



Arquitectura de software en la era de la Inteligencia Artificial

La ilusión del "hazme una aplicación"

De escribir código a diseñar, dirigir y verificar sistemas inteligentes

César Alonso Magaña Gavilanes
Software Insights

Jornadas de Vinculación 2026

Facultad de Telemática
Universidad de Colima

Ponencia virtual por Zoom

El recurso escaso cambió

La IA acelera la producción de código; la ingeniería debe acelerar la validación.

ANTES

**escribir
código**



AHORA

**diseñar +
verificar sistemas**

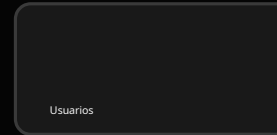
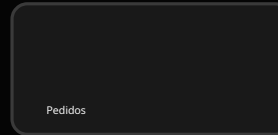
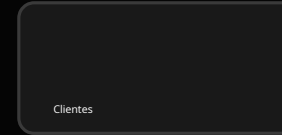
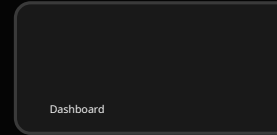
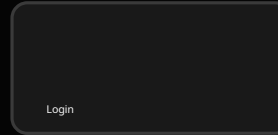
Pregunta central: si la IA puede programar más rápido, ¿qué valor aporta un ingeniero?

La ilusión del "hazme una aplicación"

Un prompt puede parecer especificación, pero normalmente no lo es.

PROMPT

"Hazme un sistema para administrar una empresa exportadora de aguacate."



**Parece terminado
porque tiene pantallas.**

Pero todavía faltan reglas, estados, permisos, datos, seguridad y operación.

Prototipo no es sistema empresarial

El prototipo responde “qué se ve”; el sistema responde “qué debe pasar”.

LO QUE LA IA GENERA RÁPIDO

pantallas
CRUD
APIs simples
componentes
datos simulados
flujo feliz

LO QUE EXIGE INGENIERÍA

reglas reales
permisos y auditoría
excepciones
trazabilidad
operación 24/7
datos vivos

ARQUITECTURA

Decisiones que convierten código en un sistema mantenible, seguro y evolutivo.

Cuando el CRUD se convierte en negocio

Un proceso empresarial vive en estados, responsables, evidencias y auditoría.



Cada transición necesita una razón verificable



Arquitectura = decisiones + trade-offs

No existe arquitectura perfecta; existe arquitectura adecuada al contexto.

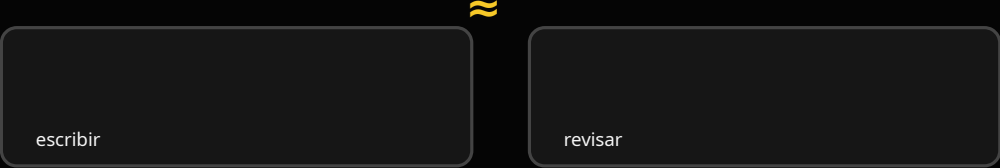


**Una IA puede darte diez respuestas convincentes y contradictorias.
El arquitecto decide según restricciones reales.**

Nueva asimetría: generar > revisar

El cuello de botella deja de ser producir código; pasa a ser comprenderlo e integrarlo.

ANTES



CON IA



regresiones

deuda técnica

seguridad rota

duplicación

dependencias

migraciones peligrosas

¿Quién verifica a la IA?

La arquitectura moderna debe ser ejecutable: reglas que la máquina pueda comprobar.



FRONTEND

```
if (role !== "ADMIN") hideButton();
```

BACKEND

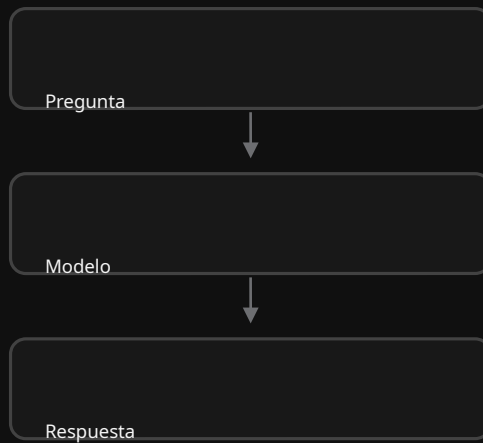
```
can(user, action, resource, context)
```

Ocultar un botón no es autorización.

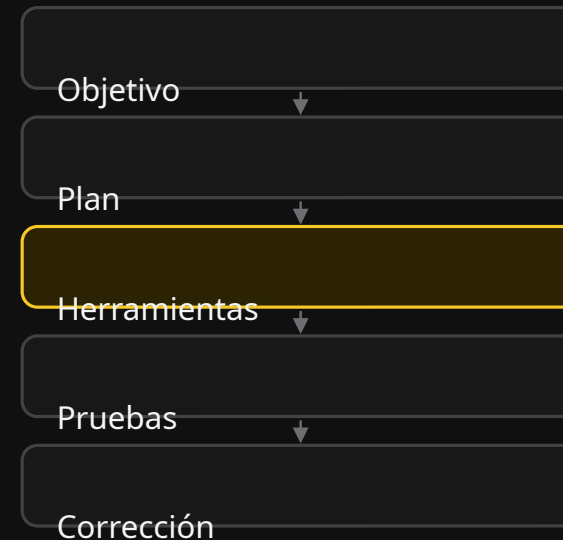
Agentes: el siguiente salto

De responder preguntas a ejecutar objetivos con herramientas.

LLM TRADICIONAL



AGENTE



Un agente puede leer archivos, modificar código, ejecutar comandos, observar fallos y volver a intentar.

El ingeniero como orquestador

Equipos humanos + agentes especializados.

Frontend Agent

Security Agent

INGENIERO / ARQUITECTO

define · divide · dirige · verifica

Backend Agent

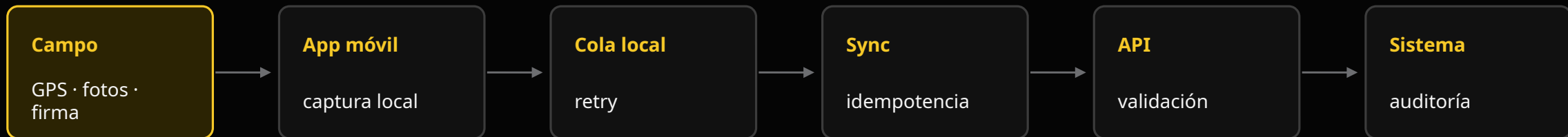
DevOps Agent

Testing Agent

**Tu valor no será escribir cada línea.
Será saber qué construir, cómo dividirlo y cómo comprobarlo.**

El mundo real no siempre tiene señal

Offline-first, edge y sincronización importan cuando el software toca operación física.



Conceptos técnicos:

IndexedDB / SQLite

compresión

conflictos

eventual consistency

edge AI

seguridad móvil

Para Telemática: redes, cloud, edge, ciberseguridad e infraestructura vuelven al centro.

De proyecto a producto SaaS

Un sistema que resuelve un problema real puede convertirse en producto.

**Pasar a SaaS no es “subirlo a Internet”.
Es diseñar aislamiento, configuración, soporte, métricas y modelo comercial.**



Esta es una ruta natural para estudiantes emprendedores y equipos técnicos.

¿Qué debería aprender un estudiante hoy?

Rutas técnicas que ganan valor con IA.

Software Architecture + AI

DDD · BPMN · eventos · system design

AI Systems & Agents

LLMs · tool calling · evals · MCP

Cloud & Distributed Systems

AWS · contenedores · observabilidad

Cybersecurity

IAM · AppSec · LLM security

Data / AI / ML

Python · RAG · MLOps

IoT / Edge / Robotics

sensores · redes · visión

No abandonen fundamentos: programación, redes, sistemas operativos, bases de datos y algoritmos.

Cómo demostrar que sabes

No coleccionen cursos: construyan evidencia pública de capacidad.

30 días

Fundamentos + IA práctica

TypeScript / Python · Git · pruebas básicas · APIs LLM

60 días

Sistemas reales

RAG · agentes · Docker · CI/CD · cloud básico

90 días

Proyecto demostrable

Auth · roles · workflow · DB · tests · deploy · agente IA

Meta: un repositorio que pueda enseñar cómo piensas, no sólo qué tutorial seguiste.

Aquí entra Software Insights

Construir tecnología y desarrollar a las personas capaces de construirla.

SOFTWARE

Desarrollo y arquitectura de sistemas empresariales.

ACADEMY

Cursos, mentoría y capacitación empresarial.

TALENT

Estancias, retos técnicos y red de colaboradores.

El objetivo no es sólo aprender herramientas. Es desarrollar criterio para crear sistemas reales.



Tres puertas para continuar

En Zoom compartiré la liga en el chat:

academy.softwareinsights.com.mx

Quiero capacitarme

Cursos presenciales y online

Busco una estancia

Desarrollo · QA · IA · DevOps

Represento una empresa

Capacitación · arquitectura · consultoría

contacto@softwareinsights.com.mx | contacto@cesaralonsomagana.com | WhatsApp 341 124 2492

cesaralonsomagana.com