



Tecnológico de Monterrey

Campus Estado de México

Modelación de sistemas multiagentes con gráficas computacionales (Gpo 302)

Evidencia 2: Actividad Integradora

Mayra Fernanda Camacho Rodríguez	A01378998
Víctor Martínez Román	A01746361
Melissa Aurora Fadanelli Ordaz	A01749483
Juan Pablo Castañeda Serrano	A01752030

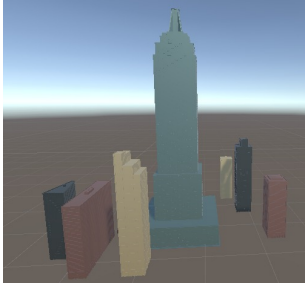
→ Descripción del medio ambiente.

El modelo representa el comportamiento vial de los agentes carros, interactuando con los agentes semáforos, dentro del entorno con los elementos; edificio y calle. Se ve representado por una cuadrícula donde los cuadrados azules representan el edificio, los cuadrados verdes dentro de estos representan los cajones de estacionamiento disponibles para los automóviles. Los autos, representados por el color azul, transitan por las cuadrículas coloreadas de gris. Por último, tenemos unos cuadros en áreas de dos que se encuentran atravesando los cuadros grises y tienen color intermitente. Esos cuadros representan los semáforos que dictan el flujo de movimiento de los coches.

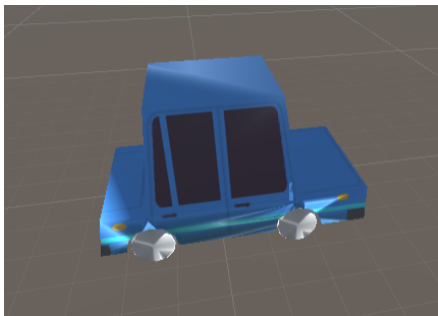
En la representación 3D de este modelo

podemos observar la representación del carro siendo remplazada del cuadro azul al siguiente modelo:

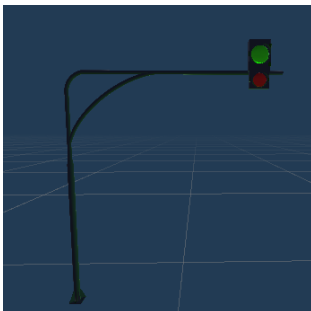
De igual manera, los edificios representados por los cuadros azules claros son representados en 3D por los siguientes modelos:



El punto verde claro que representa a las puertas en la simulación de mesa será presentado por un cuadrado negro que aparecerá en los costados de los edificios.



Por otro lado, los semáforos son representados por este modelo.



Tomando en cuenta las características de un medio ambiente que son:

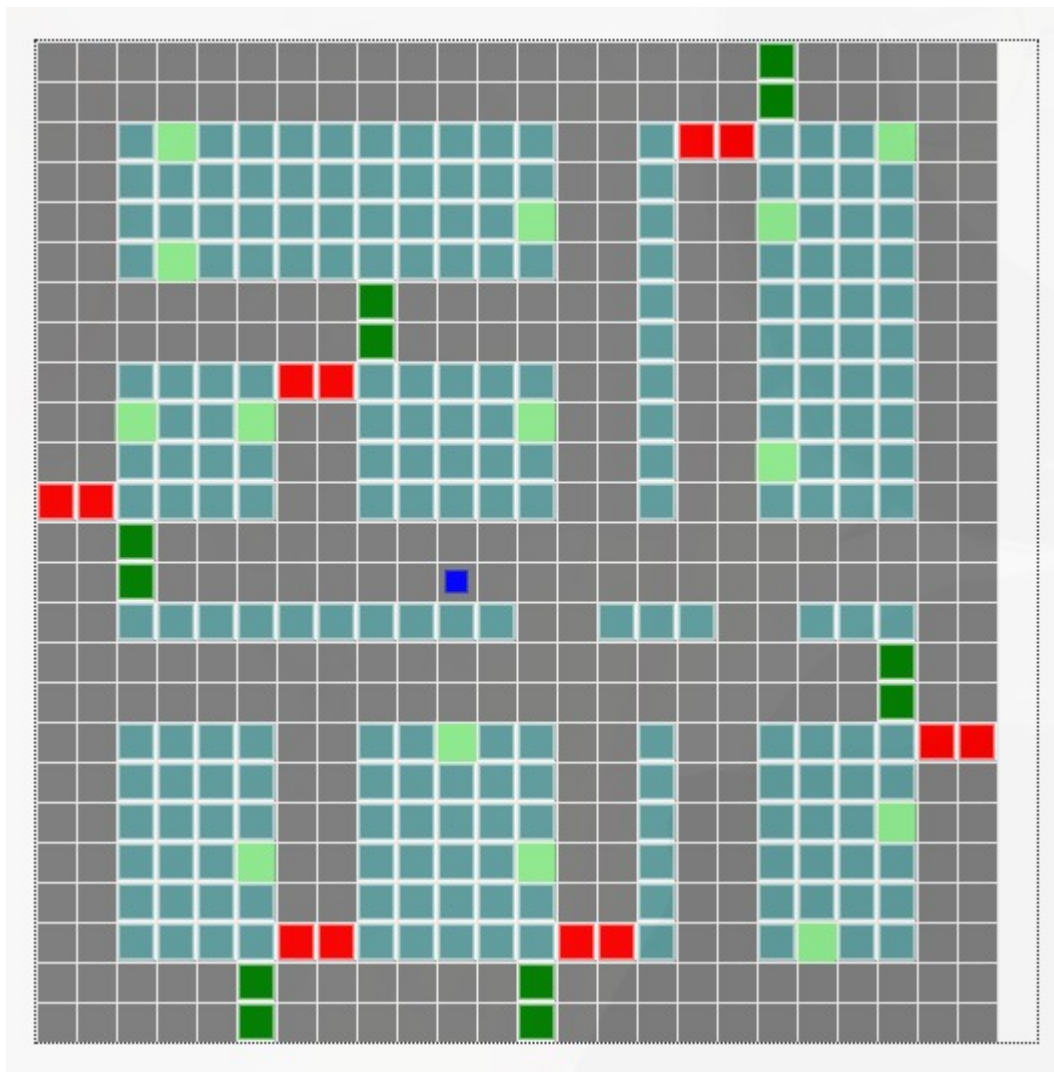
Accesible o inaccesible: Sería accesible, pues los agentes pueden explorar toda la información, y pueden comunicarse entre ellos.

Determinista o no determinista: Sería no determinista, pues no se sabe qué ruta puede tomar un agente carro y sus comportamientos varían de acuerdo a su entorno.

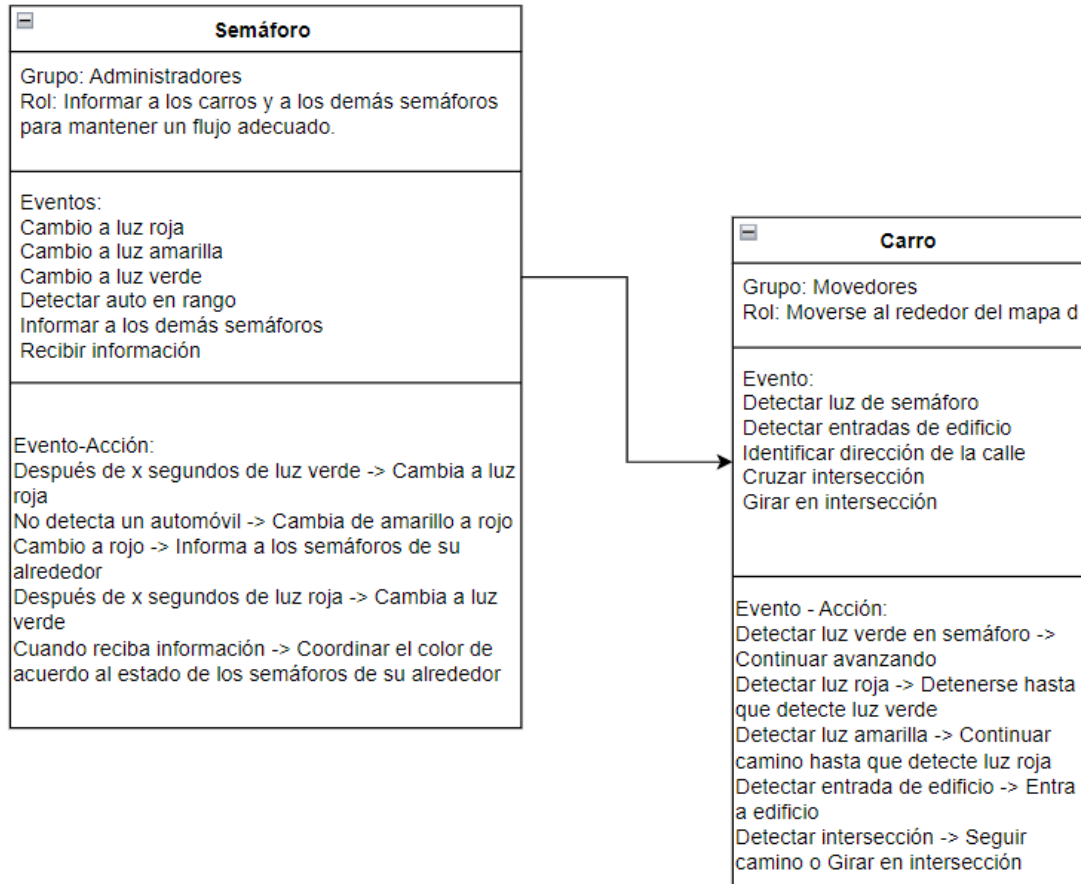
Estático o dinámico: Es un ambiente dinámico, pues los agentes se mueven y realizan acciones.

Discreto o continuo: Es continuo ya que siguen saliendo coches cuando otros lleguen a su lugar:

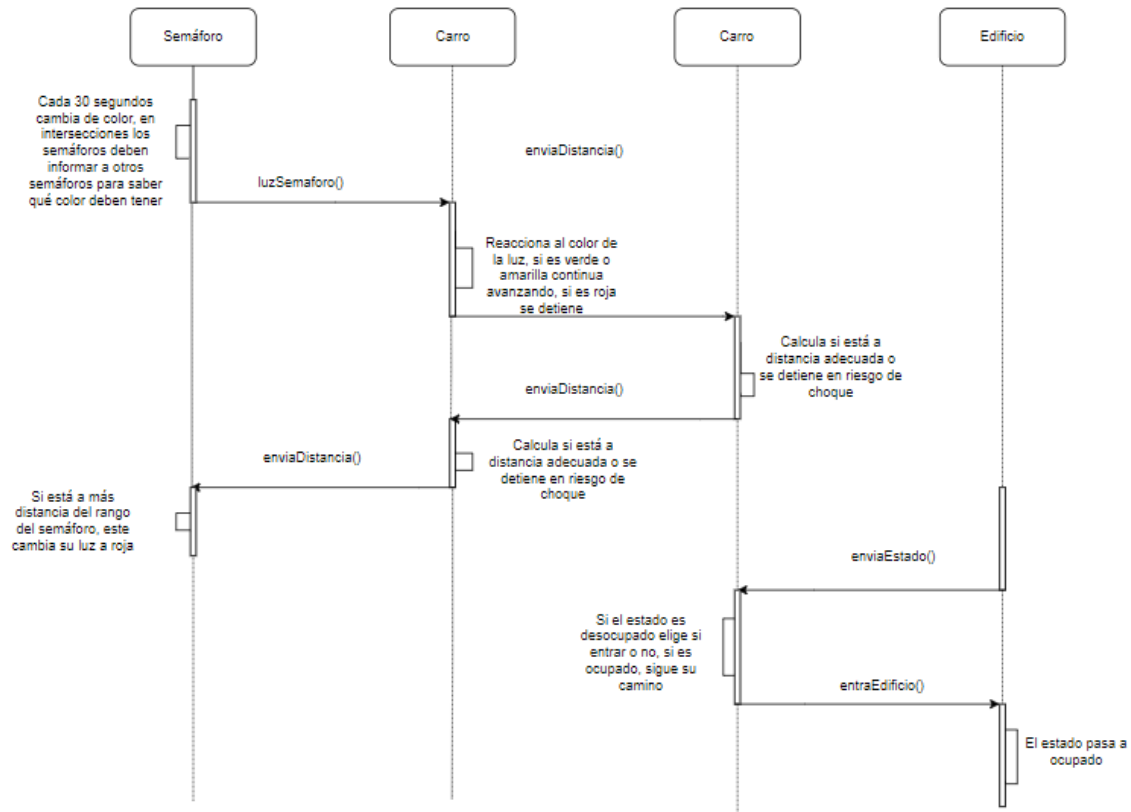
Episódico o Secuencial: El agente es episódico debido a que su experiencia es dada por episodios independientes donde existe una dependencia a la percepción del agente que solo tiene cómo propósito una única acción posterior.



→ Diagramas de Agentes (AUML)



→ Diagramas de Protocolos de Interacción



→ Plan de trabajo y aprendizaje adquirido

Para resolver esta problemática dividimos el problema en las siguientes tareas:

1. Análisis del problema
2. Definición de agentes con sus protocolos, objetivos y comportamientos
3. Planteamiento en mesa
4. Pruebas de funcionamiento en mesa
5. Conexión con flask
6. Pruebas con modelos Dummy en Unity
7. Búsqueda de modelos
8. implementación de modelos personalizados

9. Documentación final

#	Actividad	Duración	Notas
1	Análisis de problema	2hrs 45 minutos	<p>1) Se realizó una junta por medio de Discord donde se definió lo siguiente: tenemos un agente coche y otro agente semáforo que son los únicos capaces de tomar decisiones</p> <p>Los agentes interactúan entre sí cambiando información. El agente coche interactúa también con el objeto edificio.</p> <p>2) El objetivo del agente coche es llegar a un edificio.</p> <p>3) No debería haber dos agentes en una misma casilla</p> <p>4) Las calles delimitan el movimiento de los agentes.</p> <p>6) El agente semáforo debe estar organizado con los otros agentes semáforo para regular el tráfico.</p>
2	Definición de agentes	2hr 30 minutos	Se delimitó el comportamiento de los agentes

3	Definición en mesa	6hr 45 min	Se planteó la solución previamente definida en el paso anterior de los agentes en el modelo, se probó usando la librería de mesa y se comprobó que los datos estuvieran en el formato correcto para la conexión con flask.
4	conexión con flask	3 hr 35 min	Se mandaron los parámetros al modelo. Este regresa los datos de cada agente que son los necesarios para visualizar la problemática.
5	Implementación Unity	4 hrs	Siguiendo el ejemplo del código del profesor, implementamos el código para la visualización en 3D donde pusimos una clase x cada agente.
6	Pruebas	2hr 25 min	Se verificó que estuviera corriendo, se encontró un problema por no implementar la función de update y se corrigió, el programa funcionó satisfactoriamente.
7	Modelos personalizados	15 min	Se buscaron modelos y texturas que estuvieran en el formato obj, fueran gratis y estuvieran acorde a la situación problema. Después se realizaron prefabs de ellos y se implementaron. La aplicación de las paredes también fue hecha en este paso, aplicando las paredes como hijos del escenario de manera que se escalaran con él.
8	Documentación final	2 hrs	Se recopiló la documentación de la situación problema de lo aprendido en un documento de acuerdo a las especificaciones de la rúbrica.

9	Junta de revisión con equipo	45 min	Fue convocada una reunión con el equipo para revisar la entrega y ver las actividades de la semana.
---	---------------------------------------	--------	---

Mayra Fernanda Camacho Rodríguez :

Durante esta actividad tuve la oportunidad de profundizar en el conocimiento de los agentes y sistemas que vimos en clase. Aun cuando, tuve muchas actividades prácticas, como lo fue la tarea integradora, creo que no fue hasta está que gracias a las correcciones y observaciones, además del acompañamiento que tuve por parte de los profesores, que finalmente en esta actividad sentí que entendía todos los conceptos y podía aplicarlos más eficientemente. Siento que profundice más mi conocimiento en cuanto a la toma de decisiones de los agentes en cuanto a su movimiento, que fue algo que anteriormente solo tomamos como aleatorio dentro de los parámetros. Esto me fue útil para poder aprender como dentro de una simulación debe existir una delimitación de aquello que queremos lograr para poder simplificar correctamente las instrucciones que debemos darles a los agentes involucrados. Es ahí donde la documentación fue una herramienta que nos llevó de la mano y sobre todo en esta actividad, terminó siendo bastante útil. Su implementación se sintió como algo natural de nuestro proceso que facilitó nuestro entendimiento e implementación del problema en lugar de algo que teníamos que hacer solo para cubrir un rubro.

Melissa Fadanelli : Para esta actividad fue importante ver que aprendimos sobre los agentes y sus sistemas, que estos sistemas están compuestos por elementos que son capaces de interactuar entre sí y tomar decisiones, de esta manera si los colocamos en un grid podemos darles un ambiente en el que moverse y dedicarles ciertas propiedades que nos ayuden a simular un entorno real. Dentro del área de gráficas pudimos aprender cómo asignarle un ambiente tridimensional con modelos más complejos para representar de una manera más atractiva y clara el ambiente en el que nuestros agentes se mueven.

Victor Martinez Roman:

Aprendí a simular y visualizar un ambiente más complejo donde moldeamos nuestro ambiente conforme las necesidades del modelo que queremos representar, esto por medio de variables que hacen que cada simulación sea distinta con variables aleatorias que personalizan cada ejecución del código. La manipulación del ambiente es según nuestras necesidades y podemos visualizar de mejor manera ciertas situaciones, en este caso el semáforo que gracias a su cambio de color podemos saber el estado actual del agente.

Juan Pablo Castañeda Serrano:

En esta práctica no solo aprendí sobre modelación en 3 dimensiones y la búsqueda de recursos para usar en Unity, sino lo que más me llevo en esta práctica es la perfección de técnicas para modelar agentes. Y saber bien cómo programar sus acciones para que hagan su

comportamiento correcto. En la parte del código también aprendí la diferencia concreta de lo que debe y no debe de ir en el modelo, ya que en el pasado consideraba el modelo solamente como un método para colocar los agentes, y no como un facilitador de su comportamiento; sin embargo, en esta práctica entendí su uso flexible y nos sirvió de bastante para el comportamiento de todos los agentes, y que eventualmente estos conviven de manera armónica en nuestro modelo de Unity final.

LIGA AL REPOSITORIO:

- https://github.com/facebooker212/Reto_Equipo_3