



DATABASE DELIVERY 2

POL RECASENS

LLUIS MOREU

CHRISTIAN MARTÍNEZ

MARC GÁLVEZ





TABLE OF CONTENTS

- 
- 01 DATA STRUCTURE
 - 02 WRITE & READ CSV
 - 03 SHADER HEAT MAP
 - 04 UI MINIMAP



C#



DATA STRUCTURE

```
public class PlayerEventTrack
{
    static public List<Dictionary<string, object>> EventList = new List<Dictionary<string, object>>();
    static public List<Dictionary<string, object>> EventData = new List<Dictionary<string, object>>();
    static public List<Dictionary<string, object>> PositionData = new List<Dictionary<string, object>>();
}
```



WRITE CSV

```
// You can add up the values in as many cells as you want.
for (int i = 0; i < PlayerEventTrack.EventList.Count; i++)
{
    if (PlayerEventTrack.EventList[i].TryGetValue("Type", out object value))
    {
        if (value.ToString() != "Movement")

            string filePath = getPath("Position_data");

            StreamWriter if (value.ToString() == "Movement") eText(filePath);
            outStream.WriteLine(sb);
            outStream.Close();
        }
        else
        {
            Debug.Log("No Type Key found");
        }
    }
}
```



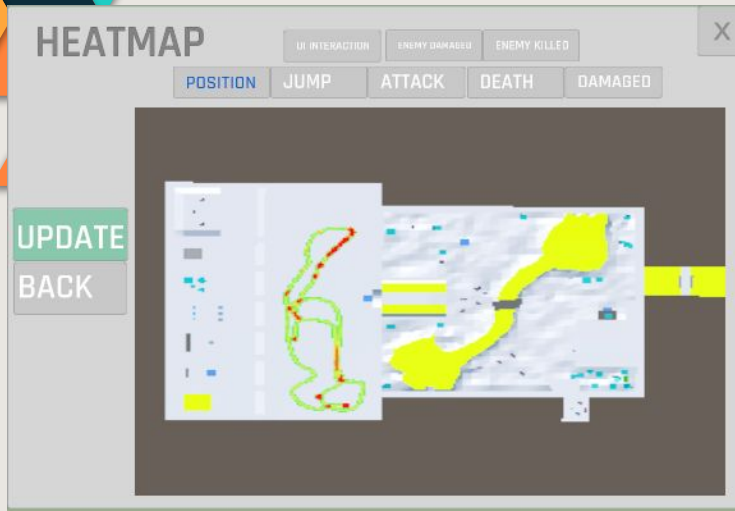


```
for (var i = 1; i < lines.Length; i++)
{
    var values = Regex.Split(lines[i], SPLIT_RE);
    if (values.Length == 0 || values[0] == "") continue;

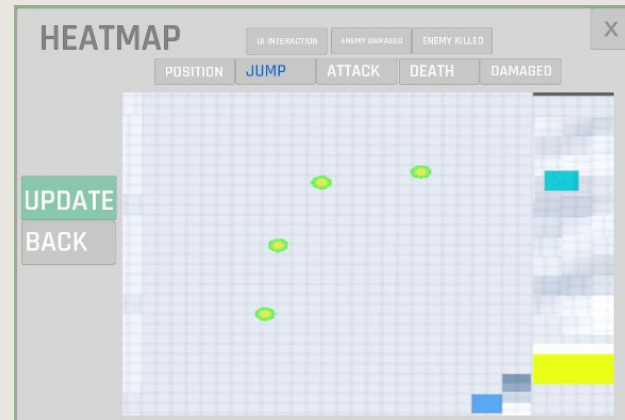
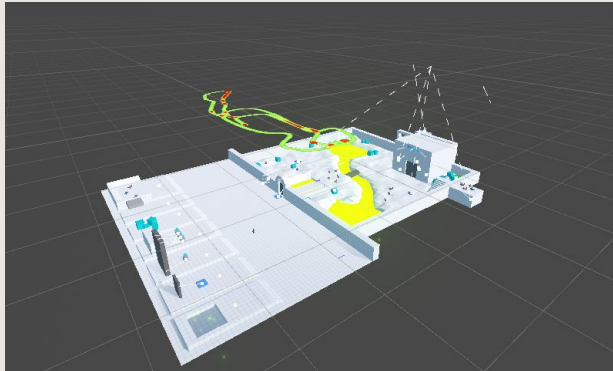
    var entry = new Dictionary<string, object>();
    for (var j = 0; j < header.Length && j < values.Length; j++)
    {
        string value = values[j];
        value = value.TrimStart(TRIM_CHARS).TrimEnd(TRIM_CHARS).Replace("\\", "").Replace(".", ",");
        object finalvalue = value;
        int n;
        float f;
        if (float.TryParse(value, out f))
        {
            finalvalue = f;
        }
        else if (int.TryParse(value, out n))
        {
            finalvalue = n;
        }
        entry[header[j]] = finalvalue;
    }
    list.Add(entry);
}
return list;
```

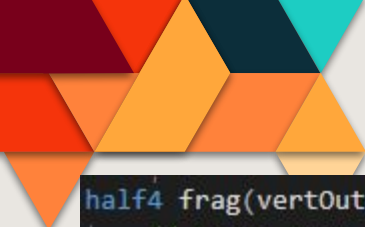
READ CSV





SHADER HEATMAP





```
half4 frag(vertOutput output) : COLOR{
    // Loops over all the points
    half h = 0;
    for (int i = 0; i < _Points_Length; i++)
    {
        // Calculates the contribution of each point
        half di = distance(output.worldPos, _Points[i].xyz);


        half ri = _Properties[i].x;
        half hi = 1 - saturate(di / ri);

        h += hi * _Properties[i].y;
    }

    // Converts (0-1) according to the heat texture
    h = saturate(h);
    half4 color = tex2D(_HeatTex, fixed2(h, 0.5));
    return color;
}
```

SHADER HEATMAP

```
uniform float4 _Points[1000];    // (x, y, z) = position
uniform float4 _Properties[1000]; // x = radius, y = intensity
```



SHADER HEATMAP

```
public void GenerateMap(HeatmapType type)
{
    var dic = PlayerEventTrack.PositionData;
    int d = 0;
    switch (type)
    {
        case HeatmapType.Position:
            count = dic.Count;

            positions = new Vector4[count];
            properties = new Vector4[count];

            for (int i = 0; i < count; ++i)
            {
                positions[i] = new Vector4((float)dic[i]["PositionX"], 0f, (float)dic[i]["PositionZ"], 0);
                properties[i] = new Vector4(1.0f, 0.5f, 0, 0);
            }
            break;
    }
}
```



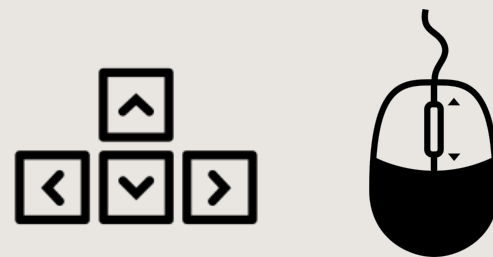
```
case HeatmapType.Death:
    dic = PlayerEventTrack.EventData;

    count = dic.Count;

    positions = new Vector4[count];
    properties = new Vector4[count];

    for (int i = 0; i < count; ++i)
    {
        if ((string)dic[i]["Type"] == "Dead")
        {
            positions[d] = new Vector4((float)dic[i]["PositionX"], 0f, (float)dic[i]["PositionZ"], 0);
            properties[d++] = new Vector4(1.0f, 1.0f, 0, 0);
        }
    }
    count = d;
    break;
```

```
material.SetInt("_Points_Length", count);
material.SetVectorArray("_Points", positions);
material.SetVectorArray("_Properties", properties);
```

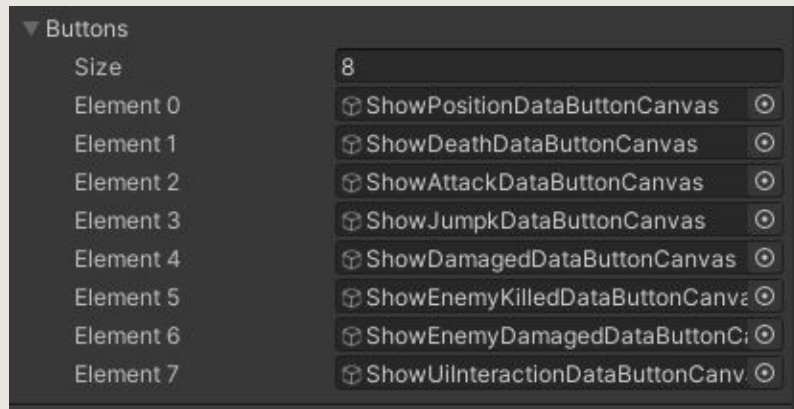


MINIMAP

- Raw Image Canvas
- Render Texture
- Camera



UI MINIMAP



HEATMAP

UI INTERACTION

ENEMY DAMAGED

ENEMY KILLED

X

POSITION

JUMP

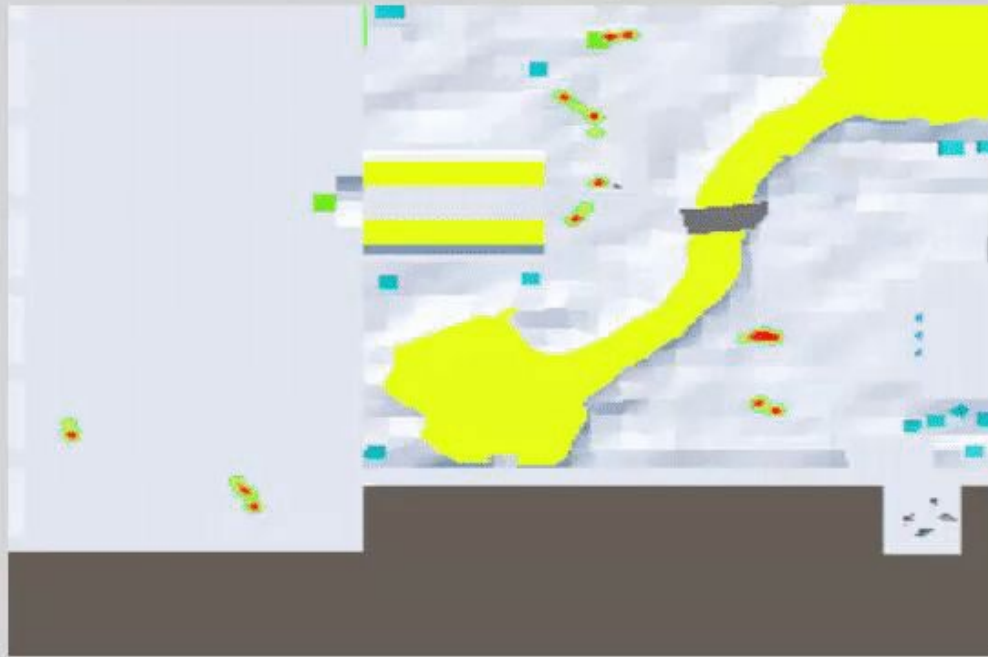
ATTACK

DEATH

DAMAGED

UPDATE

BACK





THANKS!

Reddit User Keyboard Heat Map

