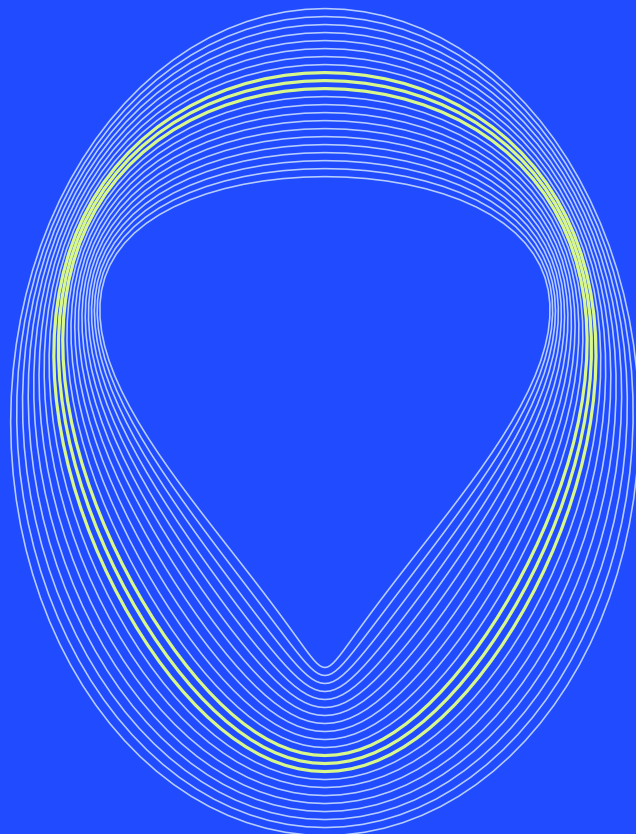


Reinventar la rueda, a propósito.

Por qué la IA vuelve a abrir la decisión de construir software
alrededor del negocio.



Resumen ejecutivo

El software debería adaptarse al negocio.

Durante años, la recomendación habitual fue comprar software y adaptar el negocio a él. La IA nos da una razón para revisar esa decisión. Nuestra hipótesis es que un menor esfuerzo de implementación puede hacer viable el software a medida para más empresas y más procesos. La oportunidad está en preservar lo que hace efectivo a un negocio, mejorar lo que genera fricción e incorporar inteligencia al trabajo.

Una decisión que vale revisar.

La IA puede cambiar el esfuerzo de implementar y ampliar dónde el software a medida resulta viable.

Dos roles para la IA.

Potenciar el desarrollo e incorporar inteligencia al trabajo de la compañía.

Una responsabilidad completa.

Evaluar adaptación, calidad, operación y mantenimiento; no solo generación de código.

El negocio es el punto de partida.

Que una categoría de software exista no significa que todas las empresas necesiten lo mismo.

“No reinventes la rueda” es un buen consejo cuando la rueda disponible sirve. En software, también se convirtió en un atajo para una decisión más difícil: cuánto debería cambiar una empresa su forma de trabajar para adaptarse al producto que puede comprar.

Un producto estándar distribuye el costo de desarrollo entre muchos clientes. Es un modelo poderoso. Pero sus procesos, estructuras de datos y prioridades también están diseñados para muchos clientes. La distancia entre ese producto y la operación de una compañía suele convertirse en configuración, integraciones, planillas, carga duplicada y trabajo por fuera del sistema.

La hipótesis de Impactia es que el desarrollo potenciado por IA puede cambiar el costo de cerrar esa distancia. Cuando implementar requiere menos esfuerzo, una aplicación a medida merece evaluarse donde antes parecía demasiado cara o lenta. La decisión debería revisarse según el valor de la adaptación, no quedar resuelta porque existe una categoría de software.

Esta es una posición sobre dónde invertir. No es una predicción de que todas las compañías reemplazarán sus sistemas. La pregunta útil es qué partes de una operación merecen software diseñado específicamente alrededor de ellas.

Construir donde la adaptación crea valor. Reutilizar donde el problema ya está resuelto.

Cambian los costos. La responsabilidad sigue.

Generar código más rápido es un componente de la entrega, no una medida de valor para el negocio.

La IA puede asistir en implementación, pruebas, documentación, exploración de código e iteración. Nuestra inferencia es que reducir ese esfuerzo puede ampliar el conjunto de proyectos que vale la pena intentar. Depende del equipo, la tarea, las integraciones y el estándar de calidad requerido.

Los resultados publicados son diversos. En tres experimentos con 4.867 desarrolladores, Cui y sus colegas encontraron un 26,08% más de tareas completadas con un asistente de código; el error estándar fue de 10,3 puntos porcentuales. Es un resultado medido en esos contextos, no una garantía de entrega. [1]

En otro contexto, el estudio de METR de principios de 2025 con 16 desarrolladores experimentados de código abierto encontró que las tareas con IA demoraron un 19% más. Su actualización de febrero de 2026 describió problemas de selección y medición que impiden tomar las nuevas estimaciones como una medida confiable de la mejora actual. [2] [3]

La investigación de DORA de 2025 describe a la IA como un amplificador de fortalezas y debilidades organizacionales. Nuestra lectura es que requisitos claros, revisión, pruebas y aprendizaje siguen siendo centrales para convertir capacidad tecnológica en entregas útiles. [4]

La evidencia no establece un multiplicador universal de productividad, un precio estándar ni un plazo garantizado de producción. Proponemos usar IA con intención, definir una primera entrega acotada y medir si mejora el proceso completo.

La velocidad y el costo deben demostrarse dentro de un alcance definido.

Adaptar el software al trabajo que vale preservar.

A medida no significa convertir cada hábito existente en una funcionalidad.

Toda compañía combina diferencias valiosas y fricción acumulada. Una regla de decisión puede condensar años de conocimiento operativo. Una tarea manual puede existir únicamente porque dos sistemas nunca se conectaron. No deberían recibir el mismo tratamiento.

Antes de diseñar la aplicación, trabajamos con quienes ejecutan el proceso y con quienes responden por sus resultados. Identificamos las decisiones relevantes, la información de la que dependen, las excepciones y qué significa completar correctamente el trabajo. Esa comprensión humana define qué debería hacer la tecnología.

Después separamos tres cosas: las restricciones que hay que respetar, las prácticas que crean una ventaja y los pasos que pueden eliminarse o rediseñarse. El objetivo es expresar una mejor operación en software. Acelerar un proceso deficiente puede ser una forma costosa de conservar el problema.

Un sistema a medida debería explicitar las reglas importantes, darle a cada rol una vista útil y conectarse con los sistemas responsables de los datos. También debería permitir que la empresa evolucione su proceso sin depender de las prioridades de un producto generalista. Eso requiere diseño y responsables de mantenimiento, además de código.

- Preservar las reglas y relaciones que hacen efectivo al negocio.
- Simplificar traspasos, duplicaciones y aprobaciones innecesarias.
- Hacer visibles las excepciones, con responsable y próxima acción.

El proceso es el punto de partida. Una mejor operación es el objetivo del diseño.

IA para construir. IA para operar.

La forma de desarrollar y las capacidades que entregamos son dos decisiones conectadas.

El desarrollo potenciado por IA se refiere al proceso de ingeniería. Ayuda a explorar, implementar e iterar; las personas siguen siendo responsables de los requisitos, la arquitectura, la revisión y las decisiones de publicación. Su valor se evalúa mediante software funcionando, calidad y esfuerzo para alcanzar un resultado aceptado.

La IA aplicada se refiere a la aplicación en sí. Puede interpretar información no estructurada, recuperar conocimiento de la empresa, proponer decisiones y coordinar acciones. Estas capacidades son útiles cuando forman parte de un proceso con contexto, permisos y mecanismos de revisión.

Cada capacidad de IA necesita una función definida. ¿Qué información puede usar? ¿Qué resultados pueden aceptarse automáticamente? ¿Cuáles requieren revisión? ¿Quién resuelve la incertidumbre? ¿Qué ocurre si un modelo o una integración deja de estar disponible? Esas decisiones convierten una capacidad prometedora en una parte utilizable de la operación.

Separamos interpretación de autoridad. Un modelo puede interpretar una solicitud mientras reglas explícitas determinan si una acción está permitida. Las acciones de mayor impacto pueden requerir aprobación, un registro de actividad y un camino de recuperación. El diseño apropiado depende del proceso y sus consecuencias.

El resultado debe medirse en el trabajo que se completa: menor esfuerzo manual, decisiones más confiables o mejor acceso a información. Una interfaz conversacional tiene valor cuando mejora ese trabajo.

La IA necesita una función, un límite y una contribución que pueda medirse.

Reutilizar las bases. Diseñar la adaptación.

El software a medida no exige reconstruir todas las capas.

A veces se presenta la decisión como comprar un producto completo o escribir todo desde cero. Una aplicación a medida puede combinar servicios de nube, identidad, bases de datos, librerías, plataformas de negocio y modelos de IA existentes.

La inversión debe concentrarse en lo específico del negocio: procesos, decisiones, comportamiento de las integraciones y experiencia de uso. Las bases maduras reducen lo que hay que desarrollar y mantener. Reutilizarlas es compatible con construir una solución para una compañía concreta.

Los sistemas existentes pueden seguir siendo las fuentes de registro. Una aplicación nueva puede conectarlos, darle al equipo un espacio de trabajo coherente e incorporar IA donde sea útil. Reemplazar un sistema central es una decisión distinta, que debe justificarse por sus beneficios, costos y requisitos de migración.

La arquitectura también necesita un plan de salida. Los acuerdos deben aclarar el acceso al código, la documentación, los datos y la infraestructura. Las integraciones y dependencias de modelos deben ser comprensibles para quienes mantendrán la solución. La adaptación tiene valor cuando le da al negocio mayor control sobre el cambio.

Bases compartidas	Identidad, infraestructura, bases de datos, sistemas existentes y modelos.
Software específico del negocio	Procesos, reglas, permisos, integraciones y experiencia de uso.
Inteligencia aplicada	Interpretación y asistencia dentro del trabajo, con los controles adecuados.

Reinventar la parte que necesita adaptarse. Conservar las bases que ya funcionan.

Una mejor decisión: comprar, extender o construir.

Elegir el menor compromiso que permita alcanzar la adaptación necesaria.

La decisión empieza por la operación. Un producto estándar puede ser la mejor respuesta para una necesidad común con poca diferenciación. Extender un sistema puede cerrar una brecha concreta. Construir una aplicación se justifica cuando el proceso es distintivo, los arreglos manuales cuestan o la empresa necesita controlar su evolución.

Las alternativas deben compararse con el mismo horizonte y alcance. Hay que incluir implementación, configuración, integración, capacitación, operación, consumo de modelos, soporte, mantenimiento y cambios esperados. Un precio inicial bajo no equivale a un costo total bajo.

La falta de adaptación también tiene un efecto económico. Registrá el costo recurrente del trabajo manual, las demoras, la recarga de información y los errores evitables. Estimá con prudencia y explicitá los supuestos. Después probalos en una entrega acotada antes de comprometer un programa más amplio.

Un sistema a medida necesita un responsable y un camino creíble de mantenimiento. Si faltan, mejorar el modelo operativo puede ser más urgente que contratar desarrollo. La IA vuelve relevante esta discusión; no la elimina.

Dirección	Cuándo tiene sentido	Qué evaluar
Comprar	El proceso estándar ya se adapta.	Configuración, integración, licencias y costo de cambiar.
Extender	Existe un sistema valioso con una brecha acotada.	Interfaces disponibles, propiedad de los datos y límites.
Construir	Lo específico del negocio justifica software propio.	Alcance, responsables y costo durante toda su vida útil.

La mejor opción se justifica por adaptación, costo total y capacidad de mantenimiento.

Medir el trabajo durante todo el ciclo de vida.

La aplicación tiene éxito cuando la operación mejora después de empezar a usarla.

Acordá la línea de base antes de desarrollar. Elegí un proceso delimitado, un responsable de negocio y pocas medidas relevantes. Registrá cuánto demora, cuánto esfuerzo manual requiere, con qué frecuencia necesita correcciones y si las personas pueden completarlo con las herramientas actuales.

Durante el desarrollo, evaluá versiones funcionando con tareas representativas y restricciones reales. Medí el esfuerzo de revisión y retrabajo además de la velocidad de implementación. Un primer borrador más rápido puede generar trabajo después; una entrega completa incluye integración, validación, despliegue y adopción.

Después de la puesta en marcha, compará con la línea de base durante un período significativo. Considerá cambios en volumen, complejidad y dotación. Incluí revisión humana, soporte, infraestructura y consumo de IA en el costo operativo. Si la comparación tiene limitaciones, explicitálas en lugar de convertirla en una promesa comercial.

Definí de antemano qué justificaría continuar, cambiar de dirección o detenerse. Un piloto útil reduce incertidumbre sobre el caso de negocio. Su objetivo es determinar si funciona en esta empresa, no confirmar una historia predeterminada sobre la IA.

- Flujo: tiempos de resolución, espera y volumen completado.
- Calidad: correcciones, excepciones y resultados aceptados.
- Adopción: uso efectivo del proceso previsto para completar el trabajo.
- Economía: desarrollo, revisión, operación, mantenimiento y evolución.

Una entrega funcionando es evidencia. Una operación mejor es el resultado.

Una nueva forma de dar servicio de software.

El recurso más valioso es entender qué merece construirse.

Impactia combina conocimiento del negocio, diseño de producto, ingeniería e IA aplicada en un mismo servicio. Usamos IA para reducir trabajo repetitivo de implementación y facilitar la iteración. La incorporamos a la aplicación cuando ayuda a mejorar la operación.

El trabajo comienza con una prioridad definida. Acordamos el proceso, las integraciones, los criterios de aceptación y un primer alcance. El equipo evalúa software funcionando desde temprano, para decidir sobre algo concreto.

Cada etapa debe producir un resultado útil: una comprensión compartida, una versión operativa, una entrega validada y un plan de soporte y evolución. Alcance, responsabilidades, plazos e inversión deben quedar en la propuesta. Los beneficios de velocidad o ahorro se demuestran en el proyecto.

Nuestra hipótesis es que esta combinación puede acercar el software a medida a más compañías y más partes de su operación. El servicio evoluciona desde aportar capacidad de implementación hacia hacerse responsable de traducir una intención de negocio en un sistema que funcione.

Reinventar la rueda tiene sentido cuando la disponible obliga al negocio a tomar el camino equivocado. Creemos que la irrupción de la IA permite volver a hacerse la pregunta, con mejores herramientas y una manera más práctica de encontrar la respuesta.

Que el negocio defina el software. Que la IA haga más viable esa adaptación.

Fuentes y alcance

Este documento presenta la hipótesis de Impactia y un marco de decisión propuesto. Los hallazgos publicados se atribuyen a sus fuentes; no son resultados de clientes de Impactia. Investigación revisada el 21 de octubre de 2026.

[1] Cui et al. (2025). Field experiments on AI-assisted software development. Microsoft Research.

[Leer la fuente original](#)

[2] METR (July 2025). Randomized study of experienced open-source developers.

[Leer la fuente original](#)

[3] METR (February 2026). Update on developer-productivity measurement and selection effects.

[Leer la fuente original](#)

[4] DORA (2025). Research on AI-assisted software delivery and organizational capabilities.

[Leer la fuente original](#)

Construyamos alrededor de tu negocio.

hello@impactia.ai

[impactia.ai](#)