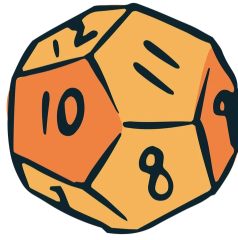


## ## RETO 1: ASISTENTE DE AVENTURAS DE ROL (Nivel Accesible)



### Objetivo

Crear un Game Master con Inteligencia Artificial que dirija una partida breve de rol para un solo jugador.

### Duración

12 horas, de 09:00 am a 21:00 pm.

### Descripción

Se proporcionarán dos documentos en PDF como referencia, con las reglas y hojas de personaje de Knave. El agente Game Master debe ser capaz de llevar una partida breve para un jugador a través de un chatbot. La partida debe contener los siguientes elementos:

- Bienvenida e identificación del jugador
- Creación de personaje, si es un personaje nuevo. Cargar un personaje ya existente. Borrar un personaje ya existente.
- Batalla contra un enemigo
- Resumen de victoria o derrota
- Actualización del personaje.

Se dejan fuera elementos de historia o roleplaying.

Algunos documentos están listos para utilizar, otros requieren cierta interpretación por parte del game master.

### Material proporcionado

- PDF de personajes
- PDF de reglas Knave
- URL de acceso a bestiario OSR
- Explicación sobre adaptación de los monstruos del bestiario a las reglas Knave

*Todos estos archivos se os darán en una carpeta de Google Drive el día del Hackaton a las 08:45. Se os enviará la URL al mismo en un correo.*

### Componentes a implementar

1. Arquitectura en No-Code en la plataforma N8N

Se utilizará N8N como plataforma en la que desarrollar la mayor parte del backend (llamadas API, carga de documentos, llamadas a base de datos, etc...) así como el agente de IA.

## 2. Agente de IA mediante API

Es crucial que el Game Master esté basado en la tecnología de Agentes y creado en N8N. Se pueden utilizar los modelos de GPT, Anthropic, Llama y/o Grok en las versiones que resulten más óptimas en términos de recursos/resultados. Se debe justificar, eso sí, el por qué se opta por uno o por otro.

## 3. Microservicio en Código Propio

Se debe implementar al menos un microservicio backend, a elección del desarrollador, escrito en Python y desplegado en Heroku.

Se puede elegir cualquier elemento del juego para llevarlo a un microservicio. Se permite el uso de IA para su creación, pero debe estar documentada. Además, el desarrollador deberá poder responder preguntas sobre su diseño y funcionalidad.

## 3. Despliegue en Heroku del microservicio que será llamado desde la arquitectura de N8N

## 4. Interfaz de Usuario

- Aquí el desarrollador podrá elegir entre crear una interfaz de chatbot desde cero y alojada en un servidor (no contará como “Microservicio en Código Propio”), o implementar el agente en una interfaz de chat ya creada como Telegram, Whatsapp, Discord... En este segundo escenario será necesario crear una interfaz por encima del chat que redirija a una de estas plataformas.

## 5. Personajes persistentes

-Se debe crear una base de datos que permita al usuario almacenar sus personajes en su estado actual, crear uno nuevo o eliminar los ya existentes.

-Por agilidad y acceso gratuito recomendamos usar Airtable o Supabase.

### **Nota adicional: Enemigos.**

-Se aporta una URL con una gran variedad de enemigos. El desarrollador podrá elegir uno al azar para el agente. Se valorará positivamente si se consigue que sea el propio Game Master el que escoja un enemigo por sí mismo, haciendo que cada partida sea diferente y si es capaz de implementar elementos de balanceo.

### **Criterios de evaluación.**

-Se valorará la creatividad, optimización y modularización de la arquitectura propuesta en N8N.

-Se valorará la calidad del código alojado en Heroku (modularidad, comentarios, pruebas unitarias).

-Se valorará la calidad del prompting utilizado para construir al agente Game Master, su robustez frente a los fallos que puedan dar otros elementos del sistema (errores en tools, en llamadas HTTP, en errores de cálculos, en errores de acceso a base de datos, etc...).

-Se valorará la capacidad de resolver problemas técnicos dentro del periodo de 12 horas para obtener el mejor resultado posible.

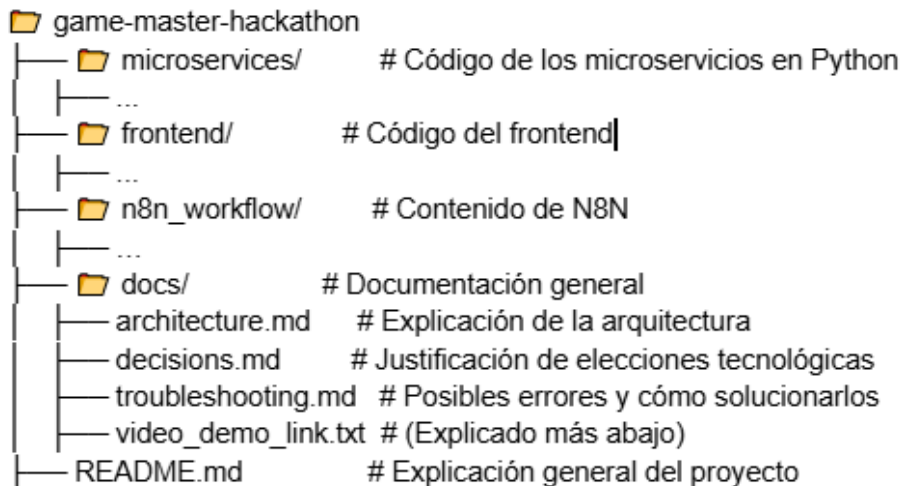
-Se evaluará la resistencia de la arquitectura contra intentos de extracción de información mediante inyecciones de prompting o ataques de spam.

-Como extra, se valorará el apartado visual de la solución que facilite su uso e invite al juego.

## Entregables

Los participantes deberán entregar un repositorio público en GitHub que contenga todo lo necesario para evaluar su solución. *(Esto aplica para los que optan por afrontar solo el RETO 1, pero también a los que quieren añadir algún extra del RETO 2 o el 3).*

### Estructura recomendada del github



Se puede modificar la estructura de ficheros según sea necesaria. Por ejemplo, si se quiere añadir un fichero para documentar lo construido en un Airtable, con su explicación de diseño, de campos, etc... puede hacerse.

De N8N se espera el JSON de los workflows, documentación explicativa junto con pantallazos y/o vídeos demostrativos de componentes.

**En el README.md debes indicar tu nombre y apellidos y dirección de correo electrónicos con los que te registraste al hackaton.**

Además, el equipo de People1st desplegará la solución y la probará con diferentes aproximaciones para comprobar el resultado durante la semana siguiente a la entrega, por lo que se recomienda tener los servicios disponibles durante esa semana.

**Adjuntar el vídeo en video\_demo\_link.txt:** Se espera una URL a un vídeo de Youtube *unlisted*. El vídeo no puede tener más de 5 minutos de duración. Debe mostrar cómo el usuario interactúa con el chatbot para resolver una partida (o las que se quieran enseñar). Si se quieren hacer cortes, añadir subtítulos, y demás, queda a elección de cada uno. Recordar que el equipo probará la solución y comprobará tiempos de carga/procesamiento/etc...

## ## RETO 2: GAME MASTER EVOLUCIONADO (*Nivel Hardcore*)



### Objetivo

Llevar el Game Master a un nivel superior con IA adaptativa, multijugador cooperativo y un sistema de reputación para mejorar la inmersión en el juego. Este reto parte de haber desarrollado el RETO 1. Se pueden añadir todas las mejoras que se quieran/puedan. Estas demostrarán un *plus* en tus habilidades, así como en tu energía creativa.

---

### Mejoras a implementar

#### 1- IA Adaptativa para el Game Master (+2 punto)

- El Game Master debe ajustar la dificultad del combate según el desempeño del jugador.
- El Game Master justificará los cambios narrativamente.

#### ¿Cómo se evalúa?

- Se revisará que la dificultad cambia dinámicamente.
  - Se valorará si el ajuste parece natural y no arbitrario.
- 

#### 2 - Modo Cooperativo: Chatbot para 2 Jugadores + Agente Game Master (+4 puntos)

- Permitir que dos jugadores jueguen juntos en el mismo chatbot.
- El Game Master debe manejar los turnos y describir lo que sucede a ambos jugadores.
- Ambos deben poder decidir sus acciones en el combate de manera fluida.
- Si la experiencia no es fluida o se atasca, no cuenta como logrado.

#### ¿Cómo se evalúa?

- Se verificará que ambos jugadores puedan participar sin errores.
- Se valorará la claridad de la narración y la gestión de turnos.

---

### **3 - Sistema de Reputación (+1 punto)**

- A medida que el jugador vence enemigos, su reputación en el mundo cambia.
- El Game Master debe generar mensajes personalizados sobre cómo se expande su fama antes o después de cada batalla, recordando eventos pasados.

#### **¿Cómo se evalúa?**

- Se verificará que el sistema registra eventos pasados y los usa para la narrativa.
- Se valorará si la reputación afecta el desarrollo del juego.

¿Quieres demostrar que estás por encima de todos los demás participantes?

## ## RETO 3: LangChain



Si de verdad estás hecho de otra pasta, tenemos este extra para tí. Es requisito indispensable haber desarrollado el RETO 1 al completo y haber implementado al menos hasta un +3 *puntos* del RETO 2.

Es crucial que dejes constancia en la documentación del funcionamiento del RETO 1 + los extras de RETO 2 aparte de demostrar los resultados del RETO 3.

### Objetivo

Mover el Agente Game Master a LangChain, alojarlo en Heroku y llamarlo desde N8N.

Importante:

- N8N seguirá gestionando la orquestación (llamadas API, base de datos, lógica de control).
- El Agente Game Master en LangChain no sustituye al microservicio Python.

### Evaluación

-Se valorará la **fluidez** de la solución con LangChain implementado.

-Se valorará que el **microservicio en LangChain esté bien estructurado** y siga las **buenas prácticas de desarrollo de APIs** (endpoints bien definidos, manejo de errores, etc.).

-Se valorará que la **arquitectura sea modular**, permitiendo añadir o quitar funciones sin alterar el flujo de N8N (por ejemplo, modificar el pool de enemigos, cambiar reglas, ajustar el acceso a bases de datos o añadir servicios como tiradas de dados).

-Se valorará una **explicación de los retos técnicos** que supuso mover el agente a código propio en vez de utilizar N8N.