

## TP 05

**Cachuflito Games Inc.** sobrevive! Todavía no se ha publicado nuestro juego anterior de FPS ("por cuestiones de timing internacional: está pronto a salir el Half Life 3" dijo el CEO) pero la empresa ya está dedicando todo su esfuerzo en una nueva idea revolucionaria del CEO. Su nombre:

### **WACA WACA!** **Wacky Aviation Combat Avant-garde X2!**

El CEO no para de hablar de que bien que la pasaba jugando al Ace Combat en la play en su casa de pibe y considera que es buen momento que le regalemos al mundo otra joya Cachuflitense.

Para esto nos ha proveído un mail con el siguiente link y las escuetas palabras "voy a estar de vacaciones"

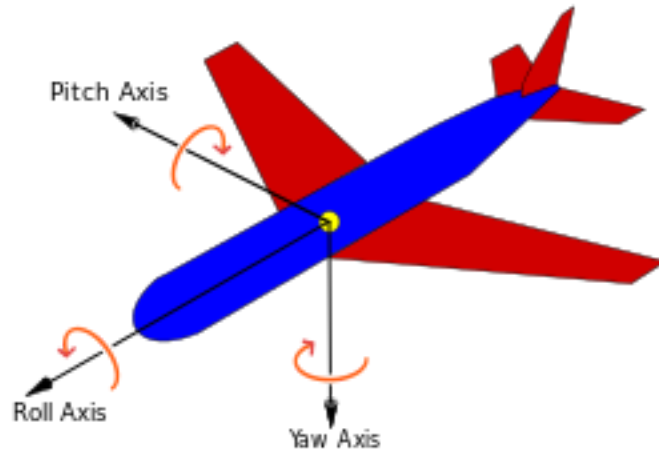
<https://www.youtube.com/watch?v=qAz5Z1LneIM>

El Game Designer, en base al extenso mail del CEO, nos armó el siguiente GDD previo a tomarse vacaciones también:

#### **GDD** (Game Design Document)

Para que el juego sea publicable (**TP aprobado**) y de esta manera generar 4 millones de dólares (**4 puntos del TP**) el juego deberá tener:

- 3 estados para ser minimamente publicable:
  - Pantalla Inicial: Logo del juego, logo de la empresa y botón para empezar el juego.
  - Pantalla de Juego: el juego propiamente dicho.
  - Pantalla Final: esta pantalla mostrará datos de puntaje e información pertinente dependiendo de cada feature implementado (cantidad de enemigos destruidos, puntaje, etc).
- El personaje se mueve como se aprecia en el video suministrado por el CEO usando el sistema de Pitch, Yaw y Roll.



Pitch y Yaw serán controlados por el input vertical y horizontal del mouse y Roll será controlado por las teclas A y D y flechas de izquierda y derecha.

La aceleración y desaceleración se controlarán con las teclas W y S y flechas arriba y abajo.

Debe haber un imagen de UI que simula una mira.

La cámara se encuentra dentro del cockpit del avión.

Si el avión se estrola contra el piso debe explotar.

- El juego debe tener 3 niveles. El nivel termina cuando destruimos a todos los enemigos. Cada uno de los niveles debe contener un Terrain diferente (ví que Unity viene con un lindo sistema de Agua en el Standard Assets. Estaría piola investigarlo).
- Hay enemigos en el nivel:  
Enemigo 01: **Aviones**. Estos aviones van en dirección al player y pueden disparar. Internamente (código) utilizan un FSM (dijo el CTO).
- El avión tiene un arma principal:  
Arma 01: **Metralleta**. La metralleta debe funcionar como se puede apreciar en el video provisto por el CEO mediante el botón izquierdo del mouse.
- En caso de usarse recursos externos (modelos 3d, texturas) se debe especificar su fuente y licencia mediante un archivo txt o dentro del juego mediante una UI.
- **Ingame UI**: El juego presenta un HUD al estar en la cámara interna del avión. Esta representará: Altura de la nave, combustible, rotación con respecto al suelo, municiones de las armas, etc (el CEO dice que cualquier cosa que sume va)

A su vez el game designer cree que si agregamos estos features el juego venderá 6 millones mas de dolares **(1 punto de TP cada item)**

- Arma 02: **Homing Missile**. Mi avión puede disparar misiles teledirigidos. Debo "lockear" a un avión enemigo antes de poder disparar el misil.

**Las armas ahora son intercambiables con los números del teclado.**

- **Sistema de Highscores**: Hay un sistema de puntaje interno del juego que no se le muestra al player. Cuando el player muere puede acceder a los highscores si su

puntaje es el suficiente. La posibilidad de ver los Highscores debe agregarse a la pantalla principal del juego (Main Menu).

- **Rear View Mirrors:** El avión tiene espejos retrovisores dentro del cockpit.
- **Sistema de Partículas:** Hay uso de sistemas de partículas en los lugares esperados. Misiles, explosiones, etc.
- **Diferentes Cámaras:** El juego debe presentar diferentes Cámaras para ver nuestro avión. Estas no deben estar pegadas al avión.  
Las diferentes cámaras son intercambiables con la rueda del mouse.
- **Enemigo 02:** Algo en el piso que dispara pew pew pew o algo así.

**Nota:** Como puede observarse en el GDD nuestro avión puede quedarse sin municiones, razón por la cual sería imposible terminar el nivel.

Al enviarle un mail al Designer sobre qué debemos hacer con esto su respuesta fue:

"No se, che. Inventen algo."

Además de todo esto es **sumamente importante** lo que dijo el CTO

El CTO, que no cobra desde hace 3 mese, gritó desde su cubículo:

"Link git, IWALP, build y sources..."

"... que compile y ejecute sin errores!"

Para luego gruñir:

"No debe verse el cursor del mouse."

"Todos los valores que sean lógicos de modificar desde el Inspector de Uinty deberán serlo (vida de mi avión, daño de los misiles y las balas, puntajes, etc). Por si el Game Designer quiere balancear alguna cosa."

"Sigan los estándares de C#!"

"Los assets que no se usen en el juego no deben estar en el repo!"

Por otro lado, se escucha internamente en los pasillos de la empresa que si el juego no está para el **Martes 30 de Abril** la empresa cierra definitivamente.