



Kelompok Moon

Maze Bounce Ball

GRAFIKA KOMPUTER - B

Anggota Kelompok



Wardatul Amalia S.
5025211006



Salsabila Fatma A.
5025211057

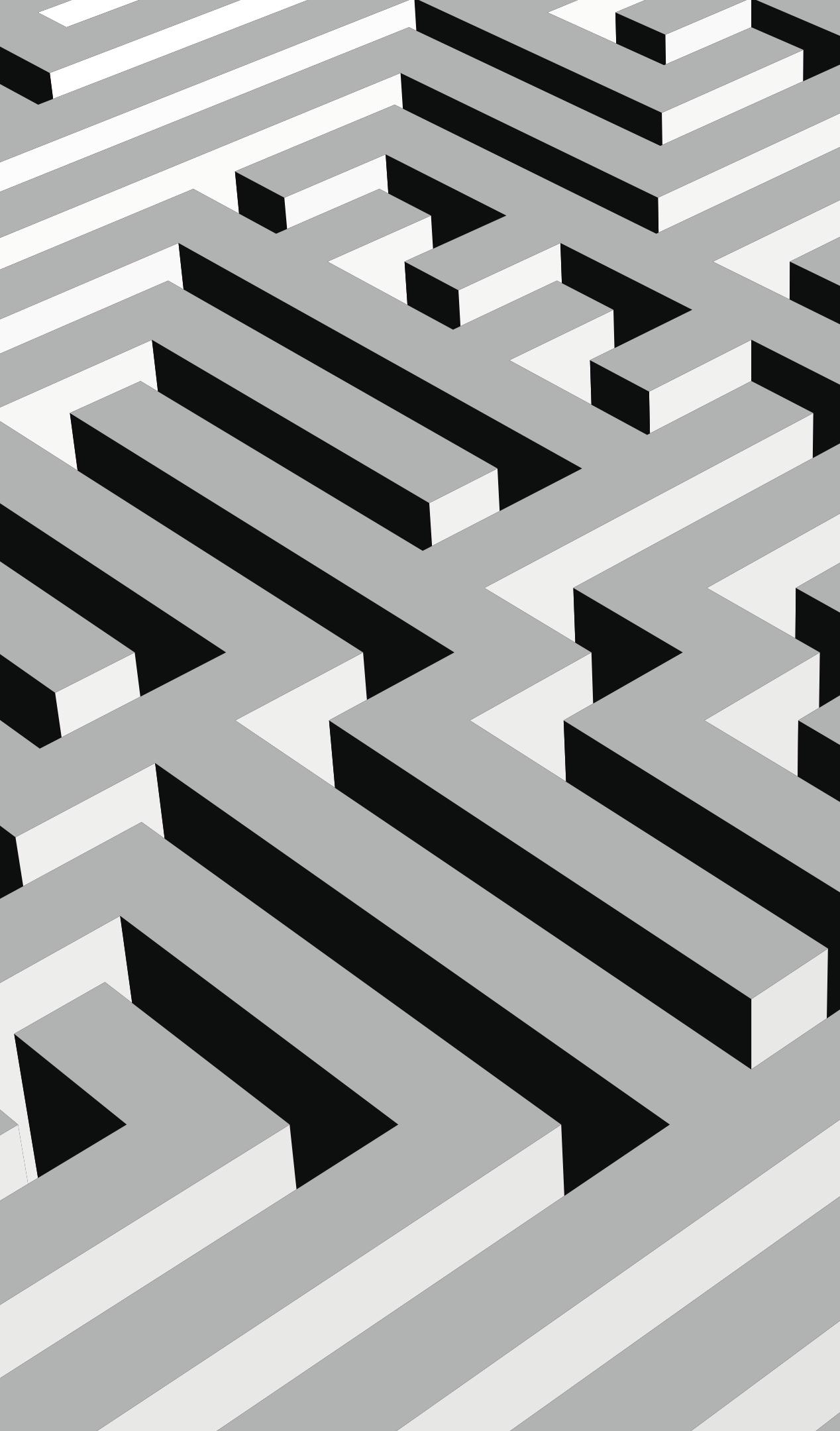


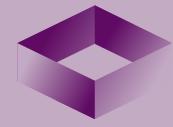
Yusna Millaturrosyidah
5025211254

List Of Content

Deskripsi Game

Fungsi Three.js yang digunakan

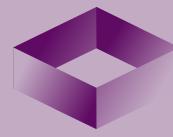




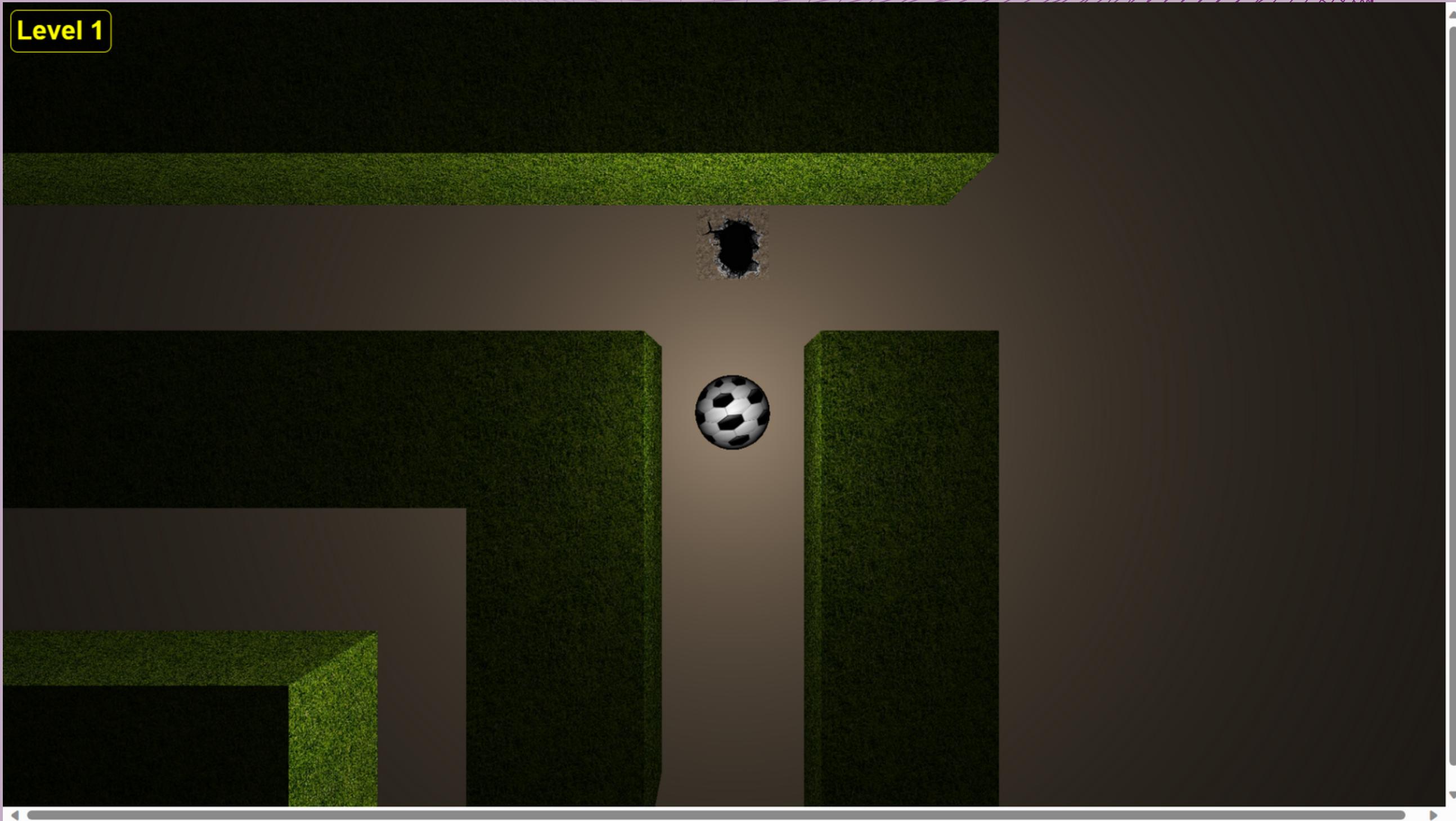
Kelompok Moon

Maze Bounce Ball adalah game 3D yang menghadirkan pengalaman bermain yang menarik dan menantang. Dalam permainan ini, player akan memasuki maze yang tujuannya adalah menggiring bola untuk berhasil melewati setiap level maze dan mencapai garis finish. Namun, ada satu rintangan utama yang harus dihindari dengan cermat, yaitu "hole" atau lubang yang berada di dekat garis finish. Jika bola terkena hole, pemain harus memulai permainan dari awal lagi.





Kelompok Moon



Texture

Di dalam program ini terdapat 4 jenis texture :

- a) ironTexture untuk bola
- b) planeTexture untuk ground
- c) brickTexture untuk maze
- d) holeTexture untuk hole

```
ironTexture = THREE.ImageUtils.loadTexture(  
    "/assets/texture/50718b9eb5268411b02e0a8167088ef9.jpg"  
)  
planeTexture = THREE.ImageUtils.loadTexture(  
    "/assets/texture/4c712fb1f820b43b1b8eac593302af6c.jpg"  
)  
brickTexture = THREE.ImageUtils.loadTexture(  
    "/assets/texture/b8b3f5ba52e906e914c5435449bf448b.jpg"  
)  
holeTexture = THREE.ImageUtils.loadTexture(  
    "/assets/texture/aykwbbm246jaokllnak.png"  
)
```

Cube Geometry

```
function generate_maze_mesh(field) {
    var dummy = new THREE.Geometry();
    for (var i = 0; i < field.dimension; i++) {
        for (var j = 0; j < field.dimension; j++) {
            if (field[i][j]) {
                var geometry = new THREE.CubeGeometry(1, 1, 1, 1, 1, 1);
                var mesh_ij = new THREE.Mesh(geometry);
                mesh_ij.position.x = i;
                mesh_ij.position.y = j;
                mesh_ij.position.z = 0.5;
                THREE.GeometryUtils.merge(dummy, mesh_ij);
            }
        }
    }
    var material = new THREE.MeshPhongMaterial({ map: brickTexture });
    var mesh = new THREE.Mesh(dummy, material);
    return mesh;
}
```

Cube Geometry digunakan untuk menampilkan maze. 1 cube maze berukuran $1 \times 1 \times 1$ dan segmen bernilai 1 di setiap sisinya. Setiap cube geometry yang dibuat, akan disusun membentuk sebuah labirin atau maze. Bentuk susunan maze akan selalu berbeda setiap fungsi generate_maze_mesh dipanggil. Di setiap level, ukuran dari keseluruhan maze akan bertambah

Sphere Geometry



```
// Add the ball.  
g = new THREE.SphereGeometry(ballRadius, 32, 16);  
m = new THREE.MeshPhongMaterial({ map: ironTexture  
})ballMesh = new THREE.Mesh(g, m);  
ballMesh.position.set(1, 1, ballRadius);  
scene.add(ballMesh);
```

Sphere Geometry digunakan untuk membentuk sebuah bola. Disini kami menggunakan ukuran radius bola sebesar 0.25 dengan widthSegments bernilai 32 dan heightSegments bernilai 16

Plane Geometry (Hole)



```
// Add a hole
g = new THREE.PlaneGeometry (0.5, 0.5);
m = m = new THREE.MeshPhongMaterial({ map: holeTexture });
hole = new THREE.Mesh(g,m);
hole.position.set(mazeDimension-2, mazeDimension-1.8, 0.1);
hole.rotation.set(Math.PI / 2, 0, 0);
scene.add(hole);
```

Salah satu penggunaan plane geometry pada program ini adalah untuk membuat area hole atau lubang. Area hole yang kami buat berukuran 0,5 x 0,5 dengan posisi 2 satuan dari garis finish

Plane Geometry (Ground)



```
// Add the ground.  
g = new THREE.PlaneGeometry(  
    mazeDimension * 10,  
    mazeDimension * 10,  
    mazeDimension,  
    mazeDimension  
)  
planeTexture.wrapS = planeTexture.wrapT = THREE.RepeatWrapping;  
planeTexture.repeat.set(mazeDimension * 5, mazeDimension * 5);  
m = new THREE.MeshPhongMaterial({ map: planeTexture });  
planeMesh = new THREE.Mesh(g, m);  
planeMesh.position.set((mazeDimension - 1) / 2, (mazeDimension - 1) / 2, 0);  
planeMesh.rotation.set(Math.PI / 2, 0, 0);  
scene.add(planeMesh);
```

Penggunaan plane geometry lainnya adalah untuk membuat tampilan ground. Plane geometry untuk ground dibuat dengan ukuran menyesuaikan ukuran field maze. Ukuran ground adalah 10 kali lipat ukuran maze.

Lighting & Camera

Jenis lighting yang digunakan adalah Point Light yang akan menyorot ke arah bola sebagai objek utama.

Jenis camera yang digunakan adalah perspective camera yang juga mengikuti pergerakan bola



```
// Add the light.  
light = new THREE.PointLight(0xffffff, 1);  
light.position.set(1, 1, 1.3);  
scene.add(light);  
  
// Add the camera.  
camera = new THREE.PerspectiveCamera(60, aspect, 1, 1000);  
camera.position.set(1, 1, 5);  
scene.add(camera);
```

Bounce

Dalam permainan ini, ketika bola menabrak maze dan masih memiliki kecepatan, maka bola akan dipantulkan ke arah sebaliknya dengan menerapkan konsep bounce dalam three.js. Bola akan dipantulkan dengan sumbu tegak lurus dengan permukaan maze

```
var inContact = false;
var contactEdge = wBall.GetContactList();
while (contactEdge) {
    var contact = contactEdge.contact;
    if (contact.IsTouching()) {
        inContact = true;
        break;
    }
    contactEdge = contactEdge.next;
}

wWorld.Step(1 / 60, 8, 3);

if (inContact && lv.y < 0) {
    lv.y = -lv.y * 0.5;

    wBall.SetLinearVelocity(lv);
}
```

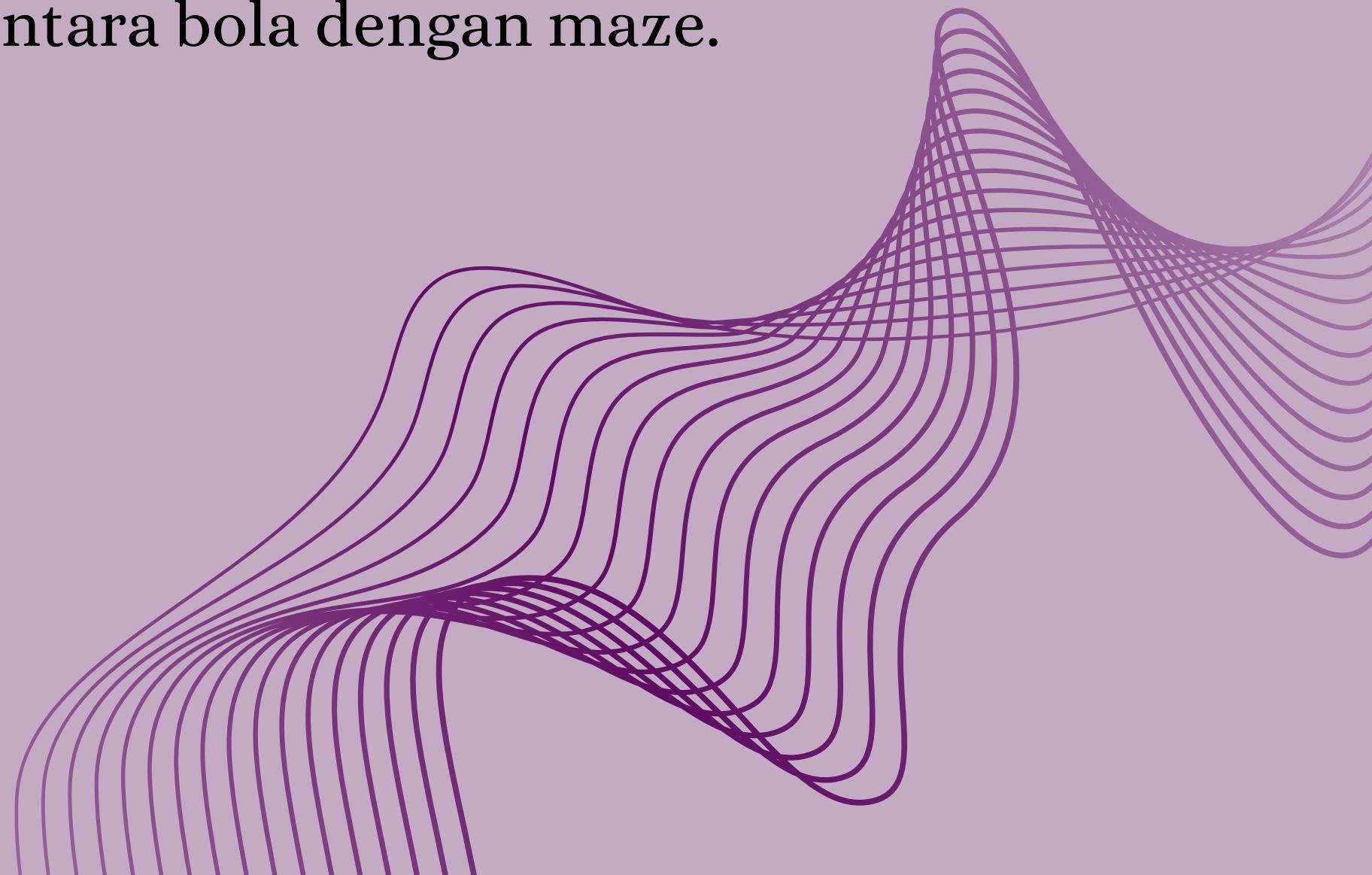
Friction



```
var lv = wBall.GetLinearVelocity();
lv.Multiply(0.95);
wBall.SetLinearVelocity(lv);

var f = new b2Vec2(
    keyAxis[0] * wBall.GetMass() * 0.25,
    keyAxis[1] * wBall.GetMass() * 0.25
);
wBall.ApplyImpulse(f, wBallGetPosition());
keyAxis = [0, 0];
```

Ketika bola menyentuh maze dan bergerak sejajar dengan permukaan maze yang bersentuhan dengan bola, maka gerak bola akan diperlambat karena adanya penerapan gesekan / friction antara bola dengan maze.



Control



```
KeyboardJS.bind.axis("left", "right", "down", "up", onMoveKey);  
$(window).resize(onResize);
```

Dalam program ini kami menggunakan tombol panah pada keyboard untuk menggerakkan axis yang menjadi patokan bola, lighting, dan camera.



Kelompok Moon

Thank You