

DOCUMENTO TÉCNICO DE PROGRAMACIÓN DEL JUEGO

CASO DE ESTUDIO: Floppy Bird

CONTEXTO:

Vamos a hacer una recreación del clásico juego Floppy Bird; el cuál fue un juego desarrollado para móviles en Hanói por el desarrollador vietnamita Nguyen Hà Đông y publicado por .GEARS Studios, Este juego consiste en que el jugador tocara la pantalla para elevar (floppy) al pájaro para esquivar las tuberías que se ponían en su camino y evitando así chocar y perder.

PRIMITIVAS:

Comenzamos primero explorando cómo podemos movernos para evitar chocar con los obstáculos del camino. Entonces programamos primero las interacciones básicas (sus mecánicas) con primitivas.

FÍSICAS:

Investigamos la lógica para determinar la dirección y gravedad que tendrá el pájaro, apareciendo los conceptos de la física de forma natural.

ENTORNO DE TRABAJO:

Aprendemos como es el entorno de desarrollo que en este caso es Processing, averiguando así el cómo funciona y su ciclo de vida.

PRINCIPIOS FUNDAMENTALES:

Abstraemos los principios de programación subyacente como las estructuras de control selectivas e iterativas.

FASE DE ANALISIS ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,

HISTORIA DE USUARIO DEL PROYECTO

HISTORIA 1: Movimiento de jugador

Como jugador yo quiero evitar que el pájaro choque o deje de volar. Así que con la tecla ESPACIO lo mantengo flotando.

HISTORIA 2: Aparición constante de postes

Como jugador yo quiero que aparezcan constantemente diferentes postes (en tamaño) para esquivarlos.

HISTORIA 3: Choque del pájaro con el poste

Como jugador yo quiero saber que al momento de chocar con un poste pierdo

DOCUMENTO TÉCNICO DE PROGRAMACIÓN DEL JUEGO

HISTORIA 4: Cronómetro

Como jugador yo quiero saber cuanto tiempo que estado sobreviviendo sin chocar y cuánto me falta para ganar

HISTORIA 5: Reiniciar

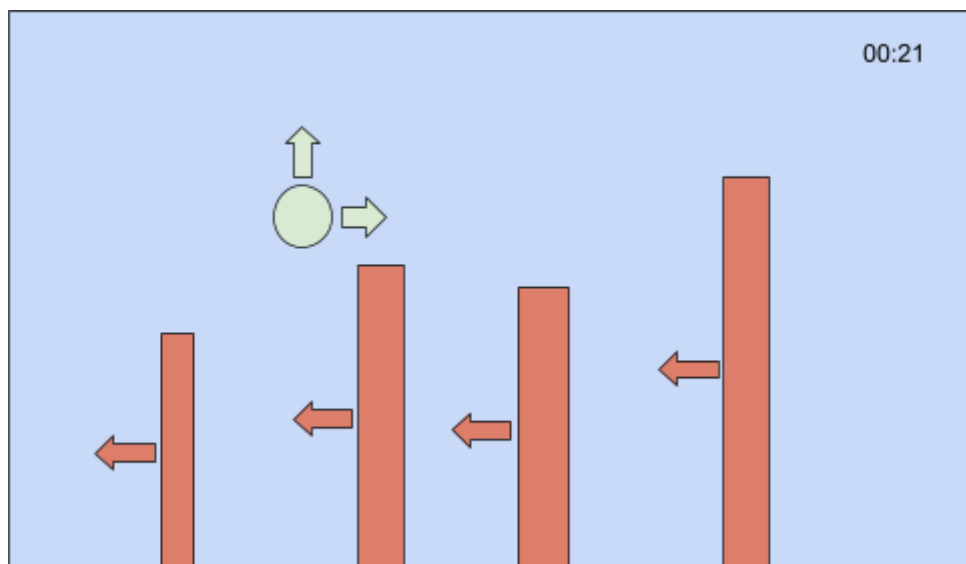
Como jugador yo quiero poder reiniciar con el enter cuando termine perdiendo

EPS (Entradas, Procesos y Salidas),,,,,,,

HISTORIA	ENTRADAS	PROCESOS	SALIDAS
1	Tecla ESPACIO	Detecta la entrada del teclado. Salta para arriba	Una nueva posicion del pajaró
2	Posición del poste	Detectar el poste y esquivar con el pájaro	Aparición automática
3	Colisión entre el pájaro y el poste	Detecta la colisión y el pájaro muere	Se pierde
4	Al iniciar el juego	Se detecta el inicio del juego y va calculando el tiempo	Los segundos que vas sobreviviendo
5	La tecla ENTER	Se detecta la tecla y se reinicia	Se resetea el juego, volviendo al tiempo inicial

BOCETOS VISUALES DEL JUEGO,,,,,,

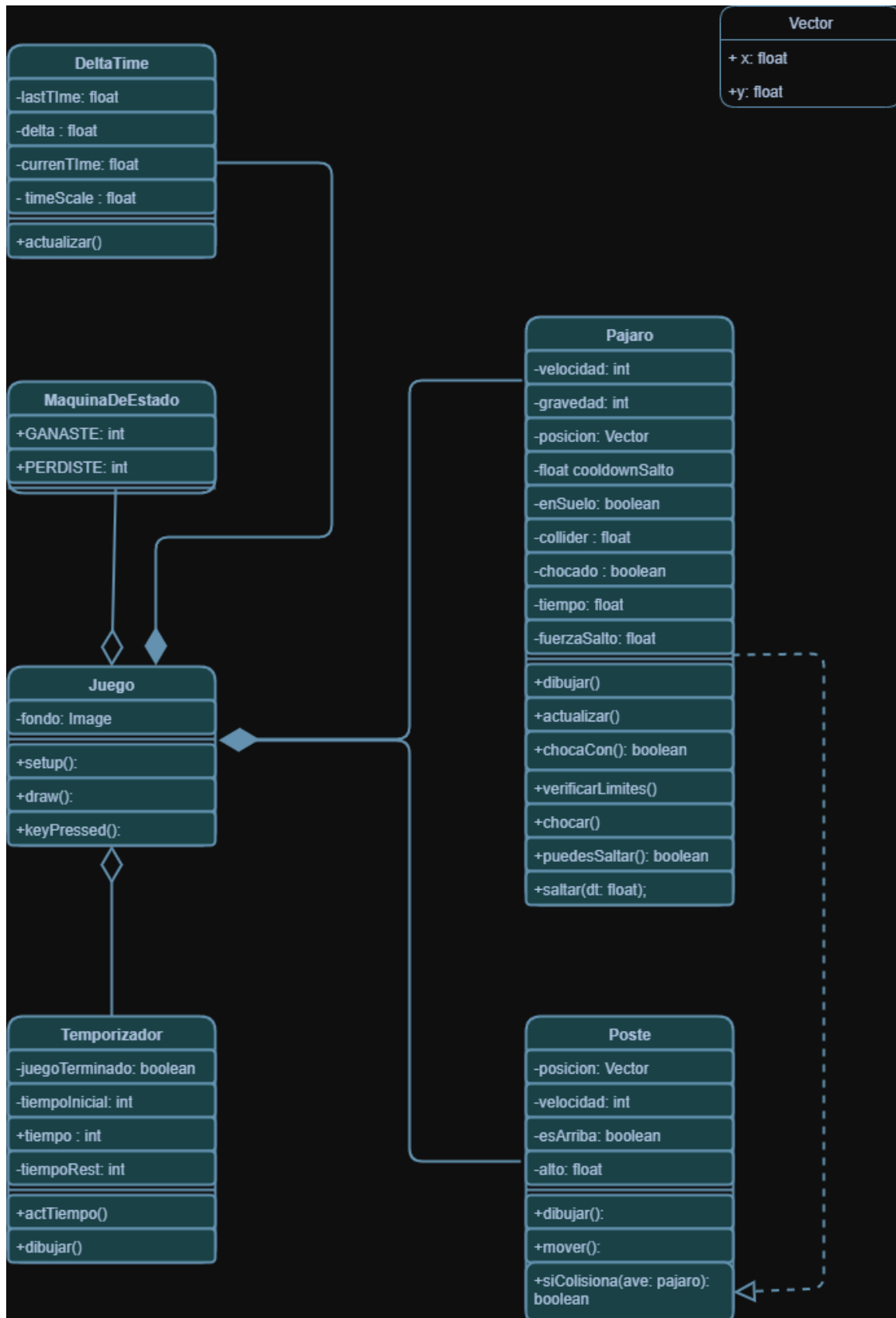
. PANTALLA DE INICIO



Como jugador yo quiero evitar que el pájaro choque o deje de volar. Así que con la tecla ESPACIO lo mantengo flotando.

DOCUMENTO TÉCNICO DE PROGRAMACIÓN DEL JUEGO

DIAGRAMA DE CLASES UML:



DOCUMENTO TÉCNICO DE PROGRAMACIÓN DEL JUEGO

FASE DE IMPLEMENTACIÓN ,,,,,,,,,,,,,,

AREAS

- ☒ **T1**-CREAR LAS CLASES POSTE Y PAJARO CON SUS ATRIBUTOS Y MÉTODOS
- ☒ **T2**-DIBUJAR AL PAJARO CON PRIMITIVAS Y SE MUEVA HORIZONTALMENTE
- ☒ **T3**-APLICARLE GRAVEDAD AL PAJARO
- ☒ **T4**-QUE CUANDO APRETE EL ESPACIO SALTE
- ☒ **T5**-DIBUJAR LOS POSTES(primitivas) Y SE MUEVAN
- ☒ **T6**-GENERAR UN ARRAY DE POSTES
- ☒ **T7**-HACER QUE TENGAN DIVERSAS ALTURAS UNOS DE OTROS
- ☒ **T9**-QUE EL PAJARO CHOQUE CON LOS POSTE
- ☒ **T10**-CUANDO CHOQUE SE MUESTRE EL ESTADO PERDIENDO
- ☒ **T11**-FINALIZADO EL TIEMPO MUESTRE EL ESTADO GANASTE
- ☒ **T12**-QUE NO SE SALGA DE LA PANTALLA
- ☐ **T13** - Cada que pasa el tiempo se acelera la velocidad de los postes

CRITERIOS DE ACEPTACIÓN

- ☒ EL PAJARO DEBE MOVERSE Y SALTAR repetidas veces
- ☒ LOS POSTE DEBEN MOSTRARSE Y MOVERSE
- ☒ LOS POSTES DEBEN TENER DIVERSAS ALTURAS
- ☒ DETECTAR LA COLISIÓN
- ☒ El Temporizador debe mostrarse correctamente
- ☒ POSTCONDICIÓN: QUE NO SE SALGA DE LA PANTALLA
- ☒ POSTCONDICIÓN: Que acelere segun el tiempo