

# I HACKATHON COMPUTACIÓN BURGOS

HACOBU 2024

8-9 de Marzo

DESAFÍO



Imagen creada con IA generativa

Organizan

**ITCL**  
CENTRO TECNOLÓGICO



UNIVERSIDAD  
DE BURGOS

NOS  
**IMPULSA**

competitividad  
empresarial

**ice**

  
Junta de  
Castilla y León

**WOLARIA**

Colaboran



Ayuntamiento  
de Burgos

**DIHBU**

Digital Innovation Hub Industry 4.0



**DIGIS<sup>3</sup>**

Co-funded by  
the European Union

Financiado por  
la Unión Europea  
NextGenerationEU

GOBIERNO DE ESPAÑA  
MINISTERIO DE INDUSTRIA, COMERCIO  
Y TURISMO

Plan de  
Recuperación,  
Transformación  
y Resiliencia

**EOI** Escuela de  
organización  
industrial

# TABLA DE CONTENIDO

---

**02**      **INTRODUCCIÓN**

---

**03**      **RETO DE REALIDAD VIRTUAL**

---

**04**      **RETO DE INTELIGENCIA  
ARTIFICIAL**

---

**05**      **EXPLICACIÓN DE VARIABLES**

---

**07**      **PUNTUACIÓN DE SOLUCIONES**

---

# INTRODUCCIÓN

La Inteligencia Artificial (IA) tiene un gran potencial de aplicación en el **campo de los videojuegos**. Uno de los casos de uso más interesantes es la capacidad del juego de usar IA para adaptar la dificultad del mismo al desempeño del jugador.

Uno de los ejemplos más famosos es *Metal Gear Solid V*, donde los enemigos empiezan a llevar casco si el jugador acierta muchos disparos en la cabeza o patrullarán en pareja si el jugador suele eliminarlos cuando están solos. De igual forma, si el jugador comienza a perder, los enemigos llevarán menos protección y se separarán del grupo. Es decir, la dificultad se adapta a la pericia del jugador.

Otro ejemplo más sutil es la saga *Mario*, donde se aplican diferentes ajustes de dificultad en función de si el jugador no es capaz de superar la dificultad en zonas concretas o si avanza muy rápido por el juego. Estos ajustes varían desde ofrecer tutoriales o invencibilidad hasta desbloquear niveles ocultos mucho más difíciles.

En HaCoBu 2024 se proponen dos retos, uno de Realidad Virtual (RV) y otro de Inteligencia Artificial. Los retos estarán relacionados y los proyectos finales podrán interactuar entre ellos, pero cada equipo solo deberá resolver uno de los dos retos.

Cada equipo deberá entregar los archivos fuentes de su herramienta, un ejecutable (si aplicase) y preparar una exposición de 5 minutos en la que tendrá que presentar su solución al jurado.

# RETO DE REALIDAD VIRTUAL

Para este reto, se propone crear un videojuego capaz de adaptarse a ciertos parámetros del jugador, como la puntuación y el tiempo. Los juegos no deberán implementar estos algoritmos de adaptación, si no que serán compatibles con los modelos desarrollados en el reto de IA.

Para poder tener unos datos uniformes y que todos los juegos sean compatibles, se propone desarrollar un juego arcade del género "Minijuego de pesca" en **Realidad Virtual** (RV). La temática del juego será libre y se valorará la originalidad de los juegos. Algunos ejemplos de este género serían la mecánica de pesca de *Dredge*, cazar Pokémon en *Pokémon Go* o *Let's Go*, o juegos más clásicos como *Catch 'Em*.

El juego deberá consistir en una ronda estilo arcade en la que el jugador deberá continuar pescando/capturando/cazando hasta que pierda, ya sea porque ha fallado muchos intentos o porque se le han escapado muchos bichos/peces/objetos, dependiendo del juego. Al terminar la ronda, el juego generará una estructura de datos con la siguiente información:

- Tiempo que ha durado la ronda.
- Puntuación conseguida total.
- Puntuación del objeto que menos puntuación le ha dado.
- Puntuación del objeto que más puntuación le ha dado.
- Número de objetos capturados.
- Número de objetos perdidos.
- Tiempo de respuesta medio.
- Tiempo de respuesta mínimo.
- Tiempo de respuesta máximo.

Al comienzo de la siguiente ronda, el juego cargará una estructura de datos que contendrá variables que permitirán modificar la dificultad del juego:

- Ratio de aparición de los objetos en el juego.
- Distancia al jugador de los objetos.
- Tamaño de los objetos.
- Ratio de recompensa de los objetos.

Estas variables se detallan más en el apartado "Explicación de variables" de este documento. Si alguna variable de dificultad no encajase con las mecánicas del juego planteado, se podría ignorar. Si algún equipo quisiera añadir más variables de entrada o de salida podría hacerlo.

Para simplificar la implementación y evitar problemas de compatibilidad, se proporcionará a los desarrolladores un proyecto de Unity con las librerías de OpenXR instaladas y una API para cargar y guardar las estructuras de datos necesarias para comunicarse con los modelos desarrollados en el reto de IA. Los equipos podrán presentar proyectos desarrollados con cualquier otra herramienta capaz de crear videojuegos de realidad virtual, como Unreal, Godot u otras.

# RETO DE INTELIGENCIA ARTIFICIAL

La gamificación es una técnica de aprendizaje que traslada la mecánica de los juegos al ámbito educativo-profesional con el fin de conseguir mejores resultados: sirve para absorber conocimientos, para mejorar alguna habilidad o para recompensar acciones concretas.

En este caso concreto se plantea el **análisis de los datos** generados en el reto de RV anterior, sobre un minijuego de pesca. Para ello se podrá modificar la dificultad de diferentes parámetros del juego:

- Ratio de aparición de los objetos en el juego.
- Distancia al jugador de los objetos.
- Tamaño de los objetos.
- Ratio de recompensa de los objetos.

La dificultad podrá variar según los siguientes valores:

- Tiempo que ha durado la ronda.
- Puntuación conseguida total.
- Puntuación del objeto que menos puntuación le ha dado.
- Puntuación del objeto que más puntuación le ha dado.
- Número de objetos capturados.
- Número de objetos perdidos.
- Tiempo de respuesta medio.
- Tiempo de respuesta mínimo.
- Tiempo de respuesta máximo.

La intención del juego será mantener al jugador interesado en el mismo, no dificultando el juego hasta valores que sean imposibles de conseguir y realizando un juego demasiado simple que no reporte satisfacción al usuario.

Para simplificar la implementación y evitar problemas de compatibilidad, se proporcionará a los desarrolladores un conjunto de datos base que coincide con las estructuras generadas por el videojuego y que permita la comunicación del modelo con los videojuegos.

Además, se proporcionará un servidor FastAPI implementado en Python que permitirá conectarse con los juegos y al que los equipos tendrán que añadir su inferencia. Por consiguiente, el objetivo será generar un modelo de la dificultad del juego y generar el data set de variables como motor de IA para el reto de RV.

# EXPLICACIÓN DE VARIABLES

A continuación, se detallan las variables empleadas en los retos que componen el presente desafío.

## VARIABLES GENERADAS POR EL JUEGO

- Tiempo que ha durado la ronda.
  - Medido como un float, en segundos.
- Puntuación conseguida total.
  - Suma de las puntuaciones que le han dado cada uno de los objetos.
  - Medido como un entero sin límites inferiores o superiores.
- Puntuación del objeto que menos puntuación le ha dado.
  - Podría ser negativa si hay objetos que resten. Por ejemplo, pescar algas en un juego de pesca.
  - Las puntuaciones de los objetos se comunicarán entre la IA y el juego como floats normalizados, entre -1 y 1. -1 representará la puntuación más baja que un objeto puede dar y 1 la más alta.
- Puntuación del objeto que más puntuación le ha dado.
  - Las puntuaciones de los objetos se comunicarán entre la IA y el juego como floats normalizados, entre -1 y 1. -1 representará la puntuación más baja que un objeto puede dar y 1 la más alta.
- Número total de objetos capturados.
  - Medido como un entero positivo.
- Número total de objetos perdidos.
  - Medido como un entero positivo.
- Tiempo de respuesta medio.
  - Medido como un float.
  - Tiempo medio que ha tardado un jugador en capturar un objeto desde que aparece.
  - No incluye los objetos que se le han escapado.
- Tiempo de respuesta mínimo.
  - Medido como un float.
  - Tiempo mínimo que ha tardado un jugador en capturar un objeto desde que ha aparecido.
- Tiempo de respuesta máximo.
  - Medido como un float.
  - Tiempo máximo que ha tardado un jugador en capturar un objeto desde que ha aparecido.
  - No incluye los objetos que se le han escapado.

## VARIABLES GENERADAS POR EL MODELO DE IA

- Ratio de aparición de los objetos en el juego.
  - Medido como un float entre 0 y 1. 0 representará la velocidad de aparición más lenta, siendo más **fácil** capturar los objetos, y 1 representará la más rápida.
- Distancia al jugador de los objetos.
  - Medido como un float entre 0 y 1. 0 representará la distancia más cercana al jugador, siendo más **fácil** capturarlo, y 1 la más lejana.
  - Esta variable debería afectar a la generación aleatoria, no aparecer todos a exactamente la misma distancia.
- Tamaño de los objetos.
  - Medido como un float entre 0 y 1. 0 representará los objetos más pequeños, siendo más **difícil** capturarlos, y 1 los más grandes.
- Ratio de recompensa de los objetos.
  - Medido como un float entre -1 y 1. Representará la proporción de objetos que aparecen con diferentes puntuaciones. -1 implicará que aparecerán más objetos con puntuaciones más bajas (o negativas), siendo más **difícil**. 1 representará objetos con puntuaciones más altas, siendo más **fácil**.

# PUNTUACIÓN DE SOLUCIONES

El jurado evaluará las soluciones presentadas, de acuerdo con los siguientes criterios:

## 1. Innovación y Creatividad (20 puntos)

- *Originalidad de la Idea (0-10 puntos):* Un concepto novedoso o un enfoque único para resolver un problema.
- *Potencial de Disrupción (0-10 puntos):* Capacidad para cambiar significativamente el campo en el que se enmarca el desafío.

## 2. Impacto y Aplicabilidad (20 puntos)

- *Solución al Problema (0-10 puntos):* Eficacia con que la solución aborda el problema.
- *Beneficio para el Usuario/Comunidad (0-10 puntos):* Valor agregado para usuarios finales o la comunidad.

## 3. Calidad de la Solución y Explicabilidad (30 puntos)

- *Desarrollo y Calidad Técnica (0-20 puntos):* Solidez técnica y calidad de la solución presentada.
- *Explicabilidad (0-10 puntos):* Claridad en cómo el sistema toma decisiones o proporciona resultados.

## 4. Presentación y Demostración (30 puntos)

- *Claridad de la Presentación (0-15 puntos):* Cómo se presenta el proyecto, sus objetivos y resultados.
- *Demostración del Producto (0-15 puntos):* Calidad de la demostración en vivo del proyecto desarrollado.

**Puntuación Total: 100 puntos**