

GUÍA DE APRENDIZAJE NO. 9

Desarrollo Asistido con IA: Dashboards, Página Web y Apps

SEMANA 9

INTRODUCCIÓN

En este módulo explorarás cómo la **inteligencia artificial** puede ayudarte a construir, en pocas horas, **aplicaciones web ligeras** que resuelven necesidades reales: una **calculadora financiera**, una **landing page** con **KPIs** y un **dashboard** interactivo. A diferencia del desarrollo tradicional —que exige escribir y depurar mucho código— aquí aprenderás a **formular prompts claros** para que asistentes como **ChatGPT, Gemini o Copilot** generen un primer prototipo que luego **probarás, ajustarás y documentarás**. El enfoque es 100% práctico: integrarás **datos, lógica básica y visualización** en productos que podrás **incrustar en Moodle o sitios institucionales**. No necesitas ser programador experto; sí, tener curiosidad, criterio para **validar resultados** (exactitud de cálculos, lectura en móvil, accesibilidad) y disposición para iterar. Al finalizar, tendrás un **kit mínimo** y una **metodología reproducible** para convertir ideas y datos en artefactos digitales útiles, aplicables tanto en **contextos educativos** como **empresariales**, y una base sólida para usar la IA de forma **responsable, ética y estratégica**.

OBJETIVOS

Objetivo General

1. Que el estudiante comprenda y aplique las posibilidades del desarrollo asistido con IA para crear aplicaciones y productos web funcionales, integrando datos, lógica de negocio y visualización de información.

Objetivos Específicos

1. Generar aplicaciones interactivas básicas (ejemplo: calculadora financiera) a través de código asistido por IA.
2. Diseñar páginas web simples que integren métricas clave, gráficos y tablas utilizando plantillas generadas por IA.
3. Construir dashboards interactivos a partir de archivos de datos (CSV, Excel o Google Sheets) empleando código generado con IA.
4. Publicar y embeber estos productos digitales en plataformas educativas (Moodle) o en páginas web institucionales



TEMARIO DETALLADO

1. Creación de Aplicaciones Interactivas con IA

- Concepto de aplicaciones web ligeras.
- Caso práctico: Calculadora Financiera.
- Entrada de parámetros: monto, tasa, plazo, meses de gracia.
- Cálculo automático de cuota (PMT).
- Generación de tabla de amortización.
- Extensión: cálculo de VAN y TIR.
- Exportación como archivo calculadora.html.

2. Desarrollo de Páginas Web Simples (Landing Pages)

- Estructura básica de un sitio estático con HTML, CSS y JavaScript.
- Incorporación de tarjetas KPI.
- Creación de un gráfico con Chart.js.
- Inclusión de tablas dinámicas o listas de datos.
- Exportación como archivo index.html.

3. Generación de Dashboards con IA

- Concepto de dashboards estáticos vs. interactivos.
- Integración de archivos de datos (CSV, Excel o Google Sheets).
- KPIs automáticos: total, promedio, registros y categorías.
- Visualizaciones con Plotly:
- Serie temporal (ventas/fechas).
- Barras por categoría.
- Resumen mensual o histograma.
- Exportación como archivo dashboard.html.

LECTURAS OBLIGATORIAS

Capítulo XV “ **Desarrollo Asistido con IA: Dashboards, Página Web y Apps** “ eBook
La Guía Definitiva sobre Inteligencia Artificial en la Educación

VIDEOS

Python: The Documentary | An origin story

https://youtu.be/GfH4QL4VqJ0?si=1ElbTe_IYxTSG1Dr

How to Build a Website With ChatGPT In 2024

<https://www.youtube.com/watch?v=Kry6548MBFY>

Build a Python Dashboard with ChatGPT Lightning Fast

https://youtu.be/gS4d0_AKQi8?si=jjwORH7_q6049Fxo

Plotly Tutorial for Beginners de Stats Wire

https://www.youtube.com/playlist?list=PLBSCvBITOLa8rf2kGkP_Bx5xXqT-er4Yq

WEB SITES

CHAT GPT-5 <https://chatgpt.com/>

GOOGLE COLAB <https://colab.google/>

ACTIVIDADES A DESARROLLAR

1. Presentación del TÓPICO : “Desarrollo Asistido con IA: Dashboards, Página Web y Apps” (1 hr)
2. LABORATORIO: DESARROLLO ASISTIDO CON IA (3hrs)

ANEXO I

LABORATORIO DESARROLLO ASISTIDO CON IA

Ejercicio 1: Creación de una Calculadora Financiera con IA

Objetivo del ejercicio:

Aprender a generar con IA una aplicación web funcional que calcule la cuota (PMT), genere tabla de amortización y permita exportar resultados, entendiendo cómo los LLM pueden convertirse en copilotos de programación.

Instrucciones:

1. Abre ChatGPT o Gemini.
2. Copia y pega el siguiente prompt completo:

Prompt detallado:

Quiero que generes un archivo HTML único llamado calculadora.html que contenga:

1. Estructura completa con HTML, CSS y JavaScript embebidos.
2. Formulario con los siguientes campos de entrada:
 - Monto del préstamo (numérico).
 - Tasa de interés anual en % (numérico).
 - Plazo en meses (numérico).
 - Meses de gracia (numérico opcional, por defecto 0).
3. Botones:
 - Calcular.
 - Limpiar formulario.
 - Exportar resultados a CSV.
4. Salidas:
 - Cuota mensual calculada con la fórmula PMT.
 - Totales: intereses pagados y monto total pagado.
 - Tabla de amortización con: Mes, Saldo inicial, Cuota, Interés, Capital, Saldo final.
5. Validaciones:
 - Ningún campo vacío.

- Los valores deben ser mayores que 0.
- Si hay error, mostrar mensaje claro en pantalla.

6. Formatos:

- Usar separadores de miles y formato moneda (es-NI, USD).
- Tasa en formato %.

7. Estilo visual:

- Responsive, diseño limpio.
- Tarjetas para los totales.
- Tabla con bordes claros.

8. Funcionalidad adicional:

- Si hay meses de gracia, capitalizar intereses en ese periodo.
- Funciones para calcular VAN y TIR con tasa de descuento configurable.
- Mostrar VAN y TIR debajo de la tabla.

9. Comentarios en el código explicando cada bloque.

Todo debe ir en un solo archivo HTML listo para abrir en navegador.

3. Guarda el código generado como `calculadora.html`.
4. Ábrelo en el navegador y realiza pruebas con distintos valores.
5. Verifica que el botón de exportar descargue un CSV con la tabla.



Reflexiona:

- ¿Qué tan claro fue el código generado?
- ¿Qué ajustes pediste a la IA para que se adaptara a tus necesidades?
- ¿Qué ventajas tiene hacerlo con IA frente a programarlo desde cero?



Resultado esperado: Un archivo `calculadora.html` que calcule cuotas, genere tabla de amortización, permita exportar resultados y muestre VAN/TIR.



Ejercicio 2: Creación de una Landing Page para Ventas

Objetivo del ejercicio:

Explorar cómo la IA puede generar una página web sencilla y funcional para presentar información de ventas con KPIs, gráficos y formulario de contacto.



Instrucciones:

1. Abre ChatGPT o Gemini.
2. Copia y pega el siguiente prompt:

🔑 Prompt detallado:

Genera un archivo HTML único llamado index.html que contenga:

1. Sección Hero:
 - Título llamativo sobre ventas o marketing.
 - Subtítulo breve.
 - Botón CTA: "Solicitar demo".
 - Imagen o ilustración representada por un placeholder.
2. Tarjetas KPI (en formato grid o flexbox) con valores de ejemplo:
 - Ventas mensuales (USD).
 - Conversión (%).
 - Ticket promedio.
 - Clientes activos.


Cada tarjeta debe mostrar el valor principal y un delta (%).
3. Gráfico con Chart.js:
 - Línea: ingresos mensuales durante 12 meses.
 - Barras: unidades vendidas en los mismos 12 meses.
 - Leyenda y tooltips activados.
 - Ejes con formato moneda y miles.
4. Sección "Beneficios":
 - 3 o 4 columnas con iconos (pueden ser emojis).
 - Cada columna debe contener un título corto y una breve descripción.
5. Formulario de contacto:
 - Campos: nombre, email, mensaje.
 - Checkbox: "Acepto políticas de privacidad".
 - Validación básica: email correcto y checkbox obligatorio.
 - Al enviar, mostrar alerta "Formulario enviado con éxito".
6. Footer:
 - Nombre institucional y datos de contacto.
7. SEO básico:
 - Title y meta description.
 - Etiquetas og (Open Graph).
 - Etiqueta viewport para mobile.
8. Estilo visual:
 - Diseño moderno, limpio, responsive.
 - Tipografía segura para web.
 - Tema claro, con variables CSS para colores.

Todo debe estar en un solo archivo HTML con CSS y JS embebidos.

3. Guarda el resultado como `index.html`.
4. Ábrelo en el navegador y revisa el diseño.
5. Modifica manualmente los valores de los KPI para simular datos propios.

Reflexiona:

- ¿Qué tan fácil fue personalizar la página?
- ¿El formulario funcionó como esperabas?
- ¿Cómo podrías usar una landing generada con IA en tu contexto laboral o académico?

 **Resultado esperado:** Un archivo `index.html` con hero, KPIs, gráfico Chart.js, beneficios, formulario de contacto y footer institucional.

Ejercicio 3: Generación de un Dashboard a partir de una Base de Datos

Objetivo del ejercicio:

Aprender a usar IA para convertir un dataset en un dashboard con KPIs y gráficos interactivos, usando librerías de visualización como Plotly.

Instrucciones:

Crea un archivo `data.csv` con columnas como:

```
fecha,region,producto,unidades,importe
2025-01-05,Managua,Curso IA,3,450
2025-01-06,León,Curso Excel,2,300
2025-01-06,Masaya,Curso IA,5,750
```

- 1.
2. Abre ChatGPT o Gemini.
3. Copia y pega este prompt:

Prompt detallado:

Genera un archivo HTML único llamado `dashboard.html` que:

1. Cargue datos desde `/data/data.csv` usando `fetch` (si hay problemas, incluye una opción alternativa con datos embebidos como JSON).
2. KPIs en tarjetas:
 - Total vendido (suma de importe).
 - Ticket promedio (promedio de importe).
 - Registros (conteo de filas).
 - Productos únicos.

3. Filtros:

- Rango de fechas (date picker).
- Selector de región.
- Al cambiar filtros, recalcular KPIs y actualizar gráficos.

4. Gráficos con Plotly:

- Serie temporal de ventas agregadas por fecha (línea).
- Barras: importe total por producto (orden descendente).
- Histograma: distribución de importes individuales.

5. Exportación:

- Botón para descargar CSV con los datos filtrados.
- Botones para exportar cada gráfico en PNG.

6. Estilo visual:

- Responsive, limpio y moderno.
- Tarjetas para KPIs con color y sombra suave.

7. Formateo:

- Usar formato moneda en es-NI (USD).
- Separadores de miles en KPIs y tooltips.

8. Comentarios en el código explicando cada bloque.

Todo en un único archivo HTML con CSS y JS embebidos.

4. Guarda el resultado como `dashboard.html`.

5. Abre en el navegador (usando Live Server para cargar el CSV).

6. Prueba aplicar filtros y exportar gráficos/CSV.



Reflexiona:

- ¿Qué tan bien integró la IA el CSV en los gráficos?
- ¿Qué limitaciones encontraste frente a un software de BI tradicional?
- ¿Cómo podrías aplicar un dashboard de este tipo en tu área?



Resultado esperado: Un archivo `dashboard.html` que muestre KPIs, 3 gráficos interactivos con Plotly, filtros y exportaciones.

ANEXO II

Mini-Proyecto Integrador con IA (Tarea en Casa)

Objetivo del ejercicio:

Aplicar de manera sencilla lo aprendido en la creación de aplicaciones con IA, generando tres productos básicos: una mini-calculadora, una landing con un KPI y un gráfico, y un pequeño dashboard con datos de ejemplo.



Instrucciones:

1. Abre ChatGPT o Gemini y prepara tres prompts (uno por cada producto).
2. Copia cada uno en el asistente y guarda los resultados como tres archivos:
`mini_calculadora.html`, `mini_landing.html` y `mini_dashboard.html`.



Prompt 1 — Mini Calculadora

Genera un archivo `mini_calculadora.html` con HTML, CSS y JS embebido que:

- Tenga dos inputs: monto y tasa %.
- Al hacer clic en "Calcular", muestre el interés anual ($\text{monto} * \text{tasa}/100$).
- Incluya botón "Limpiar".
- Diseño simple, responsive y con comentarios en el código.



Prompt 2 — Mini Landing Page

Genera un archivo `mini_landing.html` con HTML, CSS y JS embebido que:

- Muestre un título: "Ventas 2025".
- Incluya un KPI destacado (ej: Ventas: \$12,500).
- Cree un gráfico de barras con Chart.js mostrando 6 meses de ventas ficticias.
- Diseño simple y responsive.



Prompt 3 — Mini Dashboard

Genera un archivo `mini_dashboard.html` con HTML, CSS y JS embebido que:

- Use un dataset embebido en JSON con al menos 10 registros (fecha, producto, importe).
- Calcule y muestre: total vendido y número de registros.
- Grafique con Plotly:
 - Serie temporal de importe por fecha.
- Diseño básico y responsive.

3. Abre los tres archivos en tu navegador y verifica que funcionen.
4. Realiza una breve nota (media página) donde expliques:
 - Qué tan fácil fue generar y adaptar los tres archivos.
 - Qué utilidad tendría este mini-proyecto en un contexto académico o laboral.



Reflexiona:

- ¿Cuál de los tres productos fue más sencillo de generar?
- ¿Qué diferencias notaste entre usar IA y programar manualmente?
- ¿Cómo podrías mejorar cada archivo si tuvieras más tiempo?



Resultado esperado:

- Tres archivos HTML (`mini_calculadora.html`, `mini_landing.html`, `mini_dashboard.html`) con funcionalidades básicas.
- Una breve reflexión escrita (máximo una página) sobre la experiencia de integración.