



Tecnológico de Monterrey

Campus Monterrey

Instrucciones de instalación para solución del reto

Materia

Modelación de sistemas multiagentes con gráficas computacionales (Gpo 4)

Profesores

Edgar Covantes Osuna

Jorge Cruz Duarte

Integrantes

Facundo Vecchi A01283666

Nicolas Cardenas A01114959

Luis Omar Leyva A01570367

David Josué Marcial Quero A00828702

Septiembre 8, 2021

Requerimientos

Programas Requeridos:

Unity 2021.1.16f(Unity / Unity Hub)

Python 3.4+

Los archivos necesarios son los siguientes:

Para empezar las instrucciones primero debemos descargar los archivos ya sea desde Github:

<https://github.com/NicolasCV/ProyectoMultiagentes>

O de la parte de entrega del reto.

El Unitypackage de la simulación gráfica: *VisualizaciónGráficaReto.unitypackage*

La carpeta de archivos de Python: PYTHON

Instrucciones

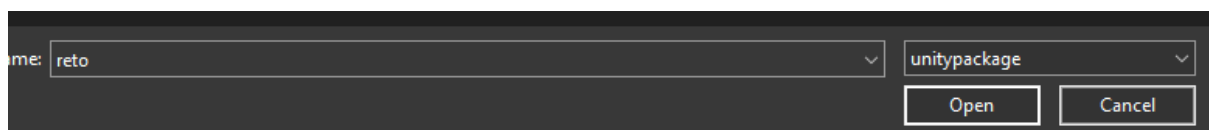
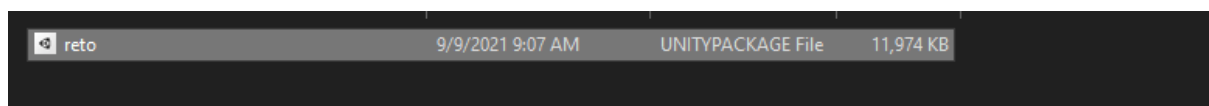
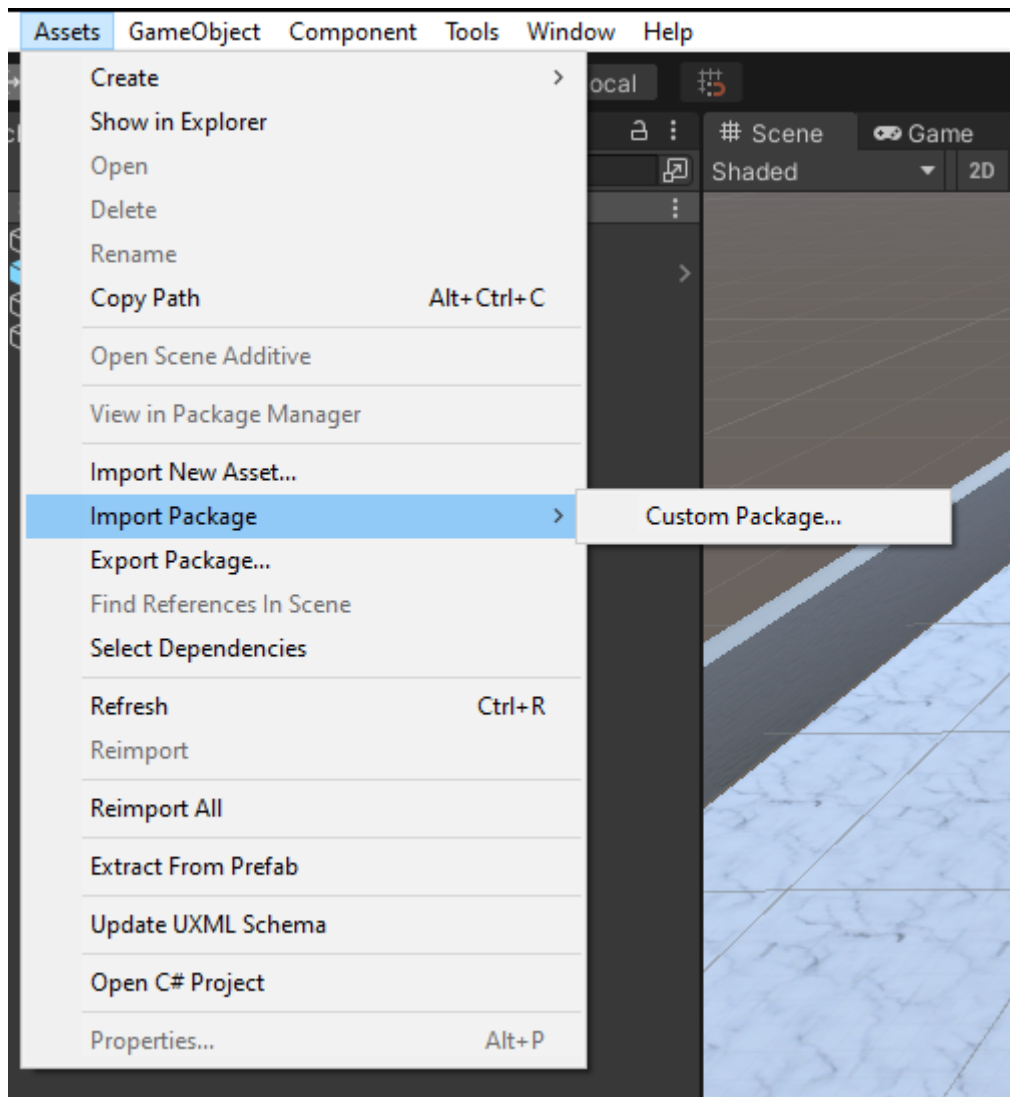
Primero que nada instala el módulo de MESA para python, asumiendo que ya cuentas con numpy y matplotlib, en caso de no tenerlas reemplazar el nombre del paquete en el siguiente comando en la terminal

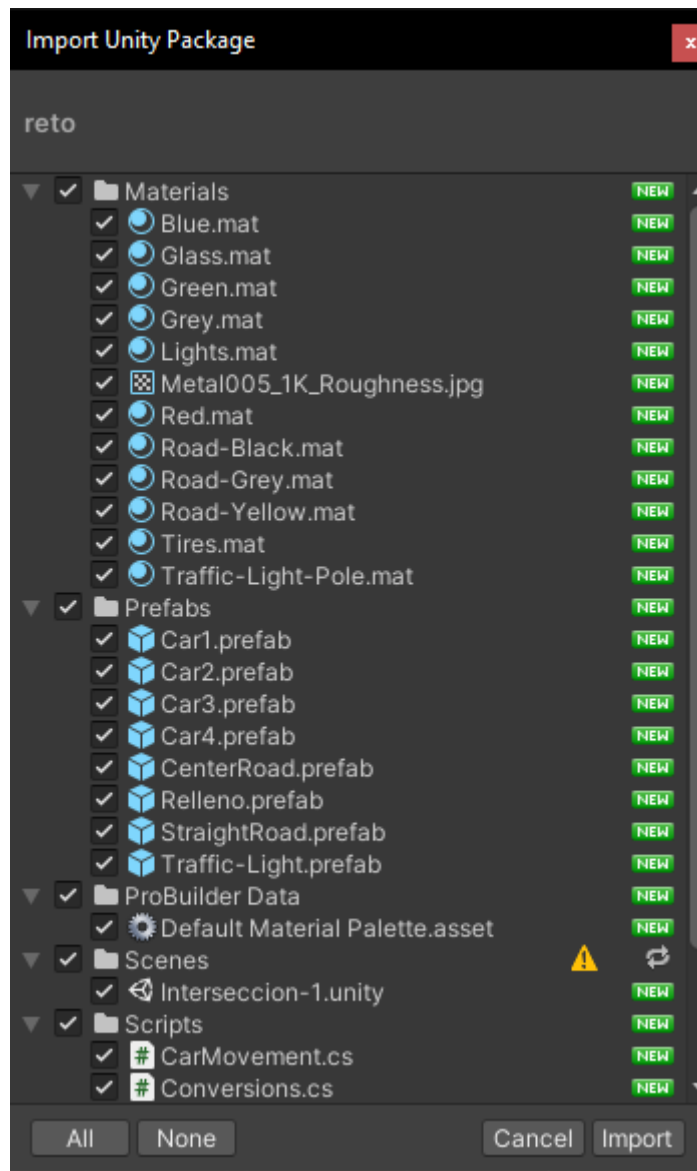
```
>pip install mesa_
```

Después es necesario abrir el archivo server.py dentro de la carpeta. Este es el servidor que le mandará las coordenadas de manera “en vivo” al programa de unity. Cuando corras el programa debe estar como en un modo “standby” que luce como lo siguiente:

```
INFO:root:Starting httpd...
```

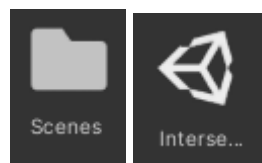
Después abrimos un nuevo proyecto de unity en la versión más reciente posible, pero es recomendado usar la 2021.1.16f por cualquier tema de compatibilidad. Después de haber hecho el nuevo proyecto. En la parte superior izquierda en la parte de *Assets*, le picamos y seleccionamos *Import Package*, aquí es donde viene en juego nuestro unitypackage, lo seleccionamos para importar e importamos todo.



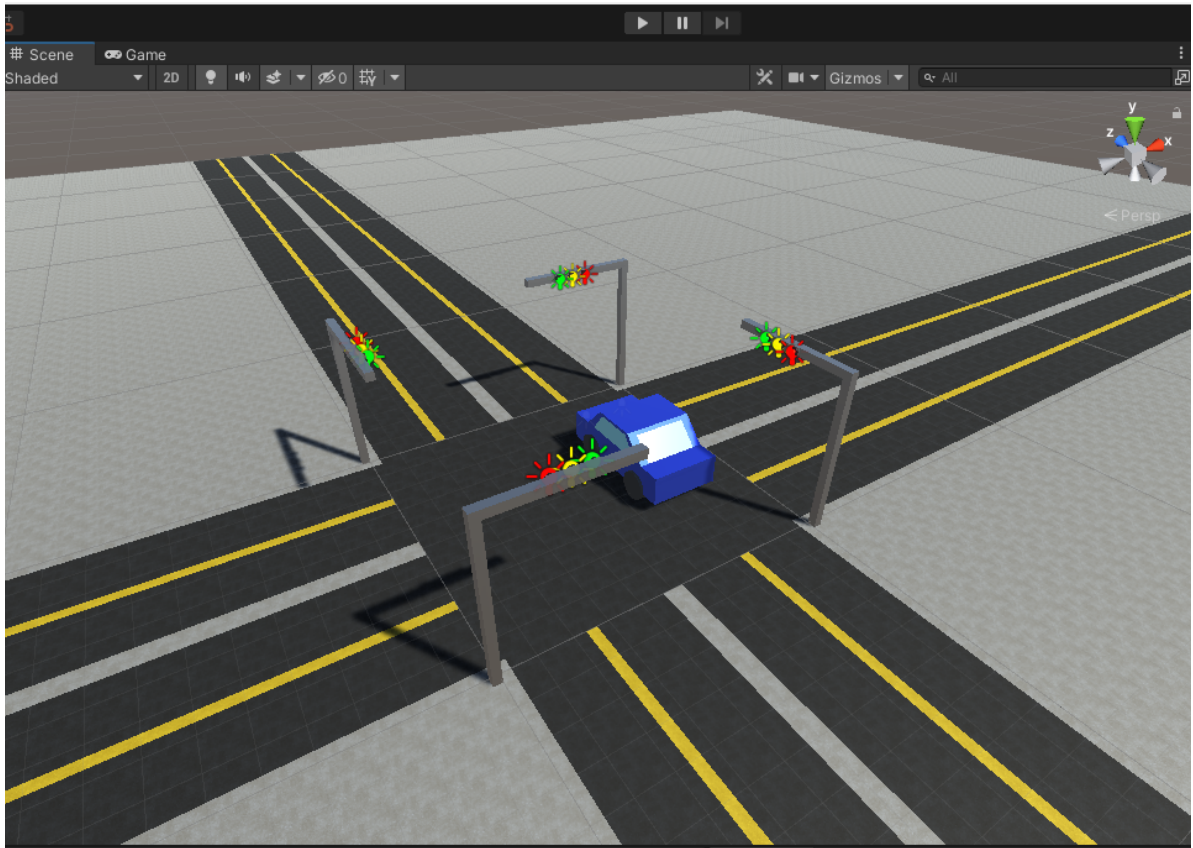


Importamos todo

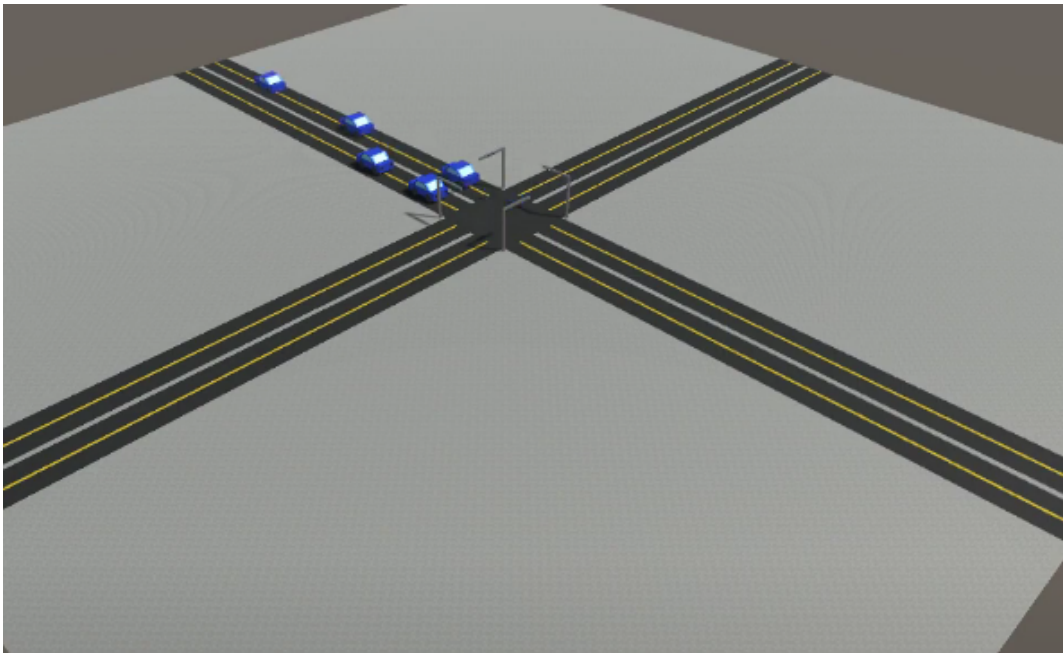
En la sección de *Assets* en la parte inferior nos vamos a la carpeta de *Scenes* y seleccionamos la escena que se llama “Intersección”, le damos doble click para activarla.



La escena debe lucir como la siguiente:



Cuando le damos play, debe lucir de la siguiente manera:



Y listo, la simulación correrá gráficamente.

Notas en casos de error

- El error de “*A Native Collection has not been disposed, resulting in a memory leak. Enable Full StackTraces to get more details.*” es normal, si corre el programa solo hay que ignorarlo
- El servidor de python debe estar pasando objetos que lucen así:

```
Body:
{"x": 3.440000057220459, "y": 0.0, "z": -15.706999778747559}
127.0.0.1 - - [09/Sep/2021 09:37:28] "POST / HTTP/1.1" 200 -
```

- Debe correr el servidor y el proyecto de unity en la red local
- Si es su red es privada pero con algunas restricciones puede que no corra el programa

Plan de trabajo

Actividades:

- ☒ ~~P1. Arranque de Proyecto~~
 - Fecha límite: 13/08/21
 - Responsables: Todos los miembros del equipo.
 - Esfuerzo:
- ☒ ~~P2. Revisión de avance~~
 - Fecha límite: 03/09/21
 - Responsables: Todos los miembros del equipo.
 - Entregables: Modelado del reto y avance del código en Python.
 - Tiempo estimado: 4 horas.
 - Tiempo real: más de 4 horas.
- ☐ P3. Entrega Final
 - Fecha límite: 10/09/21
 - Responsables: Todos los miembros del equipo.
 - Entregables: Modelado del reto, código en Python, simulación gráfica en Unity e instrucciones de instalación del proyecto.
 - Tiempo estimado: 3 días.
 - Tiempo real: 3 días.

Aprendizaje adquirido

P1

Como equipo aprendimos sobre las fortalezas y áreas de oportunidad de cada uno de los miembros del equipo lo cual nos ayudará a delegar las responsabilidades y actividades del proyecto.

P2

Al haber hecho diferentes prácticas como la actividad integradora, contamos con más experiencia tanto como para modelar los agentes pero también para programarlos en un ambiente como este. También obtuvimos experiencia con las actividades en Unity la cual nos servirá para el siguiente avance del proyecto.

P3

Ya con los modelos 3D de Unity creados para el proyecto y con la simulación de Python completa, aprendimos como hacer la conexión entre estas dos partes la cuales hemos estado trabajando durante una gran cantidad de tiempo. En esta fase juntamos todo lo realizado al igual que los documentos que se nos pidieron para realizar la entrega final. Ya contamos con la experiencia de todas las herramientas con las cuales trabajamos (Mesa,Python,Unity) pero creemos que el aprendizaje principal durante esta fase fue como realizar la conexión de la simulación a la representación gráfica mediante un servidor local.