



**Modelación de sistemas multiagentes con gráficas computacionales  
(TC2008B.2)**

**Actividad Integradora**

**Profesor:**

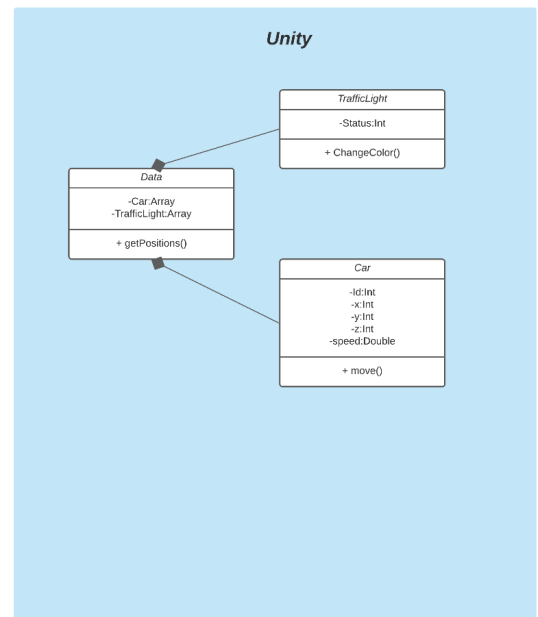
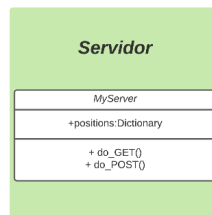
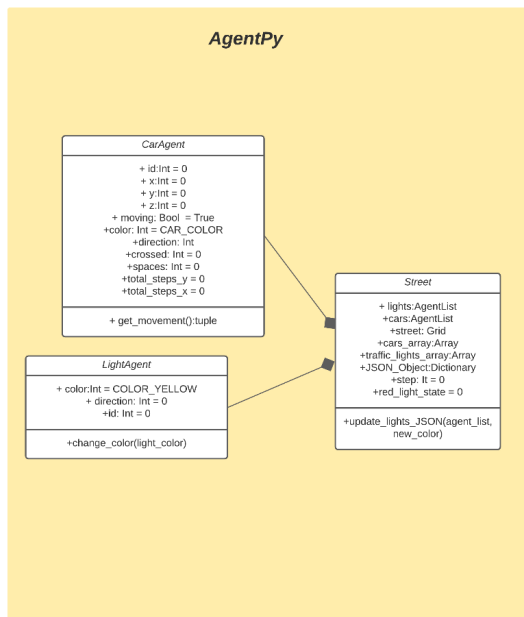
Luis Ricardo Peña Llamas

Guillermo Gabriel Rivas Aguilar

Carlos Iván López Sandoval

Rafael Alfonso Gómez González    A01733983

# Diagrama de clases



Tiempo necesario para que se complete la simulación: todos los carros llegan a salvo al final de la calle.

**50 segundos**

Cuanto tiempo de espera hubo en total en el cruce.

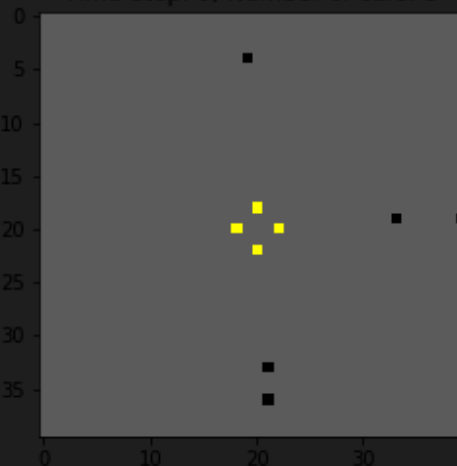
**17 segundos**

Analiza si existe una estrategia que podría disminuir el tiempo dedicado. ¿Cómo sería? Descríbela.

**Una mejor estrategia sería agregar un carril extra en cada sentido para aumentar el flujo de los vehículos. Otra estrategia sería que se agregara algún tipo de rotonda o inclusive se podría agregar un tipo puente para que el tráfico sea en su totalidad fluido y solo dependa de futuras intersecciones.**

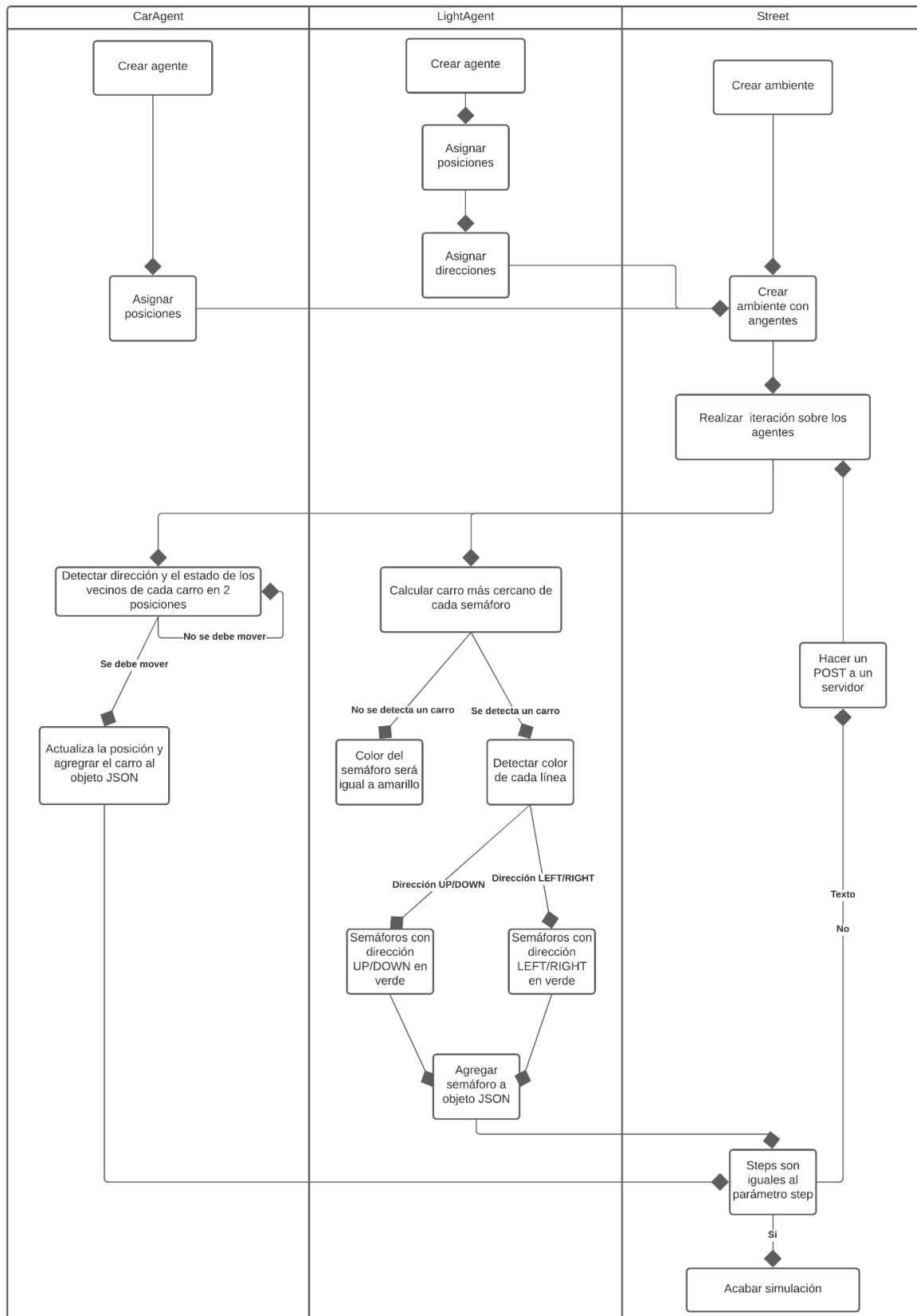
Tiempo en que los carros llegan a salvo a final de la calle: 50 segundos.  
Tiempo de espera total en el cruce: 17 segundos.

Simulation of a Street Intersection  
Time-step: 0, Number of cars: 5



● Once ● Loop ● Reflect

# Protocolo de agentes



## Estrategia cooperativa

Para lograr realizar la entrega fue necesario distribuir el trabajo y los temas relacionados al trabajo tomando en cuenta las fortalezas y debilidades de los integrantes del equipo.

Separamos el reto en tres áreas:

- Unity
- Python
- Cloud Services

En donde Unity involucraría toda la parte de creación de assets, materiales, shaders y scripts para que todo funcionara correctamente.

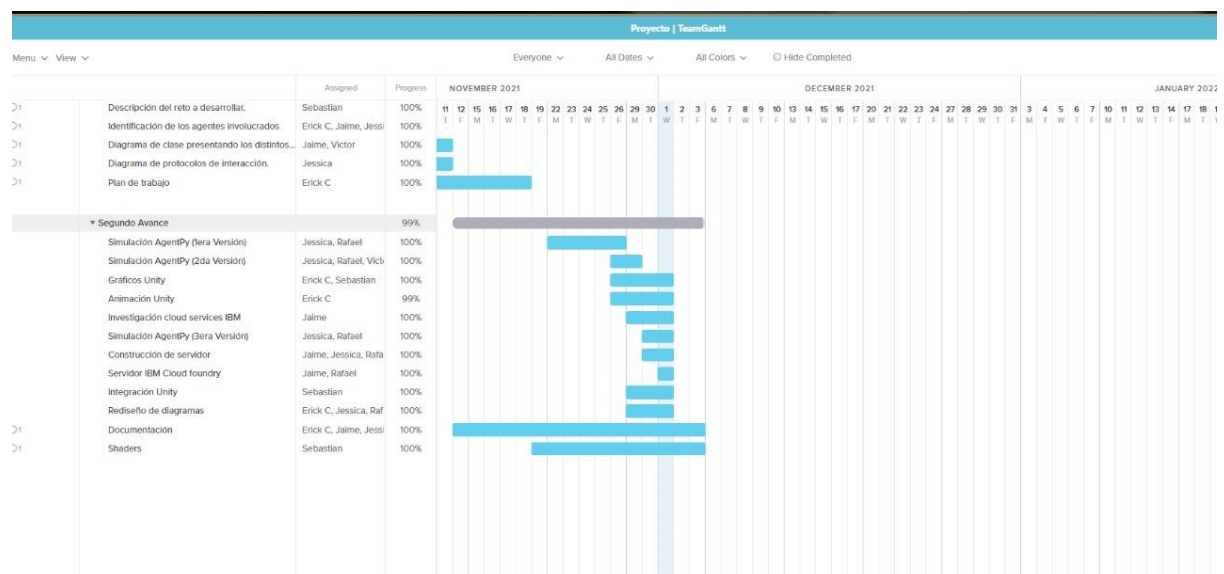
En el área de Python se encargó de la parte del servidor junto con la simulación usando *AgentPy*.

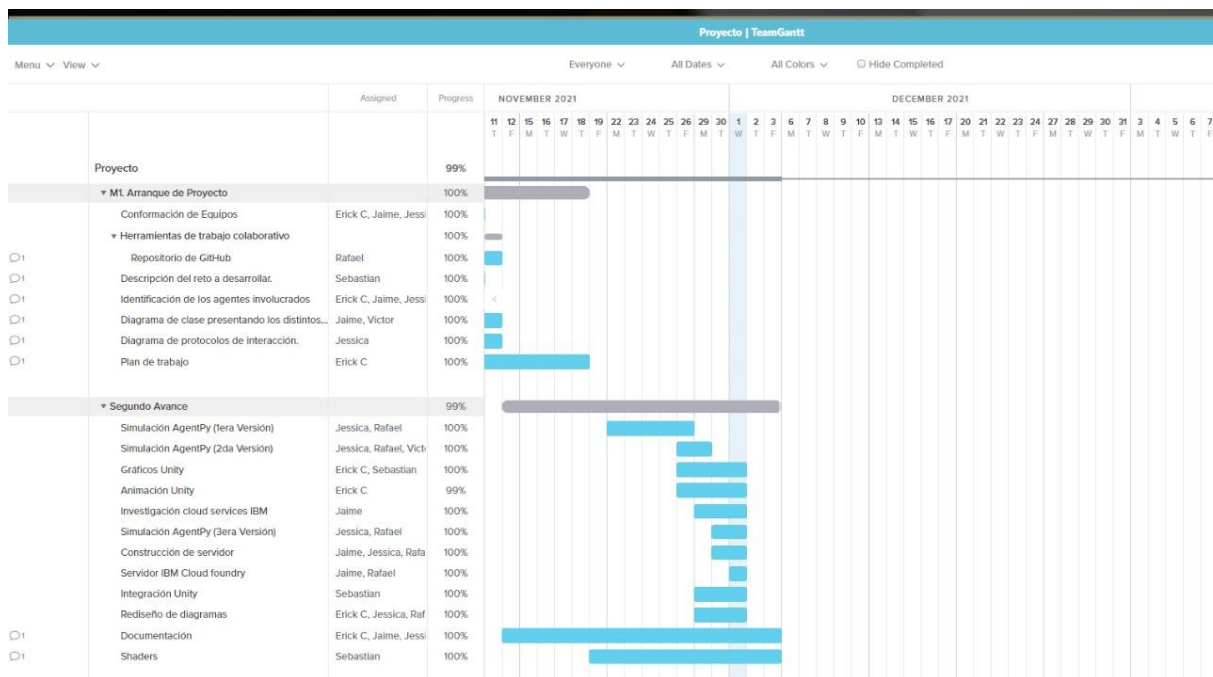
Definiendo ambos alcances empezamos trabajando en equipos repartidos de la siguiente manera:

- Unity: Sebastián y Erick
- Python: Nicole, Victor y Rafael
- Cloud Services: Jaime

Cada día se iban mandando avances junto con reuniones cortas en las que pensábamos sobre la conexión que tendrían ambas partes (frontend - backend), de igual manera se intentaba mejorar la calidad del código por parte de Python y la calidad del diseño en Unity. Siendo sinceros no hubiera sido posible acabar la actividad si no hubiésemos podido trabajar en equipo ya que a algunos se nos dificulta bastante el uso de ciertas herramientas por diversos motivos y el tiempo que le dedicamos a cada área fue bastante elevado a pesar de tener tantos integrantes en el equipo.

A continuación se muestra un diagrama de Gantt en el que tenemos todas las actividades realizadas por todos los integrantes y sus respectivas fechas.





# Reflexión

Al ir realizando el reto aprendí bastante sobre todas las áreas ya que me metí de fondo en cada una de ellas, a pesar de que en algunos momentos parecía imposible realizar la actividad se logró acabar. Entre los temas que más pude reforzar fue el de Python ya que tenía alrededor de un año que no tocaba ningún código en Python y además pude aprender sobre la plataforma de IBM de la que nunca había escuchado y me pareció inclusive mejor que AWS en cuanto a UX ya que todo se encuentra actualizado, lo cual agradezco bastante. Gracias a que todo el equipo ya había trabajado en conjunto previamente fue bastante sencillo organizarnos para las entregas, lo único que nos falló fue la calidad del código, ya que conforme se iban realizando “iteraciones” el código perdía escalabilidad. A pesar de ello al final se consiguió tener un código de calidad aceptable en todas las áreas del proyecto. Me hubiera gustado que nos enfocáramos un poco más en servicios en la nube ya que es un área que me parece fundamental y la que menos he explorado personalmente

# Anexos

[Github](#)