



Universidad  
Rey Juan Carlos

**Escuela Técnica Superior de Ingeniería Informática**

**HERRAMIENTA PARA LA OBSERVACIÓN  
Y EVALUACIÓN DE COMPORTAMIENTOS  
EMERGENTES**

**JUEGO 4x BASADO EN AGENTES**

Autor: David López Pereira

Tutor: Gustavo Recio Isasi

Cotutor: Jonathan Crespo Herrero

# Índice

1

**Introducción**

2

**Objetivos**

3

**Metodología de desarrollo**

4

**Estado del arte**

5

**Diseño**

6

**Implementación**

7

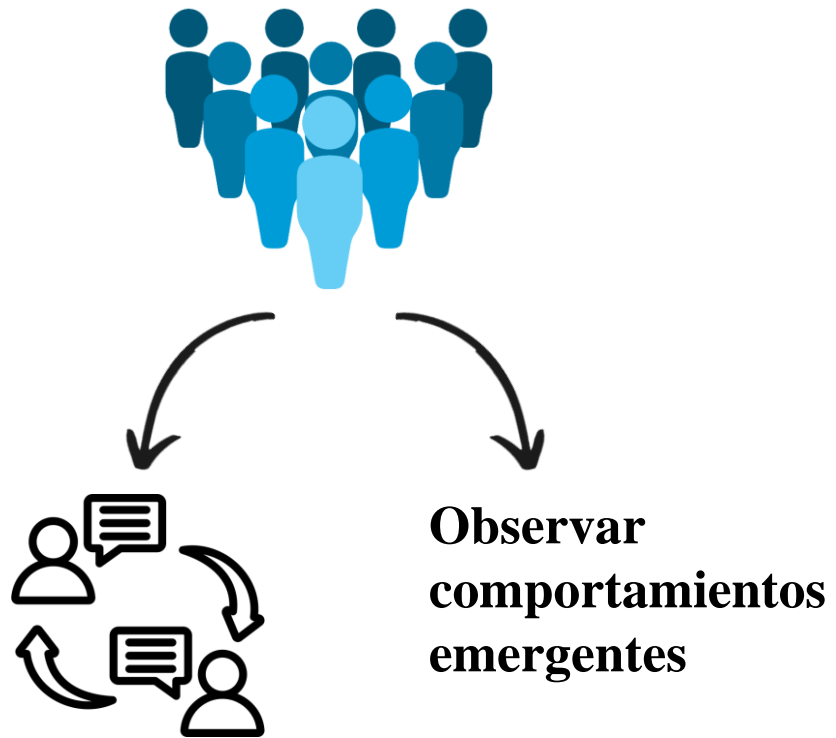
**Pruebas**

8

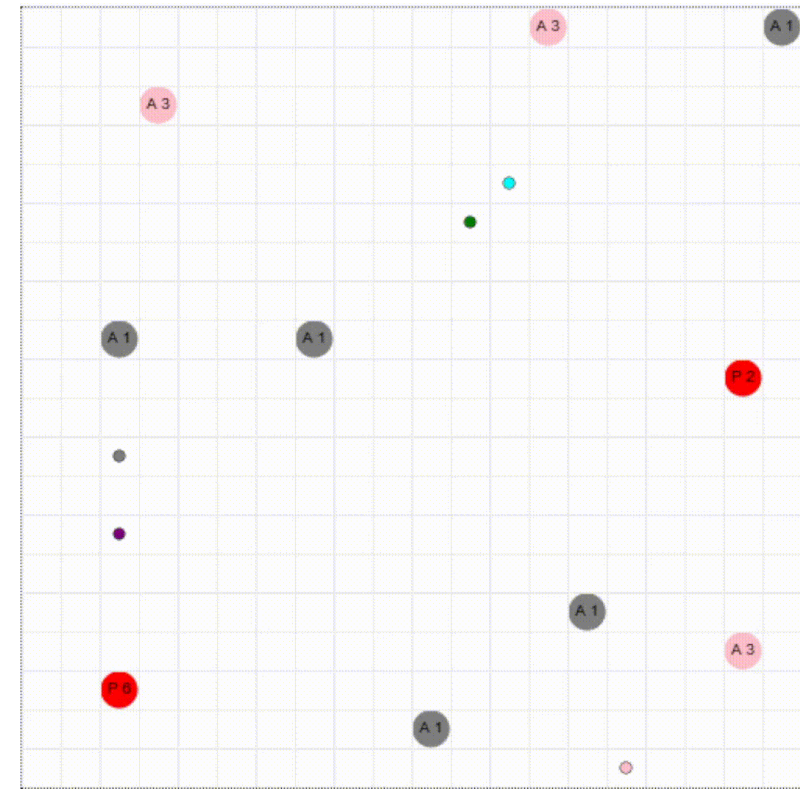
**Conclusiones**

# Introducción

¿Por qué utilizar los ABM?



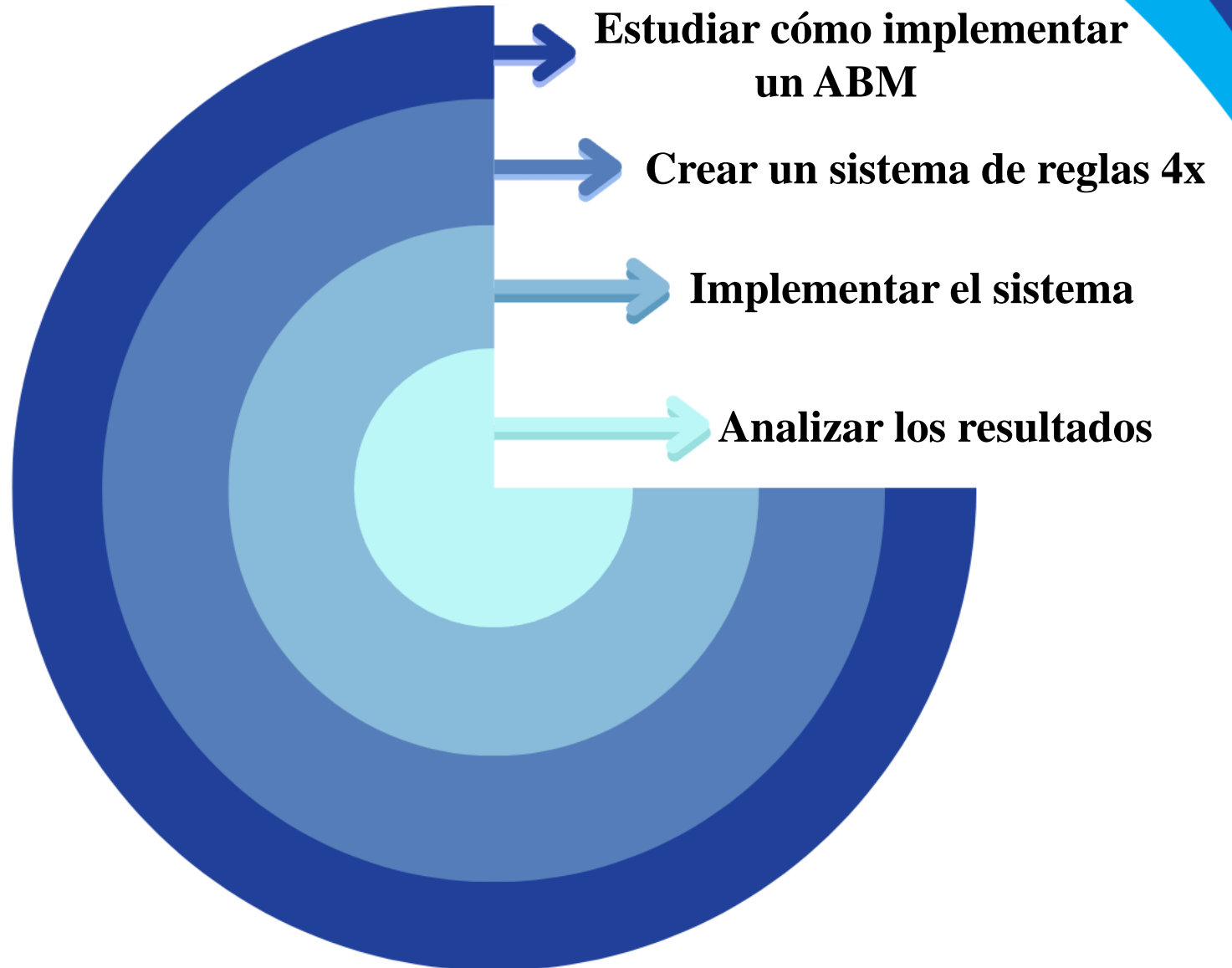
Sistema basado en agentes (ABM)



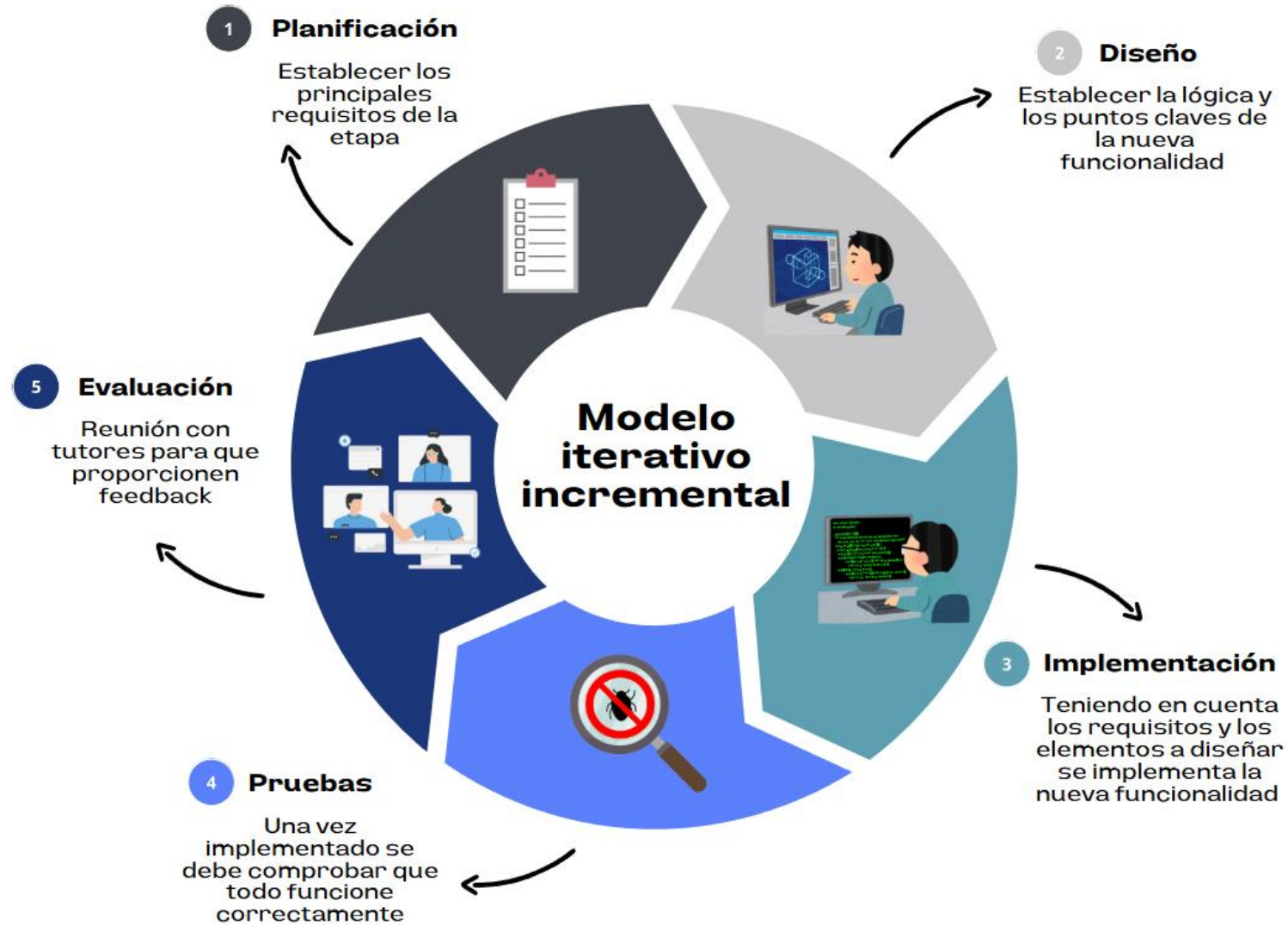
# Objetivos



**Generar ABM  
en un juego 4x**

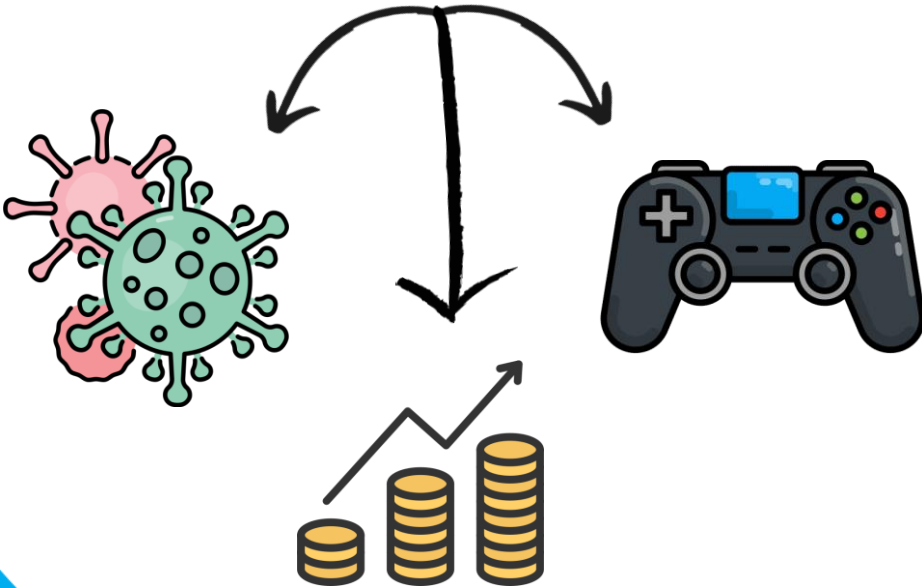


# Metodologías de desarrollo



# Estado del arte

## Trabajos relevantes



## Implementaciones

	<b>AgentPy</b>
	<b>Repast4Py</b>
	<b>PyNetLogo</b>
	<b>PyPandora</b>
	<b>Mesa</b>

# Mesa

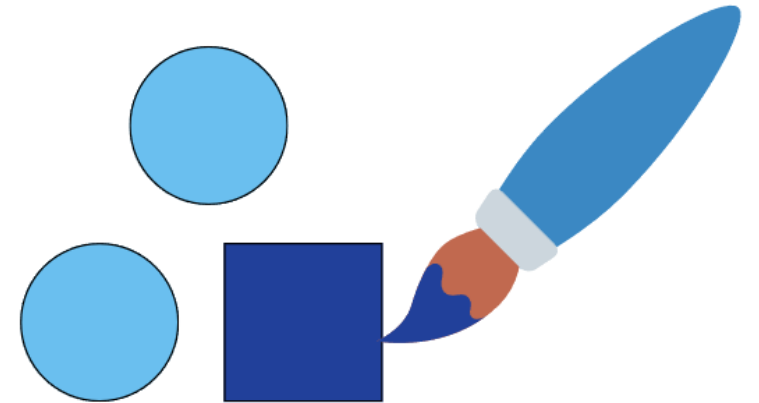
## Comunidad















## Curva de aprendizaje



## Personalización



# Diseño

Componentes del juego		
3  10 		
Atributos	Comportamientos	Otros elementos
 Oro	 Explorer	3  Tipos de arma
 Puntos Estelares	 Chaser	 Planetas conquistados
 Tecnología	 Farmer	 Mejoras de atributos
 Fábricas		



# Diseño

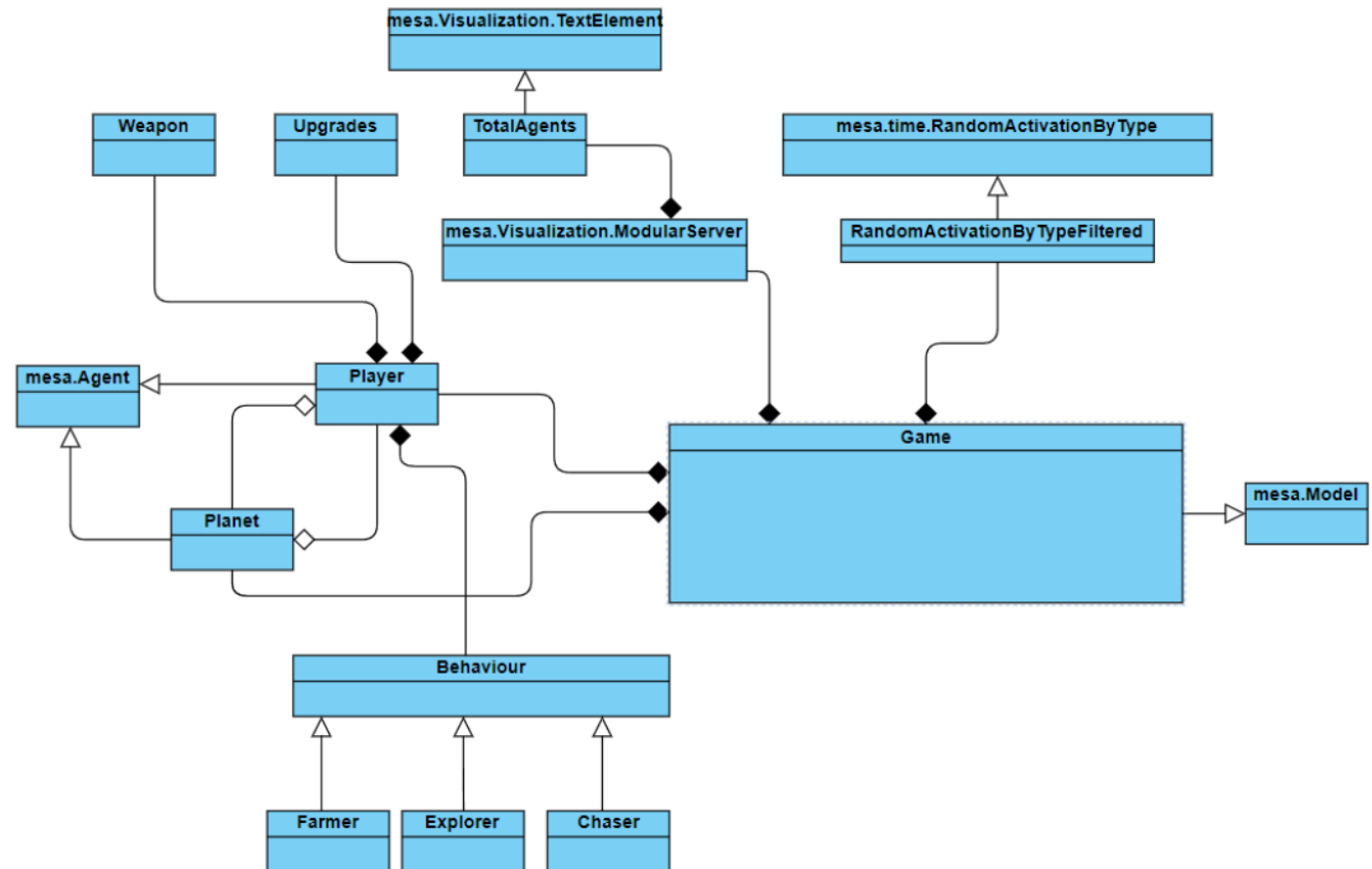
## Reglas del juego

- Cada 100 turnos 
- Cada  cuesta 25 oro  Si no puede 
- Cada acción tendrá un coste de oro
- Al luchar obtendrán un porcentaje de recursos del perdedor

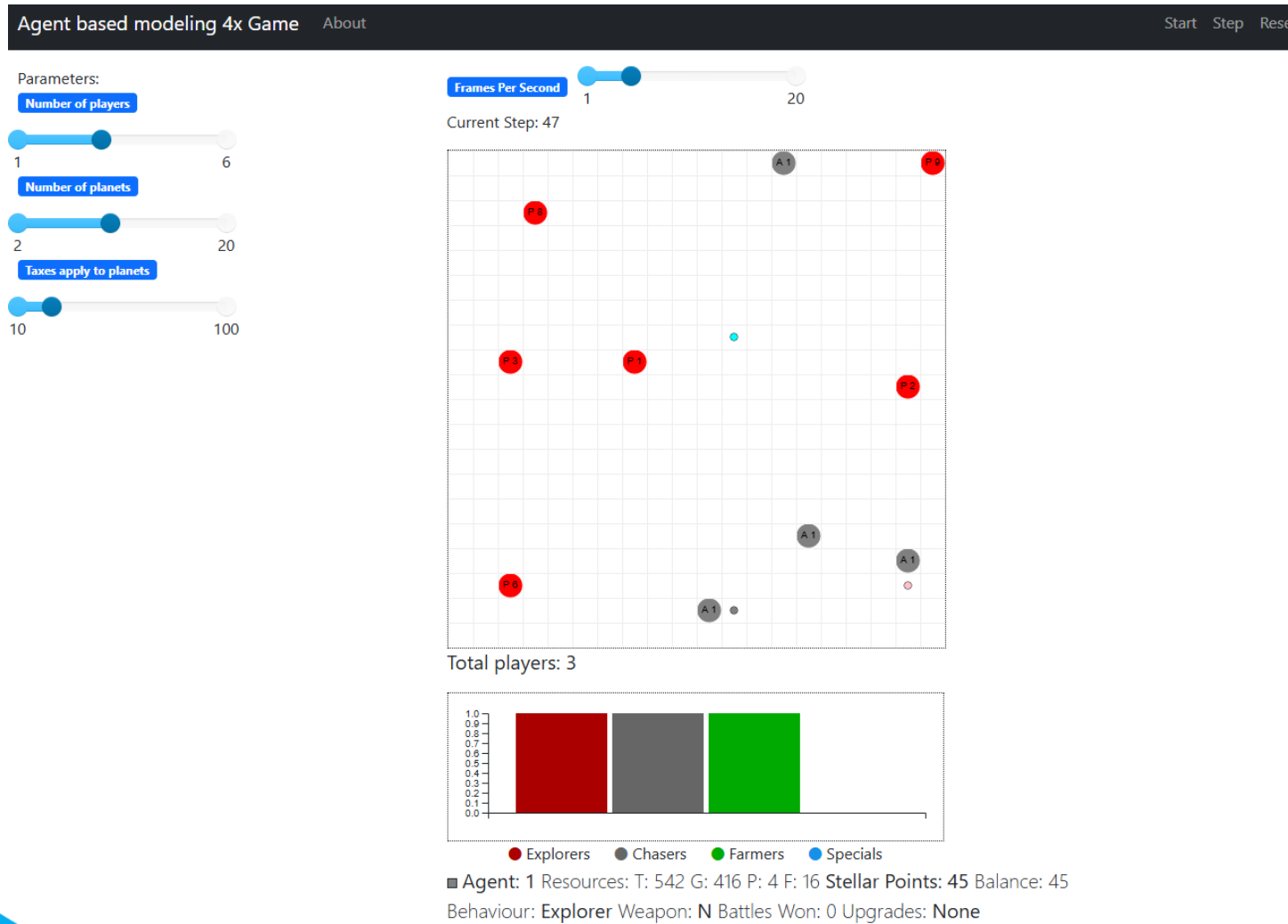
# Implementación

Diseño orientado a objetos

Diagrama UML



# Interfaz gráfica

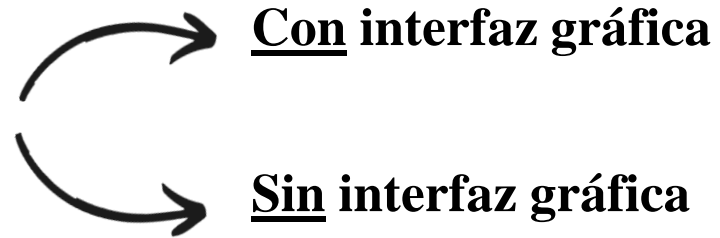


# **Pruebas y evaluación del sistema**




# Sistema por defecto

Sin ningún cambio dinámico

Dos formas de ejecución



Resultados de las 50 simulaciones sin interfaz gráfica




Agentes totales	Presencia 	Presencia 	Presencia 
716*	27 %	54 %	19 %

\*14 agentes por ejecución

# Nueva clase dinámica




Cambio dinámico en la clase Explorer en turno 20, para quitar su capacidad de huir de las luchas

Cambio en  
una instancia

Agentes totales	Presencia 	Presencia 	Presencia 
736*	26 %	54 %	20 %

\*14 agentes por ejecución

Cambio en  
toda la clase




Agentes totales	Presencia 	Presencia 	Presencia 
861**	23 %	57 %	20 %

\*\*17 agentes por ejecución

## Ejecución con cambio en primera instancia




# Nuevo atributo dinámico

Cambio en  
una instancia

Agentes totales	Presencia 	Presencia 	Presencia 
823*	26 %	53 %	21 %

\*16 agentes por ejecución

Cambio en  
toda la clase

Agentes totales	Presencia 	Presencia 	Presencia 
1.171**	32 %	29 %	39 %

\*\*23 agentes por ejecución

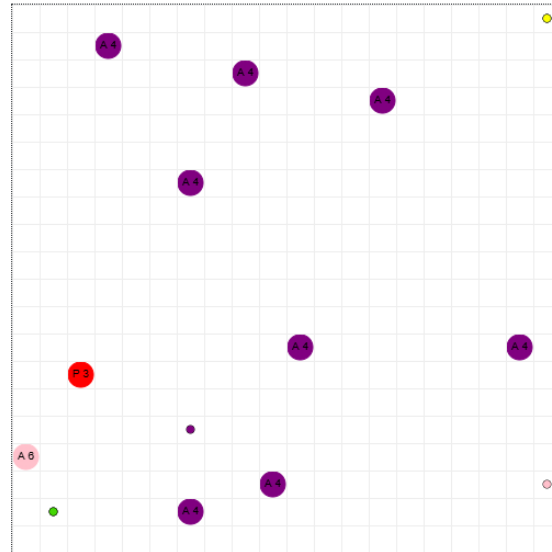


# Nuevo método dinámico

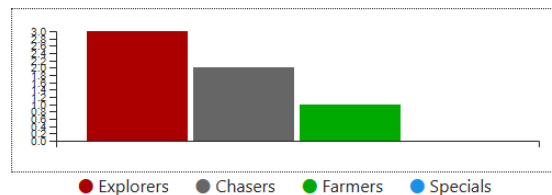
Cambio dinámico en la clase Farmer en turno 10

Comparación del turno 575 al optimizar la clase Farmer a nivel global

Current Step: 575

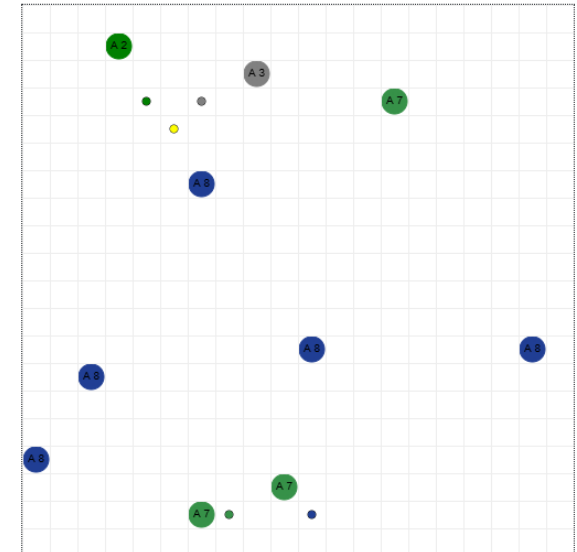


Total players: 6

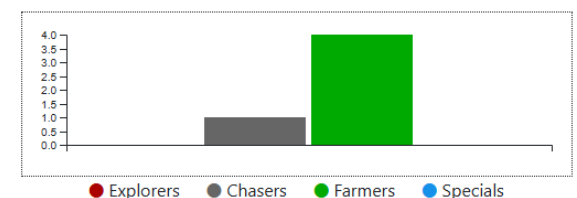


Ejecución por defecto

Current Step: 575






Total players: 5



Ejecución con cambio en toda la clase




# Nuevo método dinámico

Cambio en  
una instancia

Agentes totales	Presencia 	Presencia 	Presencia 
794*	21 %	53 %	26 %

\*16 agentes por ejecución

Cambio en  
toda la clase

Agentes totales	Presencia 	Presencia 	Presencia 
1.106**	18 %	31 %	51 %

\*\*22 agentes por ejecución




# Nuevos comportamientos

## Simulación de nuevos comportamientos emergentes

### RandomBehaviour

Comportamiento que actúa de manera aleatoria

#### Resultados

Agentes totales	Presencia 	Presencia 	Presencia 	Presencia Random
853*	19 %	42 %	16 %	23 %




\*17 agentes por ejecución

### CustomBehaviour

Comportamiento personalizado



#### Resultados

Agentes totales	Presencia 	Presencia 	Presencia 	Presencia Custom
1277**	14 %	32 %	13 %	41 %

\*\*26 agentes por ejecución

# Nuevos comportamientos

## Friendly




Evita las luchas → persigue al agente con menor balance (amigo)

Le otorga 10% oro, 5% tecnología y ambos reciben 1 punto estelar

Comportamiento emergente propio del sistema

■ Agent: 36 Resources: T: 2861 G: 2417 P: 0 F: 8 Stellar Points: -9 Balance: -9  
Behaviour: Friendly Weapon: Lasers Battles Won: 2 Upgrades: None

## Resultados

Agentes totales	Presencia 	Presencia 	Presencia 	Presencia Friendly
670*	25 %	52 %	17 %	6 %

\*13 agentes por ejecución



















# Conclusiones

## Objetivos

- ✓ Estudiar cómo implementar un ABM
- ✓ Crear un sistema de reglas 4x
- ✓ Implementar el sistema
- ✓ Analizar los resultados



## Balanceo del sistema

Tipo de ejecución	Presencia 	Presencia 	Presencia 	Agentes Totales
Por defecto	27 %	54 %	19 %	716
Cambio global en 	 -4 %	 +3%	 +1%	 +20%
Cambio global en 	 +5%	 -25%	 +20%	 +64%
Cambio global en 	 -9%	 -23%	 +32%	 +54%

# FIN

Muchas gracias por su atención

¿Hay alguna pregunta?