



Guía de Instalación - Reto Movilidad Urbana

Jesús Sebastián Jaime Oviedo A01412442

Sergio Hiroshi Carrera Monnier A01197964

Juan Daniel Rodríguez Oropeza A01411625

Jackeline Conant Rubalcava A01280544

William Frank Monroy Mamani A00829796

Premysl Pilar A01760915

Monterrey, Nuevo León, México

Diciembre 03, 2022

Ing. César Raúl García Jacas

Ing. Sebastián Ulises Adán Saldívar

Instituto Tecnológico y de Estudios Superiores de Monterrey

**Modelación de Sistemas Multiagentes con Gráficas Computacionales
(Gpo 301)**

Necesitamos los siguientes programas:

- Python 3.10
- Unity 2021.3.12f

La instalación lleva un aproximado de 5 minutos.

Link del video de instalación

Procedimiento

1. Clonar el repositorio completo de Github, donde se encuentran:
https://github.com/DannyJr08/Reto-Sistemas-Multiagentes_Movilidad-Urbana
2. MESA - Actualmente nuestra simulación de multiagentes se encuentra deployado en línea y puede ser accedido en: <http://chiron.pythonanywhere.com/> (Opcional) Para ejecutar de manera local:

- a. Copiar el código de Jupyter Notebook en un venv o en un archivo main.py
- b. Instalar Flask y en el evento de consulta al endpoint "/" agregar el siguiente código:

- i. al principio del archivo

```
from flask import Flask
from flask import jsonify
```

- ii. al final del archivo

```
app = Flask(__name__)

@app.route('/')
def handle_request():

    positions = []
    fourWayAgents = []
    response = {'agents': [], 'semaphores': [], 'semaphoreTimer': trafficLightTime}
    n_agent = 1
    n_semaphore = 1
    model.step()

    for i in model.schedule.agents:
        if isinstance(i, CarAgent):
            positions.append(i.nextPos)
            response['agents'].append(
                {'id': n_agent, 'loc': coordsToLocation[i.pos]})
            n_agent += 1
        if isinstance(i, FourWayAgent):
            fourWayAgents.append(i.greenLeft)
            fourWayAgents.append(i.greenUp)
            response['semaphores'].append(
                {'id': n_semaphore, 'greenLeft': i.greenLeft, 'greenUp': i.greenUp})
            n_semaphore += 1

    return jsonify(response)
```

- c. Ejecuta el servidor y obtener la dirección y puerto donde se levanta el servidor
 - d. Más adelante en unity, intercambia el link del API a localhost obtenido en el paso anterior
3. UNITY - Crea un proyecto 3D Core.
 4. De ahí importa el .unitypackage , arrastrando el archivo hacia el editor
 5. Palomea todo para que importe todo
 6. ¡Da play!