

¡Ya casi empezamos!

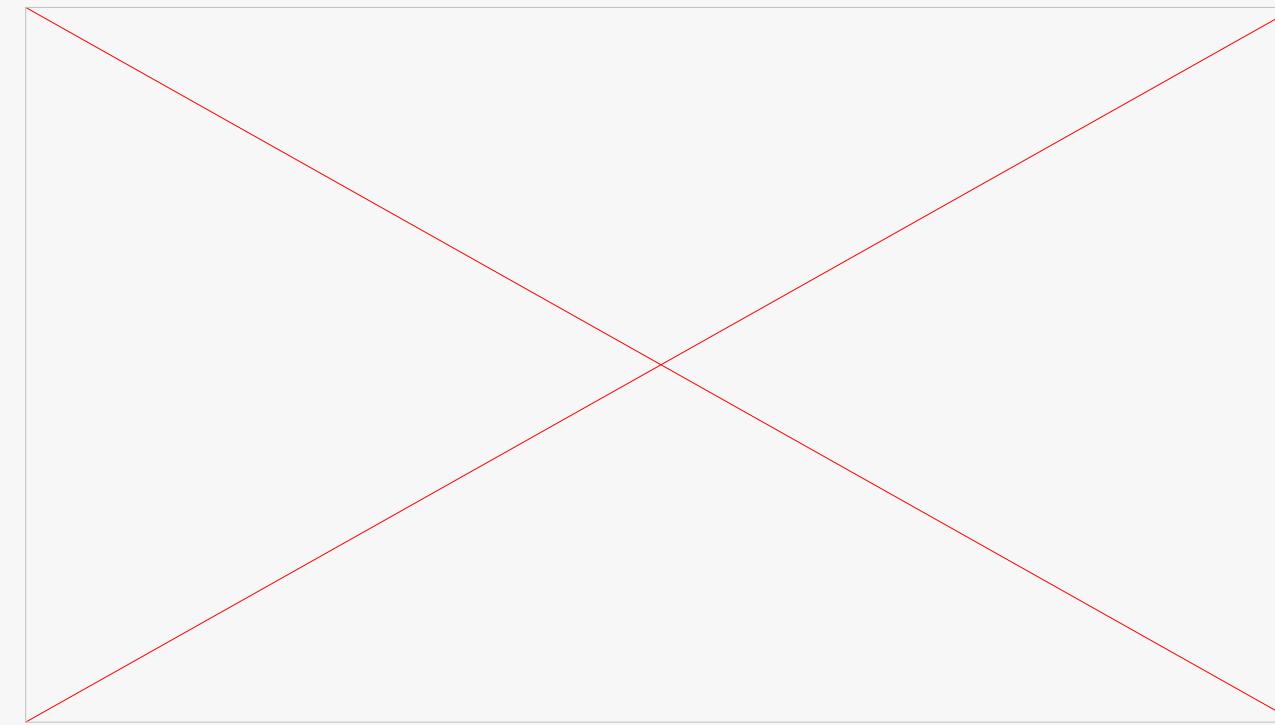


Clase 4

Fundamentos de IA y plataforma No Code



Hola! Soy Fede, un gusto



LinkedIn: Federico Apfelbaum

Hoja de Ruta



1. Introducción y objetivos
2. Máquinas que aprenden - ¿Qué es la Inteligencia Artificial?
3. Fundamentos del Prompt Engineering
4. Ejercicio práctico - Buenas prompts, marcan la diferencia
5. Riesgos de Seguridad en el uso de la IA
6. No Code - Qué es y cuándo usarlo
7. Ejercicio práctico - Crear cuenta en Make
8. Cierre, preguntas y próxima clase



1. Objetivos de la Clase

-  **Comprender el prompting, y las buenas prácticas implicadas**
-  **Entender los riesgos de seguridad y privacidad implicados, y cómo proteger información sensible**
-  **Realizar ejercicios prácticos para notar el resultado de usar buenas prácticas**
-  **Conocer las principales plataformas de IA del mercado**

2. Máquinas que aprenden

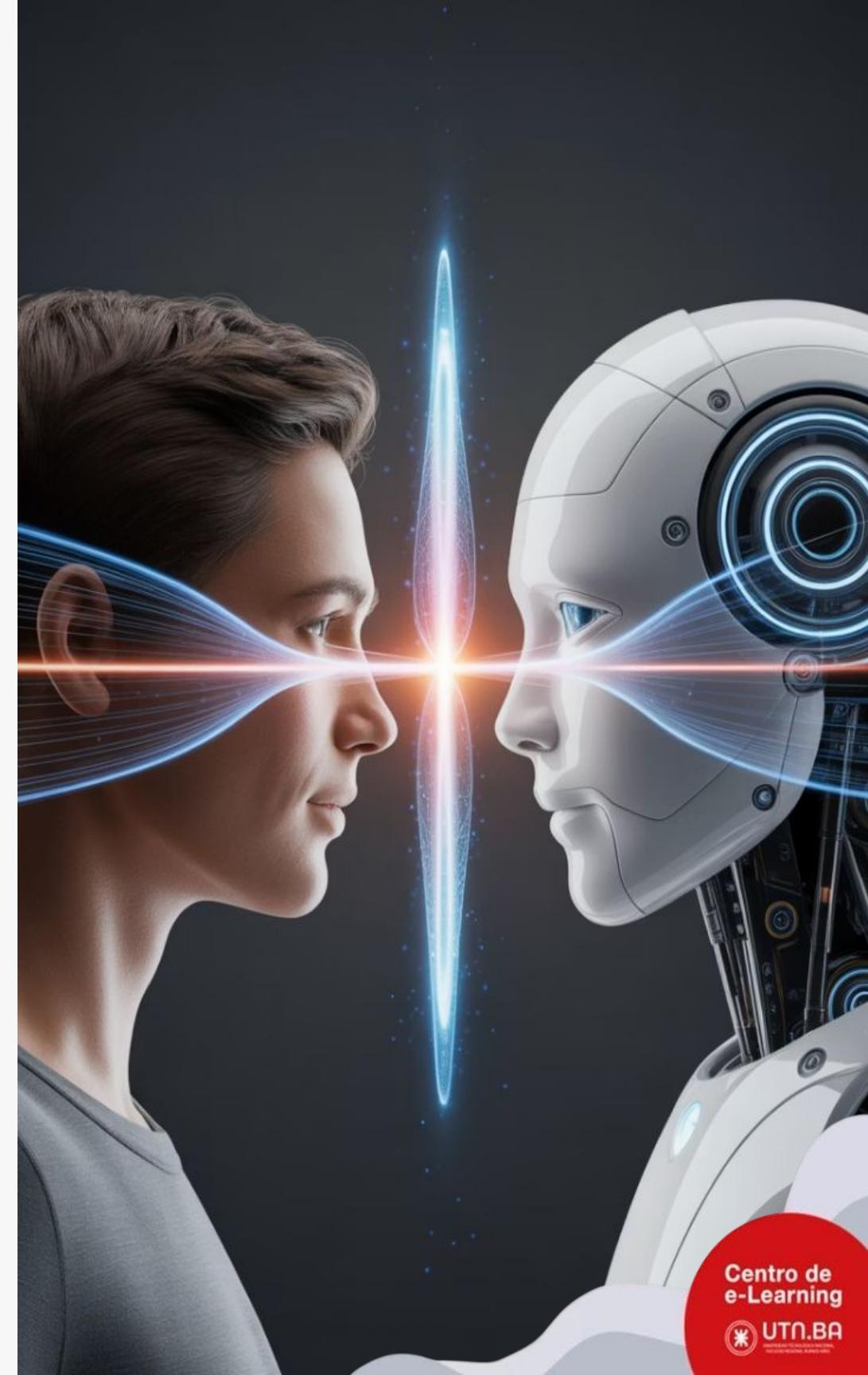
- **Datos y entrenamiento:** el modelo se alimenta de ejemplos etiquetados para aprender a reconocer patrones.
- **Extracción de patrones:** el algoritmo identifica relaciones y estructuras ocultas en los datos.
- **Retroalimentación iterativa:** se evalúa su desempeño y se ajustan parámetros para mejorar resultados.
- **Validación y generalización:** se prueba con datos nuevos para garantizar que no se sobreajuste al conjunto de entrenamiento.



Diferentes IAs



- IA Débil: sistemas entrenados para tareas específicas (recomendaciones, clasificación, chatbots).
- IA Generativa: modelos capaces de crear contenido original (texto, imágenes), interactuando mediante lenguaje natural.



3. Fundamentos del Prompt Engineering

Prompts

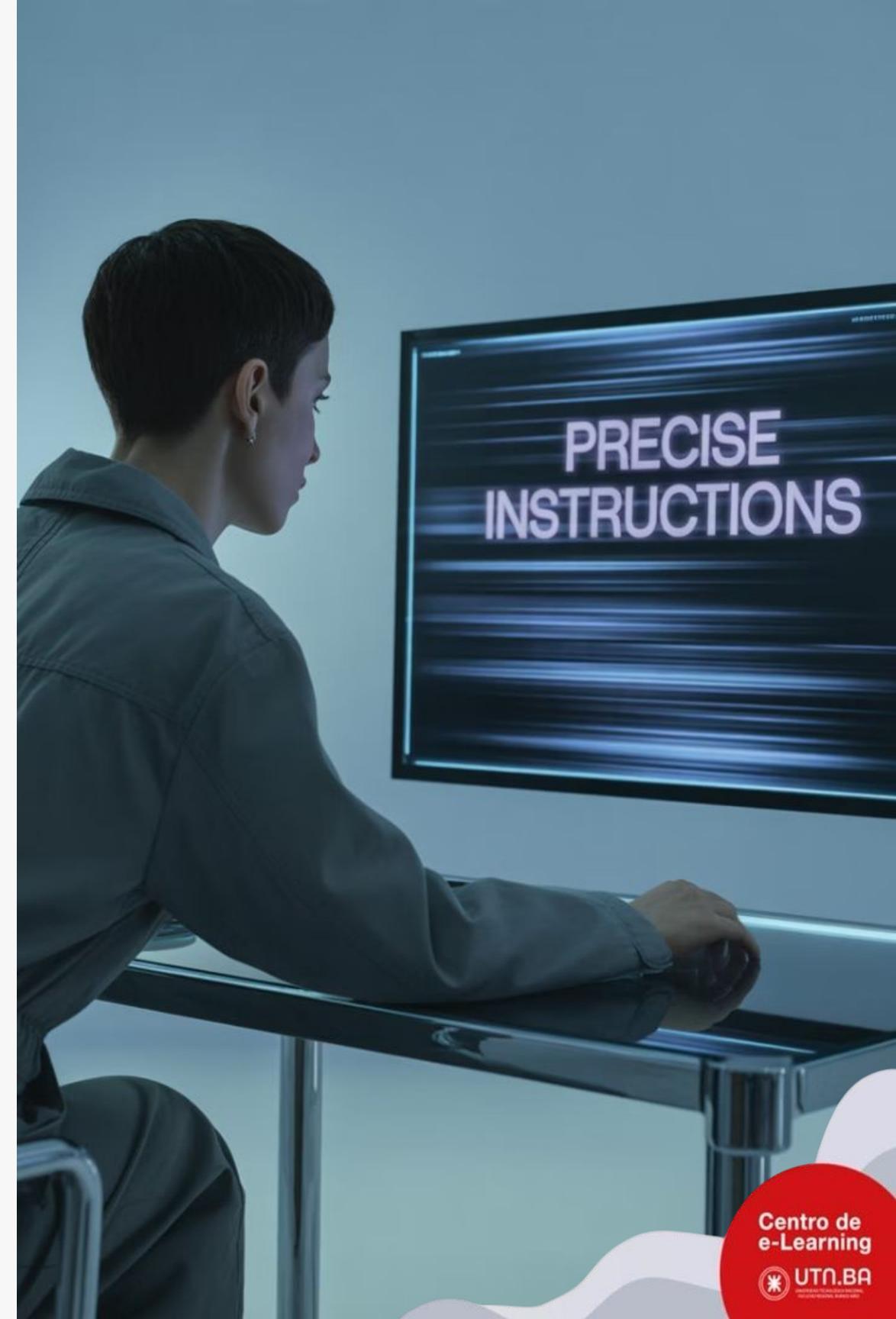
Son instrucciones, indicaciones, guías al modelo para que pueda realizar una tarea que requerimos.

Prompt Engineering

Consiste en diseñar instrucciones y entradas (inputs) para guiar al modelo hacia respuestas más precisas y útiles.

Iteración

Proceso de ajuste de la prompt basándose en respuestas previas para optimizar resultados.





Técnicas Avanzadas de Prompting



Role Prompting

Asigna un "rol" al modelo para mejorar el estilo y la relevancia.

Goal oriented

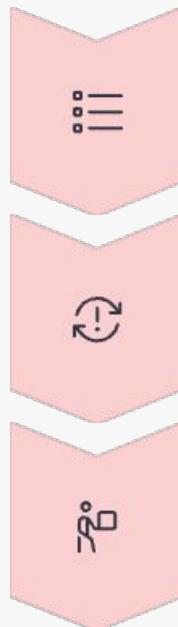
Definir claramente cuál es el objetivo que tiene el modelo al recibir esta prompt.

Chain-of-Thought

Solicita al modelo que piense paso a paso para resolver problemas complejos.



Técnicas Avanzadas de Prompting



Few-Shot Prompting

Proporciona ejemplos de entrada y salida para guiar el formato deseado.

ReAct

Combina razonamiento y acciones dentro del flujo de prompts.

Output Format

Especificar en qué formato necesitamos que nos responda el modelo (párrafo breve, informe técnico, json, tabla, etc.)

4. Ejercicio práctico

"Buenas prompts, marcan la diferencia"

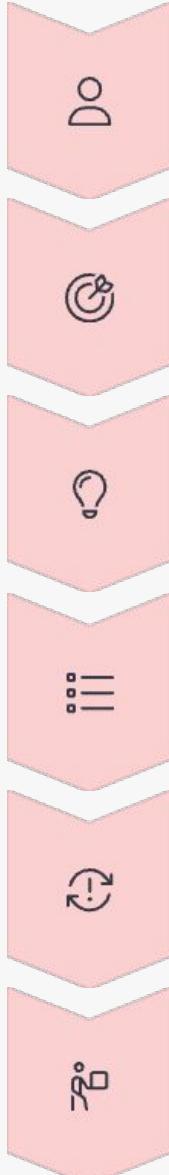
Prompt base: **Soy {rol/profesión}, ¿como puedo mejorar mi productividad"**

Ejercicio

1. Probar el resultado de la prompt base
2. Rearmar la prompt, agregando (mínimo) los siguientes aspectos:
 - Rol
 - Darle contexto al modelo
 - Aclarar formato de salida



Técnicas de prompting



Role Prompting

Asigna un "rol" al modelo para mejorar el estilo y la relevancia.

Goal oriented

Definir claramente cuál es el objetivo que tiene el modelo al recibir esta prompt.

Chain-of-Thought

Solicita al modelo que piense paso a paso para resolver problemas complejos.

Few-Shot Prompting

Proporciona ejemplos de entrada y salida para guiar el formato deseado.

ReAct

Combina razonamiento y acciones dentro del flujo de prompts.

Output Format

Especificar en qué formato necesitamos que nos responda el modelo (párrafo breve, informe técnico, json, tabla, etc.)



Ejemplo prompt:

Eres un asistente de IA especialista en productividad profesional, en el rubro de la consultoría de software.

Genera una lista de 5 consejos concretos para mejorar la productividad diaria de un profesional en el área de consultoría de software, que se dedica a la implementación de inteligencia artificial y automatización de procesos empresariales de Argentina, para clientes tanto locales como extranjeros.

Cada consejo debe incluir:

- Título breve (máx. 5 palabras).
- Descripción de cómo implementarlo (1–2 oraciones).
- Herramienta o técnica recomendada, preferentemente con versiones gratuitas.
- Tiempo estimado para ponerlo en práctica (p. ej. 5 min, 30 min, 1 día).

Necesito el informe en un lenguaje claro y directo, de no más de 120 palabras en total.



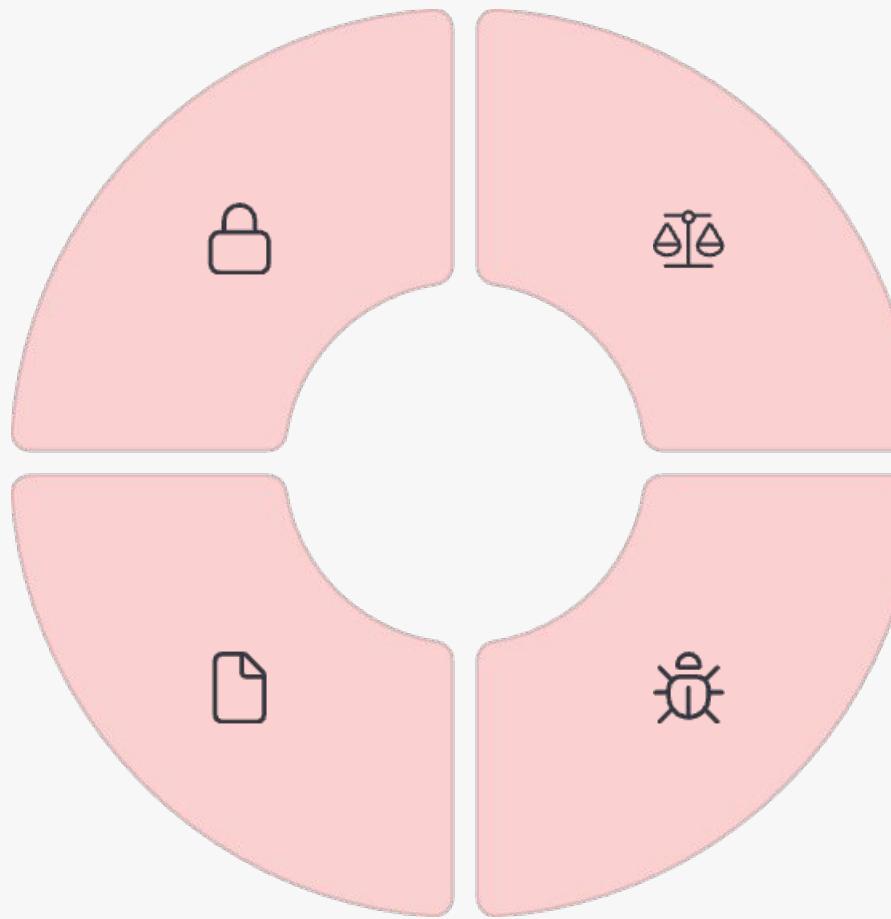
5. Riesgos de Seguridad en IA

Exposición de Datos

Información personal en prompts puede resultar en filtraciones.

Normativa

GDPR, LGPD y CCPA exigen protección de datos personales.



Sesgos

Modelos con datos históricos pueden perpetuar prejuicios.

Ataques Adversarios

Perturbaciones en la entrada pueden inducir errores críticos.



Buenas Prácticas de Seguridad

Ocultar Información Personal

Borra o cambia nombres, direcciones o detalles identificativos antes de compartir datos.

Protección de Comunicaciones

Usa conexiones seguras ([https](https://)) y almacenamiento protegido con contraseñas fuertes.

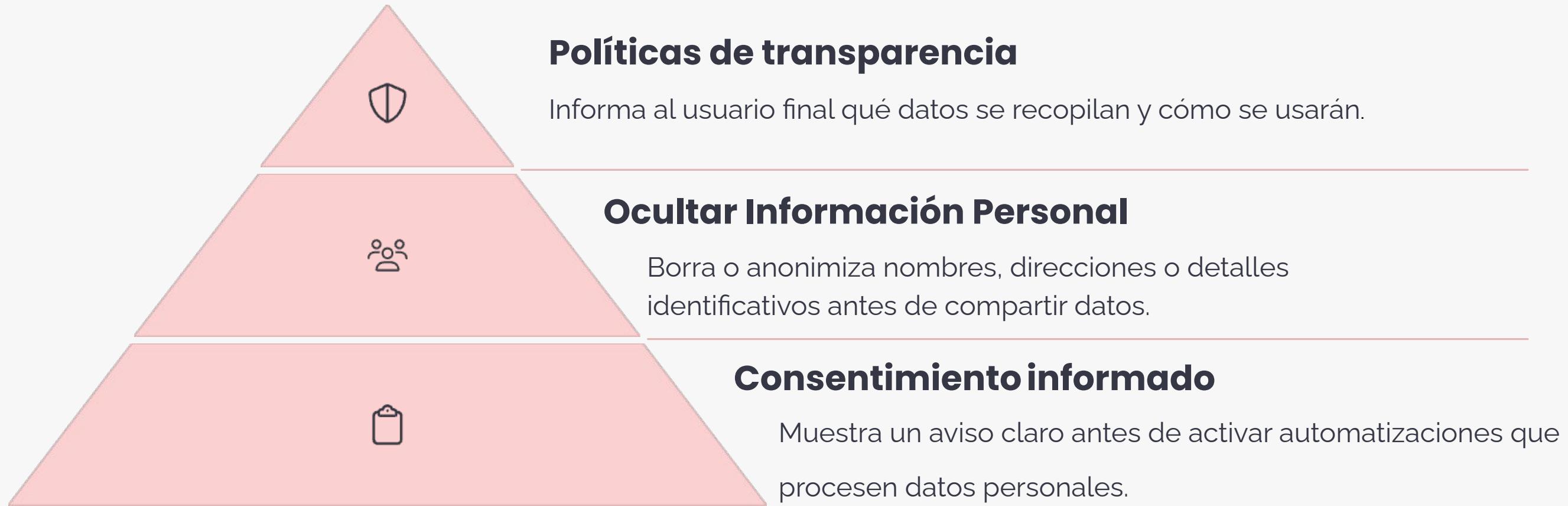
Gestión de Claves

Guarda llaves de API en sitios especiales y protegidos, no en archivos sin protección.

Permisos Mínimos

Da acceso solo a lo estrictamente necesario para minimizar impactos de errores.

Usuarios finales y privacidad de datos



6. No Code: ¿Qué es?

Las plataformas No-Code democratizan el desarrollo de software, acelerando la entrega de soluciones tecnológicas.



Interfaces Gráficas

Entornos visuales que permiten construir aplicaciones arrastrando y soltando componentes predefinidos.



Colaboración Multidisciplinar

Tanto perfiles técnicos como de negocio pueden participar activamente en el desarrollo.



Mínimo Código Manual

Reduce significativamente la necesidad de escribir código complejo, simplificando el proceso de desarrollo.

¿Cuándo usarlo?

- **Prototipado y validación rápida**

Cuando necesitas probar una idea o MVP en días (no meses) para obtener feedback temprano antes de invertir en desarrollo a medida.

- **Procesos internos y automatizaciones repetitivas**

Flujos de trabajo administrativos, integraciones entre apps (correo ↔ hojas de cálculo ↔ CRM) o generación de reportes que hoy se hacen manualmente.

- **Escasez de recursos técnicos**

Equipos sin desarrolladores disponibles, o con backlog lleno, que igualmente necesitan entregar valor rápidamente.

- **Necesidad de iteración constante**

Proyectos donde los requisitos cambian a menudo; el enfoque No Code permite ajustes en minutos sin ciclos largos de despliegue.

- **Conectividad entre sistemas “legacy”**

Cuando hace falta crear “puentes” entre plataformas que no se hablan mediante APIs estándar, evitando revisiones de código de sistemas heredados.

Indicadores de que quizá NO sea la mejor opción

- Requisitos de rendimiento **extremo** o alta concurrencia
- Lógica de negocio muy **compleja**, con reglas cambiantes y cientos de dependencias
- Necesidad de control fino sobre la seguridad o cumplimiento regulatorio estricto
- Planes de escala masiva (**millones de usuarios**) que superen los límites de la plataforma No Code

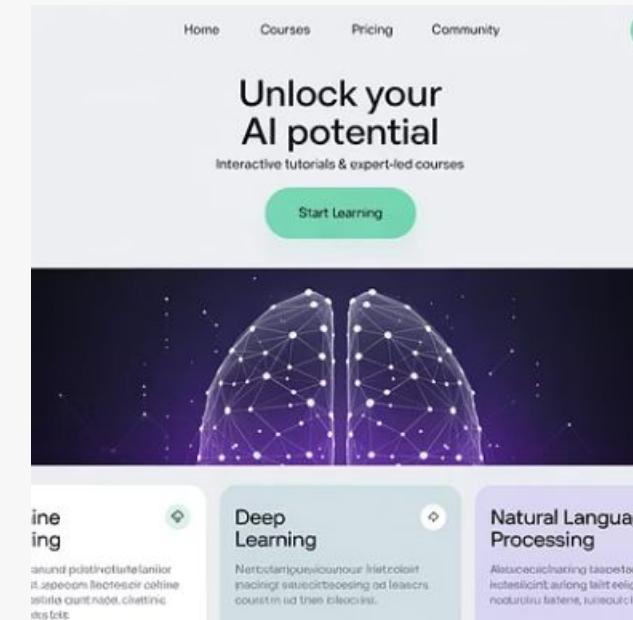
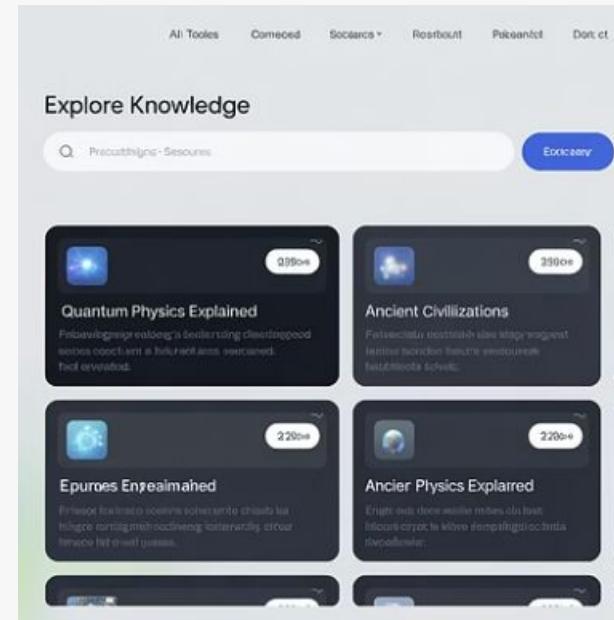
7. Ejercicio práctico

Ejercicio

1. Ingresar a <https://www.make.com/en/register>
2. Crear cuenta gratuita de Make



Recursos Adicionales



Consulta las referencias de IBM, Mervity AI, Tom's Guide, European Commission, DAIR.AI, Anthropic y OpenAI para profundizar en estos temas.

En el aula virtual, debajo de esta clase, encontrarán un video tutorial de cómo armar un cuenta en Make!

¿Preguntas?

Debatamos un poco los temas de hoy.



¡Nos vemos en la próxima clase!

Seguiremos con...

Herramientas No Code para la productividad

¡Gracias!