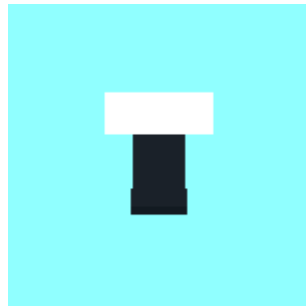


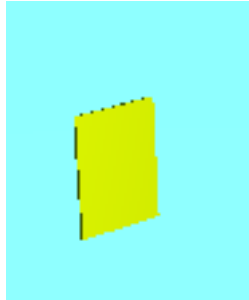
El Barquito Vago



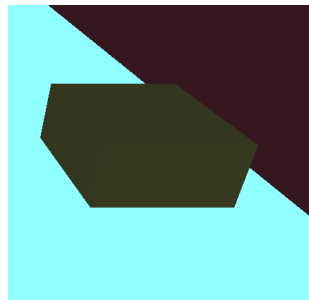
Un proyecto de SFV de
Enrique Juan Gamboa

Temática

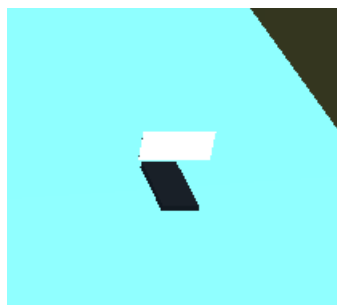
El Barquito Vago es un pequeño juego físico cuyo objetivo es llevar a un pequeño barco, a base de manipular el viento que le afecta, hasta una gran moneda de oro al final del nivel.



El reto llega a la hora de controlar el barco para evitar las rocas, ya que la inercia del mismo unido a lo que tarda el viento en cambiar hace que sea imprescindible planificar la ruta y controlar las velocidades a las que se lleva el barquito.

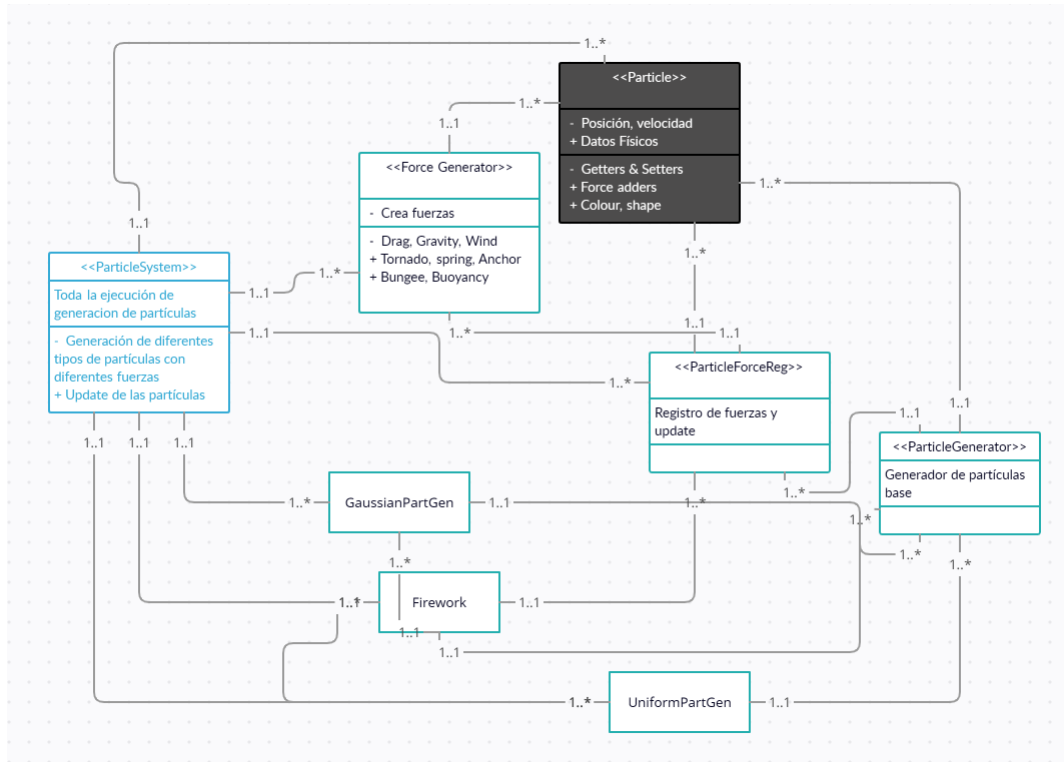


El nombre de *El Barquito Vago* realmente viene de mi abuelo, que llamaba “barcas vagas” a las que se movían sin necesidad de remar. Como en el juego ya ni controlas al barco sino que literalmente tienes que empujarlo hacia la meta hace que el nombre encaje perfectamente.

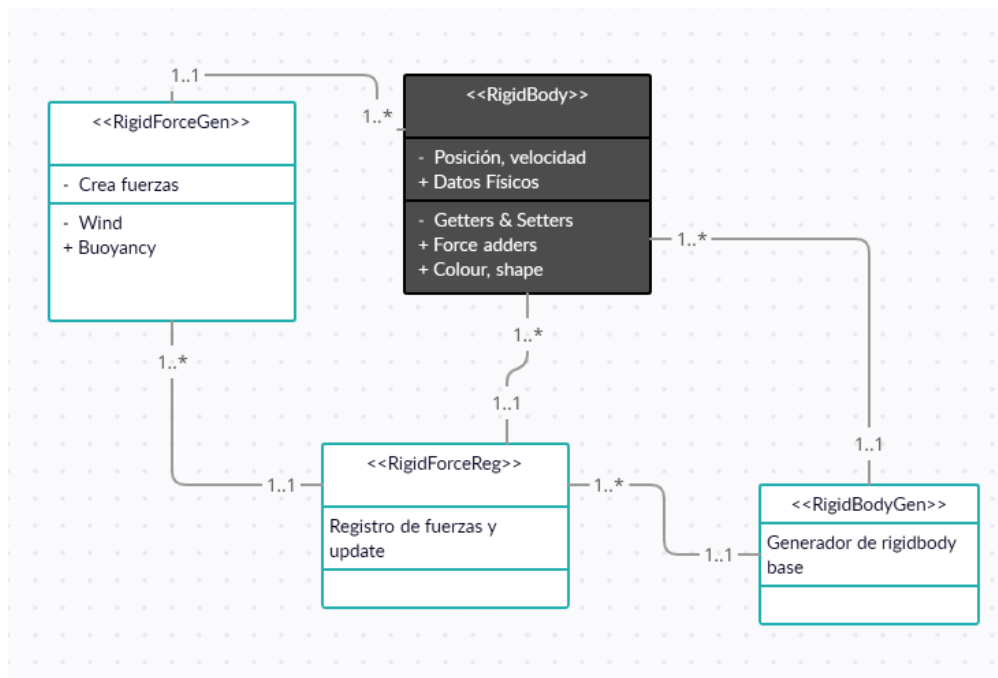


UML

Partículas:



RigidBody:



Ecuaciones usadas:

Gravedad: Se ha utilizado la gravedad tanto en los fuegos artificiales de ganar/perder la partida como en la fórmula de la flotación (de la cual es una parte esencial). Su fórmula es de una aceleración lineal en sentido negativo del eje y : $g = 9,8m/s^2$, en vector $Vg = (0, -9.8, 0)$.

Viento: el viento, clave para el control del juego, usa una fórmula incremental para ir subiendo más la velocidad del objeto afectado cuanto más está a su exposición. La fórmula usada es:

$$viento(t) = _k1 * (fuerza(t) * dirección(t) - vel(t-1)) * |fuerza(t) * dirección(t) - vel(t-1)| * (fuerza(t) * dirección(t) - vel(t-1))$$

Flotación: al tratarse de un barco, he usado una fórmula de flotación bastante compleja, y he añadido un parámetro además de la densidad del líquido, nivel del líquido, volumen y profundidad de ahogamiento: viscosidad. Viscosidad es realmente un *dampening*, pero al tratarse de líquidos le puse este nombre.

La fórmula que sigue esta fuerza a través de 4 puntos diferentes que aplican torque es:

$$\begin{aligned} & / \text{rb.y} > \text{alturaAgua} - \text{profAhogo} \Rightarrow \text{nada (actúa la gravedad sola)} \\ \text{Si} == \{ & \text{rb.y} < \text{alturaAgua} - \text{profAhogo} \Rightarrow \text{fuerza} = \text{densidad} * \text{volumen} - \text{visc} \\ & \backslash \text{ otro caso} \Rightarrow \text{fuerza} = \text{densidad} * \text{volumen} * (\text{rb.y} \\ & \quad - \text{profAhogo} - \text{alturaAgua}) * 9.8 / 2 - \text{visc} \end{aligned}$$

Efectos incorporados:

1. Flotación (buoyancy)
2. Gravedad
3. Viento
4. Fuegos artificiales (ganar)
5. Fuegos artificiales (perder)
6. Rotación (con torque)

Experimentos:

Como puse antes, he probado aplicar un elemento de viscosidad al líquido que se simula en Buoyancy, además de experimentar con algún collider que otro y reestructurar muchísimo código (de nuevo...).

Lo que más rabia me ha dado es que por mucho que lo intentara he sido incapaz de mover la cámara de manera satisfactoria, llegando al punto de que la he intentado deshabilitar para que el juego no pierda jugabilidad.