Hegyi Andrea
Informatică în limba engleza, an II, sg 4
Universitatea de Vest din Timișoara
Prof. Coord. Liviu Octavian Mafteiu-Scai

Final Report

Programming for mobile devices

Decision Wheel App

1. Abstract

In today's society we have a whole lot of choices to make at each step in our lives. Let it be big decisions or tiny decisions, like what to eat or where to go out.

People prefer to make these decisions as easily as possible, that's why a Tiny Wheel Decision App would be useful in different situations.

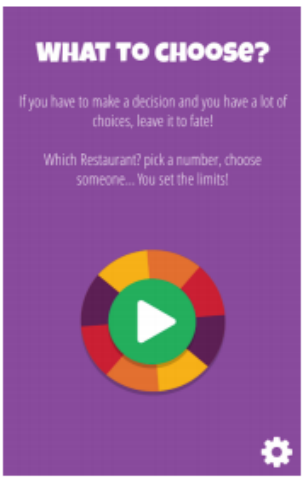
2. Motivation and users

I struggle with making even the smallest decisions and it can be annoying for both myself and my friends. I have never actually used a similar app, but I've heard about them recently and I decided to make one of my own.

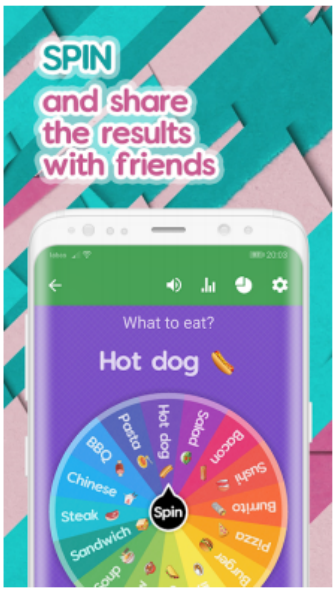
This kind of app could be used by a kid who doesn't know what game to play, by a teenager wondering where they should go out for the weekend, by a gym enthusiast who can't decide what exercises they should do or by a middle aged person confused about what to cook tonight.

3. Similar Applications

- Decision Roulette



- Spin The Wheel - Random Picker



- Daily Decision Wheel

4. My Contribution

My app has nothing more special than these apps shown above. It has a simple interface and there was no need for a database or an user authentication. My contribution was the frontend part, I decided how it should look, and the small bit of script I used.

5. User’s Manual

My application has a simple interface, the first page having a main menu where the user can go to the wheel by pressing the start button.

After pressing the START button, the application takes the user to the wheel, which is split in 8 and spins whenever the screen is touched. On this screen we have an EXIT button which takes us back to the Main Menu and an Input Field where the user can write which number he wants to have more chances of being chosen (still working).

6. App Structure

```
using System.Collections;
using UnityEngine;
using UnityEngine.UI;

Unity Script | 0 references
public class spin : MonoBehaviour
{

    private int randomValue;
    private float timeInterval;
    private bool coroutineAllowed;
    private int finalAngle;
    public InputField iField;

    public Text winText;
    public Text nText;

    // Start is called before the first frame update
    Unity Message | 0 references
    private void Start()
    {
        coroutineAllowed = true;
        iField = gameObject.GetComponent<InputField>();
    }

    // Update is called once per frame
    Unity Message | 0 references
    private void Update()
    {
        if (Input.GetMouseButtonDown(0) && coroutineAllowed)
            StartCoroutine(Spin());
    }

    1 reference
    private IEnumerator Spin()
    {
        coroutineAllowed = false;
        randomValue = Random.Range(20, 30);
        timeInterval = 0.1f;
```

```
        int rand_num = Random.Range(1, 10);

        if(rand_num >= 1 && rand_num <= 8)
        {
            switch (3)
            {
                case 1:
                    winText.text = "You got 1";
                    break;
                case 2:
                    winText.text = "You got 2";
                    break;
                case 3:
                    winText.text = "You got 3";
                    break;
                case 4:
                    winText.text = "You got 4";
                    break;
                case 5:
                    winText.text = "You got 5";
                    break;
                case 6:
                    winText.text = "You got 6";
                    break;
                case 7:
                    winText.text = "You got 7";
                    break;
                case 8:
                    winText.text = "You got 8";
                    break;
            }
        }
```

```
        for (int i = 0; i < randomValue; i++)
        {
            transform.Rotate(0, 0, 22.5f);
            if (i > Mathf.RoundToInt(randomValue * 0.5f))
                timeInterval = 0.2f;
            if (i > Mathf.RoundToInt(randomValue * 0.85f))
                timeInterval = 0.4f;
            yield return new WaitForSeconds(timeInterval);
        }

        if (Mathf.RoundToInt(transform.eulerAngles.z) % 45 != 0)
            transform.Rotate(0, 0, 22.5f);

        finalAngle = Mathf.RoundToInt(transform.eulerAngles.z);

        switch (finalAngle)
        {
            case 0:
                winText.text = "You got 1";
                break;
            case 45:
                winText.text = "You got 2";
                break;
            case 90:
                winText.text = "You got 3";
                break;
            case 135:
                winText.text = "You got 4";
                break;
            case 180:
                winText.text = "You got 5";
                break;
            case 225:
                winText.text = "You got 6";
                break;
            case 270:
                winText.text = "You got 7";
```

```
                break;
            case 225:
                winText.text = "You got 6";
                break;
            case 270:
                winText.text = "You got 7";
                break;
            case 315:
                winText.text = "You got 8";
                break;
        }

        coroutineAllowed = true;
    }
}
```