

Laboratorio Guiado

Arquitectura asistida y Diagramación C4 con GitHub Copilot

Proyecto: dotnet-architecture/eShopOnWeb

<https://github.com/NimblePros/eShopOnWeb.git>

Objetivo del laboratorio

Usar GitHub Copilot como **asistente de análisis arquitectónico** para:

- Inferir la arquitectura real desde el código
- Contrastar arquitectura declarada vs implementada
- Construir diagramas **C4 – Contexto y Contenedores**
- Desarrollar criterio arquitectónico (no solo dibujo)

Escenario

Te incorporas como arquitecto a un equipo que mantiene eShopOnWeb.
Debes entender y documentar la arquitectura actual para nuevos stakeholders.

BLOQUE 1 — Exploración arquitectónica asistida

Objetivo

Obtener una **visión macro** antes de entrar al detalle.

Ejercicio 1.1 — Identificación del tipo de sistema

Prompt básico

“¿Qué tipo de sistema es este repositorio y cuál parece ser su propósito principal?”

Resultado esperado

- E-commerce

- Backend web + frontend web
- Arquitectura en capas

Ejercicio 1.2 — Capas principales de la solución

Prompt

“Enumera las capas principales de esta solución y su responsabilidad”

Refinamiento

“¿Cómo se alinean estas capas con Clean Architecture?”

Discusión

- Web
- Application/Core
- Infrastructure

Ejercicio 1.3 — Responsabilidad por proyecto

Prompt

“Describe el rol de cada proyecto dentro de la solución”

Actividad

- Validar con estructura real del repo

BLOQUE 2 — C4 Nivel 1: Diagrama de Contexto

Objetivo

Entender el sistema y su entorno, no su implementación.

Ejercicio 2.1 — Identificación del sistema

Prompt

“Describe el sistema principal desde una perspectiva de negocio, sin entrar en detalles técnicos”

Resultado esperado

- Sistema de e-commerce
- Gestión de catálogo, carrito y pedidos

Ejercicio 2.2 — Usuarios y actores externos

Prompt

“¿Qué tipos de usuarios interactúan con este sistema?”

Refinamiento

“¿Existen sistemas externos o dependencias fuera del sistema?”

Actividad

- Identificar:
 - Usuario final
 - Administrador
 - Servicios externos (si aplica)

Ejercicio 2.3 — Construcción del C4 Contexto

Prompt

“Describe un diagrama C4 nivel contexto para este sistema”

Actividad práctica

1. Copilot genera la descripción.
2. El alumno dibuja el diagrama.
3. Se valida que no haya elementos técnicos innecesarios.

BLOQUE 3 — C4 Nivel 2: Diagrama de Contenedores

Objetivo

Visualizar **contenedores ejecutables reales**.

Ejercicio 3.1 — Identificación de contenedores

Prompt

“Identifica los principales contenedores ejecutables de esta solución”

Resultado esperado

- Web App (ASP.NET Core)
- Base de datos
- Servicios externos (si existen)

Ejercicio 3.2 — Responsabilidad de cada contenedor

Prompt

“Describe la responsabilidad de cada contenedor y cómo se comunican entre sí”

Refinamiento

“¿Qué protocolos o mecanismos de comunicación se utilizan?”

Ejercicio 3.3 — Persistencia y dependencias

Prompt

“¿Dónde se gestiona la persistencia y cómo acceden los contenedores a ella?”

Discusión

- DB como contenedor
- Infraestructura vs dominio

Ejercicio 3.4 — Construcción del C4 Contenedores

Prompt avanzado

“Describe un diagrama C4 nivel contenedores incluyendo tecnologías principales”

Actividad

- Dibujar el diagrama

- Ajustar etiquetas técnicas (HTTP, EF Core, etc.)

BLOQUE 4 — Validación arquitectónica y pensamiento crítico

Objetivo

Evitar el error clásico: **diagrama bonito, arquitectura falsa.**

Ejercicio 4.1 — Contraste arquitectura declarada vs real

Prompt

“¿La arquitectura implementada parece alineada con Clean Architecture?
¿Por qué?”

Actividad

- Revisar dependencias reales
- Detectar excepciones

Ejercicio 4.2 — Riesgos arquitectónicos

Prompt

“¿Qué riesgos arquitectónicos podrías identificar si este sistema creciera en complejidad?”

Ejemplos

- Acoplamiento Web ↔ Core
- Dominio anémico
- Infraestructura infiltrada

Ejercicio 4.3 — Qué no muestra el diagrama

Prompt avanzado

“¿Qué decisiones arquitectónicas importantes no se reflejan en un diagrama C4?”

Aprendizaje clave

- C4 no reemplaza ADRs
- Diagramas ≠ decisiones

Cierre del laboratorio

Reflexión guiada

- ¿Qué ayudó más: prompts amplios o específicos?
- ¿Dónde Copilot fue ambiguo?
- ¿Qué partes requirieron criterio humano?