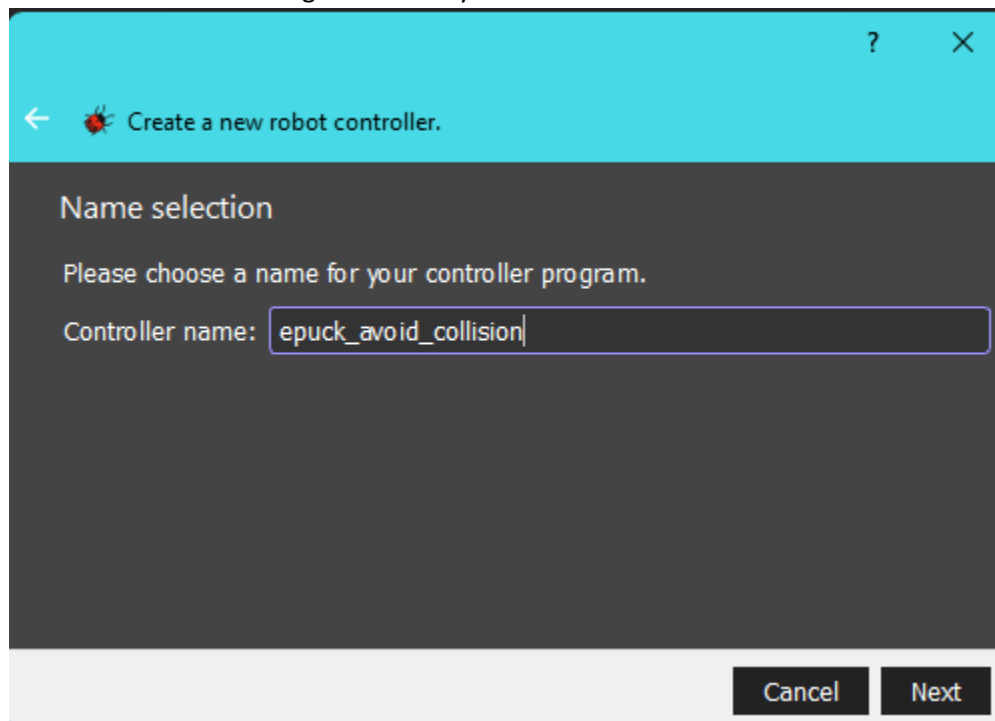


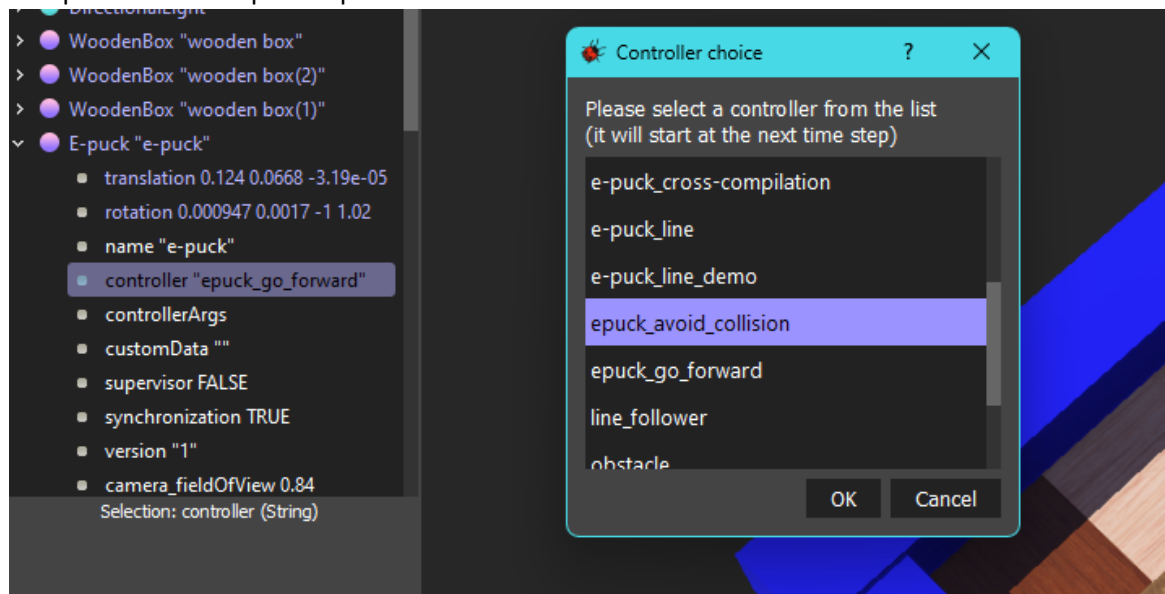
Nama : Pusaka Manggala
NIM : 1103194021

Tutorial 4 : More About Controllers

1. Duplikasi project pada tutorial 3 lalu bernama epuck_avoid_collision.wbt
2. Buat controller baru dengan bahasa Python



3. Terapkan controller pada E-puck



4. Pengkodean

from controller import Robot, DistanceSensor, Motor

```
# time in [ms] of a simulation step
TIME_STEP = 64

MAX_SPEED = 6.28

# create the Robot instance.
robot = Robot()

# initialize devices
ps = []
psNames = [
    'ps0', 'ps1', 'ps2', 'ps3',
    'ps4', 'ps5', 'ps6', 'ps7'
]

for i in range(8):
    ps.append(robot.getDevice(psNames[i]))
    ps[i].enable(TIME_STEP)

leftMotor = robot.getDevice('left wheel motor')
rightMotor = robot.getDevice('right wheel motor')
leftMotor.setPosition(float('inf'))
rightMotor.setPosition(float('inf'))
leftMotor.setVelocity(0.0)
rightMotor.setVelocity(0.0)

# feedback loop: step simulation until receiving an exit event
while robot.step(TIME_STEP) != -1:
    # read sensors outputs
    psValues = []
    for i in range(8):
        psValues.append(ps[i].getValue())

    # detect obstacles
    right_obstacle = psValues[0] > 80.0 or psValues[1] > 80.0 or psValues[2] > 80.0
    left_obstacle = psValues[5] > 80.0 or psValues[6] > 80.0 or psValues[7] > 80.0

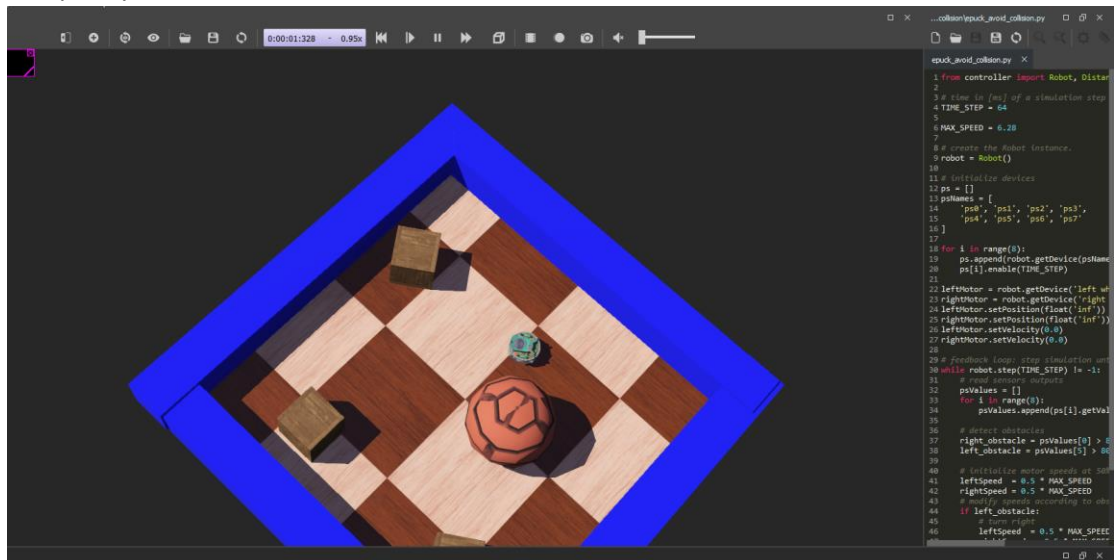
    # initialize motor speeds at 50% of MAX_SPEED.
    leftSpeed = 0.5 * MAX_SPEED
```

```

rightSpeed = 0.5 * MAX_SPEED
# modify speeds according to obstacles
if left_obstacle:
    # turn right
    leftSpeed = 0.5 * MAX_SPEED
    rightSpeed = -0.5 * MAX_SPEED
elif right_obstacle:
    # turn left
    leftSpeed = -0.5 * MAX_SPEED
    rightSpeed = 0.5 * MAX_SPEED
# write actuators inputs
leftMotor.setVelocity(leftSpeed)
rightMotor.setVelocity(rightSpeed)

```

5. Save dan run project. Robot akan berjalan dan menghindari tabrakan antar objek didepannya



6. Done.