

Modelación de sistemas multiagentes con gráficas computacionales (Gpo 523)

Equipo 5

Santiago Gabian Perez A01658280
Alan Alberto Flores Cuevas A01652453
Bruno Passarette Santos A01658904

Código Python para creación de intersección

```
main.py x
1 from .road import *
2 from .simulation import *
3 from .window import *
4
5
6 sim = Simulation()
7
8 # Add one road
9 sim.create_road((300, 98), (0, 98))
10
11 # Add multiple roads
12 sim.create_roads([
13     ((100, 98), (0, 98)),
14     ((0, 102), (300, 102)),
15     ((180, 60), (0, 60)),
16     ((220, 55), (180, 60)),
17     ((300, 30), (220, 55)),
18 ])
19
20
```

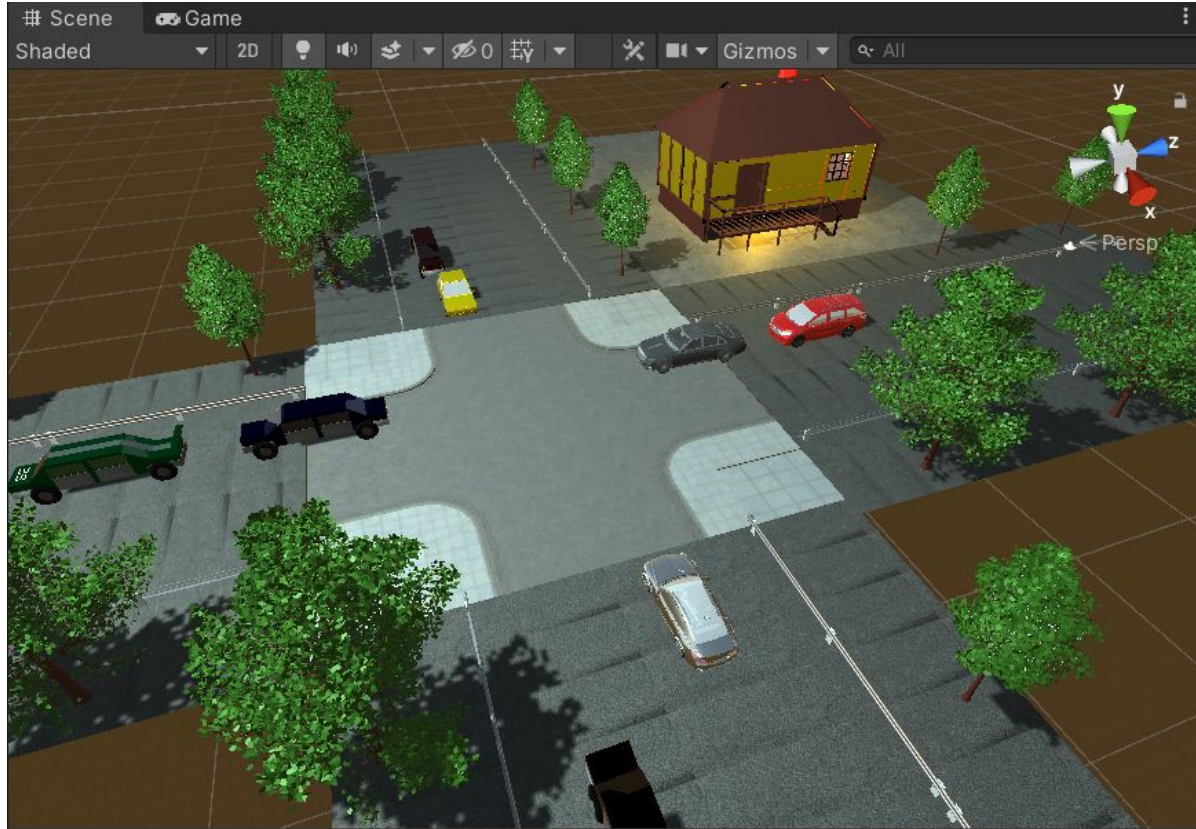
```
road.py x
1 from scipy.spatial import distance
2
3 class Road:
4     def __init__(self, start, end):
5         self.start = start
6         self.end = end
7
8         self.init_properties()
9
10    def init_properties(self):
11        self.length = distance.euclidean(self.start, self.end)
12        self.angle_sin = (self.end[1]-self.start[1]) / self.length
13        self.angle_cos = (self.end[0]-self.start[0]) / self.length
14
```

```
simulation.py x
1 from .road import Road
2
3 class Simulation:
4     def __init__(self, config={}):
5
6         self.set_default_config()
7
8
9         for attr, val in config.items():
10             setattr(self, attr, val)
11
12    def set_default_config(self):
13        self.t = 0.0
14        self.frame_count = 0
15        self.dt = 1/60
16        self.roads = []
17
18    def create_road(self, start, end):
19        road = Road(start, end)
20        self.roads.append(road)
21        return road
22
23    def create_roads(self, road_list):
24        for road in road_list:
25            self.create_road(*road)
26
27
```

Código Python para creación de intersección

```
simulation.py x
1  from .road import Road
2
3  class Simulation:
4      def __init__(self, config={}):
5          self.set_default_config()
6
7          for attr, val in config.items():
8              setattr(self, attr, val)
9
10     def set_default_config(self):
11         self.t = 0.0
12         self.frame_count = 0
13         self.dt = 1/60
14         self.roads = []
15
16     def create_road(self, start, end):
17         road = Road(start, end)
18         self.roads.append(road)
19         return road
20
21     def create_roads(self, road_list):
22         for road in road_list:
23             self.create_road(*road)
```

Unity





Plan de trabajo

Primera actualización

Introducción de los integrantes del equipo:

Responsables: Todos los integrantes cada uno realizó su introducción.

Entrega: 12 de Noviembre del 2021.

Duración: 15 minutos.

Descripción del proyecto:

Responsable: Santiago Gabian .

Entrega: 12 de Noviembre del 2021.

Duración: 20 Minutos.

Identificación de agentes:

Responsables: Todos los integrantes contribuyeron.

Entrega: 12 de Noviembre del 2021.

Duración: 30 Minutos.

Diagrama de clases:

Responsables: Bruno Passarete.

Entrega: 12 de Noviembre del 2021.

Duración: 10 Minutos.

Diagrama de protocolos:

Responsables: Alan Flores.

Entrega: 12 de Noviembre del 2021.

Duración: 40 Minutos.

Plan de trabajo

Segunda actualización

Diseño de coches e intersección en unity:

Responsables: Todos los integrantes contribuyeron.

Entrega: 18 de Noviembre del 2021.

Duración: 1 Hora.

Completo.

Avance de código de python:

Responsables: Todos los integrantes del equipo.

Entrega: 19 de Noviembre del 2021.

Duración del trabajo empleado: 3 Horas.

Status: Pendiente.

Expectativa de término: 3 Horas adicionales.