

**UNIVERSIDAD DE MANIZALES  
FACULTAD DE CIENCIAS E INGENIERÍA  
INGENIERÍA DE SISTEMAS Y TELECOMUNICACIONES**

**PROFUNDIZACIÓN I  
Inteligencia Artificial  
Phd. Carlos Betancourt Correa  
Mg. en Inteligencia Artificial  
Profesor Titular Facultad Ciencias e Ingeniería  
Investigador Asociado Colciencias  
Universidad de Manizales**

## **Laboratorio 8**

### **Exploración y Uso Avanzado de Plataformas IA, Repositorios Profesionales y Herramientas Globales para el desarrollo de IA y de SW**

Este laboratorio constituye un espacio de inmersión profunda en las plataformas y ecosistemas globales de Inteligencia Artificial más influyentes del mundo. El estudiante adquiere dominio práctico y criterio técnico sobre repositorios de modelos, datasets, asistentes inteligentes, plataformas de experimentación, notebooks avanzados y copilotos para desarrollo de software.

La metodología está fundamentada en **ABP (Aprendizaje Basado en Proyectos)**: cada recurso explorado se articula con un reto aplicado a un caso real en educación, generando evidencia concreta, prototipos funcionales y documentación profesional en GitHub.

El laboratorio desarrolla habilidades críticas:

- Búsqueda y selección avanzada de modelos IA.
- Comparación técnica entre plataformas globales.
- Construcción de prototipos inteligentes.
- Integración de herramientas IA en el ciclo de desarrollo de software.
- Documentación profesional y publicación técnica.
- Argumentación, análisis crítico y presentación.
- Trabajo en equipo

#### **Justificación:**

El acceso y uso estratégico de plataformas líderes en IA y repositorios profesionales de modelos y datasets es hoy un requisito para quienes desean diseñar soluciones innovadoras y competitivas. La industria no trabaja con una sola herramienta, sino con un ecosistema articulado de servicios, modelos y flujos de trabajo que combinan prototipos, experimentación, análisis de datos, investigación científica y despliegue de aplicaciones.

Este laboratorio permitirá a los estudiantes investigar, interactuar y dominar herramientas globales de IA, fortaleciendo su capacidad de búsqueda avanzada, análisis comparativo, evaluación crítica, aplicación técnica y generación de soluciones profesionales. Adicionalmente, se incorpora la exploración de IA especializadas en asistencia a la codificación, que están transformando la forma en que se desarrolla software, obligando a los ingenieros a asumir nuevos roles de supervisión, orquestación y evaluación de calidad. De esta forma, el laboratorio prepara al estudiante para enfrentar proyectos reales donde la IA se integra de extremo a extremo: desde la exploración de papers y modelos, hasta la construcción, documentación y publicación de prototipos funcionales.

### Objetivo General:

Desarrollar competencias avanzadas en el uso de plataformas globales de Inteligencia Artificial y repositorios de primer nivel para la búsqueda, evaluación y aplicación de modelos, datasets y herramientas IA, construyendo notebooks inteligentes, prototipos y soluciones prácticas integradas; incluyendo la exploración de asistentes de codificación, la comparación crítica de plataformas, la realización de pruebas aplicadas a un caso educativo real, y la gestión profesional de todos los entregables en un repositorio público de GitHub siguiendo buenas prácticas de documentación, versionamiento y presentación de resultados.

### Metodología ABP

El aprendizaje se organiza según la lógica del ciclo ABP:

1. **Exploración guiada:** el estudiante se apropia del ecosistema global de IA.
2. **Desarrollo del reto:** integra al menos tres plataformas para resolver un problema real.
3. **Producción del prototipo:** implementa un caso funcional documentado.

### Repositorios y plataformas a explorar (no son excluyentes):

Categoría	Plataforma	Descripción Técnica
Modelos y datasets	Hugging Face	Es el ecosistema open-source más grande del mundo para IA. Permite explorar y descargar miles de modelos LLM, visión por computador, audio, embeddings y modelos multimodales. Provee la librería <i>Transformers</i> para usar modelos de última generación, la librería <i>Datasets</i> para manipulación optimizada, y <i>Spaces</i> para desplegar demos interactivas con Gradio o Streamlit. Es la referencia global para reproducibilidad, transferencia y publicación de modelos.
Notebooks inteligentes	Google NotebookLM	Plataforma de creación de notebooks IA donde el modelo trabaja directamente con documentos cargados por el usuario (lecturas, papers, manuales técnicos). Permite generar

Categoría	Plataforma	Descripción Técnica
		explicaciones, resúmenes, evaluaciones de calidad, derivar conceptos, responder preguntas complejas y construir líneas argumentativas basadas exclusivamente en las fuentes aportadas. Funciona como un asistente de investigación contextualizado, ideal para análisis profundo de contenido técnico o académico.
Generación de prompts y prototipos	Google AI Studio	Entorno profesional para diseñar, probar y ajustar prompts para modelos generativos (Gemini, etc.). Permite autenticación, ajuste de parámetros, creación de flujos conversacionales y prototipos rápidos de asistentes. Es ideal para experimentación de <i>prompt engineering</i> , comparación de respuestas, creación de agentes especializados y pruebas rápidas antes de integrarlos a una aplicación real.
Datasets, experimentos y colaboración	Kaggle	Plataforma líder en competiciones de datos, EDA y ciencia de datos. Ofrece miles de datasets públicos, notebooks colaborativos ejecutables en GPU/TPU, foros de discusión y repositorios que permiten compartir código y resultados. Es el estándar global para aprendizaje práctico de machine learning, exploración de datos educativos, experimentación rápida y benchmarking de modelos.
Asistente de investigación	Perplexity AI	Motor de búsqueda potenciado por LLM con capacidad de citación en tiempo real. A diferencia de los chatbots tradicionales, Perplexity rastrea, verifica y enlaza fuentes confiables, permitiendo consultas profundas con evidencia. Es ideal para estado del arte, revisión sistemática, búsqueda de papers recientes, análisis de tendencias y validación académica.
Papers, código y modelos	Papers with Code	Plataforma que enlaza papers científicos peer-reviewed con sus implementaciones open-source. Permite reproducir experimentos, comparar métricas, identificar avances recientes y seguir líderes en cada área. Es fundamental para investigación aplicada, seguimiento de benchmarks, evaluación de nuevas arquitecturas y replicación de experimentos.
Modelos generativos	Replicate	Plataforma que permite ejecutar modelos generativos (imagen, texto, video, audio) mediante API o interfaz gráfica, sin necesidad

Categoría	Plataforma	Descripción Técnica
		de infraestructura local. Facilita desplegar modelos de terceros o propios, construir microservicios IA, automatizar pipelines y probar capacidades multimodales. Su enfoque en rapidez y despliegue público lo hace ideal para prototipos funcionales.
<b>Modelos open-source alternativos</b>	<b>Modelscope (Alibaba)</b>	Repositorio alternativo orientado a modelos de visión, reconocimiento de voz, NLP y modelos multimodales optimizados para hardware diverso. Incluye versiones alternativas de modelos famosos, pipelines preconfigurados y herramientas para experimentación rápida. Es útil para comparación de ecosistemas, rendimiento, compatibilidad y eficiencia.
<b>Bases de conocimiento personalizadas y RAG</b>	<b>LLaMaIndex (GPT Index)</b>	Framework especializado en construcción de asistentes basados en conocimiento propio mediante RAG (Retrieval Augmented Generation). Permite integrar PDFs, bases de datos, texto plano, APIs y páginas web para crear sistemas que consultan, razonan y responden con precisión basada en fuentes internas. Es clave para construir asistentes inteligentes con memoria, evidencias y contexto profundo.
<b>IA para codificación de software (asistentes de desarrollo)</b>	<b>GitHub Copilot</b>	Asistente de codificación basado en LLM entrenado sobre millones de repositorios. Sugiere bloques completos de código, patrones, refactorizaciones, tests unitarios y documentación. Aprenderlo permite entender cómo los LLM se integran al ciclo de desarrollo moderno y cómo optimizar flujos de trabajo de programación.
<b>IA para codificación de software (análisis y refactorización)</b>	<b>Codeium</b>	Alternativa gratuita orientada a autocompletado, explicación de código, generación de funciones y refactorización. Permite evaluar calidad, coherencia, estilo y robustez del código generado por IA. Ideal para comparar con Copilot y analizar comportamientos entre modelos.
<b>IA para prototipado de aplicaciones</b>	<b>Replit AI / Replit Ghostwriter</b>	Entorno de desarrollo en la nube con IA integrada para creación de aplicaciones completas. Es capaz de generar proyectos, archivos, rutas y funciones completas, permitiendo prototipos rápidos en minutos. Ayuda a comprender cómo la IA puede

Categoría	Plataforma	Descripción Técnica
		automatizar partes enteras del ciclo de desarrollo.
IA para desarrollo guiado por instrucciones	Claude Code (Anthropic)	Entorno conversacional donde el modelo actúa como ingeniero colaborador: analiza repositorios completos, propone mejoras, genera funciones completas, crea documentación, revisa seguridad y orienta la arquitectura. Fundamental para entender el futuro de la interacción humano-IA en desarrollo.
IA como “ingeniero autónomo” (tendencia)	Devin AI (conceptual)	Primer prototipo de “ingeniero de software autónomo”: clona repositorios, hace debugging, ejecuta scripts y gestiona pipelines. Aunque aún no está listo para producción educativa, es clave para discutir tendencias, riesgos, límites y roles futuros del ingeniero.

## Metodología:

### Paso 1: Configuración y exploración inicial

- Crear cuentas en todas las plataformas.
- Realizar un recorrido general por cada una, tomando notas de características clave.
- Documentar fortalezas, debilidades y posibles aplicaciones.

### Paso 2: Actividades detalladas por plataforma

#### 2.1 Modelos y Datasets — Hugging Face

##### Descripción Técnica

Es el ecosistema open-source más grande del mundo para IA. Permite explorar y descargar miles de modelos LLM, visión por computador, audio, embeddings y modelos multimodales. Provee la librería Transformers para usar modelos de última generación, la librería Datasets para manipulación optimizada, y Spaces para desplegar demos interactivas con Gradio o Streamlit. Es la referencia global para reproducibilidad, transferencia y publicación de modelos.

##### Desarrollo

- Explorar mínimo 3 modelos (NLP, visión, audio).
- Ejecutar un modelo descargado en un notebook.
- Construir un Hugging Face Space con demo funcional (chatbot, clasificador, generador).
- Analizar arquitectura, rendimiento y limitaciones del modelo.

##### Evidencia

- Notebook ejecutado.

- Link del Space público.
- Capturas de inferencias.
- Informe técnico en Markdown.

## **2.2 Notebooks Inteligentes — Google NotebookLM**

### **Descripción Técnica**

Plataforma donde el modelo trabaja directamente sobre documentos cargados por el usuario (papers, manuales, lecturas). Genera explicaciones, resúmenes, análisis y respuestas complejas basadas estrictamente en las fuentes cargadas. Funciona como un asistente de investigación contextualizado.

### **Desarrollo**

- Cargar un documento técnico.
- Formular 5 preguntas argumentativas sobre el texto.
- Generar resumen detallado del contenido.
- Evaluar consistencia, sesgos y profundidad de las respuestas.

### **Evidencia**

- Capturas del notebook.
- Exportación del resumen.
- PDF con análisis crítico.

## **2.3. Ingeniería de Prompts y Prototipos — Google AI Studio**

### **Descripción Técnica**

Entorno profesional para diseñar, probar y ajustar prompts para modelos generativos como Gemini. Permite autenticación, parámetros avanzados, construcción de agentes y flujos conversacionales. Es ideal para experimentación de ingeniería de prompts y prototipado rápido.

### **Desarrollo**

- Crear dos prompts profesionales:
  - Uno técnico.
  - Uno creativo.
- Generar iteraciones documentadas.
- Evaluar impacto del cambio de temperatura, top-k, top-p y contexto.

### **Evidencia**

- Capturas del proceso.
- Archivo de comparativas.
- Reflexión técnica documentada.

## **2.4 Datasets, Experimentos y Colaboración — Kaggle**

### **Descripción Técnica**

Es la plataforma líder en ciencia de datos aplicada. Incluye miles de datasets, notebooks colaborativos y competencias. Permite EDA, visualizaciones, modelos, comparaciones y publicación de kernels.

#### **Desarrollo**

- Seleccionar un dataset relevante.
- Realizar un EDA completo (limpieza, correlaciones, gráficos).
- Publicar un notebook colaborativo.
- Analizar implicaciones del dataset en un caso educativo.

#### **Evidencia**

- Link público del notebook.
- PDF con EDA.
- README en GitHub.

## **2,5 Asistente de Investigación — Perplexity AI**

#### **Descripción Técnica**

Motor de búsqueda potenciado por IA con citación verificable en tiempo real. Realiza consultas profundas con fuentes enlazadas y comprobables, ideal para estado del arte o revisión científica.

#### **Desarrollo**

- Realizar una consulta técnica compleja.
- Validar mínimo 5 fuentes proporcionadas.
- Comparar resultados frente a Google Scholar.

#### **Evidencia**

- Capturas de consulta.
- Reporte de validación de fuentes.
- Conclusiones escritas.

## **2.6 Papers, Código y Modelos — Papers With Code**

#### **Descripción Técnica**

Plataforma que enlaza papers peer-reviewed con implementaciones open-source oficiales. Es clave para reproducibilidad, seguimiento de benchmarks y replicación experimental.

#### **Desarrollo**

- Seleccionar un paper con código.
- Ejecutar parte del experimento.
- Analizar diferencias entre resultados del paper y los propios.
- Comparar métricas.

#### **Evidencia**

- Notebook con ejecución.
- Informe técnico de replicación.
- Gráficas de resultados.

## **2.7 Modelos Generativos — Replicate**

### **Descripción Técnica**

Permite ejecutar modelos generativos mediante API o interfaz gráfica sin infraestructura local. Ideal para prototipado rápido, despliegue público y pruebas multimodales.

### **Desarrollo**

- Probar dos modelos generativos (imagen, voz, texto o video).
- Construir un microservicio o script que consuma la API.
- Evaluar latencia y calidad.

### **Evidencia**

- Imágenes/textos generados.
- Script documentado.
- Carpeta /replicate/ en el repositorio.

## **2.8 Modelos Open Source Alternativos — Modelscope (Alibaba)**

### **Descripción Técnica**

Repositorio con modelos optimizados en visión, reconocimiento de voz y NLP. Ofrece pipelines preconfigurados y alternativas de modelos globales.

### **Desarrollo**

- Ejecutar mínimo un modelo.
- Comparar eficiencia vs. Hugging Face.
- Construir gráfica comparativa.

### **Evidencia**

- Notebook ejecutado.
- Gráfica comparativa.
- Análisis en Markdown.

## **2.9 Bases de Conocimiento y RAG — LLaMaIndex**

### **Descripción Técnica**

Framework especializado en asistencias basadas en documentos propios mediante RAG. Permite integrar PDFs, bases de datos, APIs y webs para respuestas basadas en evidencia.

### **Desarrollo**

- Crear una base de conocimiento con mínimo 5 documentos.
- Configurar pipeline RAG (ingestión + indexación + respuesta).
- Validar exactitud y grounding de las respuestas.

### **Evidencia**

- Notebook de implementación.
- Capturas de consultas.
- README explicando arquitectura RAG.



## **2.10 IA para Codificación de Software — GitHub Copilot**

### **Descripción Técnica**

Asistente basado en LLM integrado en IDEs. Sugiere código, test unitarios, documentación y refactorización. Representa la integración real de la IA en el ciclo moderno de programación.

### **Desarrollo**

- Crear un módulo con ayuda del asistente.
- Generar funciones, refactorización y documentación automática.
- Comparar calidad entre versión manual vs IA.

### **Evidencia**

- Capturas del IDE.
- Código comentado.
- Archivo de comparación.

## **2.11 IA para Codificación (Análisis y Refactorización) — Codeium**

### **Descripción Técnica**

Asistente gratuito para autocompletado, explicación de código y refactorización. Permite evaluar calidad y estilo del código generado por IA.

### **Desarrollo**

- Generar una función con Codeium.
- Solicitar refactorización y explicación.
- Comparar con GitHub Copilot.

### **Evidencia**

- Archivos de código.
- Informe comparativo.
- Capturas del proceso.

## **2.12 IA para Prototipado — Replit AI / Replit Ghostwriter**

### **Descripción Técnica**

IDE en la nube con IA capaz de generar proyectos completos, archivos, rutas y funciones. Ideal para prototipos rápidos y experimentación ágil.

### **Desarrollo**

- Crear un microproyecto completo (API, web app, script).
- Generar archivos y rutas con IA.
- Analizar estructura generada.

### **Evidencia**

- Proyecto ejecutable.

- Carpeta /replit/.
- Informe técnico de prototipado.

## 2.13 IA para Desarrollo Guiado — Claude Code (Anthropic)

### Descripción Técnica

Entorno conversacional donde el modelo actúa como ingeniero colaborador: analiza repositorios, propone mejoras, revisa seguridad, documenta y genera funciones completas.

### Desarrollo

- Cargar un repositorio completo.
- Solicitar mejoras estructurales.
- Generar documentación automatizada.
- Solicitar un módulo adicional creado por IA.

### Evidencia

- Capturas del chat.
- Repositorio modificado.
- Informe de mejoras.

## 14. Tendencias Futuras — Devin AI (Modelo Conceptual)

### Descripción Técnica

Primer prototipo de “ingeniero autónomo”: ejecuta pipelines, clona repos, hace debugging y gestiona tareas. No es estable para uso educativo, pero es esencial para discusión ética y futura.

### Desarrollo

- Analizar el concepto de ingeniero autónomo.
- Documentar riesgos, potencial y límites.
- Comparar con copilotos actuales.

### Evidencia

- Ensayo técnico (1–2 páginas).
- Cuadro comparativo.
- Presentación breve.

### Paso 3: Caso integrador

- Elegir un reto (propuesto por el grupo y aplicado a la educación).
- Usar al menos tres plataformas para resolverlo.
- Documentar paso a paso en un repositorio GitHub público, con README explicativo.
- Presentar resultados en informe PDF y entregar enlace del repositorio.
- Incluir análisis crítico, gráfica comparativa y recomendaciones.

### Paso 4: Presentación y reflexión final

- Elaborar una presentación (en Video).
- Compartir hallazgos y aprendizajes con el grupo.

- Responder preguntas del docente y compañeros.

## **REFLEXIÓN METACOGNITIVA**

### **1. Reflexión sobre el razonamiento analítico y técnico**

¿Qué tareas, consultas o decisiones dentro del laboratorio exigieron el uso más intenso de razonamiento analítico, verificación técnica o pensamiento lógico?

---

---

---

---

---

¿En qué momento tuviste que ir más allá de seguir instrucciones y empezar a diseñar, deducir, comparar o validar por tu cuenta?

---

---

---

---

---

### **2. Reflexión sobre corrección de supuestos y evolución conceptual**

¿Qué ideas iniciales tuviste sobre las plataformas o modelos que luego resultaron incompletas o erróneas?

---

---

---

---

---

¿Cómo cambió tu comprensión de los modelos, pipelines o herramientas después de enfrentarte a la práctica real?

---

---

---

---

---

### **3. Reflexión sobre estrategias de resolución de problemas**

¿Qué estrategias cognitivas utilizaste para resolver problemas o interpretar resultados (por ejemplo, descomposición, contrastación de fuentes, experimentación iterativa, comparación de outputs, lectura de documentación, análisis de errores, etc.)?

---

---

---

---

---

¿Qué estrategia fue la más útil y por qué?

---

---

---

---

---

#### **4. Reflexión sobre integración de plataformas y transferencia de conocimiento**

¿Qué aprendiste sobre cómo se integran entre sí las plataformas, modelos, APIs, copilotos o frameworks utilizados?

---

---

---

---

---

¿Qué decisiones debiste tomar para que el caso integrador funcionara técnicamente?

---

---

---

---

---

¿Qué conocimientos transferiste de una herramienta a otra?

---

---

---

---

---

#### **5. Reflexión sobre tu crecimiento profesional y rol como ingeniero**

¿Cómo fortaleció este laboratorio tu identidad como ingeniero o profesional en IA?

---

---

---

---

---

---

¿Qué descubriste sobre tu capacidad para aprender herramientas nuevas, integrar sistemas, evaluar tecnologías y documentar procesos?

---

---

---

---

---

---

¿En qué áreas te sientes más fuerte ahora y qué deberías seguir reforzando?

**6. Reflexión sobre el cambio de paradigma y la profesión en la era de la IA**

¿Qué cambios percibes en el rol tradicional del desarrollador de software a partir del uso de modelos generativos, copilotos de código y herramientas de automatización vistas en este laboratorio?

---

---

---

---

---

---

¿Qué tareas crees que tenderán a automatizarse y qué nuevas responsabilidades (técnicas, éticas, de supervisión, de diseño de sistemas) aparecen para los ingenieros de software e ingenieros de IA?

---

---

---

---

---

---

¿Cómo te ves a ti mismo dentro de este nuevo contexto? ¿Qué deberías aprender, fortalecer o cambiar en tu forma de trabajar para seguir siendo relevante y aportar valor en la era de la IA?

---

---

---

---

---

### Entrega obligatoria en GitHub:

Todos los entregables deben ser organizados y publicados en un repositorio GitHub con:

- README estructurado.
- Carpetas organizadas por plataforma y por el caso integrador.
- Capturas, notebooks y códigos.
- Instrucciones para la ejecución.

### Rúbrica de Evaluación:

#### Rúbrica de Evaluación Detallada:

#	Criterio / Entregable	Descripción resumida del criterio	Ponderación (%)
1	Tabla comparativa de plataformas IA	Análisis crítico, lenguaje técnico, pros/contras, escenarios de uso y recomendaciones.	15 %
2	Hugging Face (entregable)	Notebook ejecutado, Space publicado, reflexión técnica (modelo, arquitectura, rendimiento, limitaciones).	2 %
3	Google NotebookLM (entregable)	Carga de documento técnico, preguntas avanzadas, resumen generado, análisis crítico de respuestas.	2 %
4	Google AI Studio (entregable)	Diseño de prompts técnicos/creativos, iteración, análisis comparativo de salidas y parámetros.	2 %
5	Kaggle (entregable)	EDA completo (limpieza, visualizaciones, correlaciones), notebook público y documentación en GitHub.	2 %
6	Perplexity AI (entregable)	Consulta técnica compleja, validación de fuentes, comparación frente a buscadores académicos.	2 %
7	Papers with Code (entregable)	Selección de paper con código, réplica parcial de experimento, análisis de métricas y diferencias.	2 %
8	Replicate (entregable)	Prueba de dos modelos generativos, consumo vía interfaz/API, análisis de latencia y calidad de resultados.	2 %
9	Modelscope (Alibaba) (entregable)	Ejecución de un modelo alternativo, comparación de eficiencia vs Hugging Face, gráfica comparativa.	2 %
10	LLaMaIndex (entregable)	Construcción de base de conocimiento ( $\geq 5$ documentos), pipeline RAG y validación de grounding.	2 %
11	GitHub Copilot (entregable)	Generación de código, refactorización, documentación automática y comparación con versión manual.	2 %
12	Codeium (entregable)	Autocompletado, explicación de código, refactorización y contraste técnico con Copilot.	2 %
13	Replit AI / Ghostwriter (entregable)	Microproyecto (API/app/script) generado con IA, estructura de archivos y análisis del prototipado asistido.	2 %
14	Claude Code (entregable)	Análisis de repositorio, mejoras propuestas por el modelo, documentación generada y módulo adicional creado con IA.	2 %
15	Devin AI – Tendencias futuras (entregable conceptual)	Ensayo crítico sobre ingeniero autónomo, riesgos, límites, oportunidades y comparación con copilotos actuales.	2 %

#	Criterio / Entregable	Descripción resumida del criterio	Ponderación (%)
16	Caso integrador completo en GitHub	Proyecto ABP que integra $\geq 3$ plataformas, prototipo funcional, carpetas organizadas, README profesional y trazabilidad de decisiones.	25 %
17	Informe final en PDF	Redacción profesional, estructura clara, anexos técnicos, gráficos/tablas, análisis crítico y referencias adecuadas.	12 %
18	Presentación final (video)	Claridad expositiva, dominio conceptual, explicación técnica del caso integrador y capacidad para responder preguntas.	5 %
19	Reflexión metacognitiva (5 ejes)	Profundidad en el análisis del propio proceso de aprendizaje, estrategias de razonamiento, revisión de supuestos y transferencia entre herramientas.	5 %
20	Uso de IA y Prompts Empleados	Transparencia en el uso de herramientas IA, documentación de prompts, explicación de cómo se adaptaron/mejoraron las respuestas y en qué partes del trabajo se utilizaron.	10 %

**Tabla de Calificación (escala de 1 a 5):**

Calificación (0 a 5)	Desempeño	Descripción
4.6 - 5.0	Excelente	Dominio absoluto de las plataformas, evidencia completa, análisis profundo, repositorio impecable y propuesta creativa adicional.
4.0 - 4.5	Muy bueno	Cumplimiento completo, buen repositorio en GitHub, con pequeños detalles por mejorar.
3.5 - 3.9	Bueno	Entregables correctos, repositorio funcional pero con áreas de mejora en organización o documentación.
3.0 - 3.4	Aceptable	Entregas incompletas o poco organizadas, repositorio con fallas importantes en estructura o explicación.
Menor a 3.0	Deficiente	Incumplimiento de actividades clave, sin repositorio o documentación adecuada, evidencias ausentes o desordenadas.

### ***Uso de Inteligencia Artificial y Prompts***

*Cada grupo deberá documentar explícitamente el uso de herramientas de Inteligencia Artificial durante el desarrollo del laboratorio. Para ello, al final del informe se debe incluir una sección titulada “Caso de Uso de IA Aplicado”, que contenga como mínimo:*

- ***Relación de las herramientas de IA utilizadas*** (por ejemplo: ChatGPT, GitHub Copilot, Codeium, Claude, Gemini, Perplexity, etc.).
- ***Copia o resumen de los prompts principales*** empleados para generar código, notebooks, configuraciones de modelos, pipelines RAG, textos explicativos o ideas de solución.
- ***Breve explicación de cómo se adaptaron, corrigieron o complementaron las respuestas de la IA en el contexto del laboratorio*** (por ejemplo: ajustes hechos al código sugerido, modificaciones a consultas, refinamiento de prompts, incorporación de validaciones adicionales, cambios en hiperparámetros, etc.).

- **Identificación de los apartados concretos donde se aprovechó la IA**, indicando el tipo de apoyo recibido (ej.: diseño de arquitectura del prototipo, generación inicial de código en Replit o en el IDE, apoyo en la documentación del README, sugerencias de pruebas, análisis de resultados, redacción de secciones del informe, etc.).

Esta sección **no reducirá la nota por el uso de IA**; por el contrario, será valorada positivamente la **transparencia**, el **criterio técnico** y la **capacidad de mejorar críticamente** las respuestas generadas por los modelos, demostrando que la IA fue utilizada como herramienta de apoyo y no como sustituto del razonamiento propio.