

**Instituto Tecnológico y de Estudios Superiores de Monterrey,
Campus Guadalajara**



**Tecnológico
de Monterrey**

Modelación de sistemas multiagentes con gráficas computacionales

Evidencia 2 | Avances y presentación del reto - Entrega Final A

Profesores:

Iván Axel Dounce Nava

Obed Nehemías Muñoz Reynoso

Luis Raúl Guerrero

Carlos Johnatan Sandoval

Alumnos:

Iker Rolando Casillas Parra | A01641047

Cesar Simental Duenas | A01641385

Alejandro Moncada Espinosa | A01638343

Jorge Iván Sánchez González | A01761414

Martín Rivera Ahumada | A01749620

Viernes, 6 de Septiembre de 2024

Índice

Índice.....	2
1. Conformación de equipo.....	2
2. Propuesta formal.....	3
2.1 Descripción de la problemática.....	3
3. Agentes.....	4
3.1 Diagramas de clases.....	5
3.2 Propiedades.....	6
3.3 Utilidad.....	7
3.3 Diagramas de protocolos.....	8
4. Plan de trabajo.....	9
4.1 Aprendizaje adquirido.....	9
4.2 Actividades pendientes.....	9
4.3 Intervalo de actividades.....	9
5. Análisis individual:.....	9
6. Repositorio.....	10

1. Conformación de equipo

Somos el equipo Clerics de TC2008B conformado por:

Iker Rolando Casillas Parra: Mis fortalezas son la visión computacional y el pensamiento estructurado, el área en la que más necesito trabajar para mejorar es en el modelado 3D de Unity.

Mi objetivo es entregar un proyecto de buena calidad, que cumpla con todos los requisitos esperados y que sea ligero y fácil de reproducir en cualquier ambiente. Me comprometo a organizarme bien con mis compañeros y dedicar el tiempo necesario para realizar y pulir cada tarea que se pueda presentar.

Cesar Simental Dueñas: Mis fortalezas son el manejo de ambiente y modelos en Unity y programación de Scripts en este mismo programa, el área en la que más necesito trabajar es en el pensamiento de la lógica detrás de los agentes.

Mis expectativas de mi equipo en este bloque es que nuestra entrega final sea un proyecto completo y funcional, en el que todas las partes requeridas sean notables y trabajen bien en conjunto.

En equipo tendremos que definir bien las fechas límites y cómo usar nuestras fortalezas para poder complementarnos y llegar al resultado deseado.

Alejandro Moncada Espinosa: Mis fortalezas son el desarrollo de los agentes y la estructuración de las reglas y acciones para hacer que estas funcionen de una manera adecuada, el área en la que más necesito trabajar es en la visión computacional.

Mi expectativa es poder hacer una entrega en tiempo y forma demostrando los conocimientos obtenidos durante el periodo y así poder enseñar lo que cada uno pudo lograr para desarrollar un buen entregable final con todo funcionando como debe ser.

Para poder lograrlo debemos tener una muy buena comunicación para poder aclarar dudas que se tengan y poder apoyar al compañero en lo que se le dificulte al igual que ajustar bien los tiempos de trabajo para poder abarcar con todo sin estrés.

Jorge Iván Sánchez González: Mis fortalezas son el pensamiento de reglas de automatización de agentes, y mi área de oportunidad es la programación de esas ideas en agent.py

Las expectativas que tengo de este presente bloque son primero que nada cumplir con los lineamientos y tipo de proyecto de lo que espera el socio formador, seguido de eso espero poder conocer mejor cómo funcionan los agentes y cómo poderlos automatizar.

Para lograr lo anteriormente mencionado pienso que lo que debemos hacer como equipo es utilizar todas nuestras fortalezas para complementarnos y cubrir nuestras áreas de oportunidad.

Martín Rivera Ahumada: Creo que mis fortalezas incluyen el trabajo en equipo y el pensamiento lógico, lo que me permite colaborar eficazmente y resolver problemas de manera estructurada. Sin embargo, tengo áreas de oportunidad en la conceptualización de teorías complejas y en mantener el enfoque cuando manejo múltiples proyectos simultáneamente.

Como expectativas a lograr serían un proyecto basado en buenas prácticas de muy buena calidad y desempeño, trabajo en equipo. Con esto me comprometí a realizar todas mis tareas con el mejor esfuerzo y actividad, al igual que estar al pendiente de las necesidades del proyecto así como cualquier forma de cooperación entre el equipo para obtener una entrega excelente.

2. Propuesta formal

2.1 Descripción de la problemática

El patrullaje coordinado es un desafío crucial en la seguridad urbana. En un entorno de riesgo en zonas abiertas, múltiples agentes, como cámaras de vigilancia, drones y personal de seguridad, deben trabajar juntos para garantizar la seguridad y prevenir incidentes. La coordinación efectiva entre estos agentes es fundamental para detectar actividades sospechosas, responder rápidamente a emergencias y mantener la integridad de la zona.

En este proyecto, nosotros tendremos la oportunidad de diseñar e implementar un sistema multiagente que simule el patrullaje coordinado en una zona de riesgo (ya sea por daño a la integridad del personal, daño a las instalaciones o recursos, o uso indebido de instalaciones o recursos). La simulación gráfica permitirá visualizar cómo los diferentes agentes interactúan,

comparten información y toman decisiones en tiempo real. La motivación de esta simulación es resolver problemas diversos en zonas abiertas de riesgo, como la cobertura de vigilancia, minimizar los tiempos de respuesta y garantizar la seguridad de los trabajadores y la infraestructura.

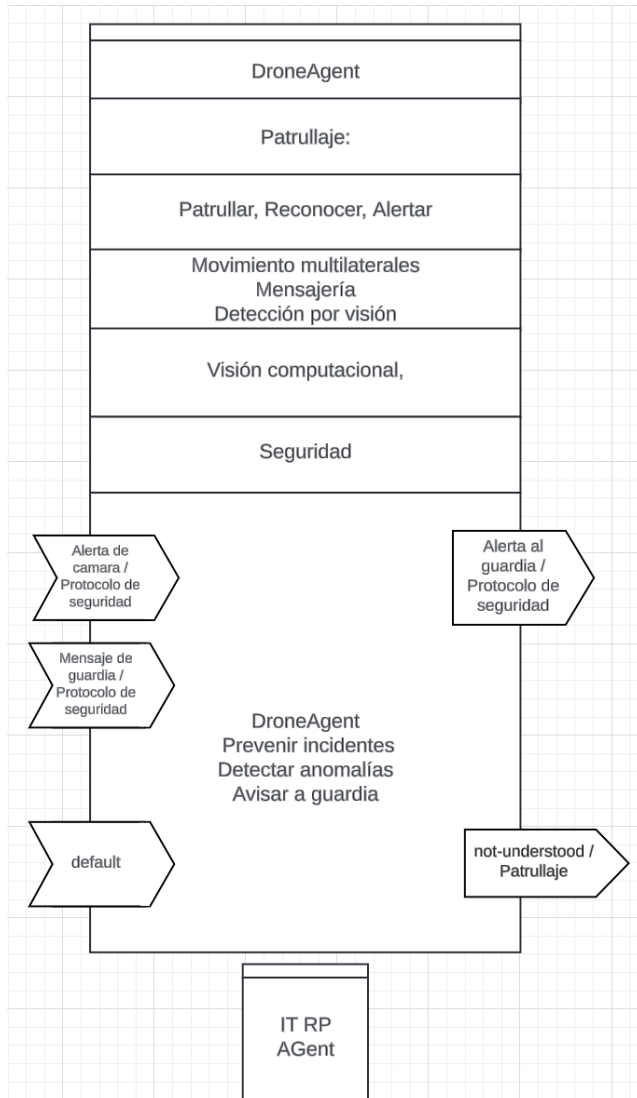
Al abordar este desafío, nuestro equipo contribuirá a mejorar la seguridad en entornos urbanos, aplicando principios de inteligencia artificial, comunicación entre agentes y estrategias de coordinación. Además, exploraremos cómo la tecnología puede mitigar riesgos y proteger a la comunidad en situaciones críticas.

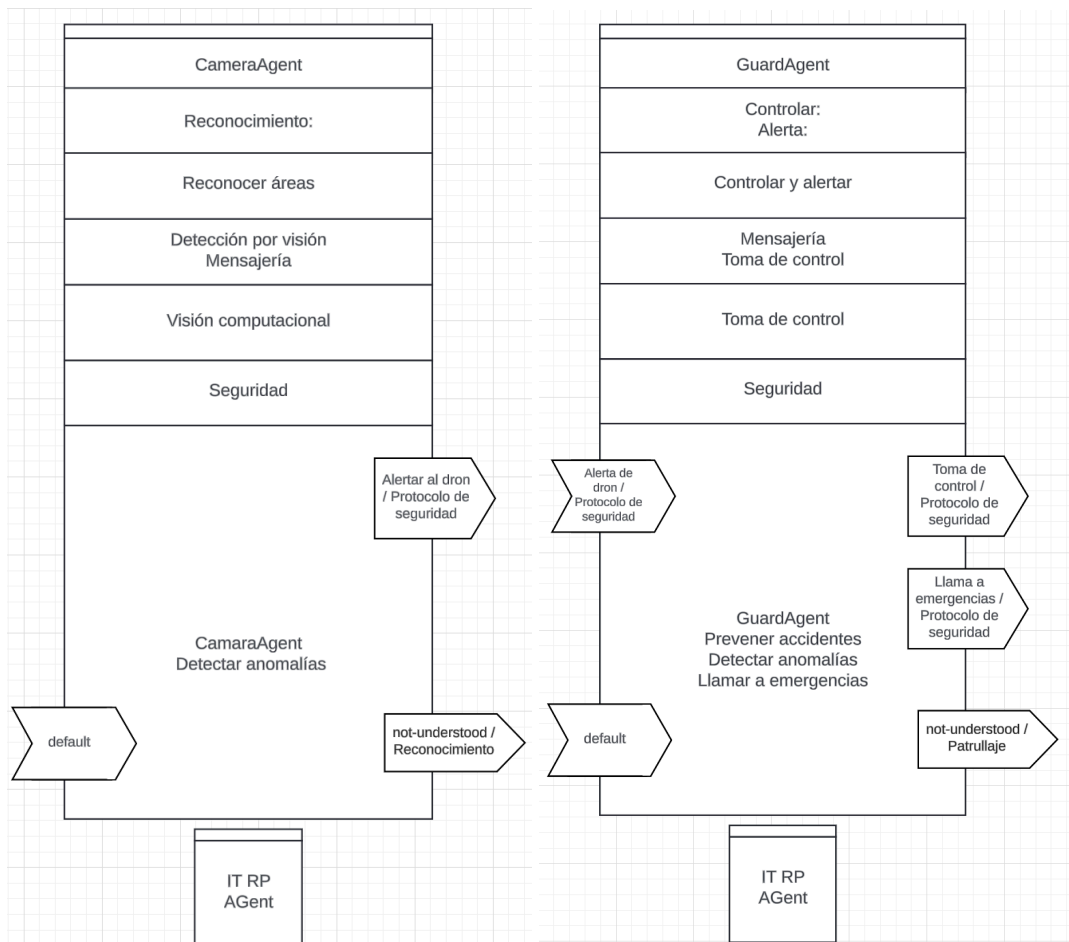
3. Agentes

Los agentes involucrados en este proyecto son los siguientes:

- **DroneAgent:** El agente Drone es capaz de identificar Boxes y Walls y moverse. Su interacción con los demás agentes es el reconocimiento y recolección de Boxes para posteriormente depositarlas en un Stacking.
- **GuardAgent:** El agente Guard es un agente estático con una posición que realiza la acción de observar por medio del dron cuando algo sospechoso ocurre para verificar si es peligroso o es una falsa alarma. Su interacción es dar un aviso general sobre lo que ocurre en la bodega.
- **CameraAgent:** El agente Camera es un agente estático con una posición donde la acción que realiza es el observar si ocurre algo sospechoso. Su interacción es avisar al guardia y a su vez al dron para que se dirija a la posición a investigar qué ocurre.

3.1 Diagramas de clases





3.2 Propiedades

Ambiente

Bodega:

- **No accesible:** Porque todos los agentes dependen de su propia visualización no conocen el mapa.
- **No determinista:** Porque se espera un resultado pero no siempre se puede lograr.
- **Estático:** No cambia el escenario, puede cambiar la ubicación del fuego cada que corre el programa pero una vez corriendo no cambia su posición.
- **Discreto:** Porque no cambia de manera continua.
- **No episódico:** Porque cada evento es individual y único, no hay nada acumulable y todo menos los agentes es estático.

Agentes

Dron:

- **Proactividad:** Busca el fuego en el ambiente.
- **Reactividad:** Cuando encuentra el fuego manda una señal.
- **Socialización:** La señal se la comunica al agente Guardia.

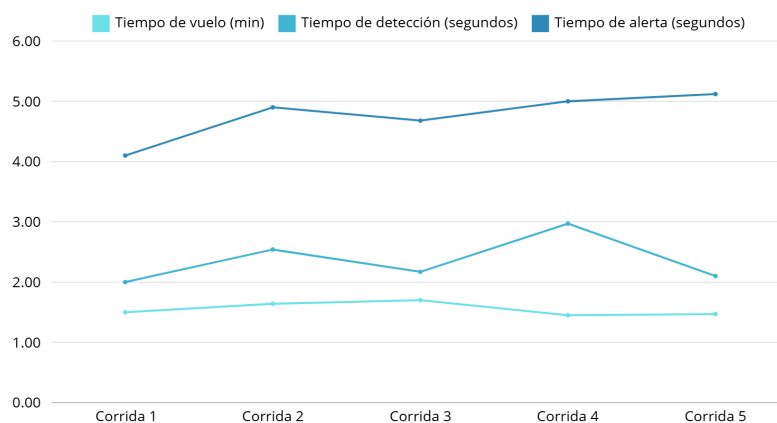
Cámara:

- **Reactividad:** Cuando detecta el fuego manda una señal.
- **Socialización:** Comunicarse con el dron al mandarle una señal.

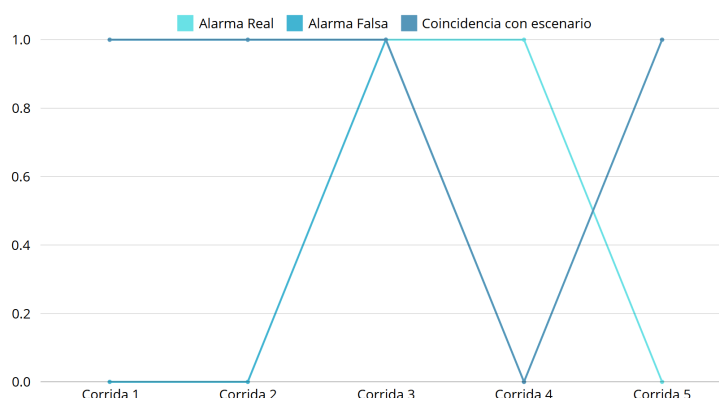
Guardia:

- **Reactividad:** Cuando recibe el mensaje del dron, le habla a los bomberos.
- **Socialización:** Maneja el dron además de que se comunican ambos.

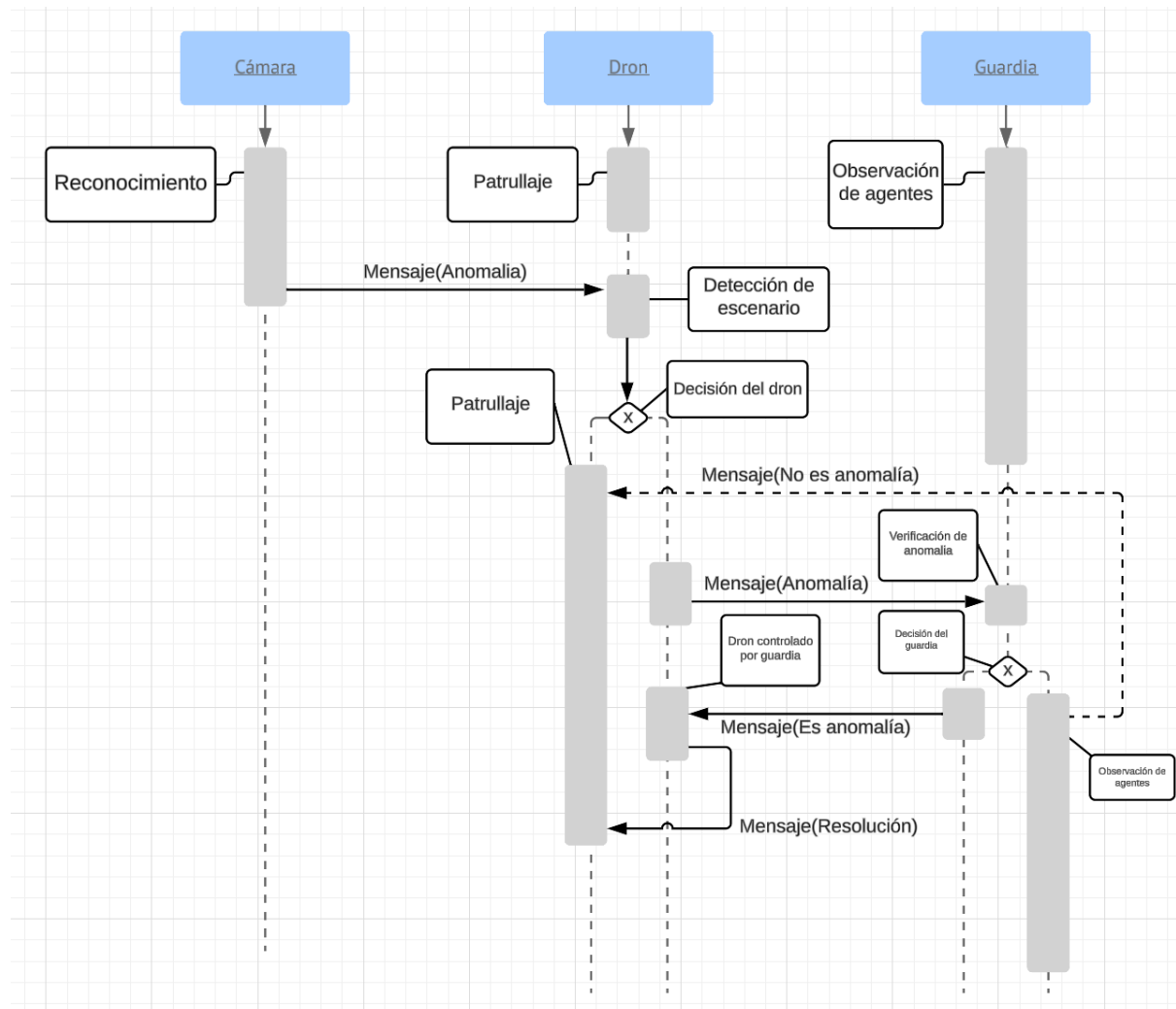
3.3 Utilidad y éxito



Para la utilidad nos basamos en el menor tiempo posible para ejecutar ciertas acciones primordiales, como lo son el tiempo de vuelo, de detección de anomalías y de alerta. Siendo un tiempo promedio aproximado (Figura superior) de 1.5 minutos para el vuelo, 2.25 para la detección y 4.5 para la alerta creemos que su utilidad se puede basar en que el 80% de las ejecuciones están por debajo del promedio de tiempos anotados previamente. Por otra parte, el éxito lo consideramos en base a dos objetivos diferentes. Por un lado el éxito del tiempo utilizado y el éxito de uso. Dado que es importante que se cumplan los tiempos acordados también es necesario que detecten bien las anomalías los agentes. Con lo que llegamos a la gráfica de utilidad para la certeza de detección (figura inferior). Y al final determinamos que el éxito depende de si la ejecución fue por debajo de los promedios de tiempo y a su vez fue certero con la detección.



3.3 Diagramas de protocolos



4. Plan de trabajo

4.1 Aprendizaje adquirido

El aprendizaje adquirido que llevamos hasta la fecha es el comportamiento de los agentes y cómo estos se relacionan e interactúan entre sí en un ambiente, con esto también hemos aprendido a delimitar reglas que regulen el comportamiento de cada uno de estos agentes así como su respectiva programación.

Por otro lado yéndonos al área del modelado, hemos aprendido a modelar utilizando Unity, esto nos da una ventaja debido a que ves situaciones más realistas con objetos más detallados que las cuadradas de colores que ofrece python para la interpretación de los agentes así como las cuadrículas negras para la creación del ambiente en que estos agentes interactúan.

Por último hemos podido desarrollar un pensamiento divergente para poder “traducir” lo que las cámaras y agentes de unity detectan así como la información que usan para interpretarlas con python y viceversa, esto para poder programar y utilizar los agentes y el entorno de unity con la herramienta de agent.py

4.2 Actividades pendientes

- Conectar múltiples cámaras al mismo tiempo para la visión computacional. Días propuestos: 3
- Ejecutar correctamente el server de unity a los agentes. Días propuestos: 3

4.3 Intervalo de actividades

- Para la conexión de las cámaras se estima la fecha del 5 de septiembre trabajando en ello: César Simental e Iker Casillas
- Para la ejecución de server de unity se estima la fecha del 5 de septiembre trabajando en ello: Alejandro Moncada e Iker Casillas

5. Análisis individual:

¿Por qué seleccionaron el modelo multiagentes utilizado? ¿Cuáles fueron las variables que se tomaron al momento de tomar la decisión? ¿Cuál es la interacción de esas variables con respecto al resultado de la simulación? ¿Por qué seleccionaron el diseño gráfico presentado? ¿Cuáles son las ventajas que encuentras en la solución final presentada? ¿Cuáles son las desventajas que existen en la solución presentada? ¿Qué modificaciones podrías hacer para reducir o eliminar las desventajas mencionadas?

Iker Rolando Casillas Parra:

Seleccionamos este modelo multiagentes porque era el más adecuado para nuestras necesidades, permitiendo definir claramente las interacciones de los agentes en el entorno. Consideramos variables como la capacidad de cada agente para realizar su tarea y la flexibilidad del entorno.

Optamos por Unity para el diseño gráfico por su realismo y facilidad para visualizar las interacciones de los agentes. Las ventajas son la claridad en los roles y rutas, pero la dificultad de integrar los servidores de Unity y la visión computacional es una desventaja. Buscaría simplificar la arquitectura para mejorar la eficiencia.

Cesar Simental Dueñas:

El modelo multiagentes fue seleccionado de esta manera tomando en cuenta el tipo de agentes y la acciones que esperábamos que realizaran.

El diseño gráfico que fue representado en Unity se dividió hacerlo como una bodega por varias cosas, las dos principales era que ya teníamos un adelanto gracias a la entrega de la evidencia 1 y también que al ser una bodega se podía hacer con mayor facilidad una matriz sin tantas variaciones.

Las ventajas de la solución final presentada fue que la función de cada agente quedó muy bien definimos, con una ruta específica para el dron y el guardia tiene su propia zona de vigilancia, algunas desventajas son lo complicado que es juntas ambos servers de agentes y visión computacional con Unity y las limitantes de esto. Las modificaciones que quisiera hacer serían sobre el tema mencionado en las desventajas, buscaría una manera óptima de juntar el agente y la visión en un solo servidor y reducir los servers necesarios para la simulación.

Alejandro Moncada Espinosa:

Utilizamos el modelo multiagentes debido a que los requisitos para poder cumplir con las tareas asignadas nos lo facilitaban los agentes al actuar con forme a reglas establecidas.

Se decidió utilizar la bodega ya que es un lugar en el cual podrían suceder acontecimientos sospechosos reales y el recorrido del dron se facilitaba de una mejor manera.

Las ventajas de esta solución final son que todas las partes esperadas como lo de unity, la visión y el agente funcionan de la manera esperada, la desventaja es que el peso de interacción hace que todo se vea más lento y tenga mayor costo de computación, por lo que se podría mejorar reduciendo las solicitudes y texturas.

Jorge Iván Sánchez González:

Utilizamos un Agent Base Model (ABM) debido a que era más sencillo programar a nuestros agentes basados en reglas que deben obedecer en lugar de un MAS que son agentes autónomos más complejos y en un ABM no necesariamente deben de ser inteligentes. Llegamos a esta conclusión tomando en cuenta diferentes factores como el corto tiempo de desarrollo, la complejidad que alcanzamos con nuestras habilidades y herramientas, así como los temas vistos en clase.

Por otro lado seleccionamos el diseño gráfico utilizado porque generalmente en los almacenes debe haber bastante seguridad porque al ser almacenes más bien hay objetos (de valor) y no personas por lo que se necesita una constante supervisión.

Las ventaja que encontramos de la solución presentada es la simplicidad del modelo y su eficacia, la desventaja es que al no ser algo tan complejo las acciones y conocimientos de los agentes son muy específicos por lo que si se busca la expansión de estos mismos existe una limitación. Y lo que haríamos en este caso para mejorar la desventaja es desarrollar más a fondo la inteligencia de los agentes para ampliar su gama de acciones y decisiones.

Martín Rivera Ahumada:

El uso del modelo multiagentes en nuestro sistema de detección de anomalías en la bodega ha sido crucial para su efectividad. Este enfoque permite que cámaras y drones trabajen conjuntamente para detectar y verificar incendios, mejorando significativamente la precisión y reduciendo la tasa de falsas alarmas. La integración de estos agentes crea un sistema robusto donde las cámaras alertan al dron, que luego valida la alerta antes de que el guardia tome una decisión final.

Aunque enfrentamos desafíos con la integración y sincronización de los agentes, estas dificultades fueron superadas mediante ajustes y pruebas exhaustivas. Esta experiencia ha resaltado cómo la colaboración entre múltiples agentes puede optimizar la respuesta a problemas críticos, mostrando la importancia de un enfoque coordinado para mejorar la fiabilidad del sistema.

6. Repositorio

<https://github.com/Mart4725/Multiagentes/new/main>

https://drive.google.com/drive/folders/1-4As7Z7Q9AiDMITnz1GT_DN0EZnO1dYw