LuckyVR Unity Take-home Assignment

Instructions

1. Review all parts of the test before beginning. If you have questions, don't hesitate to ask.
2. Please implement the project in C# using Unity 2019.4.22.f1. Set Unity to target a Desktop PC.
3. When you feel you are finished, package your Unity project in a ZIP or 7Z file and email it back to us.
4. Please attach response to optimization skill-testing question as a separate PDF to the same email.

Project: Betting on Colors

We'd like you to design and implement a simple game where players can bet on an object being either green or red. Once a bet has been placed, the color is revealed and the player either loses their bet, or wins an equivalent amount.

The player begins with 100 chips split between 10 stacks and can bet in 10 chip increments. Each stack of chips should be a different color. If the player runs out of chips, they are automatically topped up with another 100 chips.

The design is open-ended, but the app and individual chips must be 3D with UI buttons or key-presses for gameplay flow.

The main focus should be on minimizing both CPU + GPU frame-times through optimizations in scripting, data structures used, and prefab + material setup.

You can use primitives in place of proper meshes.

Networking

The game must be networked with two clients connecting to each-other and seeing their chips + bets in real-time. You may use whatever networking solution you prefer.

Optimization Skill Question

1) Explain all the ways you would optimize the following Update function:

```
private void Update()
{
    GameObject[] sceneObjects = GameObject.FindGameObjectsWithTag("scene");

    foreach (GameObject i in sceneObjects)
    {
        if (Vector3.Distance(i.transform.position, new Vector3(0,0,0)) > 5)
        {
            i.transform.position = new Vector3(0, 0, 0);
        }
        else
        {
            i.transform.position += new Vector3(1, 1, 1);
        }

    }

}
```