



Tecnológico de Monterrey

Campus Estado de México

Modelación de sistemas multiagentes con gráficas computacionales (Gpo 302)

Evidencia 1: Actividad Integradora

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

→ https://github.com/facebooker212/Actividad_Integradora_1

→ Descripción del medio ambiente:

El medio se compone principalmente de 3 agentes: Robot, Almacenamiento y Cajas. Estos agentes interactúan entre sí para poder simular un escenario donde los robots son los encargados de transportar una caja al lugar adecuado, nuestro medio ambiente tomará una cantidad determinada de agentes caja para también tomar en cuenta el tamaño de almacenamiento. El grid de nuestra simulación representa un espacio grande en el que en determinada posición el agente almacén va a estar ubicado y va a ser inamovible, al final de la simulación el almacén debe tener una caja en cada una de sus posiciones, una vez una caja se encuentre en un espacio de almacén, ya no podrá ser transportada. Esta simulación busca representar cómo funcionan los almacenes de empresas que utilizan robots o maquinaria inteligente para poder acomodar sus productos sin necesidad de intervención humana. En otro escenario se pondrían más anaqueles, diferentes tipos de agente caja con características distintas, pero en este caso es una simulación sencilla donde el único objetivo es que todas las cajas estén en el lugar adecuado.

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

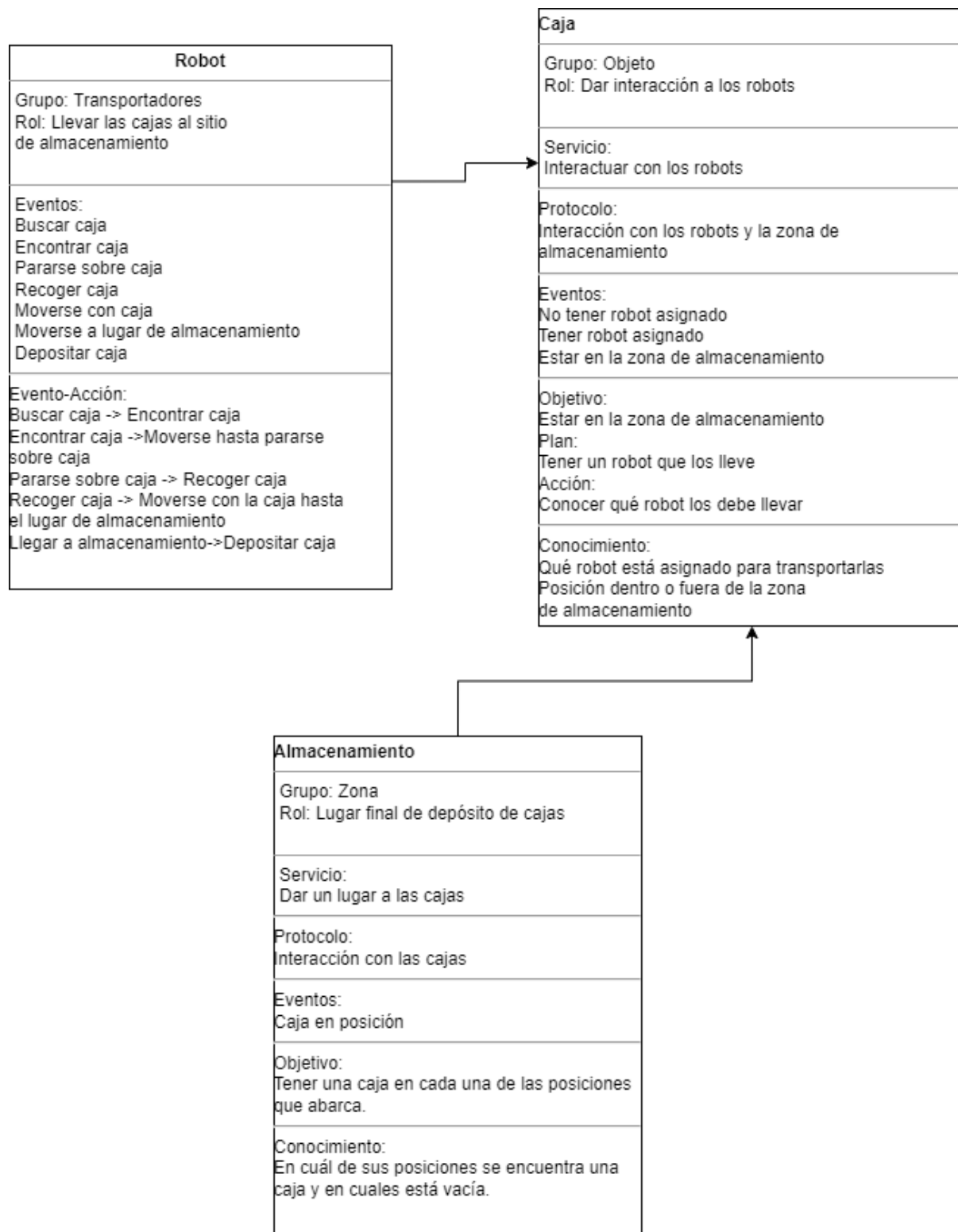
Accesible o inaccesible: Sería accesible pues los agentes pueden explorar toda la información, ya que no hay mucha.

Determinista o no determinista: Sería determinista, pues no hay diferentes salidas para los eventos descritos, si el robot encuentra una caja, la recoge y la lleva a donde debe.

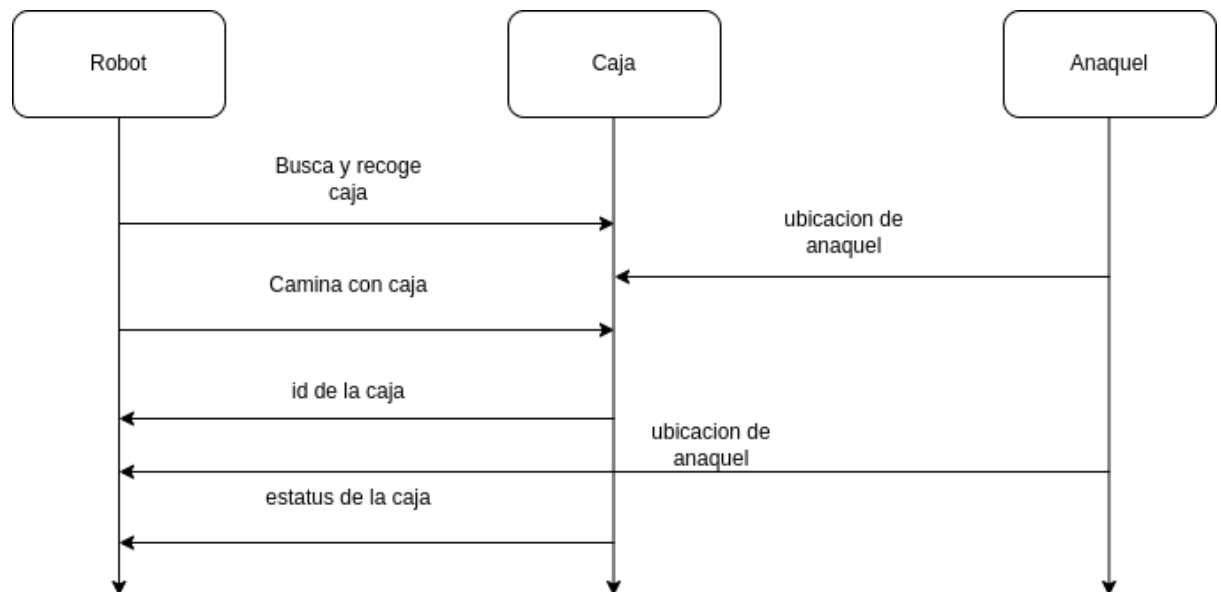
Estático o dinámico: Es un ambiente estático, pues solo cuando los agentes se mueven y realizan acciones este cambia, si no, el ambiente permanece igual.

Discreto o continuo: Se podría considerar discreto ya que como hay una cantidad determinada de cajas, una vez que las cajas estén en su lugar, ya no hay más eventos que realizar.

AUML



→ Protocolos de agentes



→ Estrategia cooperativa:

El diseño y repartición de trabajo de esta actividad lo hicimos a partir de la semana pasada. El día 14 de noviembre del 2022 tuvimos una junta donde checamos la descripción y rúbrica de la actividad integradora, a partir de ahí definimos nuestro plan de trabajo en estas etapas:

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	Fechas	Duración	Notas
1	Análisis de problema	14/11/2022	2hrs 45 minutos	<p>1)Se realizó una junta por medio de Discord donde se definió lo siguiente tenemos un agente de robot, el cual podría moverse en el plano, un cuadro , dicho plano es adaptable dependiendo de la cantidad de agentes. El robot también debería poder mover una caja a la vez y situarla en algún contenedor cercano, ubicado a los lados del mapa.</p> <p>2)El movimiento del agente debería de ser aleatorio hasta encontrar una caja.</p> <p>3)El contenedor a los lados solo debería de aguantar un máximo de 5 cajas.</p> <p>4) El robot no se mueve diagonalmente.</p> <p>5)El modelo del robot deberá ser hecho un prefab para acomodarse al código</p> <p>6) deberán existir paredes para asegurar que no se caigan los modelos.</p>
2	Definición de agentes	15/11/2022	2hr 30 minutos	Se investigó cómo realizar el comportamiento de los agentes: la caja y el robot dentro del escenario propuesto en clase.
3	Definición en mesa	15/11/2022	6hr 45 min	Se planteó la solución previamente definida en el paso anterior de los agentes en el modelo, sé 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	18/11/2022	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	19/11/2022	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	19/11/2022	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	20/11/2022	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 cómo hijos del escenario de manera que se escalaran con él.
8	Documentación final	20/11/2022	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	21/11/2022	45 min	Fue convocada una reunión con el equipo para revisar la entrega y ver las actividades de la semana.
---	------------------------------	------------	--------	---

