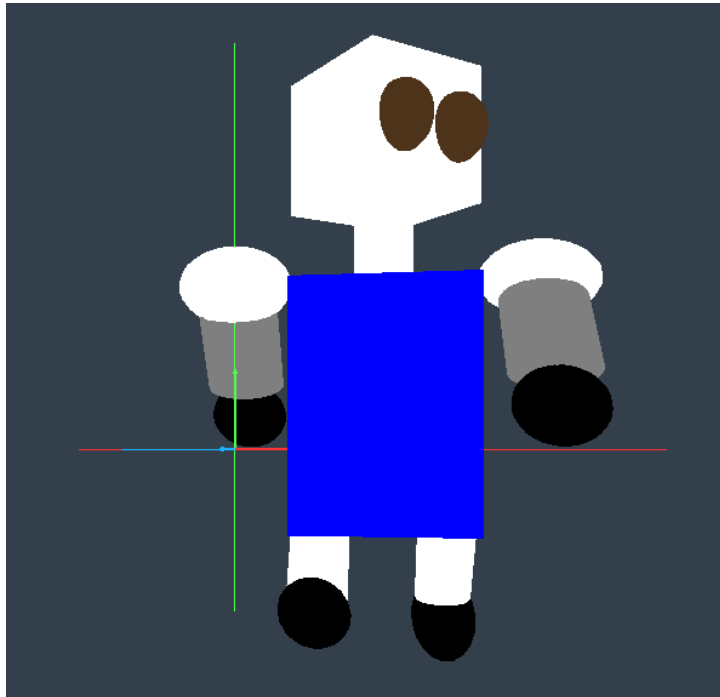


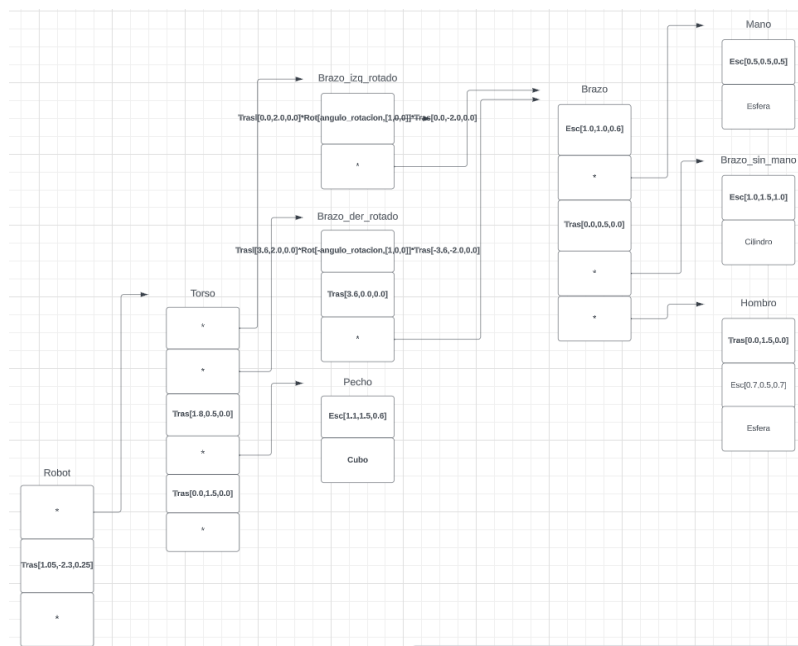
ENTREGA DEL PDF ASOCIADO A LA PRÁCTICA 3 DE IG

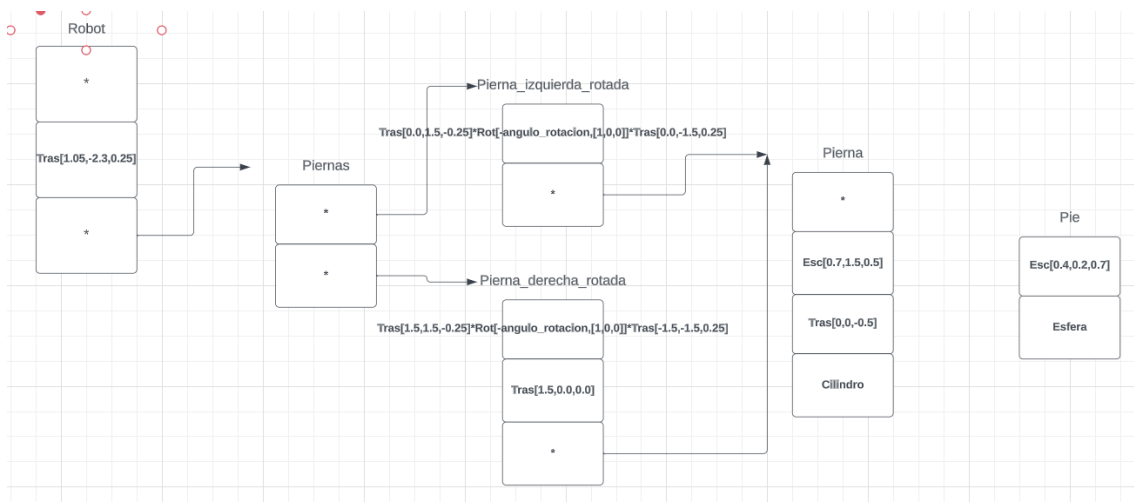
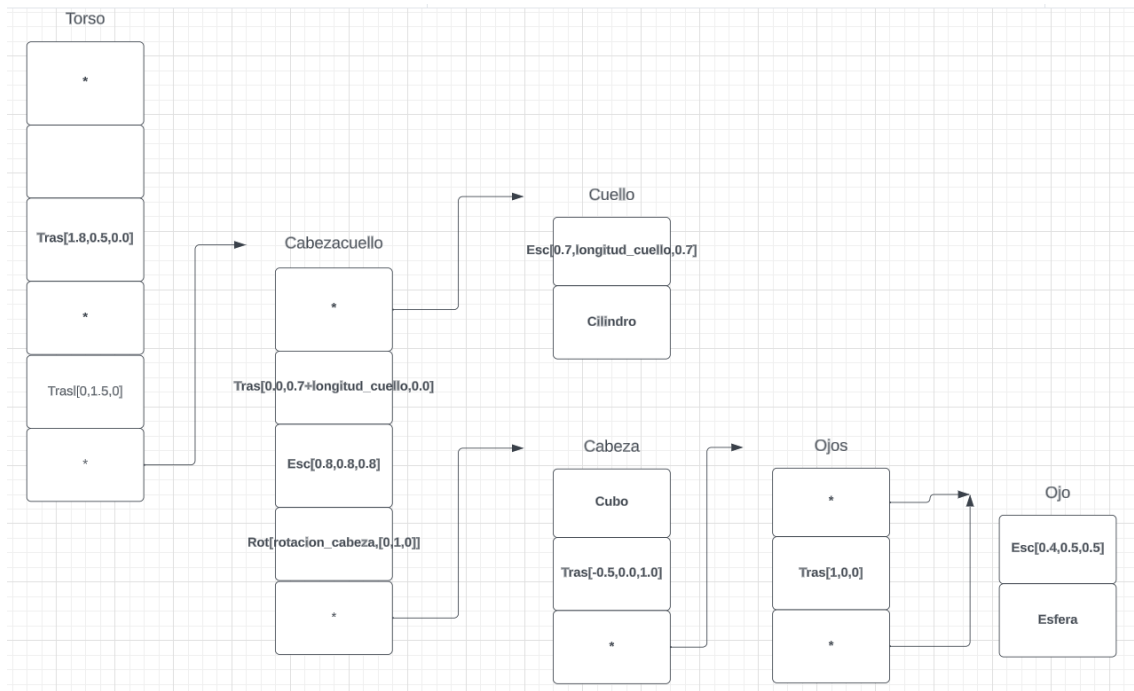
ALUMNO: Guillermo Ramblado Carrasco

DNI: 29628168M // Correo: guilleramblado@correo.ugr.es // DGIIM



1. GRAFO DE ESCENA TIPO PHIGS





2. NODOS EMPLEADOS EN EL GRAFO DE ESCENA

Nodos	Grados de libertad	Color	Construcción
Clase Robot	angulo_rotacion		73-151
	longitud_cuello		
	rotacion_cabeza		
Torso			126.132
Brazo_izq_rotado			102-104
Brazo_der_rotado			106-109
Clase Brazo			21-46
Mano		Negro	23-26

Brazo_sin_mano		Gris	35-38
Hombro			29-32
Clase Pecho		Azul	48-52
Cabezacuello			117-123
Cuello			112-114
Clase Cabeza			59-69
Ojos		Marron	60-64
Clase Ojo			54-57
Piernas			94-96
Pierna_izquierda_rotada			85-86
Pierna_derecha_rotada			89-92
Clase Pierna			43617
Pie		Negro	45.268

3. GRADOS DE LIBERTAD EMPLEADOS

Por último, entraré en detalle acerca de los diferentes grados de libertad de mi objeto 'robot' parametrizable:

1.angulo_rotacion

- Nodos con matrices de transformación dependientes de dicho parámetro:

Pierna_izquierda_rotada // Pierna_derecha_rotada: ambos nodos poseen dos matrices de rotación en torno al centro del techo del cilindro que compone la pierna, con eje de rotación un vector con la misma dirección que el eje X (y que pasa por dicho punto). Dichas matrices de rotación usan el mismo ángulo de rotación, pero con signos opuestos, consiguiendo que una pierna gire en sentido antihorario (ángulo positivo), y la otra, en sentido horario (ángulo negativo), de tal forma que conseguimos que el movimiento de las piernas se asemeje a la realidad (cuando desplazamos una hacia adelante, la otra se desplaza hacia atrás).

Braz_izq_rotado // Brazo_der_rotado: ambos nodos poseen dos matrices de rotación con respecto al centro de la esfera que forma el hombro, usando el mismo ángulo de rotación, pero con signos opuestos, como en el caso anterior de las piernas, para que el movimiento se asemeje al que realizamos en la vida real.

El ángulo de rotación empleado en brazos y piernas es el mismo, consiguiendo así que el movimiento de las extremidades sea lo más realista posible.

-Actualización de dicho parámetro:

El ángulo de rotación empleado para brazos y piernas se actualiza de forma oscilante, oscilando dentro del intervalo $[-45,45]$, con dos oscilaciones por segundo. La fórmula empleada es la misma que se nos indicó en el guion de prácticas:

- **Oscilante:** hacemos $v = a + b \sin(2\pi nt)$

Aquí $a = (v_{min} + v_{max})/2$ y $b = (v_{max} - v_{min})/2$. Ahora el valor v oscila entre v_{min} y v_{max} , con un número n (flotante) de oscilaciones por segundo. Se puede usar para rotaciones, traslaciones y escalados, de forma que sabemos que el valor v siempre está acotado por v_{min} y v_{max} .

En este caso, $n=2$ (definiendo dos oscilaciones por segundo), $v.min=-45$, $v.max=45$.

```
float min=-45.0,max=45.0;
float a = (min+max)/2;
float b = (max-min)/2;
fijarRotacionPiernasBrazos(a+b*sin(2*M_PI*2*t_sec));
```

He escogido dicho valor máximo y mínimo ya que normalmente cuando nos movemos, el ángulo de giro de brazos y piernas suele situarse entre dichos valores. De nuevo, intento conseguir un movimiento lo más realista posible.

2.longitud_cuello

- Nodos con matrices de transformación dependientes de dicho parámetro:

Cuello: posee una matriz de escalado tal que el escalado que se aplica a la componente Y es justamente la longitud del cuello indicada (depende del parámetro). Su valor inicial se situará dentro del intervalo [0.5,1], lo que implica definir la longitud del cuello.

Cabezacuello: posee una matriz de traslación tal que el desplazamiento que realizamos sobre el eje Y es justamente ' $0.7+longitud_cuello$ ', con el objetivo de ajustar la cabeza al cuello, dependiendo del valor que le hayamos dado al parámetro.

-Actualización de dicho parámetro:

Se actualizará de forma oscilante, con dos oscilaciones por segundo, empleando la misma fórmula que indiqué en el grado de libertad anterior.

```
float min=0.5,max=1.0;
float a = (min+max)/2;
float b = (max-min)/2;
fijarLongitudCuello(a+b*sin(2*M_PI*2*t_sec));
```

3.rotacion_cabeza

- Nodos con matrices de transformación dependientes de dicho parámetro:

Cabezacuello: dicho nodo posee una matriz de rotación que nos permitirá rotar la esfera que representa la cabeza del robot con respecto a su centro, con eje de rotación 'eje Y' (aplico inicialmente la rotación de la esfera centrada en el origen antes de realizar las transformaciones geométricas necesarias para posicionar correctamente la cabeza con respecto al cuerpo del robot, ya que esto simplifica mucho el proceso de calcular el punto central de la esfera sobre el cual vamos a rotar).

-Actualización de dicho parámetro:

El valor inicial de dicho grado de libertad será 0. Iremos modificando su valor de forma lineal, ya que, a diferencia de la rotación aplicada sobre brazos y piernas, permitimos que la cabeza del robot pueda rotar todo lo que se desee, aplicando la siguiente fórmula indicada en el guion de prácticas:

- **Linealmente:** hacemos $v = a + bt$.

Usamos dos valores a (valor inicial de v) y b (tasa de cambio de v por segundo). Se puede usar típicamente para rotaciones de velocidad angular constante, si v es un ángulo en radianes y b es $2\pi w$, donde w es la velocidad angular en ciclos por segundo. Se puede usar para traslaciones y escalados, pero no es aconsejable ya que el desplazamiento o el factor de escala crecerían indefinidamente con el tiempo.

He decidido que la cabeza rote 15º grados por segundo, por lo que la actualización a realizar se conseguirá de la siguiente forma:

```
fijarRotacionCabeza(15.0*t_sec);
```