

Enunciado de la práctica final

Fecha entrega: Jueves 15 de enero, 23:55 a través de AG

Práctica Final: Sistema inteligente de apoyo al brainstorming con privacidad total

Imagina que estás trabajando en un laboratorio de I+D donde la privacidad es crítica. Tu equipo necesita realizar una sesión de ideación (*brainstorming*) intensa, dibujar esquemas en la pizarra y consultar documentos confidenciales. Sin embargo, no puede permitirse que el audio, los dibujos o los documentos suban a la nube de terceros (como OpenAI o Google) por riesgo de filtraciones. Además, las sesiones suelen volverse caóticas: se pierde el foco, se olvidan datos técnicos y las ideas se mezclan sin estructura.

Para abordar estos desafíos, se propone el desarrollo de un sistema inteligente colaborativo que funciona enteramente en el navegador. Este sistema no sólo transcribe, sino que actúa como un facilitador experto utilizando la metodología de los "Seis Sombreros para Pensar" (Six Thinking Hats) mediante agentes de IA locales.

El sistema escucha, entiende bocetos en la pizarra y responde preguntas sobre documentos PDF internos, todo ello sin que un solo byte de datos salga del ordenador local.

El objetivo del proyecto es diseñar e implementar este sistema de apoyo inteligente que asista en reuniones presenciales mediante:

1. **Transcripción Privada:** Reconocimiento de habla local
2. **Razonamiento Agéntico:** Clasificación y generación de ideas estructuradas (Metodología Seis Sombreros).
3. **Multimodalidad:** Análisis inteligente de diagramas o bocetos realizados en la pantalla.
4. **Memoria Contextual (RAG):** Consultas a documentos PDF locales.

Requisitos Técnicos

El sistema debe desarrollarse utilizando tecnologías web modernas y la librería transformers.js. No se permite el uso de APIs de servidor ni servicios en la nube (pp. ej., API OpenAI). Todo debe ejecutarse en el cliente (navegador).

1. **Transcripción privada (Whisper Local):**
 - Utilizar Xenova/whisper-small (o modelos cuantizados similares) para transcribir el audio de la reunión en tiempo real.
 - Gestionar un buffer de conversación para mantener el contexto inmediato.
2. **Razonamiento agéntico ("Seis Sombreros"):**
 - Implementar un flujo de trabajo agéntico que analice la transcripción.

El sistema no debe ser un "chatbot" lineal que responde lo que sea. Debe comportarse como un equipo de expertos, cada uno con una función específica.

Para lograr esto, usaremos la metodología de los Seis Sombreros de Pensamiento (Edward de Bono), una metodología de pensamiento paralelo que divide la toma de decisiones en seis roles o perspectivas definidas (como hechos, emociones, riesgos o creatividad) para analizar un problema de manera completa, ordenada y sin conflictos.

Rol: Objetivo, neutral. Busca datos y hechos. Ejemplo: Si alguien dice "Creo que vendimos poco el mes pasado", el Sombrero Blanco busca en el PDF (RAG) y dice: "Dato: Las ventas de noviembre fueron 15.000€, un 10% menos que octubre."	Rol: Emociones, intuición. No necesita justificación. Ejemplo: Si la discusión se calienta, el Sombrero Rojo detecta: "Atención: El tono de la reunión se está volviendo agresivo. Sugiero una pausa."	Rol: Beneficios, valor, positividad. Ejemplo: "Beneficio: Si usamos blockchain, la trazabilidad será automática y transparente."
Rol: Nuevas ideas, alternativas, crecimiento. Ejemplo: "Idea: ¿Y si en lugar de blockchain usamos una base de datos distribuida simple para empezar?"	Rol: Control del proceso, organización, resúmenes. Ejemplo: "Resumen: Hemos dedicado 20 minutos al tema técnico. Falta decidir el presupuesto."	Rol: Juicio crítico, riesgos, cautela. Ejemplo: Ante una nueva idea ("Usemos blockchain"), el Sombrero Negro advierte: "Riesgo: Eso aumentará el coste de desarrollo en un 40% y no tenemos expertos en el equipo."

- El sistema debe ser capaz de clasificar o generar respuestas bajo "**roles**" o **sombreros diferentes**
- Es necesario utilizar un flujo de este tipo
 - Orquestador: Tarea zero-shot-classification para decidir que tipo de intervención se requiere. Definir algunas etiquetas candidatas, que describirán el user intent.
 - Ejecución del agente: solo se activa el agente seleccionado por el orquestador: utilizar un modelo de generación de texto (LaMini-Flan-T5 o Gwen) con un prompt dinámico de acuerdo con el rol
 - Acción (Interfaz de usuario): modificar la interfaz, por ejemplo creando notas del color correspondiente

3. Visión multimodal:

- Implementar un lienzo digital (Canvas) donde los usuarios puedan dibujar, por ejemplo, diagramas.
- Integrar un modelo multimodal (MLLM) ligero (como Janus vía WebGPU) que permita al usuario pulsar un botón "Analizar Pizarra" y recibir feedback de la

IA sobre lo dibujado (ej. "Mejorar la usabilidad de este esquema").

Sugerencias:

- Ligero, Janus:
<https://huggingface.co/onnx-community/Janus-Pro-1B-ONNX>
- Más pesado, pero con mejor capacidad de razonamiento,
Minstral-3-3B:
<https://huggingface.co/mistralai/Minstral-3-3B-Instruct-2512-ONNX>

4. Memoria RAG (Retrieval-Augmented Generation):

- Permitir al usuario arrastrar y soltar un documento PDF (p., ej. un informe técnico).
- El sistema debe "leer" el PDF, generar embeddings locales (usando modelos como all-MiniLM-L6-v2) y permitir al usuario hacer preguntas verbales sobre el documento (p. ej. "¿Cuál fue el presupuesto del año pasado?").

Evaluación (Video demostrativo)

Como parte de la evaluación final, los estudiantes deben realizar un video demostrativo de **máximo 3 minutos**. En el video, el grupo simulará una sesión de brainstorming presencial alrededor del ordenador/pantalla que ejecuta el sistema.

El video debe incluir:

1. **Introducción:** Contexto de la reunión simulada (ej. "Diseñando una nueva App").
2. **Dinámica de los sombreros:** Mostrar cómo, al discutir una idea, el sistema genera automáticamente notas, por ejemplo con críticas (Sombrero Negro) o ideas nuevas (Sombrero Verde).
3. **Interacción multimodal:** Un estudiante dibuja un esquema rápido en la pantalla y pide al sistema que lo interprete/mejore.
4. **Uso de RAG:** Un estudiante hace una pregunta fáctica cuya respuesta está en el PDF cargado, y el sistema responde correctamente.

Puntuación

La nota final se basará en la complejidad técnica, la integración de los componentes y la creatividad de la solución.

1. Transcripción en vivo (0.25 punto)

- Calidad de la integración de Whisper en el navegador y en la interfaz.
- Visualización fluida del texto transcrita ("Live captions") y su uso en el sistema.

2. Implementación de agentes "Seis Sombreros" (1.25 punto)

- Calidad y diferenciación de las respuestas de cada "sombrero". Que el Sombrero Negro suene crítico y el Verde creativo.
- ¿Se ejecutan todos a la vez o se activan bajo demanda/detección de intención? Se valorará el uso de *Zero-Shot Classification* para activar el agente correcto.
- Cómo se visualizan estas ideas en la pizarra (uso de colores, tarjetas, organización automática).

3. Multimodalidad y pizarra (1.25 punto)

- Implementación del canvas para dibujar y visualizar ideas.
- Integración exitosa del modelo de Visión-Lenguaje (MLLM).
- Capacidad del sistema para entender el contexto visual del dibujo (no solo describir, sino aportar valor).
- Uso de distintas técnicas de interacción con el texto transcrita.

4. RAG local (0.75 puntos)

- Funcionalidad de "Drag & Drop" de PDFs.
- Eficiencia en la búsqueda semántica (uso de embeddings locales).
- Precisión de la respuesta generada basada en el documento.

5. Memoria y Video (0.5 puntos)

- **Memoria técnica:** Explicación clara de la arquitectura, modelos elegidos (y por qué), y optimizaciones de rendimiento en el navegador.
- **Video:** Calidad de la narrativa, claridad en la demostración de las funcionalidades y escenificación del caso de uso.

Normas de entrega:

Uno de los integrantes del grupo entregará en AG un fichero comprimido (.zip) con:

- Código fuente: Incluye todos los archivos necesarios para ejecutar la aplicación (HTML, CSS, y JavaScript).
- Memoria (.pdf) con los nombres y NIA de cada integrante del grupo
- Video demostrativo de máximo 3 minutos