



# Tecnológico de Monterrey

**Instituto Tecnológico de Estudios Superiores de Monterrey  
Campus Monterrey**

**Modelación de sistemas multiagentes con gráficas computacionales (Gpo 302)**

## Reporte de Reto

David Alberto Salinas Escalera	A00838212
Danaé Sánchez Gutiérrez	A00836760
Victoria Valentina Marín Domínguez	A00836008
Diego Torre Damm	A01571298
Yolanda Julieta Cantú Macías	A01285390

### **Profesores:**

Dr. Jorge Mario Cruz Duarte

Dr. Jose Eduardo Ferrer

Dr. Luis Guerrero Bonilla

Monterrey, Nuevo León, a 9 de noviembre de 2024

# Índice

<b>Índice</b>	<b>2</b>
1. Conformación del Equipo	3
2. Herramientas de Trabajo Colaborativo	5
3. Propuesta Formal del Reto	6
3.1. Descripción del Reto	6
3.2. Identificación de Agentes	6
3.3. Diagrama de Clases	9
3.4. Diagrama de Protocolos de Interacción	11
3.5. Plan de Trabajo	11
3.6. Aprendizajes Adquiridos	12

## 1. Conformación del Equipo

En esta sección definimos el equipo de trabajo y las expectativas y roles de cada miembro:

### **Victoria Marín, A00836008.**

Como fortalezas reconozco el diseño de interfaces gráficas intuitivas y visualmente agradables, además de desarrollo en Unity y manejo de assets, escenarios y otras funcionalidades, en la parte de otras habilidades, considero que soy bastante responsable y que manejo una buena comunicación con el equipo. En proyectos anteriores he trabajado con Unity completamente, por lo que considero que es una de las mayores aportaciones que puedo dar al equipo y a este proyecto, y como área de oportunidad es mejorar en Python y explorar librerías nuevas y sus funciones, ya que no es el lenguaje que más uso, sin embargo es una meta que tengo para esta clase.

Mis expectativas para esta materia es aprender más a nivel técnico y aplicado el manejo de threads y procesos, además de implementación de agentes en Python y conexiones con Unity, mi meta o visión para esta clase es poder aplicarlo a nivel individual y poder integrarlo para mejorar el proyecto, además de que me gusta la idea de poder probarlo en el laboratorio con un robot real. Me comprometo a entregar mis responsabilidades o tareas con la mejor calidad posible y a tiempo y forma, además a participar activamente en codificar y diseñar agentes y la parte con la que más pudiese aportar es con el diseño de escenario e implementación en Unity.

### **Danaé Sánchez Gutiérrez, A00836760.**

Como fortalezas uno de los lenguajes de programación donde me siento más cómoda al desarrollar es Python, ya que considero su sintaxis como una de las más sencillas de entender por su simplicidad. Gracias a otra Unidad de Formación ya he trabajado con Unity, donde destacaba por el diseño de interfaces creativas al diseñar videojuegos, así como el manejo de assets como sonido, colores, sprites, etc. Además, soy una persona responsable que le gusta mantener un orden con lo que realiza, además de que mantengo una comunicación activa con mi equipo para tener en claro en todo momento las responsabilidades de cada miembro y saber cómo se interrelacionan las partes para así obtener resultados satisfactorios.

Mis expectativas de esta Unidad de Formación se centran en el diseño de escenarios de simulación con agentes, en este caso los robots, así como el control de ellos a través del código que realicemos en Unity. El entender lo que está pasando y el por qué del comportamiento de los agentes, así como identificar las necesidades en base al entorno y poder ponerlo a prueba de manera física con el uso del Robotario, ya que es una oportunidad atractiva y de la cual espero aprender mucho, pues ahí podremos observar si es que realmente nuestra simulación funciona en un entorno real. Me comprometo a cumplir con mis responsabilidades, así como trabajar en equipo de manera equitativa para que todos aprendamos sobre el manejo de multiagentes y podamos desarrollar todas las competencias necesarias involucradas en esta Unidad de Formación.

**David Alberto Salinas Escalera, A00838212**

Una de mis fortalezas es el dominio del lenguaje de Python, en el cual ya he trabajado con él durante un tiempo bastante considerable para llamarme bueno con él, desarrollando proyectos y diferentes retos que han fortalecido mis habilidades de programación y resolución de problemas. Otra fortaleza que considero que tengo es Unity, tengo un buen nivel de experiencia usándolo, ya que en proyectos pasados me encargue de la mayoría del desarrollo en esta plataforma, mientras que otros miembros del equipo se enfocaron más en la creación de la página web. Como área de oportunidad, reconozco que no tengo experiencia en machine learning, ni en campos relacionados con inteligencia artificial, aunque tengo un gran interés por aprender y desarrollarme en estos temas en el futuro

Mis expectativas para esta clase son altas, espero desarrollar habilidades en planificación de rutas, gestión de tareas y administración de recursos energéticos. Además también espero mejorar mis habilidades en unity haciendo visualizaciones en 3D de operaciones logísticas, simulando el entorno del ambiente y los movimientos de los robots. Como equipo espero fomentar un ambiente de apoyo y cooperación donde todos contribuyan al cumplimiento de los objetivos. Me comprometo a ser una persona responsable, entregando todo lo que diga que voy a hacer en tiempo y forma. También a trabajar como equipo de manera equitativa y de manera respetuosa, intentando ayudar a los demás de mi equipo con lo que necesiten.

**Diego Torre Damm A01571298**

Cuento con las fortalezas de tener experiencia desarrollando aplicaciones robustas y de alto impacto en el lenguaje python. Cuento con un conocimiento base conceptuales de machine learning, además de haber aplicado estos conocimientos para crear modelos anteriormente y tener la capacidad de aplicar estos conocimientos en ejemplos prácticos. Como áreas de oportunidad, considero que puedo desarrollar mayormente mis habilidades para trabajar con herramientas como unity para crear simulaciones de alto impacto en casos reales.

Mis expectativas para esta unidad de formación son aprender a mayor profundidad de machine learning y elaboración de sistemas integrando agentes inteligentes. Me comprometo a ofrecer y elaborar la mejor solución posible de la mano con mi equipo de trabajo. También a integrar y aplicar los conocimientos adquiridos.

**Yolanda Cantú Macías, A01285390**

Considero que una de mis fortalezas es el pensamiento analítico y el reconocimiento de patrones y comportamientos en diversas aplicaciones de la ciencia y la tecnología como el análisis estadístico, ecuaciones diferenciales, modelación de comportamientos y fenómenos físicos, entre otros. Principalmente mis intereses van de la mano con el razonamiento analítico y lógico matemático. Además tengo experiencia en diferentes áreas artísticas como

diseño, dibujo, animación, pintura, entre otros los cuales he aplicado anteriormente en la resolución de otros retos y situaciones problema a lo largo de los estudios de mi carrera. Igualmente tengo experiencia con programación orientada a objetos en python y c++ al igual que he trabajado anteriormente en unity y en creación de simulaciones principalmente de modelos y ecuaciones usando matlab y python.

Respecto a las áreas de oportunidad, reconozco que me hace falta más experiencia en la parte del uso de modelos de aprendizaje para la simulación y modelado de sistemas interconectados con múltiples agentes y factores. Otra área de oportunidad en la que me quiero enfocar es en desarrollar soluciones por medio de la integración de diversas herramientas y lenguajes de programación de manera eficiente para entregar un producto final que aproveche al máximo múltiples tecnologías.

Finalmente tengo como expectativas para este bloque el poder aprender más sobre los diferentes modelos de aprendizaje, cómo estos son aplicados y modificados de acuerdo a las necesidades del problema o del cliente. Además, espero poder aprender más sobre la incorporación de diversas tecnologías y herramientas en un producto innovador y diferenciador. Me comprometo a trabajar en conjunto con mi equipo para poder llevar a cabo el proyecto de la mejor manera trabajando equitativamente y colaborativamente.

## **2. Herramientas de Trabajo Colaborativo**

En esta sección adjuntamos la liga al repositorio de Github y de ser necesario algún otro anexo.

<https://github.com/diegodamm83/Reto-Multiagentes-EQ3>

Liga a diagramas de estado de cada agente:

[https://drive.google.com/file/d/1VL-YuLH8MygBB\\_PYKATt5QYxWzEJK9Oz/view?usp=sharing](https://drive.google.com/file/d/1VL-YuLH8MygBB_PYKATt5QYxWzEJK9Oz/view?usp=sharing)

### 3. Propuesta Formal del Reto

#### 1. Descripción del Reto

Este reto consiste en desarrollar un sistema multiagente capaz de gestionar de manera autónoma el flujo logístico en el almacén de una compañía abandonada que ha dejado de usar las instalaciones. La compañía quiere retomar el espacio abandonado y hacer que el sistema coordine varios robots autónomos para optimizar el manejo de pallets desde la cinta transportadora de la entrada, hacia los racks de almacenamiento y posteriormente hacia los racks de salida. Estos robots deben poder detectar y evitar obstáculos, planificar rutas óptimas y gestionar sus niveles de batería de forma autónoma, ajustando su velocidad y desplazamiento en función de las condiciones del entorno.

La simulación debe mostrar la operación del almacén en su totalidad, generando informes de eficiencia que permitan evaluar el rendimiento del sistema en aspectos clave como el tiempo de espera de los pallets y la ubicación de los racks. Además, se requiere una visualización en 3D que muestre detalladamente la disposición del almacén, el movimiento de los robots y sus interacciones con otros elementos, como racks, obstáculos, paredes, etc.

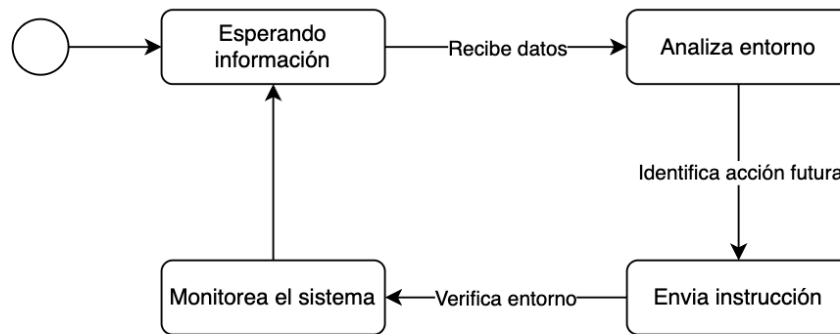
La solución, debe resolver varios desafíos de optimización logística, incluyendo el enrutamiento de vehículos, la programación de tareas, y la administración eficiente de energía para cada robot. Con estos objetivos el sistema que nos pide la empresa será una solución eficiente y práctica para la gestión de un almacén, maximizando el uso de recursos de la planta y asegurando el flujo continuo y eficiente de los productos almacenados.

#### 2. Identificación de Agentes

Los agentes que estamos considerando para este proyecto son los siguientes:

- **Master o administrador:**
  - Este agente se clasifica como basado en utilidad, donde actúa como evaluador o supervisor del ambiente y los agentes involucrados, al obtener la situación o estado actual determina las acciones que se deben tomar para que todo funcione con el performance esperado, comunicándose así con los robots, cargadores y los agentes presentes con instrucciones sobre su comportamiento.
  - Su performance se basa en que tan eficientes son las instrucciones que proporciona o comunica a los demás agentes y cómo esto impacta al ambiente, negativa o positivamente.
  - Sus sensores se componen de una cámara, los estados o recibir comunicaciones de otros agentes.
  - Sus actuadores son el medio por el que transmiten a los demás agentes las instrucciones.

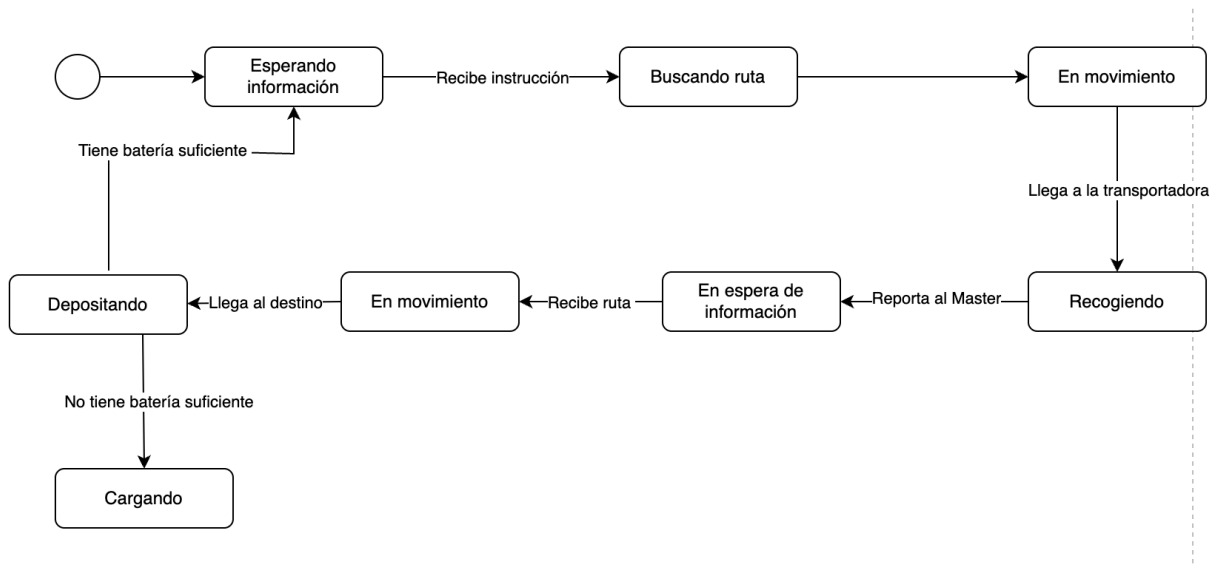
### Diagrama de estado:



#### ● Robots (LGVs):

- Los robots son agentes basados en objetivos, donde su meta es poder llevar todas las cajas a un estante válido, sin chocar entre ellos y tomando en cuenta el espacio disponible en cada estante.
- Utiliza sensores ultrasónicos y láser para detectar los obstáculos, otros robots y espacio.
- Sus actuadores son las garras o transportadores que recogen y depositan las cajas, además de ruedas omnidireccionales que le permiten moverse en cualquier dirección.
- Su rendimiento está determinado por la cantidad de cajas que puedan ser transportadas sin incidentes y además se puede considerar el tiempo en el que lo hacen.

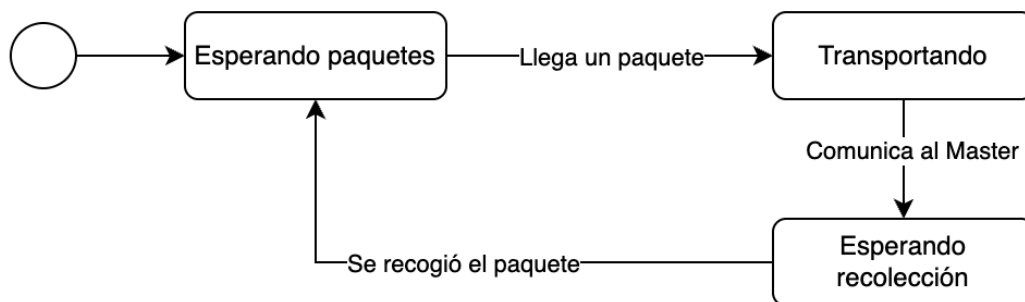
### Diagrama de estado:



- **Transportadora:**

- La transportadora es un agente reactivo simple y su función es recibir las cajas de entrada y posicionarlos para que los robots los recojan, si detecta una caja se mueve, de lo contrario no.
- Como sensores tiene detectores de presencia de las cajas, para controlar el flujo, estos pueden ser sensores de peso.
- Su actuador es la cinta transportadora que posiciona los paquetes en la zona para recogerlos.
- Su performance está basada en la eficiencia con la que entrega las cajas.

**Diagrama de estado:**

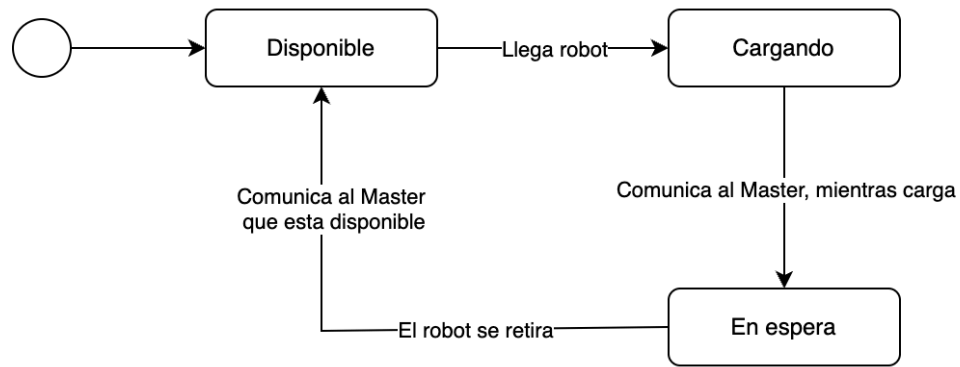


- **Cargadores:**

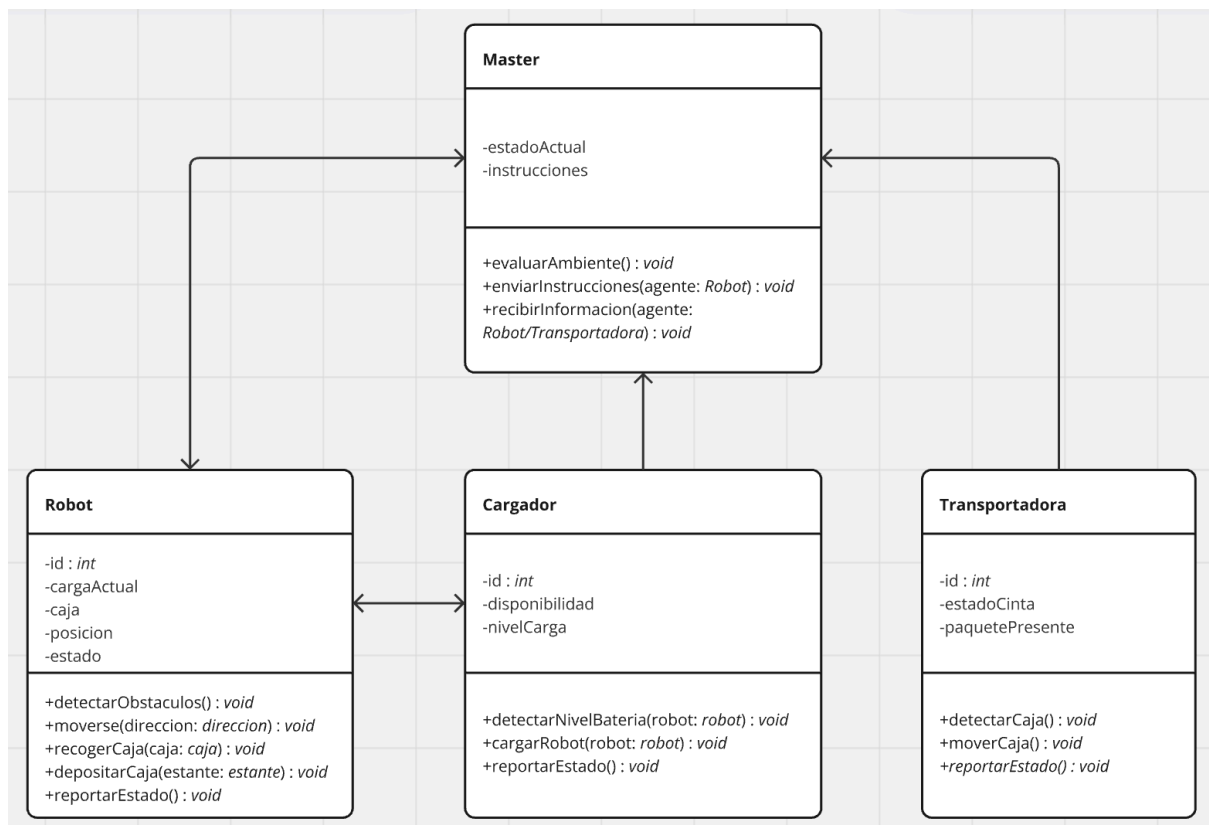
- Los cargadores son agentes basados en metas y modelos, ya que reaccionan a la presencia de los robots con bajo nivel de batería, y distribuyen la carga basándose en el modelo que tienen los robots de carga-descarga y su batería. También controlan el estado de los robots y el suyo con disponibilidad.
- Tiene sensores de proximidad y comunicación de estados de la batería de los robots, también sensores de disponibilidad.
- Sus actuadores son los conectores de carga, que permiten que el robot se cargue a la batería necesaria.
- Su performance se basa en cómo los robots mantienen la batería en relación a las tareas y tiempo, es decir que puedan optimizar la carga para rendir con el menor número de interrupciones posibles.

**Diagrama de estado:**





### 3. Diagrama de Clases



#### Master (Administrador):

- **Atributos:**
  - **estadoActual**: Estado actual del entorno y los agentes.
  - **instrucciones**: Lista de instrucciones para cada agente.
- **Métodos:**
  - **evaluarAmbiente()**: Evalúa el estado actual del ambiente y determina las acciones.

- `enviarInstrucciones(agent)`: Envía instrucciones a un agente específico.
- `recibirInformacion(agent)`: Recibe datos del estado actual de otros agentes.
- **Interacciones**: Se comunica directamente con **Robots**, **Cargadores**, y **Transportadora** para supervisar y dar instrucciones.

#### Robot (LGV):

- **Atributos**:
  - `cargaActual`: Nivel de batería actual.
  - `caja`: Caja actual transportada (si aplica).
  - `posicion`: Posición actual en el espacio.
  - `estado`: Estado de movimiento o transporte.
- **Métodos**:
  - `detectarObstaculos()`: Usa sensores ultrasónicos y láser para detectar obstáculos y otros robots.
  - `moverse(direccion)`: Actúa sobre sus ruedas omnidireccionales para moverse.
  - `recogerCaja(caja)`: Usa su actuador para recoger una caja.
  - `depositarCaja(estante)`: Deposita la caja en un estante adecuado.
  - `reportarEstado()`: Reporta su estado actual al **Master**.
- **Interacciones**:
  - Se comunica con el **Master** para recibir instrucciones y reportar su estado.
  - Puede interactuar con **Cargadores** cuando necesita recargar.

#### Transportadora:

- **Atributos**:
  - `estadoCinta`: Estado de la cinta (moviéndose o detenida).
  - `paquetePresente`: Indica si hay un paquete en la cinta.
- **Métodos**:
  - `detectarCaja()`: Usa sensores para verificar la presencia de una caja.
  - `moverCaja()`: Activa la cinta transportadora para mover el paquete a la posición de recolección.
  - `reportarEstado()`: Comunica su estado al **Master**.
- **Interacciones**:
  - Reporta al **Máster** si se encuentra en movimiento o en espera de un paquete.

#### Cargador:

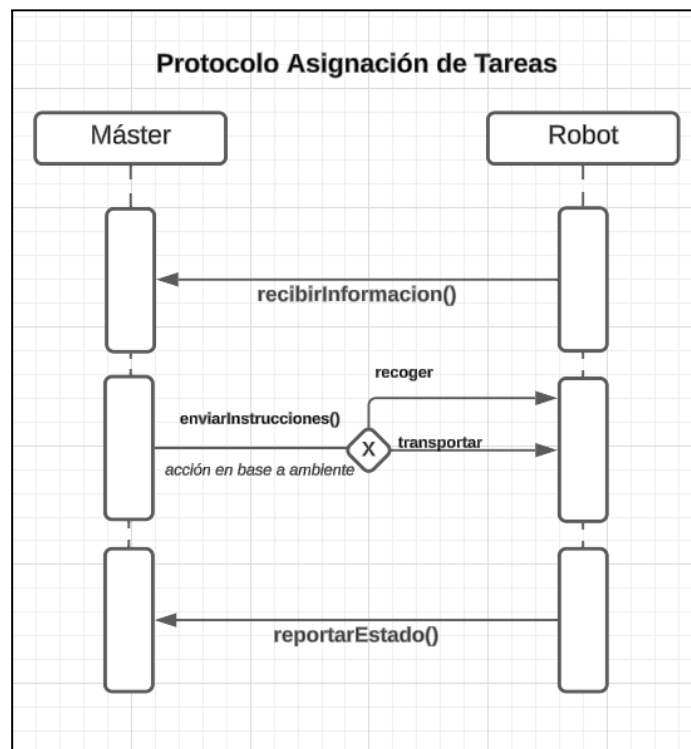
- **Atributos**:
  - `disponibilidad`: Estado actual de disponibilidad del cargador.

- **nivelCarga**: Nivel de carga que puede proporcionar.
- **Métodos**:
  - **detectarNivelBateria(robot)**: Revisa el nivel de batería de un robot cercano.
  - **cargarRobot(robot)**: Usa el conector de carga para recargar un robot.
  - **reportarEstado()**: Comunica su disponibilidad al **Máster**.
- **Interacciones**:
  - Envía información al **Máster** sobre su disponibilidad.

#### 4. Diagrama de Protocolos de Interacción

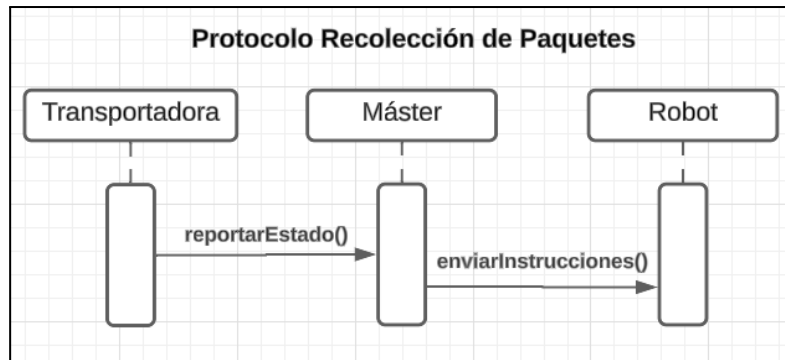
##### Protocolo de asignación de tareas

Protocolo de interacción entre los agentes administrador/máster y robots. Un robot envía información por medio de su método reportar estado, información que es recibida por el máster por medio del método recibir información. Posteriormente el administrador evaluará el ambiente y en base a ello asignará una de dos instrucciones: recoger un paquete, transportar un paquete a su destino. El robot realizará las instrucciones dadas por medio de los métodos de moverse, recoger caja y detectar obstáculos y cuando termine dicha acción se reportará nuevamente el estado de este al administrador.



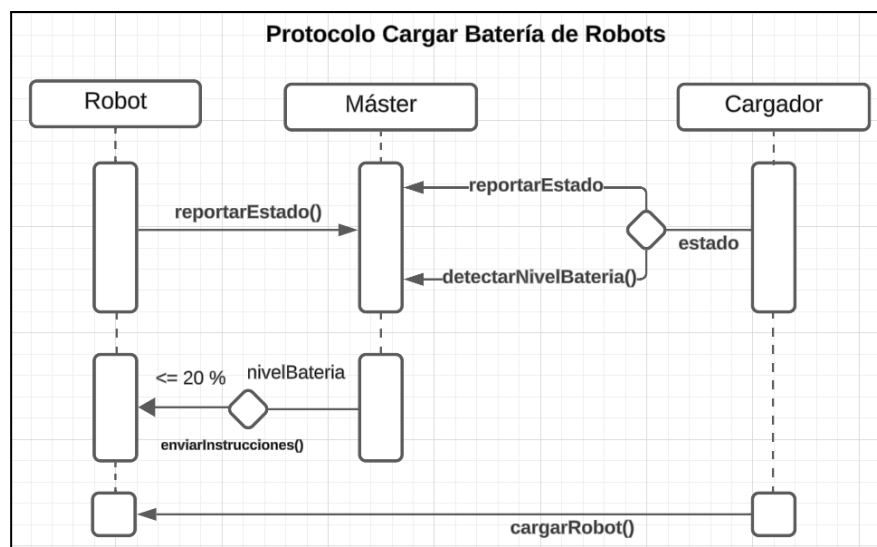
##### Protocolo de recolección de paquetes

Protocolo de interacción entre los agentes de transportadora, máster y robot. Inicialmente el agente de transportadora reportará su estado al administrador, con la información recibida, el administrador evaluará el ambiente considerando los robots disponibles y posteriormente enviará instrucciones al robot elegido para recoger el paquete, enviando las instrucciones por medio del método enviarInstrucciones.



### Protocolo cargar batería de robots

Protocolo de interacción entre los agentes de robot, máster y cargador. Inicialmente los agentes de cargador y robot enviarán información al máster para que este pueda evaluar el ambiente. En el reportar estado del robot se toma en cuenta que se envía información de su estado actual de batería. Simultáneamente en el periodo inicial de interacción entre los agentes, el cargador puede hacer de cero a dos cosas, se declara una condicional de or inclusivo que depende del estado del cargador de manera que se llamen únicamente los métodos necesarios dependiendo de si el cargador está en uso o no. El método de detectarNivelBateria se envía únicamente en los casos donde el estado del cargador es disponible y se usa para comunicarle al máster que puede enviar un robot a ser cargado. En el estado del robot el máster recibe información sobre su batería y si ésta se encuentra por debajo del 20% se le envía al robot las instrucciones de ir a la estación de carga. Finalmente en la estación de carga el cargador detecta la presencia de un robot a ser cargado y ejecuta su método cargar robot que no solo inicia a cargar la batería del robot sino que actualiza el estado del cargador a no disponible.



## 5. Plan de Trabajo

Esta sección contiene un listado de tareas pendientes, incluyendo las fechas y miembros del equipo asignados.

Actividad	Fechas	Encargados
Crear modelos de simulación gráfica en unity (Prioridad)	5 días (15 Nov)	Victoria y Danaé
Crear primer modelado del entorno de simulación con python (Prioridad)	5 días (15 Nov)	Diego y David
Conexión entre modelo multiagente en python y simulación gráfica	1 semana (17 Nov)	Todo el equipo
Elaborar simulación para prueba de semana 4 en CETEC	4 días	Todo el equipo
Hacer pruebas y correcciones a la simulación de para semana 4	1 semana	Todo el equipo
Prueba Final robot CETEC	1 semana (18 Nov)	Todo el equipo
Entrega documento final	2 semanas y media (27 Nov)	Todo el equipo
Entrega simulación final	2 semanas y media (27 Nov)	Todo el equipo

## 6. Aprendizajes Adquiridos

En nuestra primera entrega, abordamos la clasificación de los agentes según sus funciones específicas dentro del sistema y su rol en la automatización de la planta. Nos enfocamos en cómo estructurar la modelación de estos agentes, identificando sus interacciones, atributos y requisitos para posteriormente integrarlos de manera efectiva en la simulación.

Como equipo, realizamos un análisis de nuestras fortalezas y áreas de mejora, lo cual nos permitió distribuir responsabilidades y diseñar un método de trabajo eficiente y colaborativo, y establecimos un flujo de trabajo que facilitara el seguimiento de tareas y el cumplimiento de tiempos.

También definimos y evaluamos el ambiente en el que estará el sistema multiagente, considerando tanto sus limitaciones como sus características operativas. Evaluamos los parámetros para las reglas y comportamientos de cada agente, definiendo sus sensores y actuadores de acuerdo con las funciones específicas de cada tipo de agente en la simulación. Además, definimos las bases de evaluación para el rendimiento de cada agente y de forma colectiva el funcionamiento del almacén.

## **Referencias**

Odell, J., Parunak, H. V. D., & Bauer, B. (2000, July). Extending UML for agents. In *Proceedings of the agent-oriented information systems workshop at the 17th national conference on artificial intelligence* (pp. 3-17).

Bauer, B., Müller, J. P., & Odell, J. (2001). Agent UML: A formalism for specifying multiagent software systems. *International journal of software engineering and knowledge engineering*, 11(03), 207-230.