

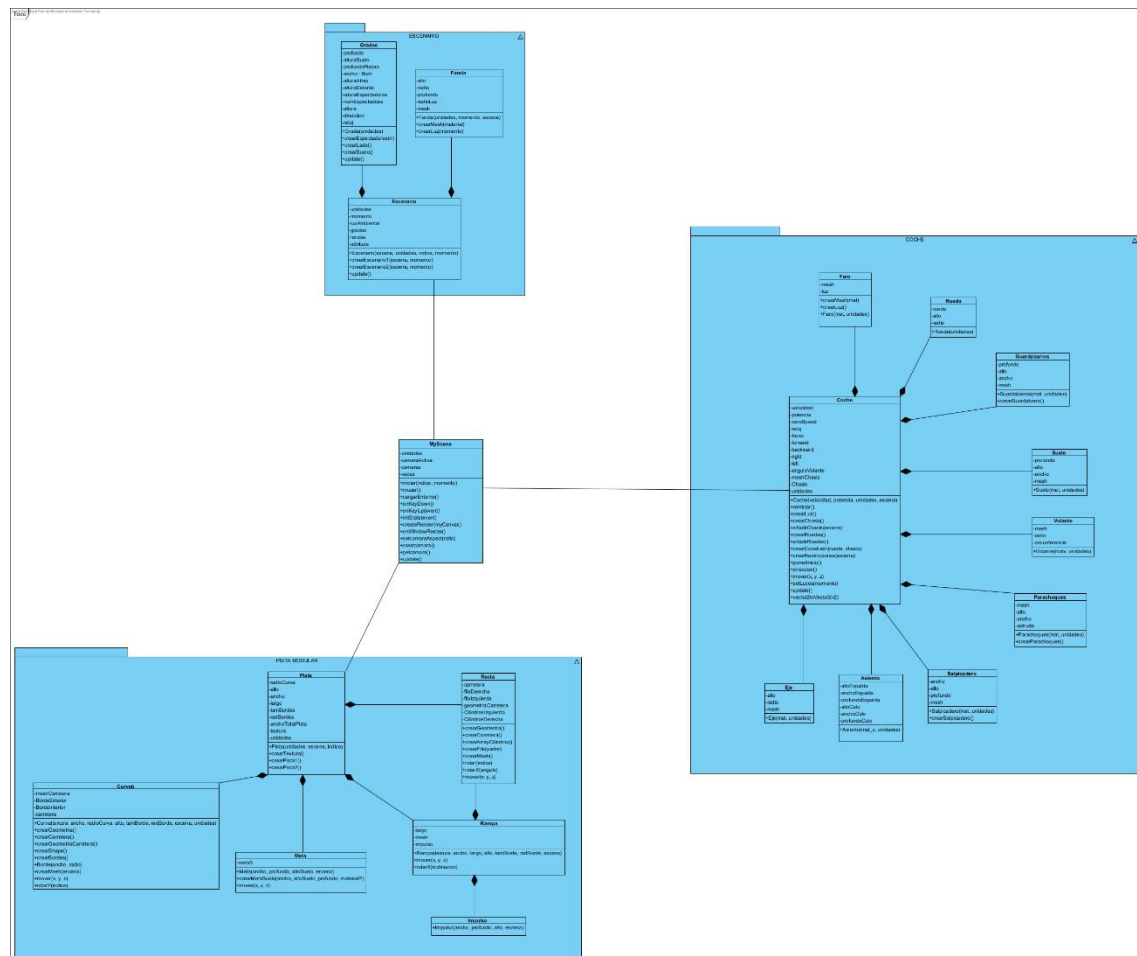
DOCUMENTACION PRACTICA 2

Descripción

He hecho un juego de karting con pista modular y un escenario mínimamente animado.

Primeramente cada usuario podría escoger el tipo de pista y el momento en el que la quiere hacer (noche o día) y empezará la carrera desde la posición de salida y tendrá que recorrer el circuito hasta volver a la meta para ganar

Diagrama de clases



Dificultades

Primero las pienso enumerar y luego como las conseguí superar

1. Poner físicas a las curvas
2. Orientar la cámara superior
3. Impulsar hacia delante al coche
4. Cámara en posición del piloto

5. Animación del volante
6. Caja de colisión del chasis
7. Colisión de las Ruedas con el chasis
8. Ruedas girando sin sentido
9. Movilidad de las partes de la pista para su colocación

1. Debido a que el profesor me indico que un `convexMesh` podría ser terriblemente ineficiente se me ocurrió pasar de una barrera únicamente plana a que estuviera formado por cilindros, es por eso que únicamente tenía que hacer una función que recorriera la curva interior o exterior colocando los cilindros. No obstante, como `physijs` no funciona correctamente si mueves la geometría previa, no pudo ser simplemente desplazar la geometría hasta la curva y luego ir colocando en relación con la lejanía de esta al centro de rotación, sino que se tuvo que ir colocando cada uno por medio de `position.set` y formulas geométricas

2. Para esto se me ocurrió basarme en como tenía montado el coche y calcular la diferencia entre el parachoques trasero y el delantero en coordenadas globales por medio de un método y así en cada `getcamara()` en el que la camara activa fuera esta coger dicha dirección del coche y modificar el atributo `up.x` y `up.z` con `dir.x` y `dir.z`

3. Usándome del método del anterior para obtener la dirección del coche, únicamente tenía que aplicar dicho impulso en la dirección completa del coche (`x,y,z`)

4. Debido a que colgar la camara del coche no me funciono, me vi obligado a actualizar el `getcamara()` de tal forma que dependiendo de la camara que estuviera activa colocase la activa un offset determinado con respecto al coche

5. Para animar el volante hice uso de los valores que movían las ruedas a derecha e izquierda e hice unas transiciones que rotasen el mismo a derecha o a izquierda siguiendo a las ruedas y en caso de que “soltara el volante” este retomaría la posición inicial, como las ruedas

6. Para lograr una cabina física compleja sin tener en cuenta las limitaciones propias de `Physijs` de las formas, use el `BoxHelper` de todo el chasis. El problema es que la geometría generada por el mismo se encontraba centrada en el origen, aunque el Chasis estuviera posado sobre el plano `xz`. Como `Physijs` no permite mover la geometría y posteriormente hacer el `BoxMesh()` y si quería que encajaran, tendría que mover el `BoxMesh` hasta que se encontrara posado sobre el eje `xz` y así que cuadrara con el Chasis. Esto ultimo no pudo ser posible ya que el chasis colgaba del `BoxMesh`, y por mucho que moviera primero el `BoxMesh` y luego lo colgara, estos seguían manteniendo la distancia, por lo que me vi obligado a mover todo el Chasis hasta abajo para que encajaran

7. Para evitar este hecho y darle un poco mas de “realismo” al coche, hice unos “ejes” en los que no formarían parte del `BoxHelper` que calcularía el tamaño de la caja `Physijs` del Chasis de esa forma lograba una física acertada para la cabina y unas ruedas girando unidas a un chasis y no flotando en el aire.

8. Este problema fue un verdadero quebradero de cabeza ya que seguía al pie de la letra como estaba hecho en el ejemplo. Pero se ve que el `angularMotor` de `Physijs` sigue las normas de orden de `THREE` y teniendo al coche orientado en el eje `z` y teniendo la rotación de esta en el eje `x` provocaba que estas no giraran como unas ruedas normales, por lo que me vi obligado a rotar todo el chasis, orientarlo en el eje `x` como tenía el profesor en su ejemplo y modificar los ángulos de giro y restricciones y ya si funcionó perfectamente

8. Teniendo en cuenta que la curva , recta y rampa son varios `physijs` bajo una misma clase me vi obligado a hacer rotaciones poco ortodoxas con estas dando solo posibilidad a la recta a quedarse orientada con el eje x o eje z (0,1) y la curva rotar ($\pi/2$, π , $3\pi/2$, 2π). Quiero aclarar que intentar meterlo todo en un `Object3D` auxiliar y rotar este con un simple `rotatio`.y no funcionó.

Resumen

A grandes rasgos, mi practica no tiene una gran funcionalidad, ya que lo único que he implementado al final de "interacción" ha sido la meta y el turbo, que decidí solo dejar uno al final de la rampa, pero me siento enormemente orgulloso del diseño de las clases y el modelaje de todos los elementos de la escena que son plenamente hechos a mano por mi y es por ello que tampoco se les ha podido dar una buena textura debido a como tiene implementado las texturas en los cilindros `THREE` y en los extrudes que es principalmente de lo que esta formado todo mi mundo