

Apuntes Curso de Desarrollo con IA

DÍA 1

Índice

¿Por qué la IA es tu mejor aliada para Programar?	3
El nuevo panorama: La IA ya está aquí (y eso es bueno)	3
El Programador sigue siendo el piloto: La analogía del Copiloto	5
La IA como potenciador: 3 ejemplos de uso para tu día a día	6
Taller #1: Aprendizaje Acelerado con IA	6
Caso Práctico: Decisión de Stack Tecnológico (Python vs. JavaScript)	7
Guía Paso a Paso para Dominar NotebookLM	7
Paso 1: Creando tu Cuaderno de Conocimiento	7
Paso 2: Alimentando tu IA con Fuentes Confiables	8
Añadiendo Cursos en Video (YouTube)	9
Incorporando Documentación Oficial	10
Paso 3: Dialogando con tus datos a través del Chat	13
Paso 4: Otras herramientas de NotebookLM	14
Otras Ideas para Aplicar NotebookLM	15
Taller #2: Domina la ingeniería de prompts para devs	16
Caso Práctico: Análisis de un Repositorio en GitHub	16
Taller #3: Convierte un boceto en una web lista para producción	20
Caso Práctico: Creación de un portfolio de desarrollador	20
Características Clave de Firebase Studio	22
Conclusiones	23
Súmate a nuestro directo del día 2	24

¿Por qué la IA es tu mejor aliada para Programar?

La idea de la Inteligencia Artificial puede ser intimidante, especialmente cuando estás dando tus primeros pasos en el mundo de la programación. Es fácil percibir esta revolución con algo de miedo, pensando que viene a reemplazar el talento humano. Y si sientes eso, es normal.

Sin embargo, el propósito de esta guía es demostrarte exactamente lo contrario: la IA no es tu reemplazo, sino la herramienta más poderosa que ha llegado para potenciar tu ingenio y acelerar tu camino como desarrollador.

El nuevo panorama: La IA ya está aquí (y eso es bueno)

La Inteligencia Artificial no es una promesa futurista, sino una realidad integrada en el presente del desarrollo de software. Para entender el cambio, es crucial observar el contexto actual del sector y aceptar que esta es una herramienta que llegó para quedarse.

Las Cifras de la Revolución

Los siguientes datos, basados en estudios masivos a desarrolladores, pintan un cuadro claro de la situación actual.

45% del código ya es escrito por IA:

- **Por qué es importante:** Esto no significa que la mitad del trabajo haya desaparecido. Significa que la colaboración con la IA ya es el estándar de la industria. Aprender a usarla no es una opción para el futuro, es una habilidad esencial para tu primer día de trabajo.

55% del proceso de desarrollo se ve acelerado:

- **Por qué es importante:** La IA no compite contigo, compite contra el tedio. Su objetivo es automatizar las partes repetitivas y lentas para que tú puedas dedicar tu energía mental a la arquitectura, la lógica de negocio y la resolución creativa de problemas, que es donde realmente aportas valor.

90% de los desarrolladores sienten mayor productividad y satisfacción laboral:

- **Por qué es importante:** Al delegar tareas repetitivas o complejas a la IA, los desarrolladores pueden enfocarse en resolver problemas de mayor nivel y en los aspectos más creativos del trabajo. Esto no solo aumenta la productividad, sino también la satisfacción profesional.

Una Evolución, no una Extinción

El miedo al cambio es natural, pero nuestro sector siempre ha estado en constante evolución. La IA es simplemente el siguiente paso en esa progresión.

Así como los editores de código avanzados, los frameworks y los lenguajes de alto nivel fueron una evolución que nos hizo más eficientes, la IA es el siguiente paso lógico en nuestra caja de herramientas.

La IA es la nueva realidad, pero esto no disminuye tu valor; al contrario, lo redefine. La herramienta más potente del mundo es inútil sin un experto que la dirija. Tú sigues siendo el piloto.

El Programador sigue siendo el piloto:

La analogía del Copiloto

La metáfora más acertada para describir la relación entre un desarrollador y la IA es la de un piloto y su copiloto. La IA es un asistente increíblemente capaz, pero necesita ser dirigida por un experto que tenga los conocimientos fundamentales y la visión del proyecto.

¿Quién dirige la nave?

Una IA, por sí sola, no sabe qué construir, cuál es el objetivo del negocio o cómo debe ser la arquitectura del sistema. Simplemente genera respuestas basadas en patrones. El verdadero valor del desarrollador reside en sus fundamentos y su capacidad para guiar a la herramienta. Nosotros somos los pilotos.

"Yo no le puedo decir a la IA hacia dónde tiene que ir si yo no sé hacia dónde tiene que ir."

El Peligro de "Solo Aceptar"

El peligro real, y el error que delata a un aficionado, es lo que llamamos **"vibe coding"**. Consiste en lanzar un prompt, copiar el resultado sin entenderlo y cruzar los dedos. Un profesional, en cambio, utiliza la IA guiándola con su conocimiento, audita cada línea de código sugerida y se asegura de que la solución no sólo **"funcione"**, sino que sea robusta, eficiente y cumpla con los objetivos del proyecto.

La IA es tu copiloto, no un piloto automático al que le cedes el control total.

Entender eso es el primer paso. Ahora, veamos qué instrumentos te proporciona la IA para componer software de forma más rápida, inteligente y creativa.

La IA como potenciador: 3 ejemplos de uso para tu día a día

Aquí es donde la teoría se convierte en práctica. La IA no es solo un asistente de código; es un set de superpoderes que transformará tu forma de aprender, crear y resolver problemas.

Taller #1: Aprendizaje Acelerado con IA

Como desarrolladores, nos enfrentamos a herramientas increíblemente poderosas como **ChatGPT** o **Gemini**, pero estas tienen un problema fundamental: **por defecto, no tienes control sobre sus fuentes de información.**

Estos modelos han sido entrenados con datos masivos de internet, y al generar una respuesta, pueden mezclar información correcta con datos obsoletos, incorrectos o simplemente inventados, un fenómeno conocido como **"alucinación"**.

Aquí es donde entra [NotebookLM](#), una herramienta gratuita de Google diseñada para darte el control total.

Con NotebookLM, tú decides cuáles son las fuentes de información, acotando el conocimiento para obtener respuestas precisas y confiables.

El objetivo de este taller es que pierdas ese miedo y aprendas, paso a paso, cómo usar NotebookLM para analizar información de distintas fuentes y tomar una decisión tecnológica informada, actuando como un verdadero experto.

A través de un escenario práctico, descubrirás cómo transformar horas de investigación en minutos de análisis inteligente, poniendo la IA a trabajar para ti.

Caso Práctico: Decisión de Stack Tecnológico (Python vs. JavaScript)

Todo desarrollador, en algún punto de su carrera, se enfrenta a una decisión crucial: ¿qué lenguaje de programación debería aprender o utilizar para un nuevo proyecto? Es una pregunta que requiere analizar pros, contras, curvas de aprendizaje y ecosistemas tecnológicos.

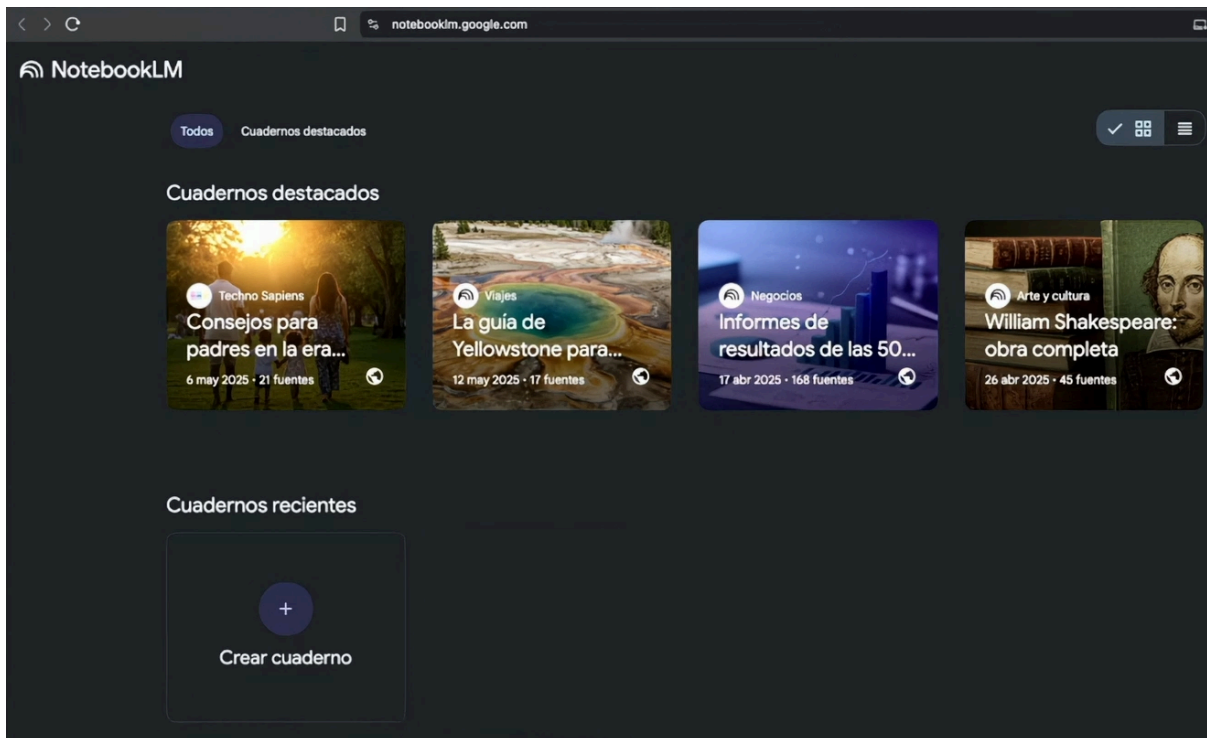
Para este taller, utilizaremos el clásico desafío de comparar Python y JavaScript. Nuestro objetivo no es declarar un ganador, sino utilizar este escenario realista para aprender un proceso de análisis que podrás aplicar a cualquier duda tecnológica que tengas, desde comparar frameworks hasta elegir una base de datos.

Guía Paso a Paso para Dominar NotebookLM

Paso 1: Creando tu Cuaderno de Conocimiento

El primer paso es el más sencillo: **crear un nuevo cuaderno en NotebookLM**. Al hacerlo, te encontrarás con un espacio completamente vacío. Este cuaderno es tu "lienzo en blanco", un entorno de IA donde tú, y solo tú, tienes el control total.

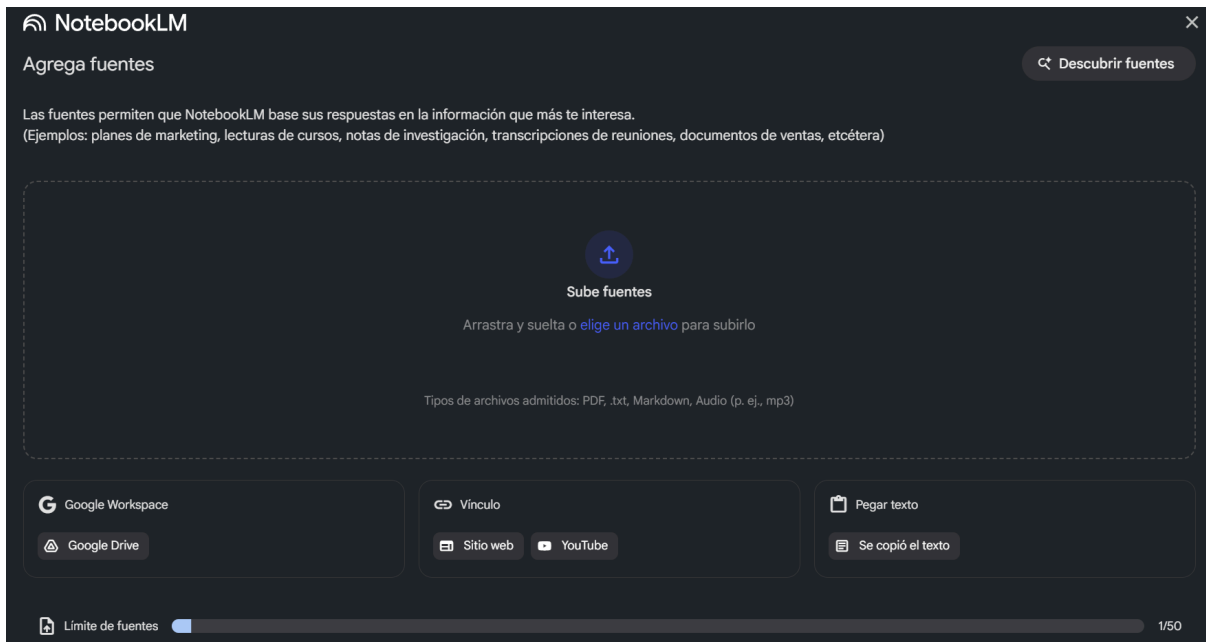
Esto es fundamental. El punto de partida es un conocimiento cero; la IA solo sabrá lo que tú decidas enseñarle.



Paso 2: Alimentando tu IA con Fuentes Confiables

Ahora es el momento de añadir las "**fuentes**" que conformarán la base de conocimiento de tu IA. NotebookLM es increíblemente versátil y te permite añadir información de diversos formatos.

- Videos de YouTube: Pega un enlace y la IA procesará la transcripción completa.
- Sitios Web: Proporciona una URL para que analice el contenido de una página.
- Documentos de Google Drive: Conecta tu cuenta y añade textos, presentaciones, etc.
- Archivos PDF y de Texto: Sube documentos directamente desde tu ordenador.



Para nuestro taller, vamos a añadir dos tipos de fuentes clave: cursos en video y documentación oficial.

Añadiendo Cursos en Video (YouTube)

Imagina que quieres basar tu decisión en el contenido de cursos completos. En lugar de ver horas de video, puedes simplemente añadir los enlaces. Para nuestro ejemplo, añadiremos:

- [Curso de Python desde cero](#) (10 horas de duración).
- [Curso de JavaScript](#) (más de 6 horas de duración).

NotebookLM procesará estos videos en segundos, extrayendo toda la información clave y dejándola lista para ser consultada. ¿Te das cuenta de las horas de trabajo que acabas de ahorrarte?



Incorporando Documentación Oficial

Para complementar los videos, es fundamental contar con la documentación oficial. Usando la función “**Descubrir**” de NotebookLM, podemos encontrar y añadir las fuentes más confiables. Para nuestro caso, seleccionaremos:

- La documentación de MDN (Mozilla Developer Network) para JavaScript.
- El sitio web oficial de Python.org para Python.

Descubre fuentes

¿Cuáles son tus intereses?

Busca la documentación oficial de JS y Python

Busca fuentes en:

☒ La Web

☐ Google Drive

Tengo curiosidad

Enviar



Una vez que tus fuentes están cargadas, comienza la verdadera magia: el diálogo inteligente con tus datos.

Paso 3: Dialogando con tus datos a través del Chat

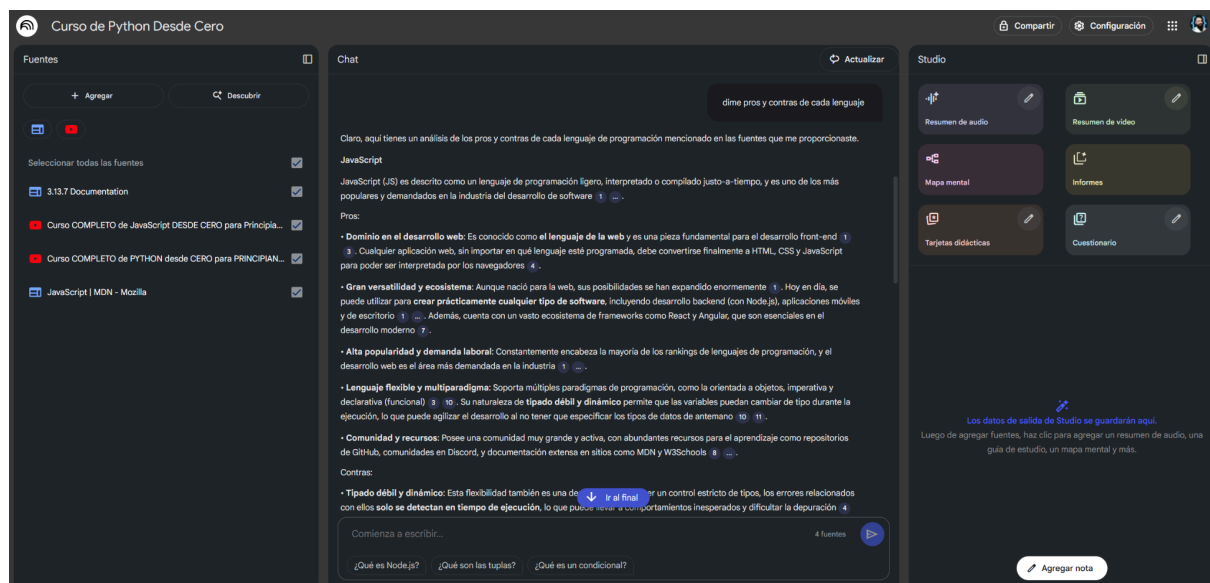
Con tus fuentes listas, puedes empezar a hacer preguntas en el chat. La IA responderá basándose exclusivamente en la información que le proporcionaste.

Vamos a lanzarle una pregunta directa:

"dime pros y contras de cada lenguaje"

La IA analizará las casi 20 horas de video y los cientos de páginas de documentación para darte una respuesta sintetizada.

La característica más importante de estas respuestas son las **citas numeradas** que aparecen junto a cada afirmación. Estos números **[1]**, **[2]**, etc., son enlaces que te llevan al punto exacto de la fuente (ya sea el minuto de un video o el párrafo de una web) de donde se extrajo la información. Esto elimina por completo las alucinaciones y te da total confianza en la respuesta.



Paso 4: Otras herramientas de NotebookLM

NotebookLM va mucho más allá de un simple chat. Ofrece un conjunto de herramientas diseñadas para transformar el consumo pasivo de datos en una experiencia de aprendizaje activa. Estas funciones se adaptan a diferentes estilos y situaciones, permitiéndote sintetizar la información de formas innovadoras.

Herramienta	Descripción	Beneficio para el Aprendizaje
Mapa Mental	Organiza visualmente los conceptos clave, características y recursos de cada lenguaje.	Te permite comparar de un solo vistazo las estructuras de ambos lenguajes, facilitando la identificación de similitudes y diferencias.
Resumen de Video/Audio	Genera un podcast o un video corto (ej. de 5 minutos) que resume toda la información de las fuentes.	Perfecto para repasar conceptos clave mientras viajas o haces otras actividades. Condensa horas de material en un formato fácil de consumir.
Tarjetas Didácticas (Flashcards)	Crea automáticamente una baraja de tarjetas con preguntas y respuestas sobre los conceptos fundamentales (ej. "68 tarjetas para conceptos clave").	Ideal para reforzar el aprendizaje activo y autoevaluar tu conocimiento sobre los temas más importantes de cada tecnología.
Cuestionario	Genera una prueba formal (ej. de 15 preguntas) para evaluar tu comprensión sobre el material cargado.	Te ayuda a medir de manera objetiva cuánto has aprendido y a identificar las áreas que necesitas reforzar antes de tomar una decisión.

Otras Ideas para Aplicar NotebookLM

Comparar lenguajes es solo el comienzo. El verdadero poder de NotebookLM radica en adaptarlo a tus necesidades específicas como desarrollador.

Puedes utilizar esta técnica para:

1. **Depurar código:** Alimenta a NotebookLM con la documentación de tu propia aplicación y fragmentos de tu código. Luego, pídele ayuda para encontrar errores con un contexto que ninguna otra IA generalista tendrá.
2. **Aprender sobre APIs o comparar frameworks:** ¿Necesitas entender rápidamente una nueva API o decidir entre React y Vue? Carga sus documentaciones oficiales y deja que NotebookLM te genere resúmenes comparativos y mapas mentales.
3. **Refactorizar o migrar código:** Proporcióname la base de código existente como fuente para que aprenda sobre ella y te asista en tareas complejas de migración o mejora de la arquitectura.
4. **Analizar incidencias:** Sube logs de errores o reportes de incidencias para que la IA te ayude a identificar patrones y a entender la causa raíz de un problema mucho más rápido.

Taller #2: Domina la ingeniería de prompts para devs

La ingeniería de prompts ha madurado rápidamente.

La idea inicial de buscar el "**prompt perfecto**" y único ha dado paso a un enfoque más dinámico y estratégico: un proceso iterativo que dialoga con la IA.

Los "**modelos razonadores**" más recientes pueden procesar instrucciones complejas, auditar su propia "**cadena de pensamiento**" y colaborar con el usuario para refinar las consultas.

Esta capacidad permite realizar análisis profundos de herramientas, librerías y repositorios de código complejos, extrayendo conocimiento estratégico que antes requería semanas de investigación manual.

Tareas como la "**Investigación Profunda**", que se ejecutan durante varios minutos, permiten al modelo realizar un análisis exhaustivo que es cualitativamente superior a las respuestas instantáneas.

Caso Práctico: Análisis de un Repositorio en GitHub

Para ilustrar este flujo de trabajo, utilizaremos como ejemplo el análisis de la librería de visión artificial [Ultralytics](https://github.com/ultralytics/ultralytics). El objetivo es pasar de una comprensión superficial a un dominio profundo del proyecto en el menor tiempo posible, utilizando un enfoque de prompting escalonado.

Prompt #1 (Gemini 2.5 Flash):

*"Haz un análisis de esta librería de código
<https://github.com/ultralytics/ultralytics>"*

Resultado obtenido: Una respuesta básica y general sobre la funcionalidad del repositorio. Describe qué es Yolo, pero carece de la profundidad necesaria para un desarrollador.

Prompt #2 (ChatGPT 5 Thinking):

*"Haz un análisis de esta librería de código
<https://github.com/ultralytics/ultralytics>"*

Resultado obtenido: Una respuesta mucho más estructurada con pros y contras. Crucialmente, permite auditar la "cadena de pensamiento" del modelo, revelando las fuentes consultadas.

Prompt #3 (Cómo refinar prompts):

"Quiero un prompt refinado que me ayude a entender cómo está estructurado el código de un repo de Github"

RESPUESTA 1:

Actúa como un asistente experto en análisis de repositorios de GitHub.

Tengo un repositorio de código y quiero comprender claramente su estructura y organización interna.

Tareas que debes realizar:

- 1. Genera un mapa de carpetas y archivos principales, indicando qué rol cumple cada uno (ejemplo: src/ → código fuente, tests/ → pruebas, docs/ → documentación, etc.).*
- 2. Explica los puntos de entrada del código (ejemplo: main.py, __init__.py, app.js) y cómo se conectan con el resto del proyecto.*
- 3. Resume los módulos o paquetes más importantes, indicando qué funcionalidad resuelven.*
- 4. Señala los archivos de configuración clave (ejemplo: requirements.txt, pyproject.toml, package.json, Dockerfile, Makefile) y para qué sirven.*
- 5. Identifica la lógica central del proyecto (dónde ocurre lo más importante) y su relación con utilidades, helpers o librerías.*
- 6. Si existe, describe cómo están organizadas las pruebas y ejemplos de uso.*

7. *Propón una explicación sencilla en lenguaje no técnico (para managers o personas sin background de programación).*

Entrega el resultado en formato claro y estructurado con títulos, listas y, si es posible, un pequeño esquema en árbol para la estructura de carpetas.

RESPUESTA 2:

Rol: Actúa como un ingeniero de software senior especializado en análisis de repositorios de GitHub.

Objetivo: Entregar un análisis estructurado del repo {URL_repo}, en formato claro y accionable.

Instrucciones de salida:

1. *Mapa del repo*
 - *Tabla con columnas: [Ruta/Archivo] [Responsabilidad principal] [Lenguaje/stack] [Interno/Público]*
 - *Identifica qué carpetas son núcleo, cuáles son soporte (tests, utils, docs) y cuáles son configuraciones.*
2. *Flujo de ejecución principal*
 - *Explica cómo arranca la aplicación (entrypoints: main, index, app.py, etc.).*
 - *Resume en 150 palabras o menos el camino típico desde la entrada (ej: request HTTP) hasta la salida (ej: respuesta JSON).*
3. *Dependencias y APIs*
 - *Lista principales dependencias externas y su rol (framework, base de datos, auth, etc.).*
 - *Señala módulos expuestos como API pública (endpoints, librerías exportadas).*
4. *Puntos críticos / Hotspots*
 - *Archivos con más complejidad, acoplamiento o centralidad.*
 - *Señala 2–3 riesgos técnicos y sugiere un refactor incremental.*
5. *Siguiente paso recomendado*

- *Indica cómo un developer nuevo puede hacer su primera contribución útil (ej: mejorar tests, actualizar docs, añadir linters).*

Requisitos:

- *Usa viñetas, tablas o diagramas de texto si ayudan.*
- *Si hay información faltante en el repo, márcala como "No disponible en fuentes".*
- *No expliques tu proceso interno, entrega solo el resultado final en formato legible.*

Resultado obtenido: análisis técnico que identifica los puntos de entrada, dependencias y "hotspots" críticos del código.

Prompt #4 (Gemini + Deep Research):

"Analiza el repositorio de código <https://github.com/ultralytics/ultralytics> y cuéntame todo lo que puedas sobre el mismo"

Resultado obtenido: Se invoca una tarea que se ejecuta durante más de 15 minutos. El resultado es un documento completo de **19 páginas**, un informe profesional con resumen ejecutivo, análisis de arquitectura y todas las fuentes citadas.

Sinergia con Bases de Conocimiento

Este flujo de trabajo crea una poderosa sinergia con la fase anterior. El documento de 19 páginas generado por la "Investigación Profunda", junto con las fuentes citadas, puede ser utilizado como material de entrada para una nueva base de conocimiento en NotebookLM.

Esto crea un "gurú" personal e hiperespecializado sobre la librería, listo para consultas complejas sobre su implementación o arquitectura.

Taller #3: Convierte un boceto en una web lista para producción

La IA es la herramienta perfecta para derribar "el síndrome de la hoja en blanco". Con herramientas como [Firebase Studio](#), puedes transformar una simple idea visual en un prototipo funcional en cuestión de minutos.

El flujo es asombrosamente sencillo:

1. Partes de un simple boceto o dibujo de una página web, hecho en un papel o una herramienta digital.
2. Le pides a la IA que cree una aplicación web basada en esa imagen y le proporcionas el texto descriptivo (*copy*).
3. La IA genera todo el código (HTML, CSS, JavaScript), la estructura de componentes y el *stack* tecnológico (por ejemplo, con Next.js y Tailwind CSS), creando una aplicación web completamente funcional.
4. Con un solo clic en el botón de "Publicar", la aplicación se despliega y se vuelve accesible en internet a través de una URL pública.

Este proceso te permite validar ideas y construir un Producto Mínimo Viable (MVP) en una fracción del tiempo que tomaría tradicionalmente.

Caso Práctico: Creación de un portfolio de desarrollador

Utilizando el ejemplo de la creación de un portfolio personal para un desarrollador, veremos en tres pasos cómo pasar de un simple boceto a una aplicación web funcional y desplegada.

1. **La Conceptualización Visual:** El punto de partida no es el código, sino una idea visual. Se puede utilizar un boceto simple de la interfaz de usuario,

dibujado en una herramienta digital como [Excalidraw](#) o incluso en una servilleta de papel.



2. **La Creación del Contenido:** Paralelamente, se utiliza un LLM (por ejemplo Gemini 2.5) para generar el texto (copy) que poblará la página web. La clave aquí es asignarle a la IA una persona experta mediante un prompt como **"Actúa como un experto en SEO y copywriting..."**, para obtener textos optimizados y profesionales.

Prompt para crear el copy, SEO-friendly de tu portfolio

Quiero crear una página web que sirva como portfolio de mis proyectos como desarrollador de software.

Para que la web sea atractiva e indexable en buscadores, actúa como un experto en SEO y diseño UI/UX para crear el texto (copy) de la web.

Información sobre mí:

- Mi nombre es [__]
- Soy Ingeniero de Software con más de [__] años de experiencia en [__]
- He trabajado en [__]
- Actualmente Freelance Software Developer
- Tengo experiencia con [__]

- Mis proyectos en github más importantes son: [__]

Genera el copy que debe tener la página web, para que pueda copiarlo y pegarlo a un LLM que sea capaz de generarme el sitio.

3. **El Desarrollo y Despliegue automatizado con Firebase Studio:** Con el boceto visual y el contenido textual listos, se utiliza Firebase Studio como la herramienta central de ejecución. Se le proporciona al sistema la imagen del boceto y el texto del copy a través de un prompt, y la IA procede a generar una aplicación web completa.

Prompt para Firebase Studio:

Crea una página web a modo de portfolio para un programador freelance. Basate en el diseño que te adjunto como imagen. [Adjuntar captura de pantalla del diseño de la web]

El copy que debe tener la página es el siguiente: [Copiar la salida del prompt anterior]

Características Clave de Firebase Studio

Firebase Studio ejemplifica una nueva clase de Plataformas Integradas de Desarrollo y Despliegue (IDDPs), que orquestan todo el proceso de desarrollo y despliegue en un entorno unificado con capacidades notables:

- **Generación de "Blueprint":** La IA propone una arquitectura completa (stack tecnológico como Next.js y Tailwind CSS, paleta de colores, tipografía) que es totalmente personalizable por el desarrollador antes de la generación final del código.

- **Entorno de Desarrollo en la Nube:** Proporciona un editor completo (similar a VS Code) en el navegador para revisar y modificar los archivos generados, junto a una ventana de chat para continuar iterando mediante prompts.
- **Despliegue con un Clic:** El botón "Publish" despliega la aplicación directamente a la infraestructura de producción de Firebase, generando una URL pública y accesible al instante.

Conclusiones

El futuro del desarrollo es una colaboración sinérgica entre la inteligencia humana y la artificial.

El desarrollador, armado con sus conocimientos fundamentales, dirige y audita a la IA para aprender, crear y desplegar soluciones a una velocidad sin precedentes, volviéndose más valioso que nunca.

Lejos de ser una amenaza, la IA refuerza el sector, haciendo a los desarrolladores más productivos y satisfechos en su trabajo.

Súmate a nuestro directo del día 2

ACCEDE AQUÍ