player.transform.localScale += new Vector3(_desiredScale.x, _desiredScale.y, 0);
    }

    //Increase Dough ball speed
    private void IncreaseSpeed()
    {
        _playerSpeed += 0.5f;
    }
}
```



```
    void Update ()
    {
        _currentSize = _player.transform.localScale;
        _currentSpeed = _playerSpeed;

        DetectSwipe();

        if (_player.transform.localScale.x >= _maxSize.x)
        {
            _player.transform.localScale = _maxSize;
        }

        if (_currentSpeed >= _maxSpeed)
        {
            _playerSpeed = _maxSpeed;
        }
    }
```

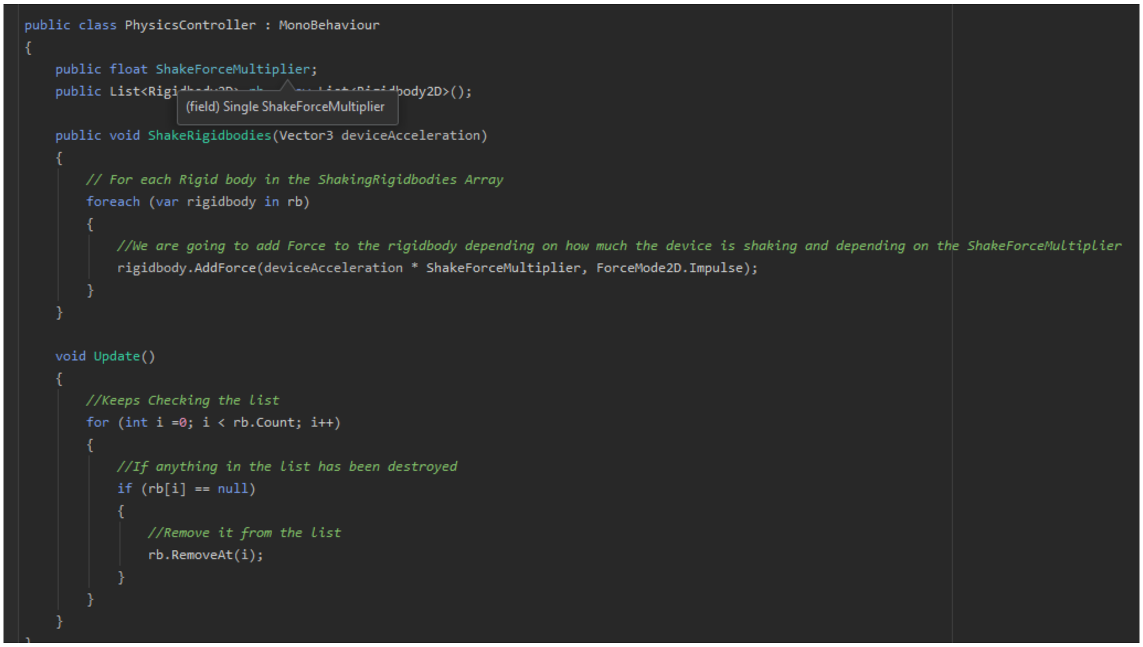
Shake Controls & Physics Controls



```
[RequireComponent(typeof(PhysicsController))]
public class ShakeDetector : MonoBehaviour
{
    public Animator camAnim;
    //If the Acceleration is above the threshold in this variable then the shake will happen
    public float ShakeDetectionThreshold;
    //Makes it so we detect shakes a certain amount of time per second
    public float MinShakeInterval;

    private float sqrShakeDetectionThreshold;
    private float timeSinceLastShake;

    //The GameObject assigned here will be the pop up timer text that will spawn in
    public GameObject Sprinkles;
    private PhysicsController physicsController;
    // Start is called before the first frame update
    void Start()
    {
        sqrShakeDetectionThreshold = Mathf.Pow(ShakeDetectionThreshold, 2);
        physicsController = GetComponent<PhysicsController>();
    }

    // Update is called once per frame
    void Update()
    {

        //if (Input.acceleration.sqrMagnitude >= sqrShakeDetectionThreshold && Time.unscaledTime >= timeSinceLastShake + MinShakeInterval)
        //{
        //    physicsController.ShakeRigidbodies(Input.acceleration);
        //    timeSinceLastShake = Time.unscaledTime;
        //}

        if (Input.acceleration.sqrMagnitude >= 3)
        {
            Debug.Log("Shake Detected");
            FindObjectOfType<AudioManager>().Play("Shaker");
            //Randomly spawns in Sprinkles between 2 points
            TimeIndicator indicator = Instantiate(Sprinkles, new Vector3(Random.Range(-2.52f, 2.88f), 3, 0), Quaternion.identity).GetComponent<TimeIndicator>();
            camAnim.SetTrigger("shake");
        }

    }
}
```

With the controls on shaking, for mobile the way it works is that when the player shakes their phone then depending how fast they are shaking their phone then they will spawn the sprinkles so that it can land on the donut. This is only because of the variable “ShakeDetectionThreshold” and if the player shakes above the threshold given the spawning will occur.

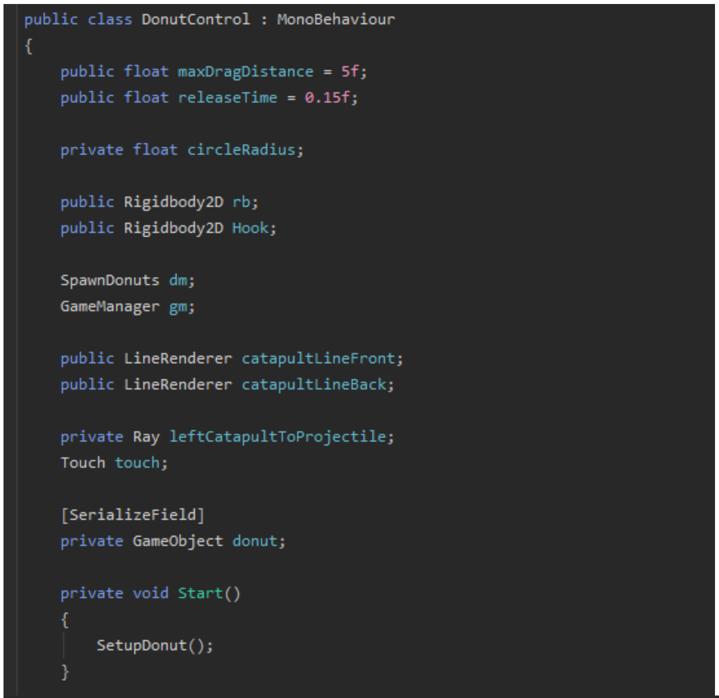
The Shaking controls also require another script which is the Physics Controller and this handles the rigidbodies for each sprinkle that has been spawned. This script works based on how much the phone is being shaken and the faster it is being shaken the more faster the sprinkle will fall due to the rigidbody.



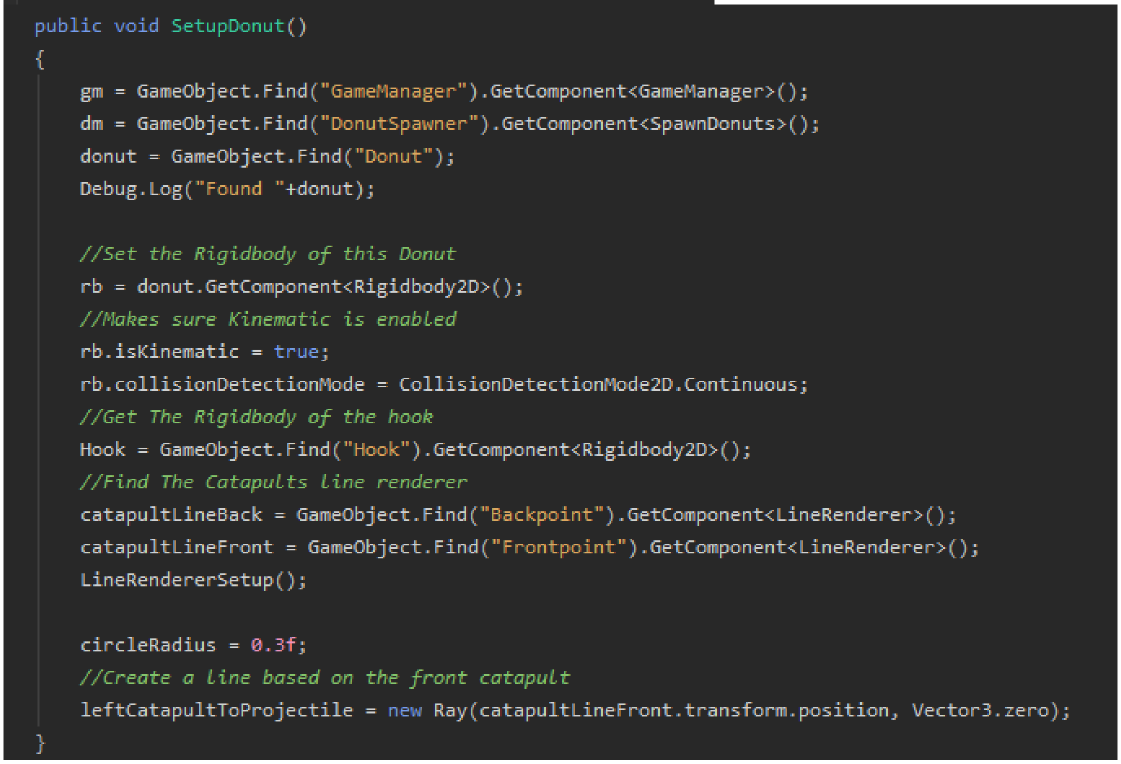
```
public class PhysicsController : MonoBehaviour
{
    public float ShakeForceMultiplier;
    public List<Rigi[illegible]body2D>();

    public void ShakeRigidbodies(Vector3 deviceAcceleration)
    {
        // For each Rigid body in the ShakingRigidbodies Array
        foreach (var rigidbody in rb)
        {
            //We are going to add Force to the rigidbody depending on how much the device is shaking and depending on the ShakeForceMultiplier
            rigidbody.AddForce(deviceAcceleration * ShakeForceMultiplier, ForceMode2D.Impulse);
        }
    }

    void Update()
    {
        //Keeps Checking the list
        for (int i =0; i < rb.Count; i++)
        {
            //If anything in the list has been destroyed
            if (rb[i] == null)
            {
                //Remove it from the list
                rb.RemoveAt(i);
            }
        }
    }
```

Drag and Shoot

The way the dragging and shooting on mobile works is that when the player puts their finger anywhere on the screen then the ball on the catapult will move to that position up until it reaches the max drag distance and this will be held in the variables “touch” and “maxDragDistance”. The max drag distance is set to 5 and the further the player drags the power is sent to the catapult so that the ball can launch higher. The touch variable once registered will also be converted to the world coordinates so that the position of the finger can be correctly placed to be the same position in the game. There is also a SetupDonut() function which grabs all the donuts information from the previous level and sets it up on the next level so that the donut can have the appropriate settings such as a rigidbody for it to fall down after being shot.

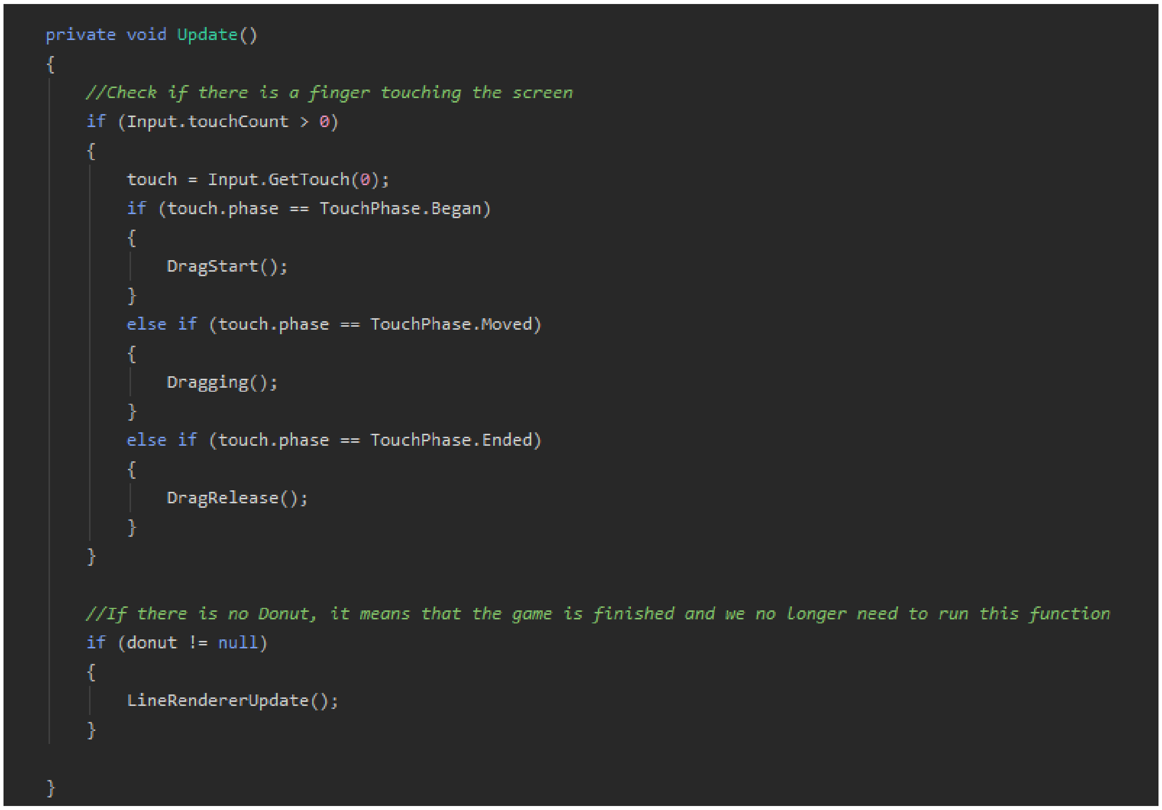


```
public class DonutControl : MonoBehaviour
{
    public float maxDragDistance = 5f;
    public float releaseTime = 0.15f;

    private float circleRadius;

    public Rigidbody2D rb;
    public Rigidbody2D Hook;

    SpawnDonuts dm;
    GameManager gm;

    public LineRenderer catapultLineFront;
    public LineRenderer catapultLineBack;

    private Ray leftCatapultToProjectile;
    Touch touch;

    [SerializeField]
    private GameObject donut;

    private void Start()
    {
        SetupDonut();
    }
```

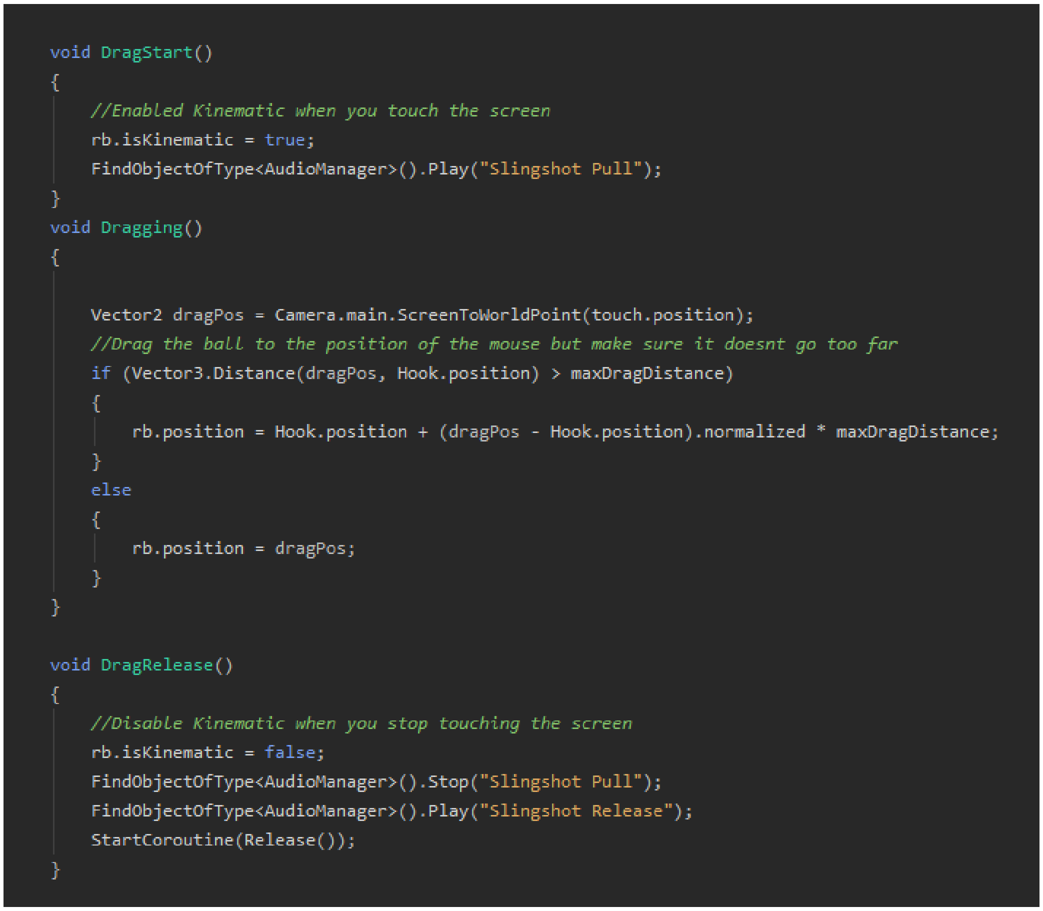


```
public void SetupDonut()
{
    gm = GameObject.Find("GameManager").GetComponent<GameManager>();
    dm = GameObject.Find("DonutSpawner").GetComponent<SpawnDonuts>();
    donut = GameObject.Find("Donut");
    Debug.Log("Found "+donut);

    //Set the Rigidbody of this Donut
    rb = donut.GetComponent<Rigidbody2D>();
    //Makes sure Kinematic is enabled
    rb.isKinematic = true;
    rb.collisionDetectionMode = CollisionDetectionMode2D.Continuous;
    //Get The Rigidbody of the hook
    Hook = GameObject.Find("Hook").GetComponent<Rigidbody2D>();
    //Find The Catapults line renderer
    catapultLineBack = GameObject.Find("Backpoint").GetComponent<LineRenderer>();
    catapultLineFront = GameObject.Find("Frontpoint").GetComponent<LineRenderer>();
    LineRendererSetup();

    circleRadius = 0.3f;
    //Create a line based on the front catapult
    leftCatapultToProjectile = new Ray(catapultLineFront.transform.position, Vector3.zero);
}
```

Since the game knows the player's touch position, it can check whether or not the player is ready to go and start dragging or if they have released their finger from the phone and this is done on the Update() function since the game will always be checking for player input. When the player touches the screen it will also call the DragStart() function so that if the player wants to keep moving the ball the game will call another function related to the script. If the player is currently still moving their finger after touching their phone it means they are still able to drag the ball to whatever position they would like to include the hook so that the ball can be slingshotted and this is called by the Dragging() function. Finally when the player releases their finger from their screen it will call the DragRelease() function which will call another method which is an IEnumerator Release() function and this allows the function to wait a couple of seconds before the donut can be released as well turning off the rubber band from the slingshot and then respawning the donut and turning everything back on again and this will continuously happen until the last donut has been shot by the player.

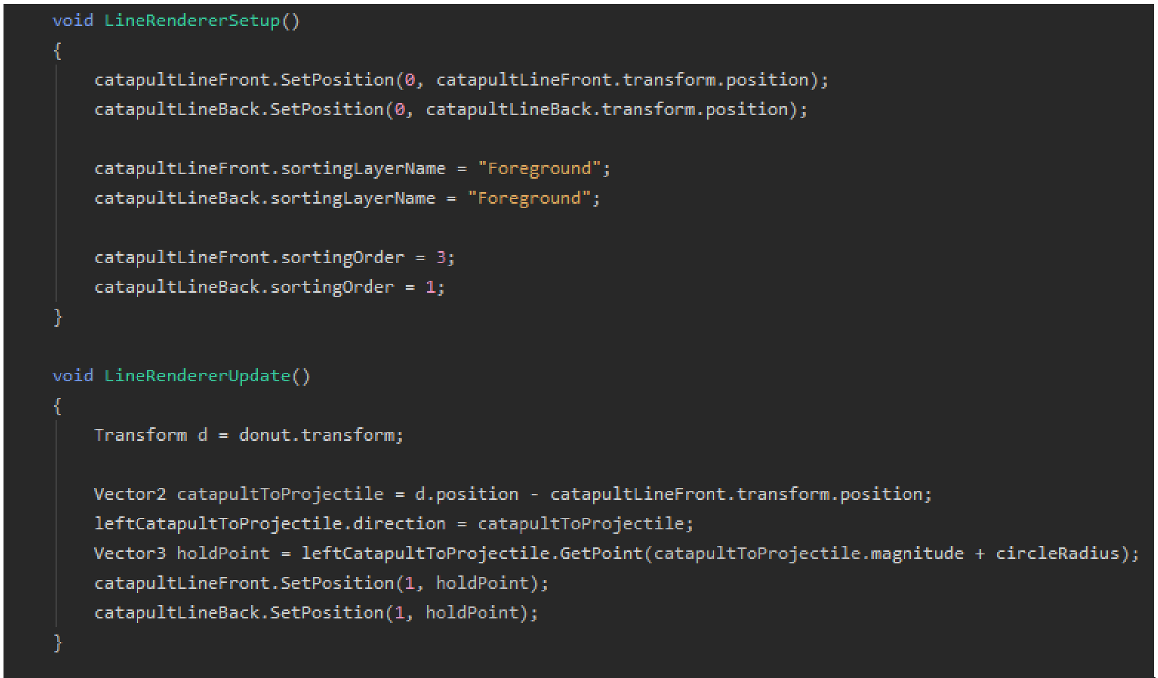


```
private void Update()
{
    //Check if there is a finger touching the screen
    if (Input.touchCount > 0)
    {
        touch = Input.GetTouch(0);
        if (touch.phase == TouchPhase.Began)
        {
            DragStart();
        }
        else if (touch.phase == TouchPhase.Moved)
        {
            Dragging();
        }
        else if (touch.phase == TouchPhase.Ended)
        {
            DragRelease();
        }
    }

    //If there is no Donut, it means that the game is finished and we no longer need to run this function
    if (donut != null)
    {
        LineRendererUpdate();
    }

}
```

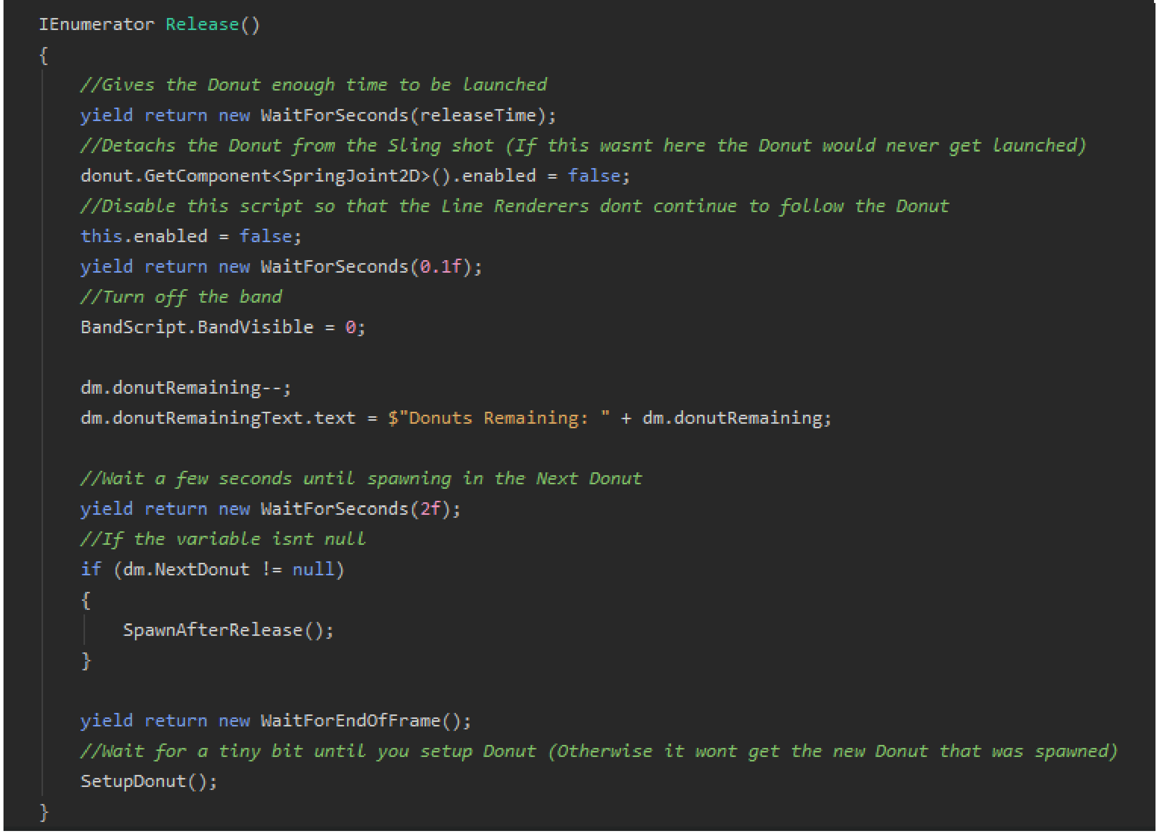


```
void DragStart()
{
    //Enabled Kinematic when you touch the screen
    rb.isKinematic = true;
    FindObjectOfType<AudioManager>().Play("Slingshot Pull");
}
void Dragging()
{

    Vector2 dragPos = Camera.main.ScreenToWorldPoint(touch.position);
    //Drag the ball to the position of the mouse but make sure it doesnt go too far
    if (Vector3.Distance(dragPos, Hook.position) > maxDragDistance)
    {
        rb.position = Hook.position + (dragPos - Hook.position).normalized * maxDragDistance;
    }
    else
    {
        rb.position = dragPos;
    }
}

void DragRelease()
{
    //Disable Kinematic when you stop touching the screen
    rb.isKinematic = false;
    FindObjectOfType<AudioManager>().Stop("Slingshot Pull");
    FindObjectOfType<AudioManager>().Play("Slingshot Release");
    StartCoroutine(Release());
}
```

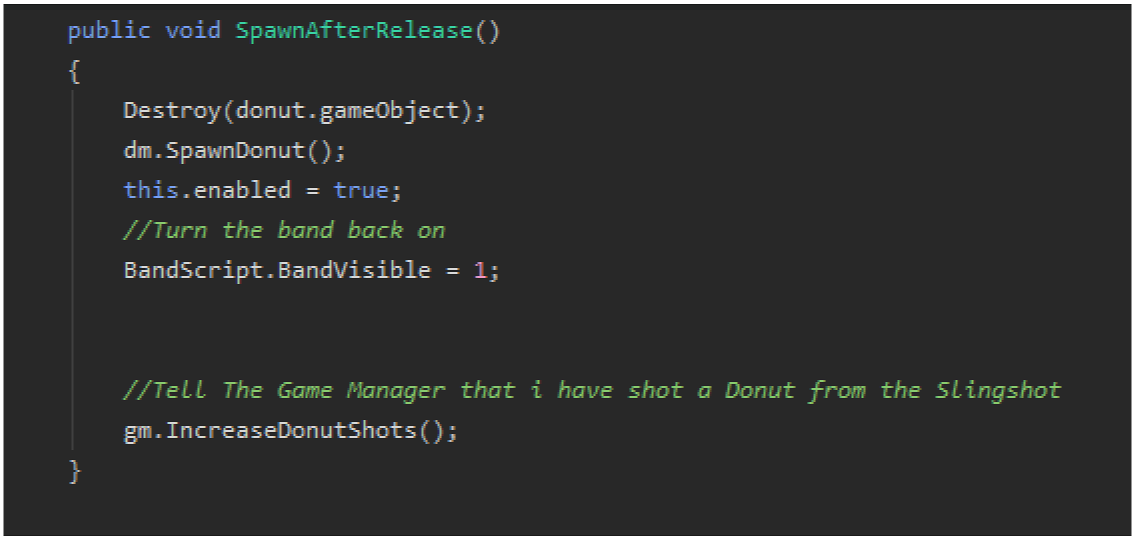
The way the rubber bands are setup is by Line Renderers and they will match from the catapults position and will be matched with the center of the ball's position and this is so that the ball won't hang and instead start of normally so that the player is able to drag the ball and get the correct power generated from dragging. The LineRendererUpdate() function is used to update the rubber bands position to where the ball is currently positioned and can correctly be resetted when the donut is respawned. The LineRendererSetup() function is used to set up the rubberband at the start of the game. Finally the SpawnAfterRelease() function is called in the IEnumerator and this will also turn the rubber band back on after respawning the donut back at the spawn point.



```
    void LineRendererSetup()
    {
        catapultLineFront.SetPosition(0, catapultLineFront.transform.position);
        catapultLineBack.SetPosition(0, catapultLineBack.transform.position);

        catapultLineFront.sortingLayerName = "Foreground";
        catapultLineBack.sortingLayerName = "Foreground";

        catapultLineFront.sortingOrder = 3;
        catapultLineBack.sortingOrder = 1;
    }

    void LineRendererUpdate()
    {
        Transform d = donut.transform;

        Vector2 catapultToProjectile = d.position - catapultLineFront.transform.position;
        leftCatapultToProjectile.direction = catapultToProjectile;
        Vector3 holdPoint = leftCatapultToProjectile.GetPoint(catapultToProjectile.magnitude + circleRadius);
        catapultLineFront.SetPosition(1, holdPoint);
        catapultLineBack.SetPosition(1, holdPoint);
    }
```



```
IEnumerator Release()
{
    //Gives the Donut enough time to be launched
    yield return new WaitForSeconds(releaseTime);
    //Detachs the Donut from the Sling shot (If this wasnt here the Donut would never get launched)
    donut.GetComponent<SpringJoint2D>().enabled = false;
    //Disable this script so that the Line Renderers dont continue to follow the Donut
    this.enabled = false;
    yield return new WaitForSeconds(0.1f);
    //Turn off the band
    BandScript.BandVisible = 0;

    dm.donutRemaining--;
    dm.donutRemainingText.text = $"Donuts Remaining: " + dm.donutRemaining;

    //Wait a few seconds until spawning in the Next Donut
    yield return new WaitForSeconds(2f);
    //If the variable isnt null
    if (dm.NextDonut != null)
    {
        SpawnAfterRelease();
    }

    yield return new WaitForEndOfFrame();
    //Wait for a tiny bit until you setup Donut (Otherwise it wont get the new Donut that was spawned)
    SetupDonut();
}
```



```
public void SpawnAfterRelease()
{
    Destroy(donut.gameObject);
    dm.SpawnDonut();
    this.enabled = true;
    //Turn the band back on
    BandScript.BandVisible = 1;


    //Tell The Game Manager that i have shot a Donut from the Slingshot
    gm.IncreaseDonutShots();
}
```

Characters

The main object in the game is of course the donut that the player will create in the game. They are able to choose from many flavors creating different donuts for each experience they get. They can choose whether they want sprinkles or not, or whether they want a chocolate donut or a rainbow donut, The main design of the donut is up for the player to decide which gives them a much more fun experience.

Gameplay

The player's main activity is to get through each donut process which represents a level and they have to beat the level so that the donut can evolve. Once it gets to the end the player can package the donut and see their finished product.

Winning state of each level:

Ingredient Level: Each level has its win state and loses state. For example, in the ingredients level, the player will win if they get all the correct amount of ingredients in the correct container within the set time, however, if they don't then they will lose and they can restart the level.

Roll A Dough Level: With the Roll A Dough level in order for the player to win they would have to get the dough ball to the finish line within the time limit given however there are also certain obstacles that can stop the dough ball from progressing and this ranges from a rock to a donut cutter which will instantly destroy the dough ball and the player will lose.

Icing/Sprinkles Level: For the sprinkles/Icing Level the player will always win as the main objective for the player is to customize their Donut. The level starts off with the Player being able to choose a flavor of icing for their Donut, once they have chosen a flavor an icing droplet will drop onto the Donut and there will be icing on the Donut.

Once the Icing has been applied to the Donut the player will have 10 seconds to shake their phone which will spawn in Sprinkles that will then land on the Donut, after the 10 seconds are over the mini-game is over.

Packing Level: In the Packing Level in order for the player to win they would have to at least get 1 donut that they have created into the packet that is falling down and swinging at the same time and to do this, they would have to shoot the donut on to it so that it can be safely packed. After you have shot all your donuts, then depending if you have gotten at least one donut into a packet then you win the minigame and have fully completed the entire game.

Losing state of each level:

Ingredient Level: With the Ingredient Level, in order for the player to lose they would have to run out of time and not fully fill up every ingredient needed to make the dough ball.

Roll A Dough Level: The player will lose in this minigame if the player does not reach the finish line within the given time.

Icing/Sprinkles Level: There is no losing state for this minigame since we want the player to always create their finished donut so that it can be showcased and be ready for the next minigame.

Packing Level: To lose in the Packing mini-game the player would have to miss all their shots on the packets that are flying in and falling down.

Checkpoints: There are checkpoints every time the player passes a minigame until they get to the end and when they get to any minigame they are able to restart and it will load the current minigame that they are on and they would be able to resume there.

How do we keep A Losing Player Engaged: We keep the losing player engaged by adding in a rating system, so if the player does not reach the max amount of stars given for each level then they can restart the game so that they can achieve the highest level there is for each level and beating the game with each level having the maximum stars earned. Also, the game is highly interactive so the player will always want to try and beat their times or scores for all minigames in this game and this is the main way to keep a losing player engaged.

Game Objectives & Rewards

The objective of the game is to complete all the available levels with the highest rating possible. Each level will have a set amount of mini-games that must be completed in order to finish the whole level, these mini-games include the Ingredients Mini game, the Roll A Ball mini-game, the sprinkles mini-game, and the packing mini-game.

Each of these mini-games has its own challenges and penalties but the reward of them is pretty much the same in that you will get a rating and in how you will be getting a Donut.

Challenges And Penalties Of Each Mini-Game:

Ingredients Mini-Game: The Challenge for the Ingredients mini-game is that you will need to successfully put each ingredient into the right container within the time limit. If you put the right ingredient into the right container you will get time added onto your timer but if you put the wrong ingredient into the wrong container then you will receive a penalty and will lose 10 seconds from the timer.

The challenges in the ingredients mini-game aren't only putting the ingredients into the right container and trying to not make the Timer hit 0, there is also the speed of the conveyor belt which its speed will depend on the level of difficulty. The harder the level the faster the conveyor belt that the Ingredients come from and if you just let the Ingredient go down the Conveyor belt then you will lose time.

Roll A Ball Mini-Game: The Challenge of the Roll A Ball mini-game is that you will need to make your dough ball as big as possible by rolling it past the obstacles and making sure you get to the finish line before time runs out. There are 2 different types of obstacles: a rock and a donut cutter. If the dough ball runs into the rock obstacle, then the size of the Donut will reset to its starting size. If the dough ball runs into the donut cutter obstacle then you will just lose the game and will have to retry.

Sprinkles Mini-Game: There are no challenges or penalties in this mini-game, all you need to do is pick an icing flavor and then add sprinkles to your Donut by shaking your phone.

Packing Mini-Game: The Challenge of the Packing Mini-Game is that you will need to shoot the Donut from a slingshot into the Packages that are floating down, when the Donut goes into the package then your score will increase. There isn't really a penalty if you miss the donut it just means that you won't be able to get the max score but if you miss all your Donuts then you will lose the mini-game and will have to retry the mini-game.

Audio

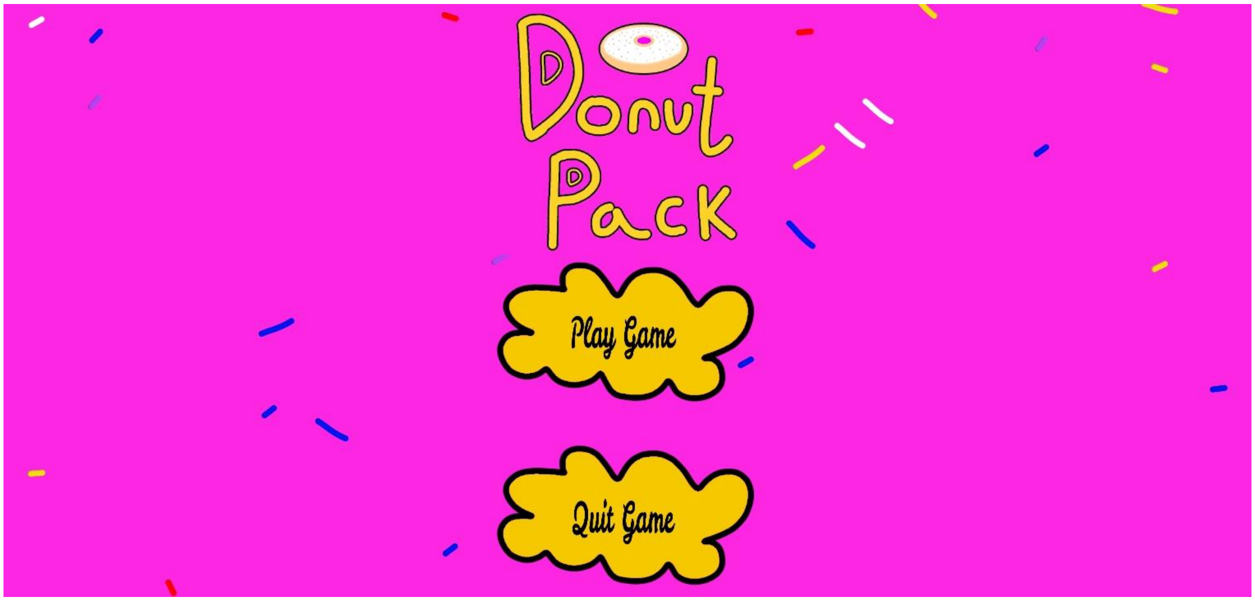
The music and sound effects varies between different levels and they all provide a very nice and immersive experience for example, when the right ingredient is put into the right container a cheerful sound will play that indicates the right choice was made but if you put the wrong ingredient into the wrong container then a deep sounding sound effect will play that indicated you made the wrong choice.

List Of Sounds:

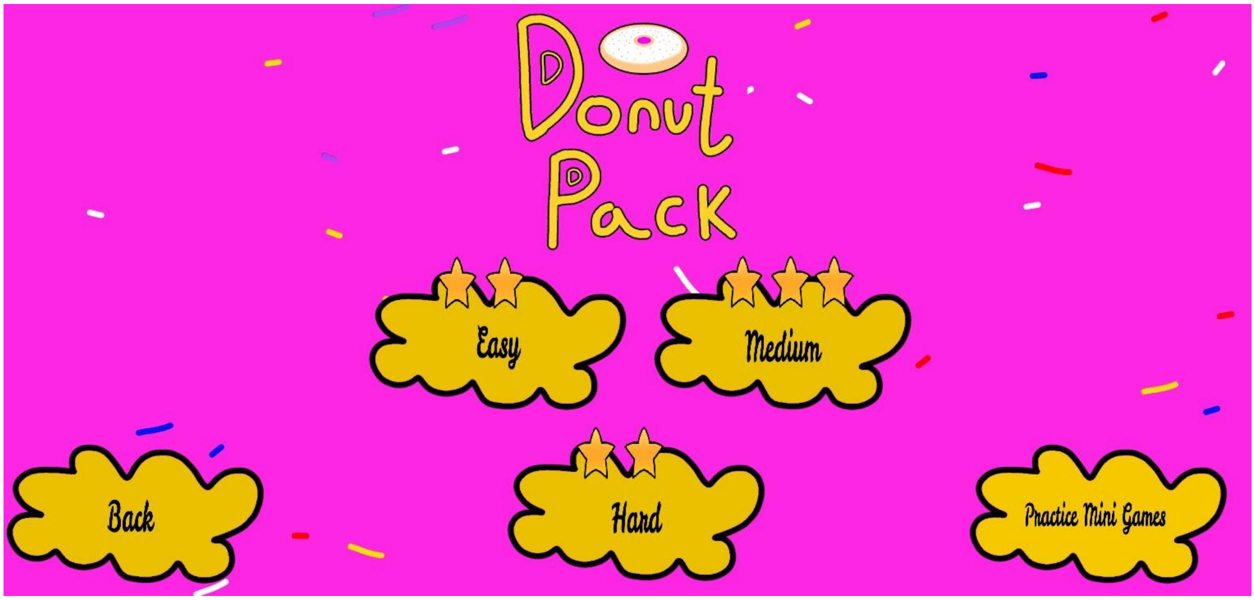
Button Pressing SFX - Made By Rhys Durrant (A Friend)
Wrong Container SFX - Made By Rhys Durrant (A Friend)
BGM For Ingredients Mini Game - Made By Rhys Durrant (A Friend)
BGM For Roll A Ball Mini Game - Made By Rhys Durrant (A Friend)
BGM For Sprinkles Mini Game - Made By Rhys Durrant (A Friend)
BGM For Final Rating Scene - Made By Rhys Durrant (A Friend)
Right Container SFX - Came From unadamlar
Clapping SFX - Came From SFX
Shaker SFX - Came From Asparagus_P
Splatter SFX - Came From gprosse
Icing Machine SFX - Came From deleted_user_4401185
Single Wrapping Paper Tear - Came From freesfx.co.uk

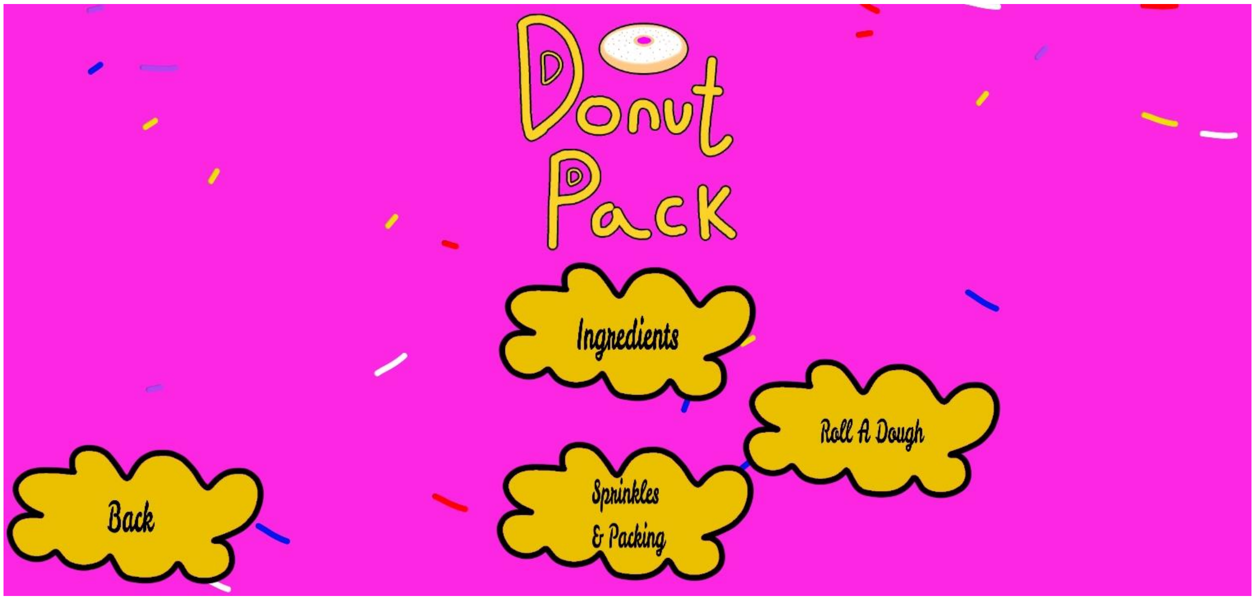
User Interface

The user interface for the game is quite simple and understandable for the player to get going.

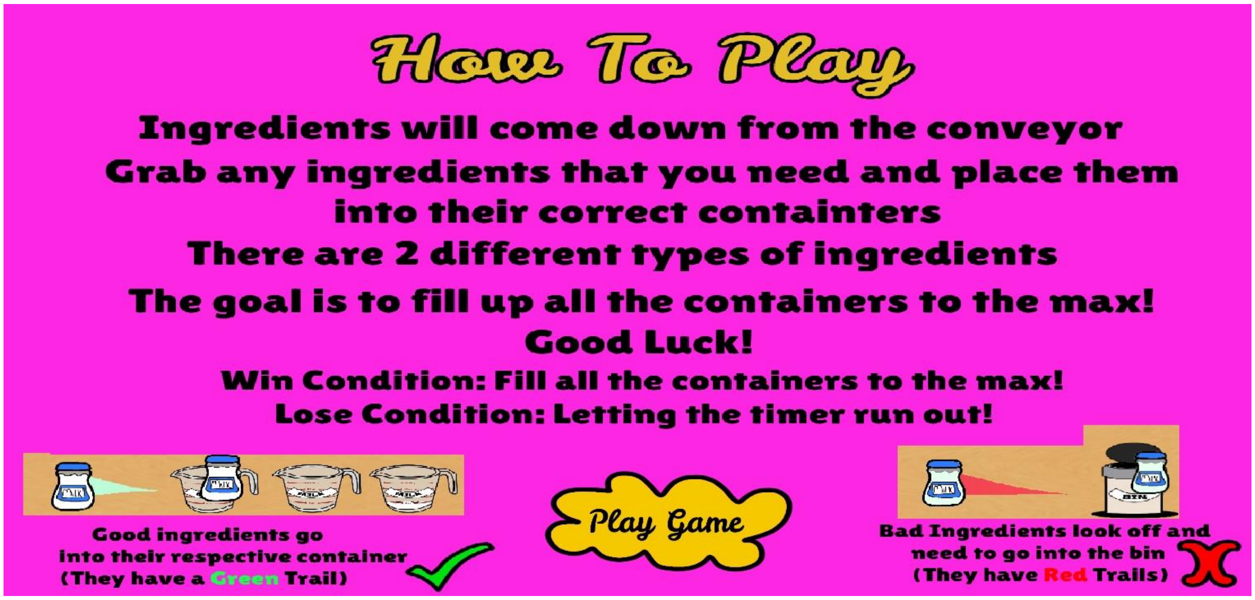


The player is prompted at the main menu to start the game or quit the game! If they quit the game the application will close itself.





Once the player clicks to play the game they will be prompt will difficulties to creating their donut as well as being able to practice each level individually to get better at it.



How To Play

Ingredients will come down from the conveyor
Grab any ingredients that you need and place them into their correct containters
There are 2 different types of ingredients
The goal is to fill up all the containers to the max!
Good Luck!

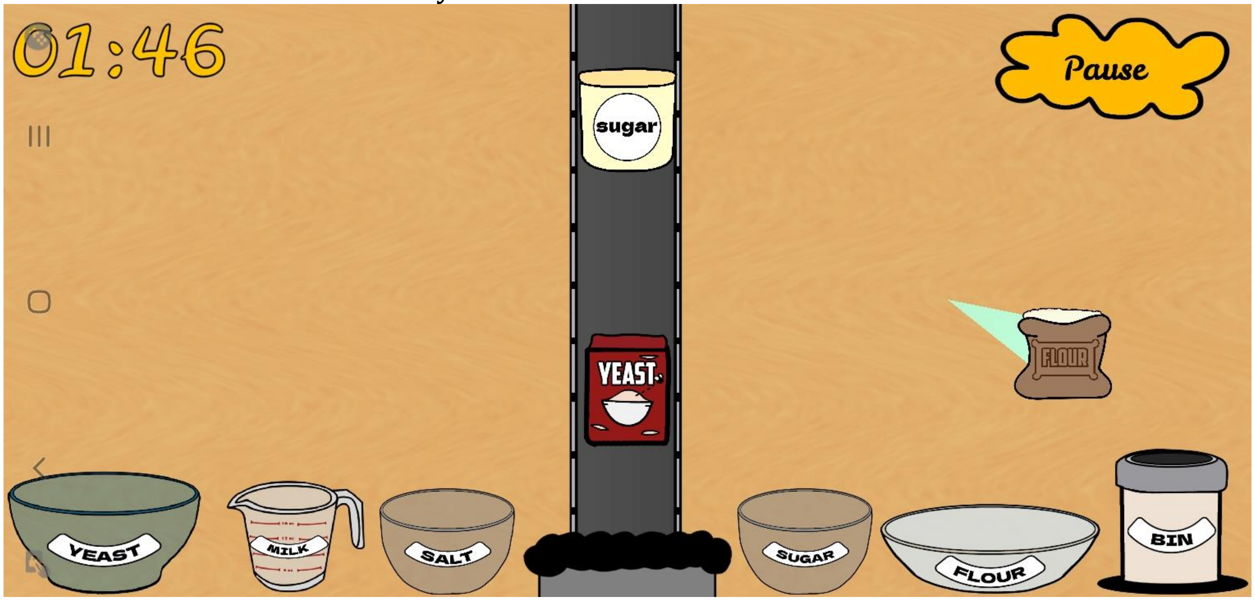
Win Condition: Fill all the containers to the max!
Lose Condition: Letting the timer run out!

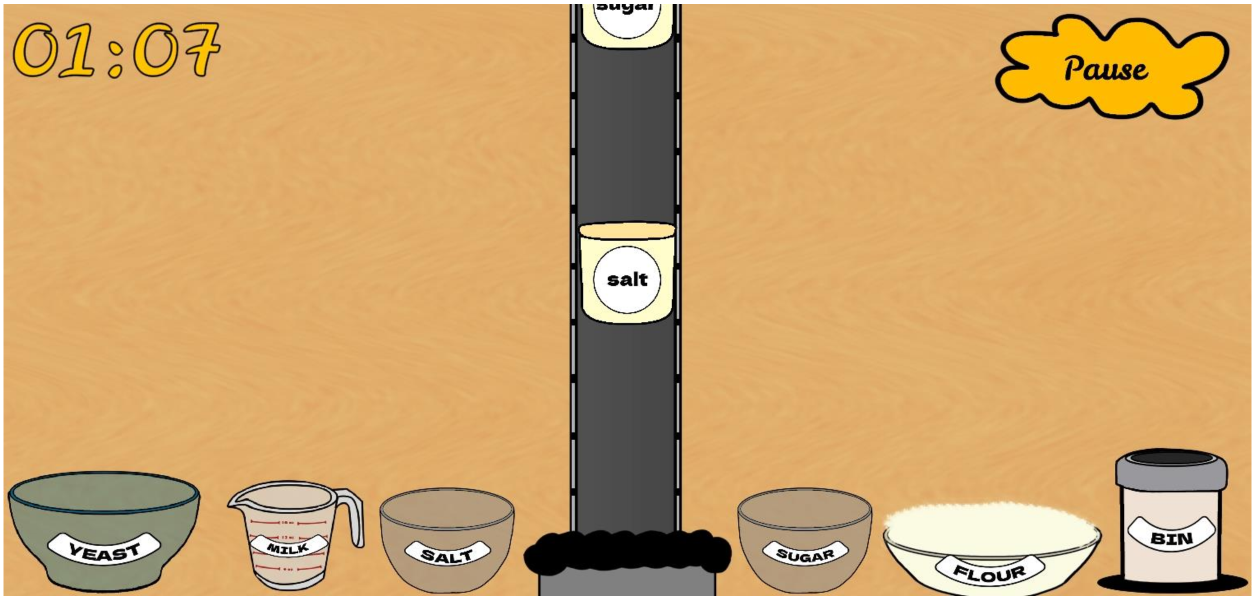
Good ingredients go into their respective container (They have a Green Trail)

Bad Ingredients look off and need to go into the bin (They have Red Trails)

Before heading into levels players will be prompt with a how-to-play scene regardless of whether they know how to play the game or not to understand how the level works as well as knowing the win and losing conditions.

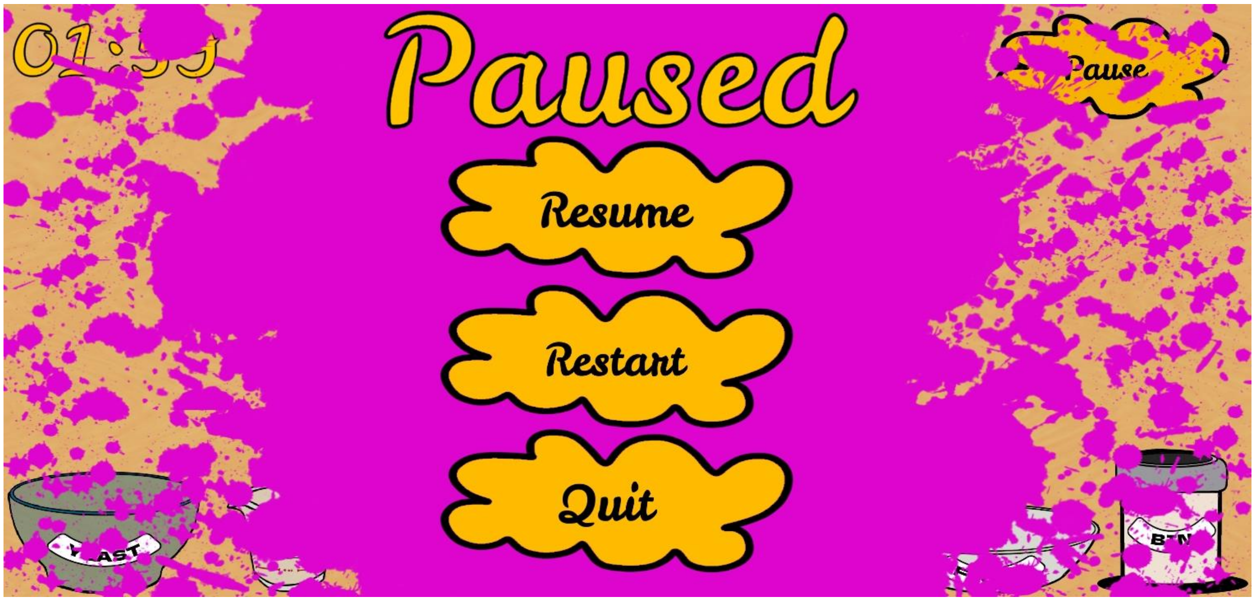
The Items have UI representation when being hold on to understand whether the item is correct or not so that when the player picks up an item they will know if it is good to pot into the bowl or to throw it away in the bin.



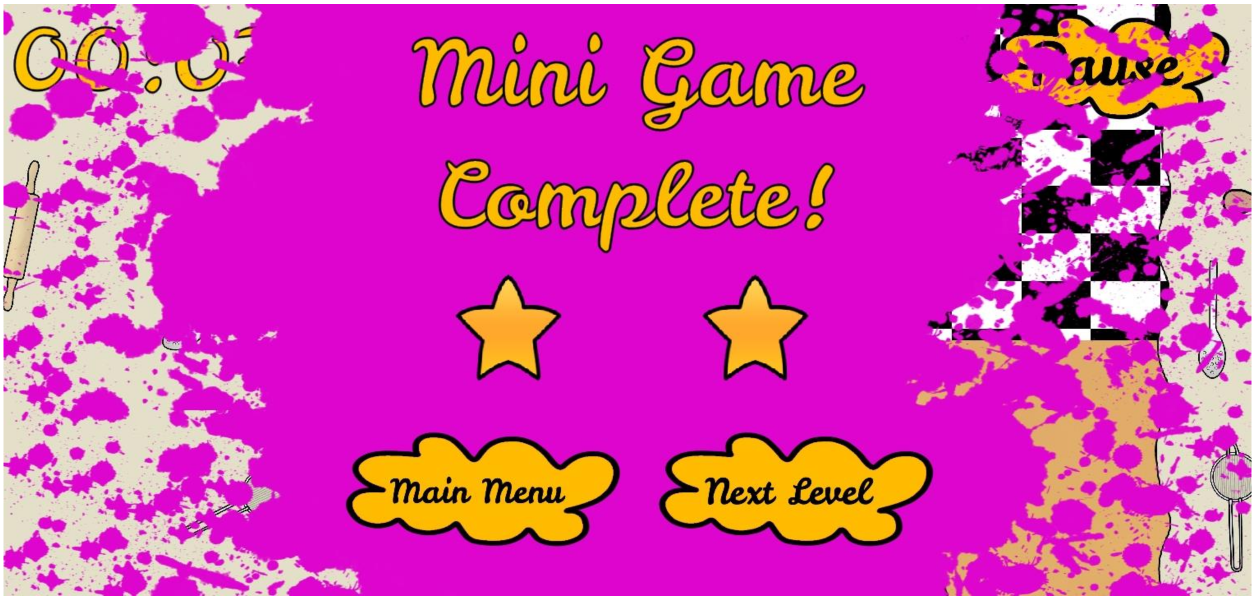


If correct the game will display a bonus time score as well as filling up the ingredient bowl they added it to.

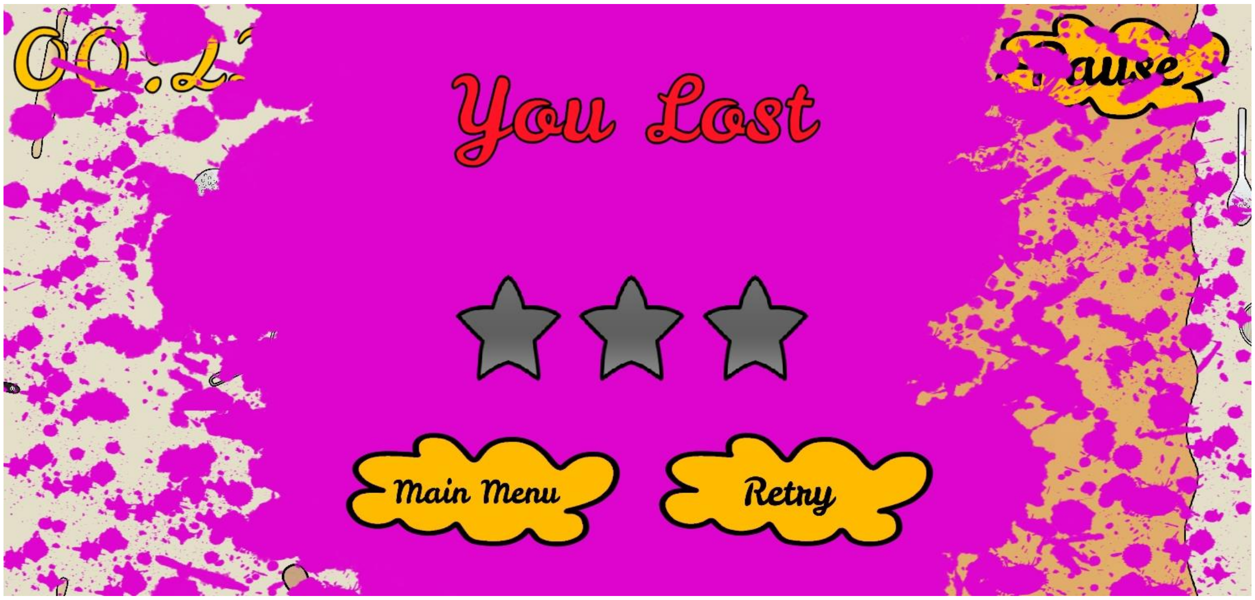
Depending on the difficulty the bowl will fill at different paces (This is the easy difficulty therefore the bowl has filled in one go as that is what the player only needs for the easiest difficulty).



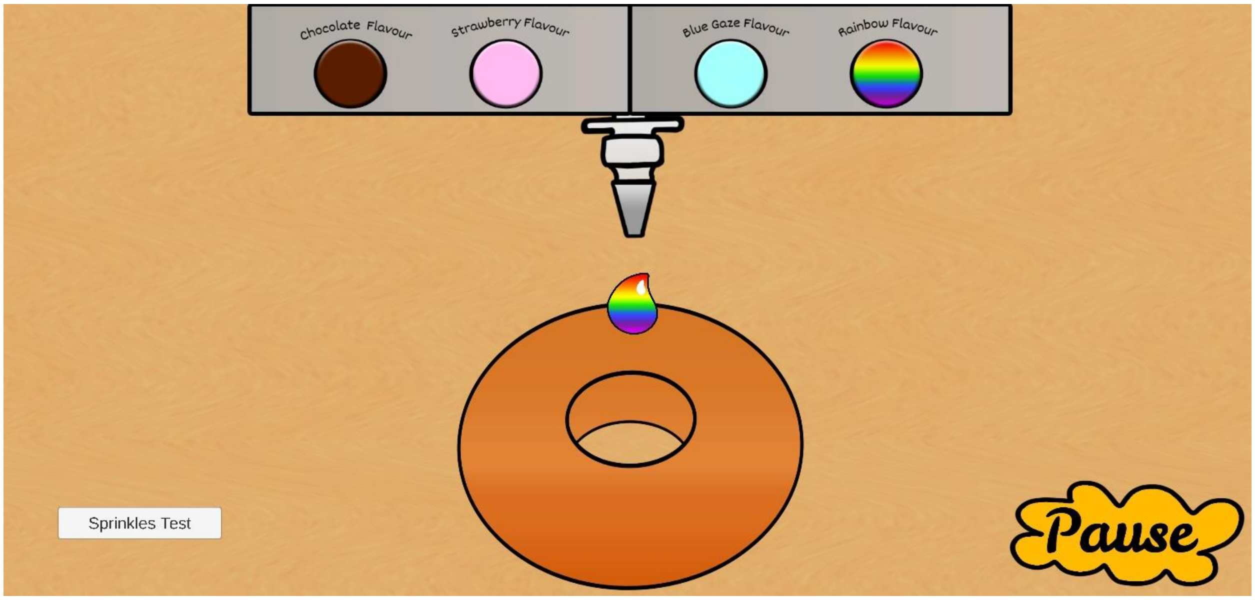
If a player is to pause during the game they are prompted with 3 buttons which are resuming the game, restarting the level they are currently playing, or quitting the game and returning to the main menu.

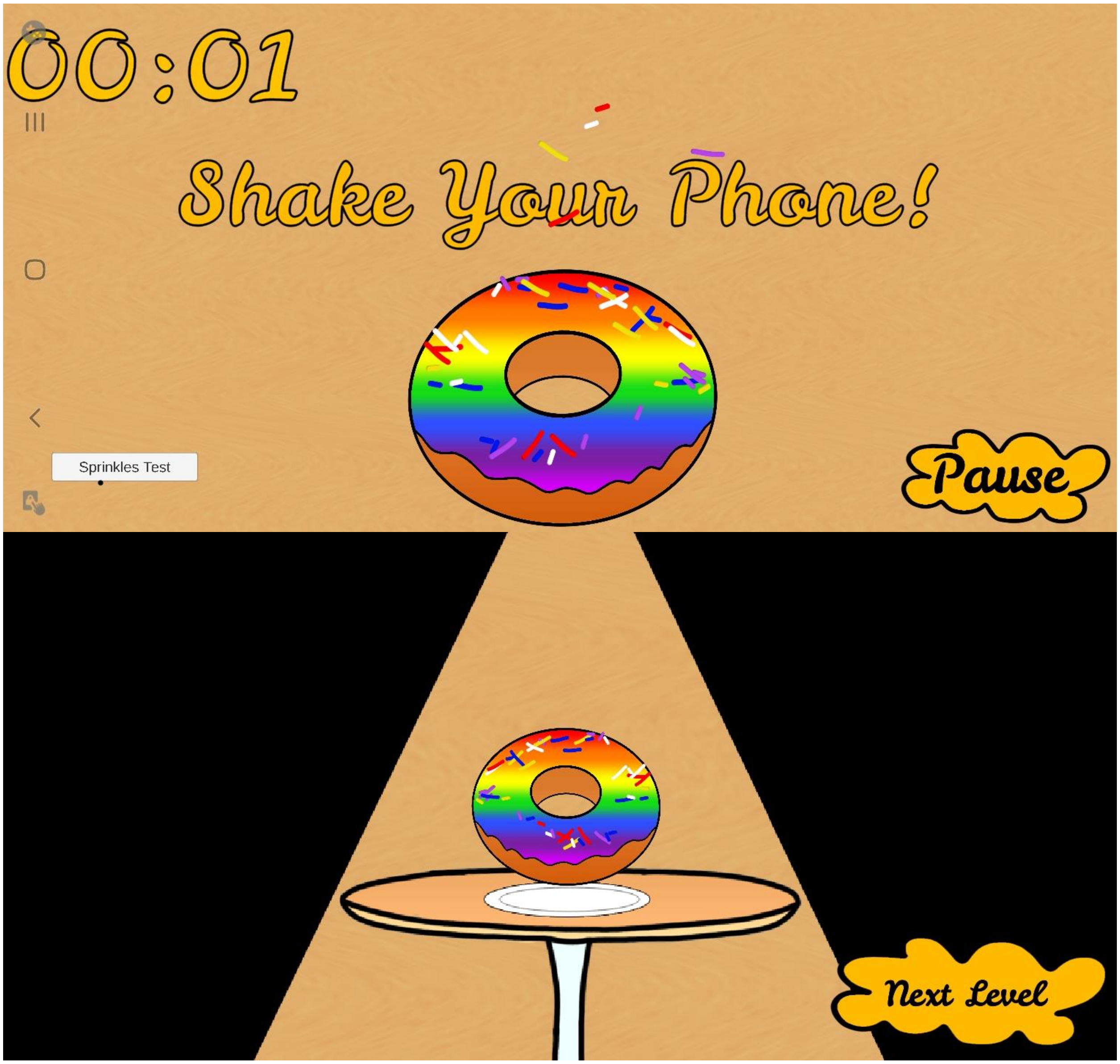


If they proceed to beat a mini-game they will be graded on how well they did using a rating system. If they play flawlessly the game will proceed to give them 3 stars. If they make a few mistakes the game will give them 2 stars and if they perform poorly but still are able to win they will earn 1 star. They will be prompted to return to the menu or head to the next level.

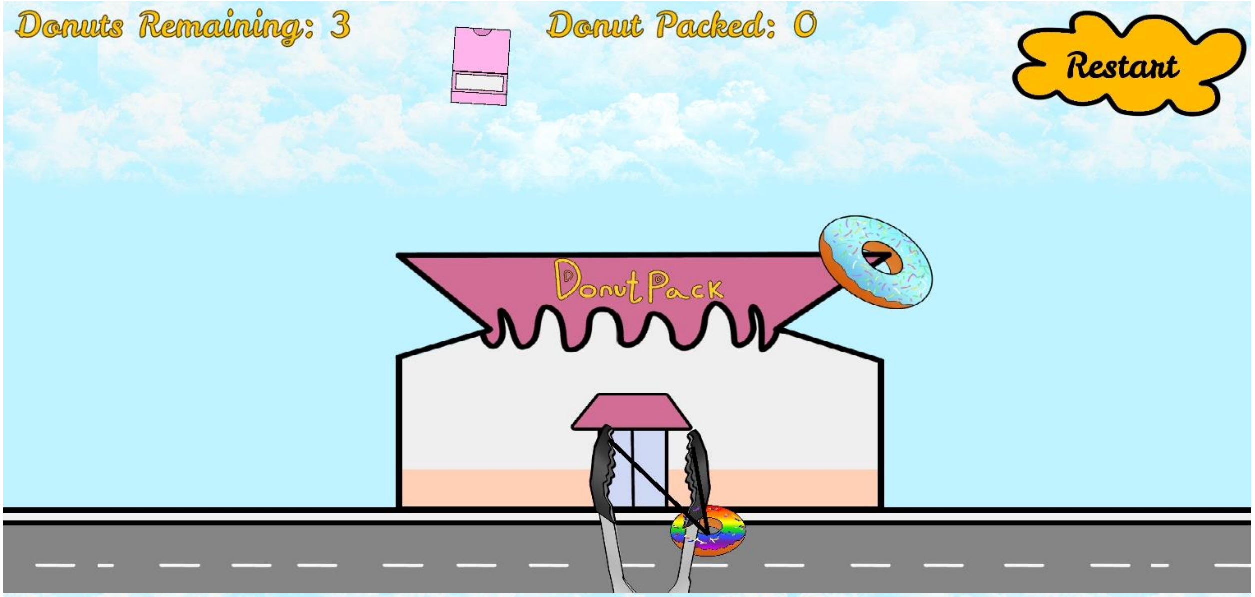


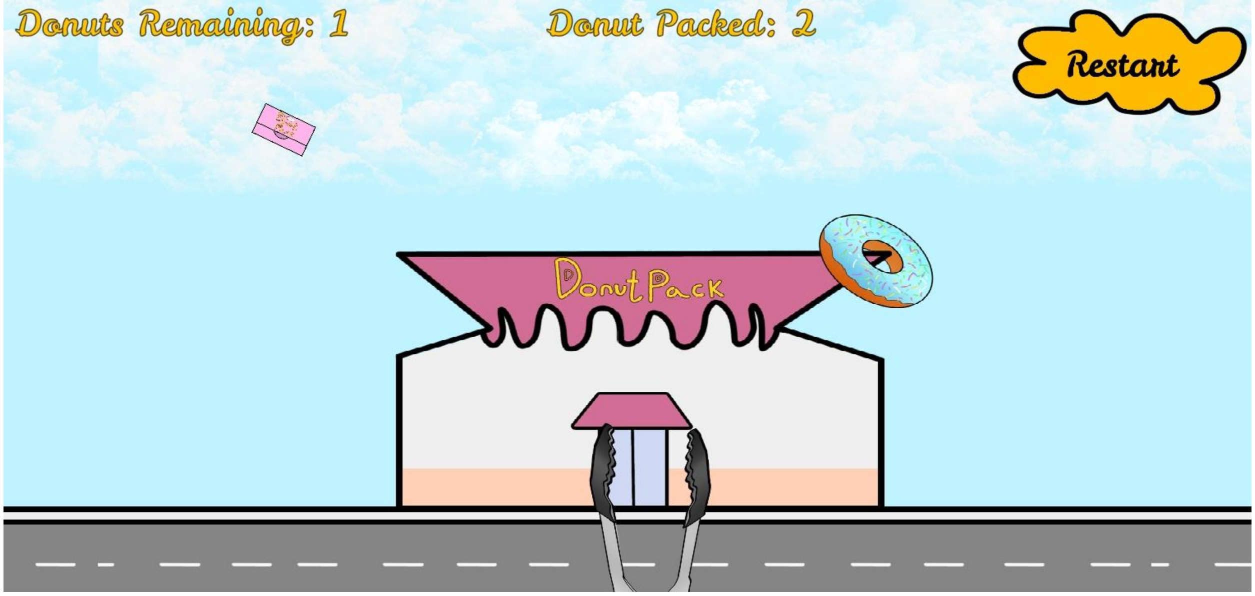
If the player is to lose in a mini game they will be prompted with the power to retry the mission or return to the main menu as well as knowing they did not receive any stars for their victory.


Chocolate Flavour
Strawberry Flavour
Blue Gaze Flavour
Rainbow Flavour
Sprinkles Test
Pause

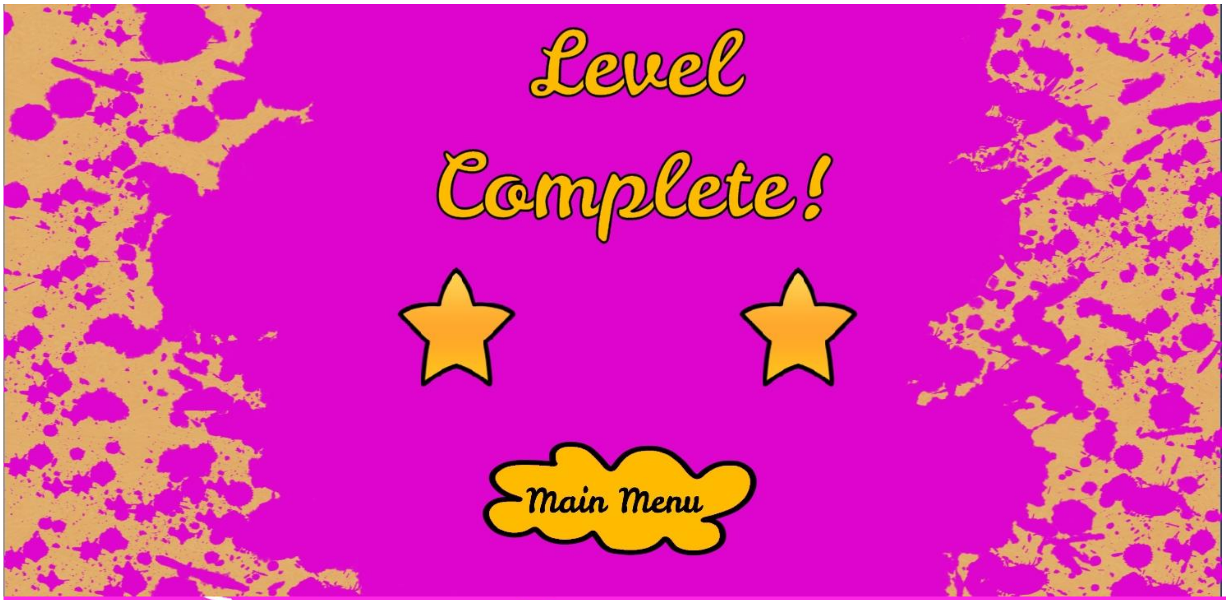


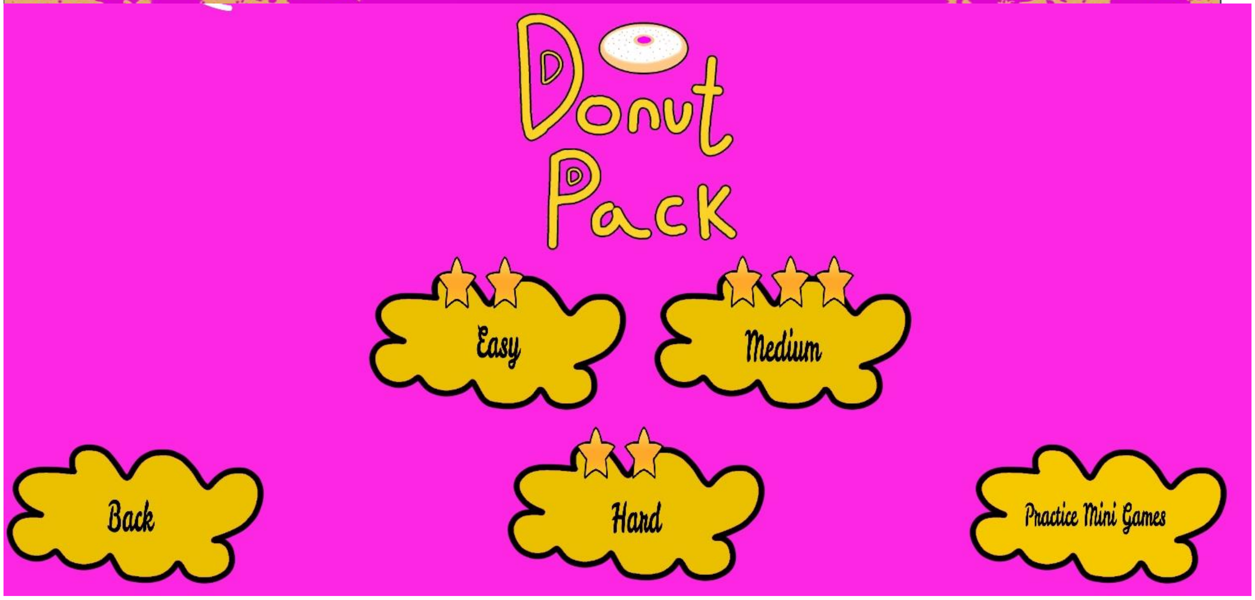
Not every mini-game is rated. The frosting and icing scene is there for the player to create their donut and show it off. Once choosing a flavor the player is even prompted to shake the phone with a big text telling them to do so within the timer.





When it comes to the packing game the player can see that they are stretching the tong slinger in the direction they wanna shoot with visible proof. If the donut collides with the box the player will know as the box will close as well as knowing how many donuts they packed.





When beating every level the player is prompted with a level completed scene where they will see all their overall rating with the final level rating. If they get 3 stars on every level they will receive a 3-star rating for the level of difficulty they have beaten. If they have a few 2 stars then they will receive 2 stars and if they only get one star they will be given a one-star rating for the difficulty. The player can also see in the main menu how many stars they got overall in each difficulty.

References

Programming/Video Harvard References

- Answers.unity.com. 2021. Change material of Particle System? - Unity Answers. [online] Available at: <https://answers.unity.com/questions/939421/change-material-of-particle-system.html> [Accessed 27 November 2021].

- Answers.unity.com. 2021. Convert int to string - Unity Answers. [online] Available at: <https://answers.unity.com/questions/55781/convert-int-to-string.html> [Accessed 14 December 2021].

- Answers.unity.com. 2021. Default Null Value for Key Not Found in Dictionary? - Unity Answers. [online] Available at: <https://answers.unity.com/questions/197156/default-null-value-for-key-not-found-in-dictionary.html> [Accessed 14 December 2021].

- Answers.unity.com. 2021. Grab a specific item from a list - Unity Answers. [online] Available at: <https://answers.unity.com/questions/1081177/grab-a-specific-item-from-a-list.html> [Accessed 4 November 2021].

- Answers.unity.com. 2021. how can i make a List of rigidbody - Unity Answers. [online] Available at: <https://answers.unity.com/questions/1433543/how-can-i-make-a-list.html> [Accessed 14 October 2021].

- Answers.unity.com. 2021. How delete or remove a component of an GameObject? - Unity Answers. [online] Available at: <https://answers.unity.com/questions/378930/how-delete-or-remove-a-component-of-an-gameobject.html> [Accessed 6 December 2021].

- Answers.unity.com. 2021. How to add a Script to a GameObject during Runtime [2016] - Unity Answers. [online] Available at: <https://answers.unity.com/questions/1136397/how-to-add-a-script-to-a-gameobject-during-runtime.html> [Accessed 6 December 2021].

- Answers.unity.com. 2021. How to avoid scaling heritage when parenting. - Unity Answers. [online] Available at: <https://answers.unity.com/questions/147816/how-to-avoid-scaling-heritage-when-parenting.html> [Accessed 19 October 2021].

- Answers.unity.com. 2021. How to GameObject.FindGameObjectsWithTag within children of a specific GameObject? - Unity Answers. [online] Available at: <https://answers.unity.com/questions/1197131/how-to-gameobjectfindgameobjectswithtag-within-chi.html> [Accessed 14 December 2021].

- Answers.unity.com. 2021. how to know when list is empty? - Unity Answers. [online] Available at: <https://answers.unity.com/questions/1072078/how-to-know-when-list-is-empty.html> [Accessed 4 November 2021].

- Answers.unity.com. 2021. Is it possible to get a prefab object from its asset path? - Unity Answers. [online] Available at: <https://answers.unity.com/questions/313398/is-it-possible-to-get-a-prefab-object-from-its-ass.html> [Accessed 22 November 2021].

- Answers.unity.com. 2021. Landscape mode only - Unity Answers. [online] Available at: <https://answers.unity.com/questions/774186/landscape-mode-only.html> [Accessed 14 October 2021].

- Answers.unity.com. 2021. Mobile Game Different Screen Size Issue - Unity Answers. [online] Available at: <https://answers.unity.com/questions/1731052/mobile-game-different-screen-size-issue.html> [Accessed 27 October 2021].

- Answers.unity.com. 2021. OnTriggerEnter2D called twice sometimes when IsTrigger is ON - Unity Answers. [online] Available at: <https://answers.unity.com/questions/1531176/ontriggerenter2d-called-twice-sometimes-when-istri.html> [Accessed 4 November 2021].

- Answers.unity.com. 2021. Place UI element in background (Behind ALLLL game objects) - Unity Answers. [online] Available at: <https://answers.unity.com/questions/1632682/place-ui-element-in-background-behind-allll-game-o.html> [Accessed 14 December 2021].

- Answers.unity.com. 2021. "Set-up Android SDK path to make Android remote work" 2019.1.1 - Unity Answers. [online] Available at: <https://answers.unity.com/questions/1628598/set-up-android-sdk-path-to-make-android-remote-wor.html> [Accessed 13 October 2021].

- Answers.unity.com. 2021. Shuriken particles unlimited lifetime - Unity Answers. [online] Available at: <https://answers.unity.com/questions/250745/shuriken-particles-unlimited-lifetime.html> [Accessed 17 October 2021].

- Answers.unity.com. 2021. Why does my GUI Text look so small on my phone but normal in Unity Editor? - Unity Answers. [online] Available at: <https://answers.unity.com/questions/831151/why-does-my-gui-text-look-so-small-on-my-phone-but.html> [Accessed 29 November 2021].

- Answers.unity.com. 2021. Why is ARM64 greyed out? How do I turn it back on to meet the Android 64 bit requirements? - Unity Answers. [online] Available at: <https://answers.unity.com/questions/1627961/why-is-arm64-greyed-out-how-do-i-turn-it-back-on-t.html> [Accessed 25 October 2021].

- array, D., Castor, K. and Sinclair, C., 2021. Disabling all gameobjects in an array. [online] Stack Overflow. Available at:

<https://stackoverflow.com/questions/32259268/disabling-all-gameobjects-in-an-array> [Accessed 27 November 2021].

- Develop, M., Toma, D., Wheeler, D. and ersan, a., 2021. How to make a countdown timer in Unity (in minutes + seconds) - Game Dev Beginner. [online] Game Dev Beginner. Available at: <https://gamedevbeginner.com/how-to-make-countdown-timer-in-unity-minutes-seconds/> [Accessed 16 October 2021].

- Dotnetperls.com. 2021. C# Convert Dictionary to String (Write to File). [online] Available at: <https://www.dotnetperls.com/convert-dictionary-string> [Accessed 14 December 2021].

- Freesound. 2021. Correct Choice by unadamlar. [online] Available at: <https://freesound.org/people/unadamlar/sounds/476178/> [Accessed 14 December 2021].

- Freesound. 2021. Generator #1.wav by deleted_user_4401185. [online] Available at: <https://freesound.org/people/deleted_user_4401185/sounds/243735/> [Accessed 14 December 2021].

- Freesound. 2021. Rolling Metal Conveyor Belt.wav by Razzvio. [online] Available at: <https://freesound.org/people/Razzvio/sounds/79573/> [Accessed 12 December 2021].

- Freesound. 2021. shaker.aif by Asparagus_P. [online] Available at: <https://freesound.org/people/Asparagus_P/sounds/342301/> [Accessed 12 December 2021].

- Freesound. 2021. Slingshot 2.wav by olver. [online] Available at: <https://freesound.org/people/olver/sounds/513933/> [Accessed 14 December 2021].

- Freesound. 2021. Splatter by dersuperanton. [online] Available at: <https://freesound.org/people/dersuperanton/sounds/434479/> [Accessed 14 December 2021].

- Game Dev Beginner. 2021. How to change a Sprite from a script in Unity (with examples) - Game Dev Beginner. [online] Available at: <https://gamedevbeginner.com/how-to-change-a-sprite-from-a-script-in-unity-with-examples/> [Accessed 17 November 2021].

- Sciencing. 2021. How to Calculate a 10 Percent Discount. [online] Available at: <https://sciencing.com/calculate-10-percent-discount-6397158.html> [Accessed 23 November 2021].

- Technologies, U., 2021. Unity - Scripting API: Collider.enabled. [online] Docs.unity3d.com. Available at: <https://docs.unity3d.com/ScriptReference/Collider-enabled.html> [Accessed 19 October 2021].

- Technologies, U., 2021. Unity - Scripting API: GameObject.SetActive. [online] Docs.unity3d.com. Available at: <https://docs.unity3d.com/ScriptReference/GameObject.SetActive.html> [Accessed 31 October 2021].

- Technologies, U., 2021. Unity - Scripting API: MonoBehaviour.OnTriggerEnter2D(Collider2D). [online] Docs.unity3d.com. Available at: <https://docs.unity3d.com/ScriptReference/MonoBehaviour.OnTriggerEnter2D.html> [Accessed 14 October 2021].

- Technologies, U., 2021. Unity - Scripting API: ParticleSystem. [online] Docs.unity3d.com. Available at: <https://docs.unity3d.com/ScriptReference/ParticleSystem.html> [Accessed 28 November 2021].

- Technologies, U., 2021. Unity - Scripting API: ParticleSystem.main. [online] Docs.unity3d.com. Available at: <https://docs.unity3d.com/ScriptReference/ParticleSystem-main.html> [Accessed 28 November 2021].

- Technologies, U., 2021. Unity - Scripting API: ParticleSystem.textureSheetAnimation. [online] Docs.unity3d.com. Available at: <https://docs.unity3d.com/ScriptReference/ParticleSystem-textureSheetAnimation.html> [Accessed 28 November 2021].

- Technologies, U., 2021. Unity - Scripting API: Physics.IgnoreLayerCollision. [online] Docs.unity3d.com. Available at: <https://docs.unity3d.com/ScriptReference/Physics.IgnoreLayerCollision.html> [Accessed 3 December 2021].

- Technologies, U., 2021. Unity - Scripting API: Random.Range. [online] Docs.unity3d.com. Available at: <https://docs.unity3d.com/ScriptReference/Random.Range.html> [Accessed 25 October 2021].

- Technologies, U., 2021. Unity - Scripting API: SceneManagement.Scene.name. [online] Docs.unity3d.com. Available at: <https://docs.unity3d.com/ScriptReference/SceneManagement.Scene-name.html> [Accessed 22 November 2021].

- Technologies, U., 2021. Unity - Scripting API: TextureSheetAnimationModule. [online] Docs.unity3d.com. Available at:

<https://docs.unity3d.com/ScriptReference/ParticleSystem.TextureSheetAnimationModule.html> [Accessed 28 November 2021].

- Technologies, U., 2021. Unity - Scripting API: Transform.GetChild. [online] Docs.unity3d.com. Available at: <https://docs.unity3d.com/ScriptReference/Transform.GetChild.html> [Accessed 31 October 2021].

- unity, C., 2021. Change range of particle system start colours in unity. [online] Stack Overflow. Available at: <https://stackoverflow.com/questions/54736142/change-range-of-particle-system-start-colours-in-unity> [Accessed 27 November 2021].

- Unity Forum. 2021. 2D Sprite always behind background. [online] Available at: <https://forum.unity.com/threads/2d-sprite-always-behind-background.421276/> [Accessed 17 October 2021].

- Unity Forum. 2021. Change your app's target API level to at least 26. [online] Available at: <https://forum.unity.com/threads/change-your-apps-target-api-level-to-at-least-26.583162/> [Accessed 25 October 2021].

- Unity Forum. 2021. How can I change Key's Value in dictionary. [online] Available at: <https://forum.unity.com/threads/how-can-i-change-keys-value-in-dictionary.512622/> [Accessed 14 December 2021].

- Unity Forum. 2021. How to scale particle system?. [online] Available at: <https://forum.unity.com/threads/how-to-scale-particle-system.476616/> [Accessed 5 December 2021].

- Unity Forum. 2021. Search for objects in a list... [online] Available at: <https://forum.unity.com/threads/search-for-objects-in-a-list.467064/> [Accessed 4 November 2021].

- Youtube.com. 2021. 75% Of A Number. Working Out 75 Percent Of Something By Converting 75% Into A Fraction.. [online] Available at: <https://www.youtube.com/watch?v=WJFbbNuHFcE&ab_channel=maths3000> [Accessed 24 November 2021].

- Youtube.com. 2021. Clapping Sound Effects. [online] Available at: <https://www.youtube.com/watch?v=jDOrc8FmDy4&ab_channel=SFX> [Accessed 15 December 2021].

- Youtube.com. 2021. Drag and Drop for Mobile & Desktop in Unity. [online] Available at: <https://www.youtube.com/watch?v=FdxvTcHJiA8&ab_channel=DevLeonardo> [Accessed 14 October 2021].

- Youtube.com. 2021. Drag and Shoot || Angry Bird Shoot || Unity 2020 tutorial. [online] Available at: <https://www.youtube.com/watch?v=-E76wczofA8&ab_channel=VDOGAMES> [Accessed 2 December 2021].

- Youtube.com. 2021. How to EASILY Build to an Android Phone in Unity. [online] Available at: <https://www.youtube.com/watch?v=Nb62z3J4A_A> [Accessed 13 October 2021].

- Youtube.com. 2021. HOW TO MAKE 2D PARTICLE EFFECTS - UNITY TUTORIAL. [online] Available at: <https://www.youtube.com/watch?v=_z68_0oC_0o&ab_channel=Blackthornprod> [Accessed 27 November 2021].

- Youtube.com. 2021. How to make DAMAGE POPUPS in 5 Minutes! - Unity. [online] Available at: <https://www.youtube.com/watch?v=I2j6mQpCrWE&ab_channel=BimzyDev> [Accessed 17 October 2021].

- Youtube.com. 2021. How To Make Your Game Look The Same On All Mobile Screen Sizes - Unity Mobile Game Development. [online] Available at: <https://www.youtube.com/watch?v=KxwxZea0KAg&ab_channel=AwesomeTuts> [Accessed 3 December 2021].

- Youtube.com. 2021. How To Spawn Gameobject At Random Position Excluding Player Position And Pick It Up | Unity Tutorial. [online] Available at: <https://www.youtube.com/watch?v=IXDvl8aTM_M&ab_channel=AlexanderZotov> [Accessed 19 October 2021].

- Youtube.com. 2021. Making an IOS/Android game in UNITY - Beginner Tutorial - #1. [online] Available at: <https://www.youtube.com/watch?v=CGleQZVgdN4> [Accessed 13 October 2021].

- Youtube.com. 2021. SAVE & LOAD SYSTEM in Unity. [online] Available at: <https://www.youtube.com/watch?v=XOjd_qU2Ido&ab_channel=Brackeys> [Accessed 14 December 2021].

- Youtube.com. 2021. SCREEN SHAKE WITH NO CODE - EASY UNITY TUTORIAL. [online] Available at: <https://www.youtube.com/watch?v=N24MhfeoUpE&ab_channel=Blackthornprod> [Accessed 15 December 2021].

- Youtube.com. 2021. Shake Detecion for Mobile Devices in Unity (Android, iOS). [online] Available at:

<https://www.youtube.com/watch?v=CPGZZUjTMhU&t=207s&ab_channel=ResoCoder> [Accessed 14 October 2021].

- Youtube.com. 2021. TOUCH Controlled Drag & Shoot in Unity Mobile | 2D Game Dev Tutorial. [online] Available at: <https://www.youtube.com/watch?v=Q-_J9S6NaC0&ab_channel=MuddyWolf> [Accessed 28 November 2021].

- Youtube.com. 2021. Unity tutorial - Particle collision tutorial. [online] Available at: <https://www.youtube.com/watch?v=zMgSq3O8FE0&t=10s&ab_channel=GreenMaskGames> [Accessed 17 October 2021].

- Youtube.com. 2021. Unity 5 - Setting Up OnParticleCollision(). [online] Available at: <https://www.youtube.com/watch?v=LEhOiLW_API&ab_channel=Allen%27sUnityWorld> [Accessed 18 October 2021].

- Youtube.com. 2021. Unity Mobile - Swipe Controls - Unity 3D [Tutorial]|Tap touch|Detect Right/Left Swipe&Up/Down Swipe. [online] Available at: <https://www.youtube.com/watch?v=haCWs5cg3cY> [Accessed 22 October 2021]

Youtube.com. 2016. *New Super Mario Bros. DS - All Variety Minigames*. [online] Available at: <https://www.youtube.com/watch?v=h1FKzGk80Bo&t=290s> [Accessed 10 October 2021].

Youtube.com. 2016. *New Super Mario Bros. DS - All Minigames*. [online] Available at: <https://www.youtube.com/watch?v=I1JDGVRL0sk&t=17s> [Accessed 10 October 2021].

2020. *3D Conveyor Belts in Unity Tutorial*. [video] Available at: <https://www.youtube.com/watch?v=rQyUACEyAVw&t=353s> [Accessed 11 November 2021].

Design Havard References

Brusheezy. 2021. *Free Spray Photoshop Brushes 12*. [online] Available at: <https://www.brusheezy.com/brushes/58777-free-spray-photoshop-brushes-12> [Accessed 13 October 2021].

hive modern furniture. 2009. *alessi tonale small bowl 4 pack*. [online] Available at:

<https://hivemodern.com/pages/product6968/alessi-david-chipperfield-tonale-small-bowl-dc03> [Accessed 17 October 2021].

Baamboozle.com. 2021. *Find It Fast | Baamboozle*. [online] Available at: <https://www.baamboozle.com/game/108518> [Accessed 17 October 2021].

Tsurkan, T., 2021. *Vector Measuring Cup Kitchen Item Utensils Stock Vector (Royalty Free) 1712517343*. [online] Shutterstock. Available at: <https://www.shutterstock.com/image-vector/vector-measuring-cup-kitchen-item-utensils-1712517343> [Accessed 18 October 2021].

White, J., 2013. *A side view of a Pyrex measuring jug Stock Photo - Alamy*. [online] Alamy.com. Available at: <https://www.alamy.com/a-side-view-of-a-pyrex-measuring-jug-image60818289.html> [Accessed 18 October 2021].

Dreamstime. 2021. *Funny Milk Bottle*. [online] Available at: <https://www.dreamstime.com/stock-illustration-funny-milk-bottle-red-cap-small-eyes-big-smile-red-sticker-handwritten-lettering-isolated-yellow-image53855605> [Accessed 20 October 2021].

County Dairies. 2021. *Semi Skimmed 189ml*. [online] Available at: <https://www.countydairies.co.uk/products/milk/semi-skimmed-189ml> [Accessed 20 October 2021].

Argos. 2021. *Argos*. [online] Available at: <https://www.argos.co.uk/product/8423524> [Accessed 25 October 2021].

Label, Y., 2021. *Yeast 1 Pound Bag Red Label*. [online] Kitchenconservatory.com. Available at: <https://www.kitchenconservatory.com/Yeast-1-Pound-Bag-Red-Label-P9836.aspx> [Accessed 30 October 2021].

Walmart.com. 2021. *Red Star Active Dry Yeast 2 lb. bag*. [online] Available at: <https://www.walmart.com/ip/Red-Star-Active-Dry-Yeast-2-lb-bag/31081703> [Accessed 30 October 2021].

IKEA. 2021. *UPPLAGA white, Serving bowl - IKEA*. [online] Available at: <https://www.ikea.com/gb/en/p/upplaga-serving-bowl-white-10424715/> [Accessed 1 November 2021].

Amazon.co.uk. 2021. *IKEA RUNDLIG - Serving bowl, white bamboo, white - 30 cm*. [online] Available at: <https://www.amazon.co.uk/IKEA-RUNDLIG-Serving-white-bamboo/dp/B00GMMFZPA> [Accessed 1 November 2021].

Dreamstime. 2021. *Wheat flour in burlap sack bag on white background*. [online] Available at: <https://www.dreamstime.com/wheat-flour-burlap-sack-bag-white-background-

image124909798> [Accessed 1 November 2021].

Henrywatson.com. 2021. *Henry Watson's Potteries - Search results for: 'canister+replacement+seal'*. [online] Available at: <https://www.henrywatson.com/catalogsearch/result/?q=canister%2Breplacement%2Bseal> [Accessed 1 November 2021].

Brusheezy. 2021. *Flour Brushes*. [online] Available at: <https://www.brusheezy.com/brushes/59502-flour-brushes> [Accessed 1 November 2021].

Todt, G., 2021. *Leckerli One Font · 1001 Fonts*. [online] 1001 Fonts. Available at: <https://www.1001fonts.com/leckerli-one-font.html> [Accessed 29 November 2021].

Etsy.com. 2021. *Dozen Donut Favor Boxes Single Donut Box Donut Shop Party | Etsy UK*. [online] Available at: <https://www.etsy.com/uk/listing/511429879/dozen-donut-favor-boxes-single-donut-box> [Accessed 6 December 2021].

Nisbets.co.uk. 2021. *Heavy Duty Strainer 14" - C832 - Buy Online at Nisbets*. [online] Available at: <https://www.nisbets.co.uk/heavy-duty-strainer-14in/c832?vatToggle=incvat&plaid=1&cm_mmc=PLA-_-14736808781-_-129030693073-_-C832&cm_mmca1=go_14736808781_129030693073_547522581565_pla-1451228316394_c_> [Accessed 8 December 2021].

Handler, R., 2021. *These Fluffy Glazed Doughnuts Taste Just Like Krispy Kremes*. [online] Delish. Available at: <https://www.delish.com/uk/cooking/recipes/a30165545/how-to-make-donuts-at-home/> [Accessed 8 December 2021].

Kela.de. 2021. *Rolling pin Maribor | Kela Online Shop*. [online] Available at: <https://www.kela.de/en/rolling-pin-maribor/item-1-17300.html> [Accessed 8 December 2021].

IKEA. 2021. *IKEA 365+ HJÄLTE stainless steel, black, Balloon whisk - IKEA*. [online] Available at: <https://www.ikea.com/gb/en/p/ikea-365-hjaelte-balloon-whisk-stainless-steel-black-30158678/> [Accessed 8 December 2021].

Robinson, P., 2019. *An eggcellent design - creation.com*. [online] Creation.com. Available at: <https://creation.com/eggshell-design> [Accessed 8 December 2021].

John Lewis. 2021. *John Lewis & Partners Wooden Spoon, FSC-Certified (Beech Wood), Natural*. [online] Available at: <https://www.johnlewis.com/anyday-john-lewis-partners-wooden-spoon-fsc-certified-beech-wood-natural/p230594037> [Accessed 8 December 2021].

Amazon.co.uk. 2021. *Mrs. Anderson's Baking Donut Cutter with Handle, Stainless Steel, 3-Inches x 3-Inches*. [online] Available at: <https://www.amazon.co.uk/Harold-Andersons-

Baking-Stainless-Cutter/dp/B008ZUB8MQ> [Accessed 8 December 2021].

AliExpress. 2021. *10.04£ |Stainless steel mesh oil spoon scoop dipping large spoon noodles colander net kitchen accessories|large scoop colander|kitchen colanderlarge colander - AliExpress*. [online] Available at: <https://www.aliexpress.com/item/32709681046.html> [Accessed 9 December 2021].

Amazon. 2021. *JB Prince Gray Kunz Perforated Spoon - Small*. [online] Available at: <https://www.amazon.co.uk/JB-Prince-Gray-Perforated-Spoon/dp/B0187ZEI52> [Accessed 9 December 2021].

IKEA. 2021. *TILLÄMPAD stainless steel, Tongs - IKEA*. [online] Available at: <https://www.ikea.com/gb/en/p/tillaempad-tongs-stainless-steel-20452114/> [Accessed 12 December 2021].

Brusheezy. 2021. *24 Clouds*. [online] Available at: <https://www.brusheezy.com/brushes/2187-24-clouds> [Accessed 12 December 2021].

Now, K., 2014. *Brossard,Quebec, Canada - August 27, 2014: Krispy Kreme Doughnuts...*. [online] iStock. Available at: <https://www.istockphoto.com/photo/krispy-kreme-doughnuts-store-gm510105313-46797526> [Accessed 12 December 2021].

Balance, H., 2016. *How To Check Your Krispy Kreme Gift Card Balance*. [online] www.MyGiftCardSite.com. Available at: <https://www.mygiftcardsite.us.com/how-to-check-your-krispy-kreme-gift-card-balance/> [Accessed 12 December 2021].

Tripadvisor. 2020. *Photo: Store front view*. [online] Available at: <https://en.tripadvisor.com.hk/LocationPhotoDirectLink-g55055-d1918483-i298046118-Sal_s_Family_Pizza-Franklin_Tennessee.html> [Accessed 12 December 2021].

Vectors, R. and Vectors, F., 2021. *Shop front or store view vector image on VectorStock*. [online] VectorStock. Available at: <https://www.vectorstock.com/royalty-free-vector/shop-front-or-store-view-vector-11865271> [Accessed 12 December 2021].

Maisonsdumonde.com. 2021. *Dining table L 100 cm | Maisons du Monde*. [online] Available at: <https://www.maisonsdumonde.com/UK/en/p/dining-table-l-100-cm-circle-165906.htm> [Accessed 14 December 2021].