Fluffy Bird Postmortem

**\* Player**

Es el encargado de implementar las interacciones de colisión con todos los otros objetos. Se decidió realizar el movimiento mediante rigidbody.AddForce cuando el jugador hace click izquierdo. La velocidad se limita mediante una velocidad máxima que aumenta de a poco a medida que el score va creciendo.

Se utilizaron tags para todos los objetos de manera tal que los eventos de OnTrigger… puedan reconocer más fácilmente cual es el collider que interactúa con el pájarito.

El jugador pierde cuando el pájaro toca el suelo, sale por arriba de la cámara o choca contra algún pipe.

La estrella proveerá inmunidad solo contra pipes, pero no contra el suelo o cuando el jugador va muy arriba.

Se agregaron animaciones para la estrella y cuando el jugador pierde

**\* Obstacles**

Los obstáculos se generan desde una corutina del GameManager que varía el tiempo según el score actual del jugador. A mayor score, más rápido aparecen obstáculos hasta un cierto máximo.

Los obstáculos tienen una altura variable y tambien tienen un 7% de posibilidad de tener una estrella cerca del gap entre el pipe de arriba y el de abajo.

Se implementó un pool de pipes que reutiliza los pipes inactivos y de esa manera se evita instanciar demasiados. Los pipes se inactivan una vez que el jugador los deja atrás

Cada vez que aparece un nuevo obstaculo se randomizan todas las variables nuevamente si está siendo reutilizado

**\* Ground**

Se utilizan dos objetos de suelo que van intercambiando posiciones de acuerdo a la posición del jugador. A medida que el pájaro avanza, cada pieza del suelo se acomoda donde no se vea nada desde la cámara.

**\* Camera & UI**

La cámara simplemente se mueve con la posición x del jugador

La UI se encarga de mostrar el score actual y el boton de restart en caso de que el jugador pierda. Para el boton, se utilizó el componente standard de Button de UI de Unity

**\* GameManager**

Se encarga de modificar el score, el flujo de instanciación de obstáculos y coordinar la serie de pasos a seguir durante el game over y el restart

**\* Consideraciones**

Se decidió que la interacción entre los elementos más importantes se realizaría mediante MonoSingletons. En un juego más grande y con más elementos me inclinaría por utilizar referencias y asignarlas al GameManager, lo que sería más ordenado, pero para este caso en particular de esta manera me pareció adecuado.