**Castillo**

**Presentado por: Kenneth Forero 6000168**

Se crearon 3 primitivas las cuales se duplicaron 4 veces, a cada una se le dio su respectiva posición y rotación.

* Para el cono se usó un contador para que este se pudiese hacer, se utilizaron cosenos y senos para armar dicha figura:

var geoCono = new THREE.Geometry();

var zTemp = 0;

var res = 50.0;

for( var i = 0; i < res; i++){

for( var j = 0; j < res; j++){

var punto = new THREE.Vector3();

punto.x = (2.0 \* (res - i) / res ) \* Math.cos( ( j \* 2.0 \* Math.PI ) / res) ;

punto.y = 3.0 \* i / res;

punto.z = (2.0 \* (res - i) / res ) \* Math.sin( ( j \* 2.0 \* Math.PI ) / res) ;

geoCono.vertices.push( punto );

}

}

* Para el cilindro se utilizó el mismo método, pero sin tener que ir restando:

var geoCilindro = new THREE.Geometry();

var zTemp = 0;

var res = 25.0;

for( var i = 0; i < res; i++){

for( var j = 0; j < res; j++){

var punto = new THREE.Vector3();

punto.x = 2 \* Math.cos( ( j \* 2 \* Math.PI )/res);

punto.y = 5 \* i/res;

punto.z = 2 \* Math.sin( ( j \* 2 \* Math.PI )/res);

geoCilindro.vertices.push( punto );

}

}

* Para el muro fue mucho mas sencillo ya que era un plano y no requeria la variable “z” ni tampoco de variables trigonométricas:

var geoMuro = new THREE.Geometry();

var zTemp = 0;

var res = 25.0;

for( var i = 0; i < res; i++){

for( var j = 0; j < res; j++){

var punto = new THREE.Vector3();

punto.x = (16 \*( j \* 1 )/res);

punto.y = 5 \* i/res;

geoMuro.vertices.push( punto );

}

}