5 semanas | En vivo + material grabado · 15 cupos | Claude Code o Codex | Seguridad incluida

Programar con agentes de IA como un equipo senior

Armás tu sistema de trabajo con agentes: memoria, reglas y revisores. Y aprendés a comprobar lo que el agente dice que hizo.

PARA QUIÉN

Programadores junior y semisenior, freelancers técnicos y fundadores técnicos que ya usan un agente de programación y quieren llevarlo a producción sin sustos.

CÓMO TRABAJAN LOS AGENTES DE IA

Trabajás con Claude Code o Codex sobre tu propio repositorio. El agente propone y escribe; vos definís las reglas, lo hacés revisar por otro agente y verificás el cambio real.

Cómo se cursa: Cada semana tenés material grabado para estudiar y practicar antes de la clase. La clase es en vivo y sirve de apoyo: resolvemos dudas, explicamos lo que se complica y enseñamos lo justo. Las clases en vivo no se graban.

Soporte incluido: Todas las capacitaciones incluyen 3 meses de soporte, ayuda y acompañamiento ante dudas, desde la última clase.

Lo que te llevas

- Tu sistema de trabajo con el agente: memoria externa, reglas y revisores.
- Una auditoría cruzada de tu repositorio, hecha por otro participante.
- Una lista de chequeo de seguridad de tu flujo: secretos, permisos y paquetes.
- El hábito de verificar el cambio real, no el resumen del agente.

Dicta Julián Bagilet

Más de 19 años en tecnología. Estudio de software en Villa General Belgrano, Córdoba.

Temario

1

SEMANA 1

Memoria externa y reglas

Separás el mecanismo del conocimiento. Armás notas, un índice y un arranque de sesión liviano para que el agente no empiece ciego.

QUÉ VEMOS

- Por qué el agente empieza cada sesión sin recordar nada y cómo una carpeta de notas en Obsidian le devuelve el contexto de tu proyecto.
- La diferencia entre mecanismo (cómo se guarda y se carga la memoria) y conocimiento (lo que sabés de tu proyecto).
- Cómo escribir reglas cortas y comprobables, como "no toques las migraciones sin avisarme", en vez de párrafos vagos.
- Cómo armar un arranque liviano: un índice de pocas líneas que apunta a las notas, para no gastar el espacio del agente. Armás tu carpeta de Obsidian con notas por proyecto, decisiones y pendientes.

Al terminar la semana: Terminás la semana con tu memoria externa en Obsidian funcionando, con índice y reglas, y probada en una sesión nueva del agente.

2

SEMANA 2

Verificar, no creer

Revisores que no construyeron lo que revisan. Comparás el cambio real contra lo que el agente dijo: cerrado no es hecho.

QUÉ VEMOS

- Qué significa que cerrado no es hecho: el agente dice que arregló algo, pero nadie corrió la prueba ni abrió la pantalla.
- Por qué el revisor debe ser otro agente con otras instrucciones: quien construyó tiende a confirmar su propio trabajo.
- Cómo comparar los archivos realmente cambiados con el resumen del agente y detectar cambios que no pediste o que faltan.
- Cómo escribir un criterio de aceptación antes de pedir el cambio, para que verificar no dependa de una opinión.

Al terminar la semana: Terminás la semana con un revisor configurado y un cambio real verificado contra su criterio, no contra el resumen.

Temario (continuación)

3

SEMANA 3 SEGURIDAD

Permisos del agente y cadena de suministro

Limitás lo que puede hacer el agente y sus conectores. Usás un escáner de secretos antes de cada commit y elegís paquetes con criterio.

QUÉ VEMOS

- Qué puede hacer un agente en tu computadora: leer archivos, ejecutar comandos, tocar bases. Cómo darle solo lo que necesita cada tarea.
- Qué es un conector y por qué darle acceso de lectura es muy distinto a darle acceso de escritura.
- Cómo funciona un escáner de secretos que mira cada commit y frena la subida si encuentra una clave pegada en el código.
- Qué es la cadena de suministro: paquetes con nombre casi igual al verdadero. Cómo revisar autor, antigüedad y origen antes de instalar.

Al terminar la semana: Terminás la semana con permisos mínimos configurados, el escáner de secretos activo y tu criterio para elegir paquetes escrito.

4

SEMANA 4 SEGURIDAD

Supabase con agentes: base de datos, accesos y funciones

Armás la base de datos de tu proyecto con el agente, con reglas de acceso comprobadas y un conector de solo lectura.

QUÉ VEMOS

- Qué te da Supabase en un solo lugar: base de datos, inicio de sesión, archivos y funciones en el servidor. Cuándo conviene y cuándo no.
- Cómo pedirle al agente cambios en la base con migraciones: archivos que quedan en tu repositorio, se pueden revisar y se pueden deshacer, en vez de tocar tablas a mano.
- Reglas de acceso por fila (RLS): qué son, por qué una tabla sin reglas queda abierta a cualquiera y cómo probarlas con dos usuarios.
- Cómo conectar al agente con tu proyecto de Supabase en modo solo lectura, y por qué la clave de servicio no va nunca en el chat ni en el código de la pantalla.

Al terminar la semana: Terminás la semana con tu base en Supabase creada por migraciones, las reglas de acceso comprobadas y el agente conectado en solo lectura.

Temario (continuación)

5

SEMANA 5

Lo determinístico no se delega

Qué decide el código y qué redacta el modelo. Hacés el pre-mortem de tu proyecto y la auditoría cruzada entre participantes.

QUÉ VEMOS

- Qué decide el código y qué redacta el modelo: un precio o un permiso se programan, el borrador de un mail lo escribe la IA.
- Cómo encontrar en tu proyecto lugares donde el modelo decide algo que debería resolver una regla fija.
- Qué es un pre-mortem: imaginar que el proyecto falló dentro de unos meses y listar los motivos antes de que ocurra.
- Cómo funciona la auditoría cruzada: otro participante revisa tu repositorio con tu lista de chequeo y vos revisás el suyo.

Al terminar la semana: Terminás con el pre-mortem de tu proyecto, la auditoría cruzada hecha y una lista corta de lo que se corrige primero.

Seguridad incluida en esta capacitación

- Permisos mínimos para el agente y para los conectores que le das.
- Criterio para elegir paquetes y servidores de herramientas sin caer en uno falso.
- Casos públicos documentados de ataques a la cadena de suministro, explicados desde la defensa.
- Escáner de secretos antes de cada commit, para que una clave no llegue al historial.
- Reglas de acceso de Supabase comprobadas con dos usuarios, y el agente conectado a tu base solo en modo lectura.
- Una persona aprueba lo que no tiene vuelta atrás.

Qué no se enseña ni se hace

- ✕ Agentes sobre sistemas o cuentas ajenas.
- ✕ Cómo evadir los términos de un proveedor.
- ✕ Promesas de hacerlo todo diez veces más rápido.

Requisitos

Necesitás experiencia previa programando y trabajar sobre un repositorio propio. Hace falta un plan pago de Claude o de Codex (OpenAI), no incluido en el precio.

Inversión y formas de pago

Precio de la capacitación

$ 180.000

En vivo + material grabado · 15 cupos. Valores en pesos argentinos.

mercado pago

CUOTAS **3 cuotas de $ 60.000**
sin interés, con tarjeta, por Mercado Pago

TRANSFERENCIA **$ 153.000**
15% de descuento pagando por transferencia bancaria

Cómo inscribirte

1. **Escribinos**
Contanos qué querés lograr por WhatsApp.

2. **Confirmamos cupo y fecha**
Te avisamos apenas hay fecha y lugar en la cohorte.

3. **Reservá tu lugar**
Elegís cuotas o transferencia y recibís la confirmación por escrito.

WhatsApp +54 9 3546 54-6063

julianbagilet.com/capacitaciones/ · Instagram @mangopips_ok · Villa General Belgrano, Córdoba

Condiciones

Valores vigentes hasta el 31 de enero de 2027. Certificado de aprobación del estudio de Julián Bagilet, sin validez oficial. Derecho de arrepentimiento de 10 días corridos en compras a distancia por personas (Ley 24.240, art. 34). Si una cohorte no llega al mínimo de inscriptos, se reprograma o se devuelve el 100% de lo abonado. Los casos de las clases se explican como patrones, sin nombres ni datos de clientes.