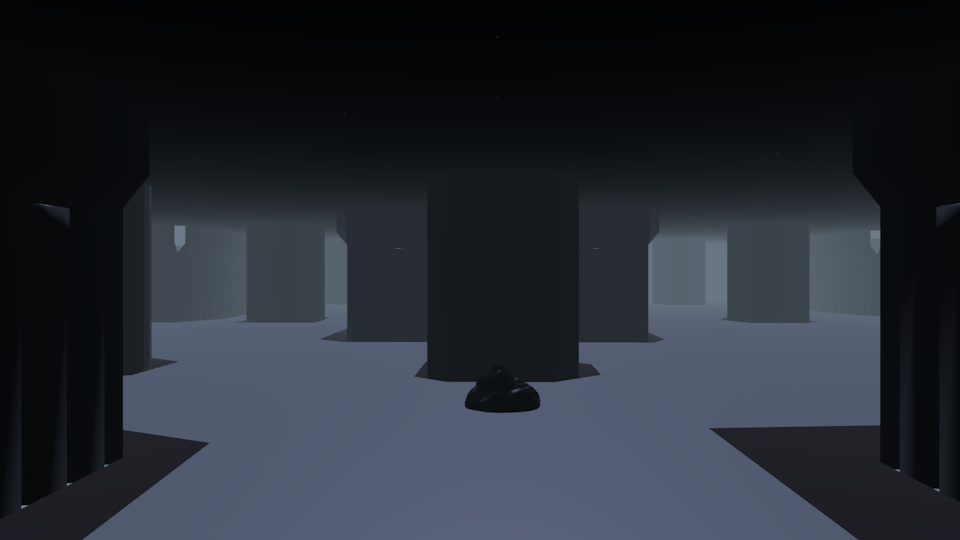
**Judul dan tema game**

Judul dari game ini adalah me vs ovaltine. Adalah game dimana user melawan komputer artificial itelegence. User diberikan tugas untuk cepat mengumpulkan trashbag dimana ai sebagai pengawas kegiatan user saat mengambil trash bag tersebut. Ai ini berbentuk oval berjumlah 10 oval memiliki garis awal penunjuk jalan, namun bila user terdeteksi melewati garis atau berada disamping kanan dan kiri garis tersebut maka ai akan mengejar dan memakanmu. Game ini akan memacu adrenalin karena berpacu dengan waktu dan jumlah. Jumlahnya 20

**Konsep game**

Ai akan menggunakan konsep greedy dalam penggunaan lajur jalannya dan mampu mengalahkan user dengan cepat

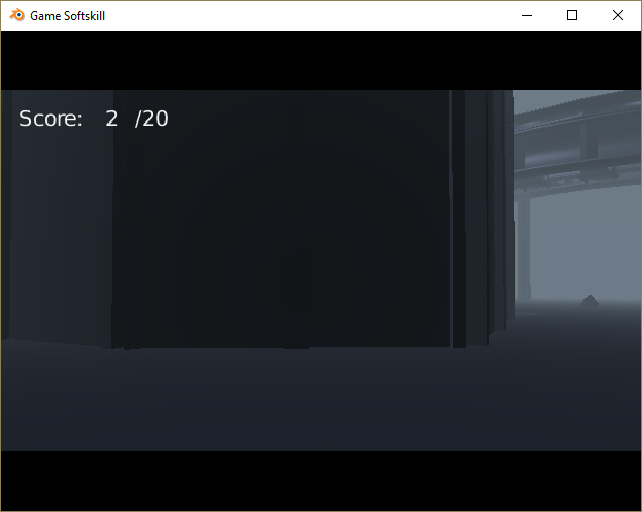
**Rancangan game**



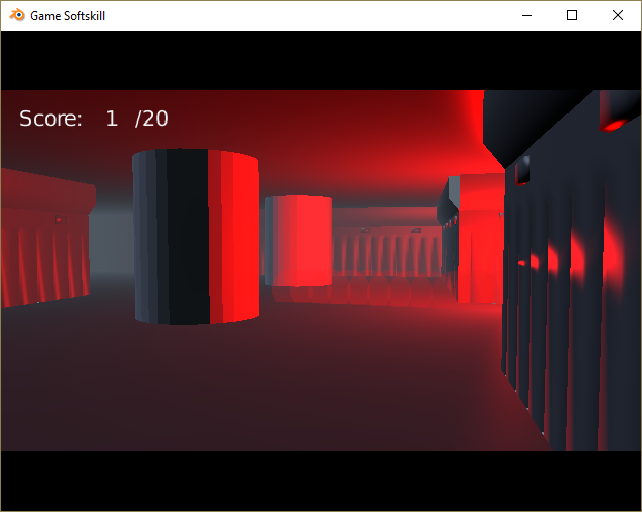
Gambar diatas merupakan tampilan awal render image dari tampilan start. Untuk memainkan game ini tinggal menggerakan wasd dan menggunakan mouse untuk menggerakkan free point of view user. Kemudian user langsung mengumpulkan trash bag berjumlah 20. Jika sudah semua tanpa dimakan ai maka dikatakan menang. Namun jika dimakan ai maka dikatakan ulang.



Adalah gambar tampilan awal game saat kita sudah merender game pada game logic



Gambar diatas adalah ketika trash bag sudah diambil 2 tanpa sepengetahuan AI



Gambar diatas adalah ketika AI berbentuk oval mengintai. User saat itu tidak dimakan karena user ngumpet diantara shape yang sudah dibuat

**Ini adalah kodingan AI di Blender yang menggunakan bahasa Python**

import bge

from bge import render as r

import math

cont = bge.logic.getCurrentController()

own = cont.owner

mouse = cont.sensors["Mouse"]

parent = own.parent

#set speed for camera movement

sensitivity = 0.05

#set camera rotation limits

high\_limit = 180

low\_limit = 0

h = r.getWindowHeight()//2

w = r.getWindowWidth()//2

x = (h - mouse.position[0])\*sensitivity

y = (w - mouse.position[1])\*sensitivity

if own["startup"]:

r.setMousePosition(h, w)

own ["startup"] = False

else:

rot = own.localOrientation.to\_euler()

pitch = abs(math.degrees(rot[0]))

if high\_limit > (pitch+y) > low\_limit:

pitch += y

elif (pitch+y) < low\_limit:

pitch = low\_limit

elif (pitch+y) > high\_limit:

pitch = high\_limit

rot[0] = math.radians(pitch)

own.localOrientation = rot.to\_matrix()

parentRot = parent.localOrientation.to\_euler()

yaw = math.degrees(parentRot[2]) + x

parentRot[2] = math.radians(yaw)

parent.localOrientation = parentRot.to\_matrix()

r.setMousePosition(h, w)