

SOMMAIRE :

- I- Les choix techniques
- II- Difficultés rencontrées
- III- La construction des quilles

I- Les choix techniques

Création de la piste :

Pour créer la piste, nous avons créé 2 plans horizontaux et avons appliqué un quadrillage de couleur blanche.

Ensuite nous avons créé les parois de la piste grâce à un plan vertical pour le côté ouest et un rectangle délimité par 4 points pour le côté est.

Création des boules :

Nous avons tout d'abord utilisé la fonction sphère qui permet de concevoir la sphère.

Pour le motif de la boule nous avons conçu une fonction qui permet de faire des cercles autour de la boule de bowling.

On applique ensuite le motif à la sphère dans la fonction boule et l'on regroupe les 2 entités grâce à la fonction group de THREE.js.

Création des quilles :

Pour créer les quilles nous avons décidé d'ajouter un fichier Quille.js à notre projet qui permettra d'ajouter toutes les fonctions relatives à la création des quilles.

Nous avons divisé les quilles en 3 parties : la base de la quille, le corps de la quille et la tête.

Des schémas expliquant la construction seront affichés en partie III.

Création de l'animation :

Nous avons tout d'abord défini les 2 trajectoires que le joueur pourra choisir.

En fonction de la trajectoire choisie, on trace une courbe de Bézier rectiligne si le choix est le numéro 1 ou alors on trace une courbe de Bézier définie par 4 points sur un plan non rectiligne.

On récupère ensuite les points de la courbe tracée et on fait bouger la boule grâce à la fonction setInterval qui exécute les instructions suivant un temps donné.

II- Difficultés rencontrées

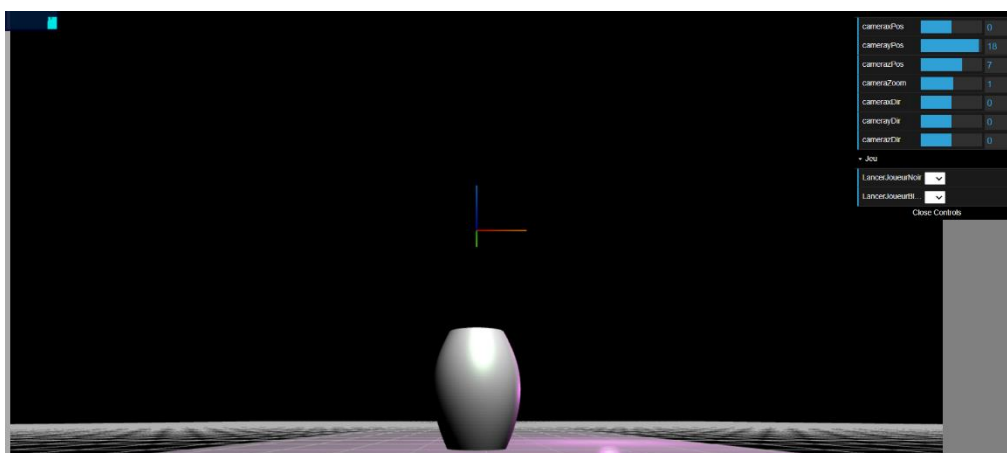
La plus grande difficulté a été celle de l'animation. En effet, après avoir tracé les différentes courbes de Bézier qui sert de trajectoire, nous n'arrivions pas à faire suivre cette trajectoire à la boule de bowling.

Nous avons également pris beaucoup de temps pour savoir à quel instant la boule touche une quille ou non.

III- La construction des quilles

1^{ère} étape :

La construction de la base de la quille :

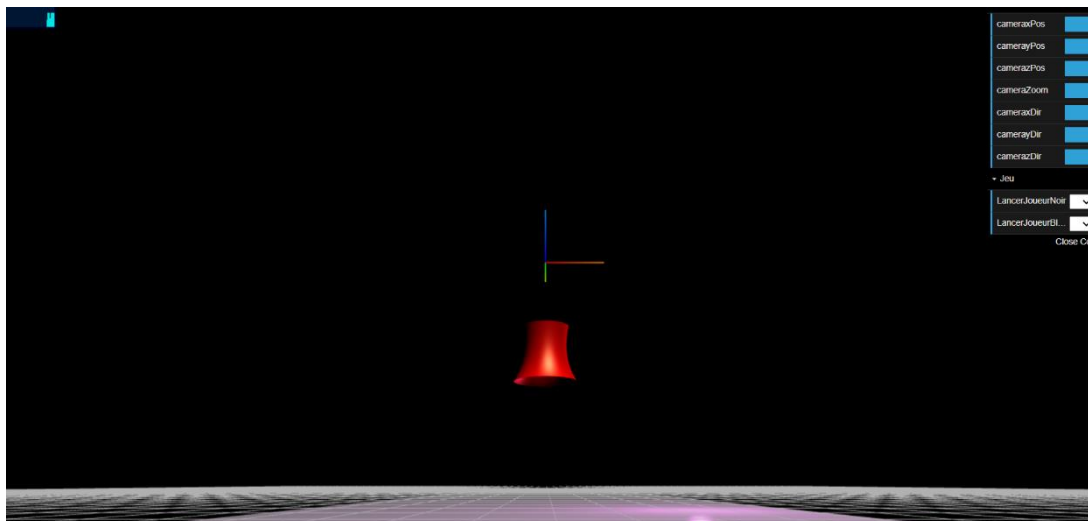


Code associé à la construction de la base de la quille :

```
3 function base(scene,nbPtCB,nbPtRot,coul,P0,P1,P2,P3){
4
5   let tabP= new Array(4);
6   for (let k=0;k<tabP.length;k++){
7     tabP[k]= new THREE.Vector3(0,0,0);
8   }
9   tabP[0]=P0;
10  tabP[1]=P1;
11  tabP[2]=P2;
12  tabP[3]=P3;
13
14  let lathe1 = latheBezTab(scene,nbPtCB,nbPtRot,tabP,coul,0.95,false);
15
16  lathe1.rotateX(Math.PI/2);
17  //lathe1.translateZ(25);
18  //lathe1.translateY(-4);
19  scene.add(lathe1);
20  return lathe1;
21 }
22
```

2^{ème} étape :

La construction du corps de la quille :



Code associé à la construction du corps de la quille :

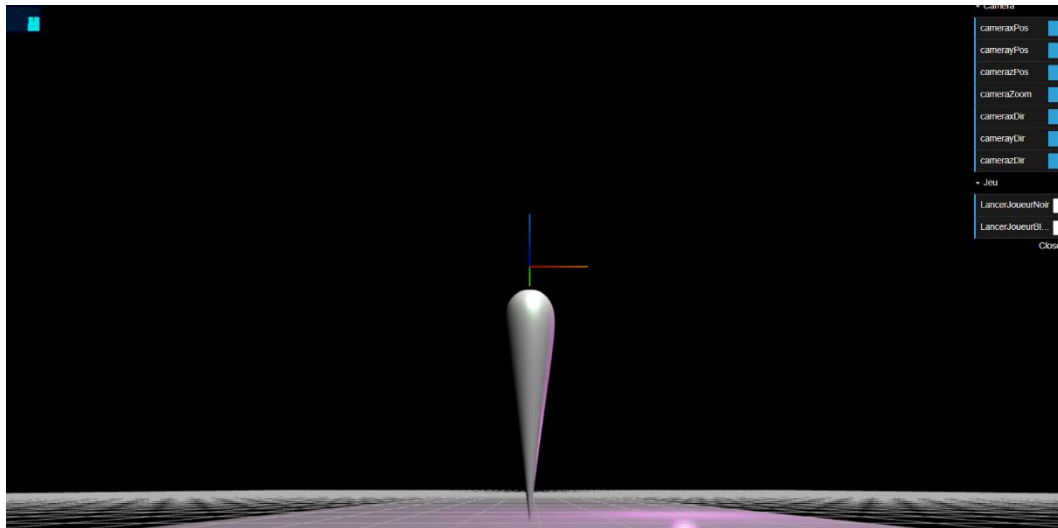
```
function body(scene,nbPtCB,nbPtRot,coul,P0,P1,P2,P3)
{
  let tabP= new Array(4);
  for (let k=0;k<tabP.length;k++){
    tabP[k]= new THREE.Vector3(0,0,0);
  }
  tabP[0]=P0;
  tabP[1]=P1;
  tabP[2]=P2;
  tabP[3]=P3;

  let lathe1 = latheBezTab(scene,nbPtCB,nbPtRot,tabP,coul,0.95,false);
  lathe1.rotateX(Math.PI/2);

  scene.add(lathe1);
  return lathe1;
}
```

3^{ème} étape :

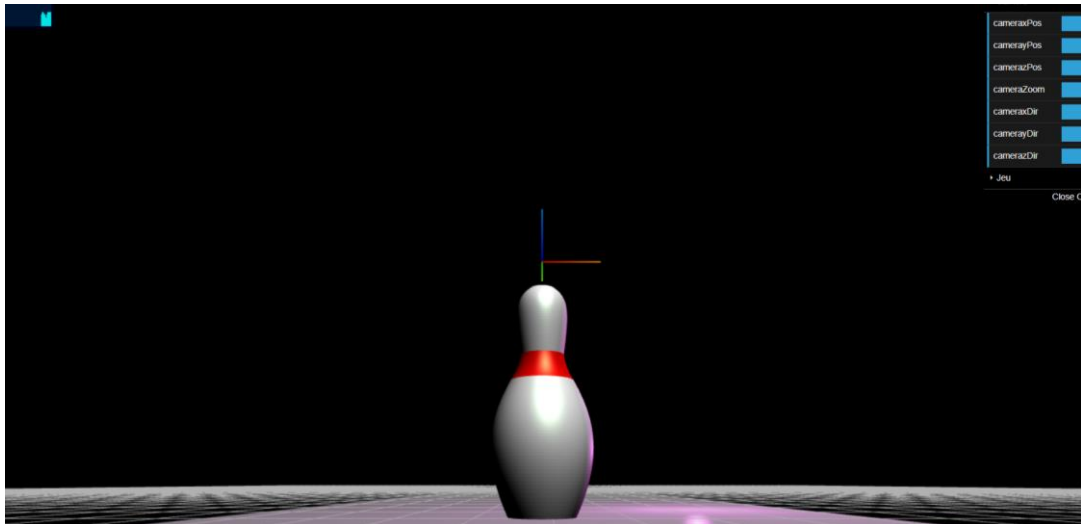
La construction de la tête de la quille :



Code associé à la construction de la tête de la quille :

```
43  
44 function head(scene,nbPtCB,nbePtRot,M3,N1,N2,N3){  
45  
46   let tabP3= new Array(4);  
47   for (let k=0;k<tabP3.length;k++){  
48     tabP3[k]= new THREE.Vector3(0,0,0);  
49   }  
50   tabP3[0]=M3;  
51   tabP3[1]=N1;  
52   tabP3[2]=N2;  
53   tabP3[3]=N3;  
54  
55   let lathe3 = latheBezTab(scene,nbPtCB,nbePtRot,tabP3,"#FFFFFF",0.95,false);  
56   lathe3.rotateX(Math.PI/2);  
57   scene.add(lathe3);  
58  
59   return lathe3;  
60  
61 }  
62
```

On regroupe alors les 3 parties de la quille ce qui donne :



Code associé à la construction de la quille entière :

```

62
63 function moveQuille(quille,movX,movY, movZ)
64 {
65     quille.translateX(movX);
66     quille.translateY(movY);
67     quille.translateZ(movZ);
68 }
69
70 function Quille(scene,P)
71 {
72     Let P0 = new THREE.Vector3(0.5,0.02);
73     Let P1 = new THREE.Vector3(1,1);
74     Let P2 = new THREE.Vector3(0.71,1.57);
75     Let P3 = new THREE.Vector3(0.5,1.984);
76
77     Let M1 = new THREE.Vector3(0.29,2.29);
78     Let M2 = new THREE.Vector3(0.33,2.65);
79     Let M3 = new THREE.Vector3(0.374,2.88);
80
81     Let N1 = new THREE.Vector3(0.6,3.21);
82     Let N2 = new THREE.Vector3(0.2,3.517);
83     Let N3 = new THREE.Vector3(0,4);
84
85     Let H=head(scene,50,150,"#FFFFFF",M3,N1,N2,N3);
86     Let b=body(scene,50,150,"#FF0000",P3,M1,M2,M3);
87     Let B=Base(scene,50,150,"#FFFFFF",P0,P1,P2,P3);
88
89     Let quille = new THREE.Group();
90     quille.add(H);
91     quille.add(b);
92     quille.add(B);
93     moveQuille(quille,P.x,P.y,P.z);
94
95     return quille;
96
97
98 }

```

Pour trouver chaque point de contrôle des courbes de Bézier nous avons utilisé l'outil Desmos Bezier qui nous renvoie les coordonnées des points de contrôle en fonction de la courbe que l'on souhaite : <https://www.desmos.com/calculator/d1ofwre0fr?lang=fr>

