Entelgy

La IA no va solo de generar código

Va de reducir lo que te cuesta desarrollar y evolucionar software.

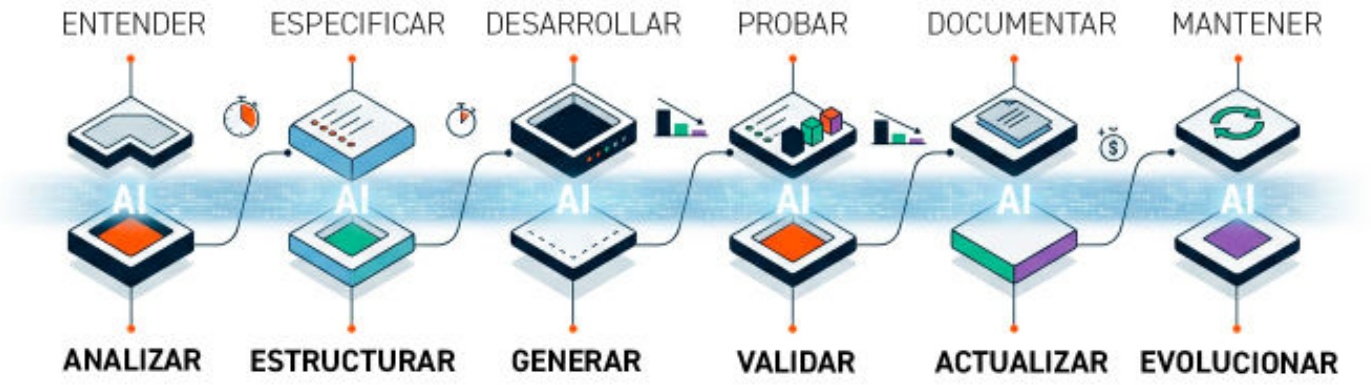
Guía práctica para quien decide cómo modernizar aplicaciones y transformar el desarrollo con IA.

6 PÁG. | 5 MIN DE LECTURA | DEL LEGACY A SPEC IA

entelgy.com

¿Dónde estás gastando realmente tu presupuesto de desarrollo?

Ese es el cambio que proponemos abordar en este documento. No se trata únicamente de incorporar IA a la programación, sino de analizar cómo puede intervenir en todo el ciclo de vida del software para reducir tiempos, eliminar tareas de bajo valor, aprovechar mejor el conocimiento de los equipos y hacer económicamente viable la modernización de aplicaciones que hasta ahora resultaban demasiado complejas o costosas de transformar.



¿Estás utilizando IA donde realmente está el coste?

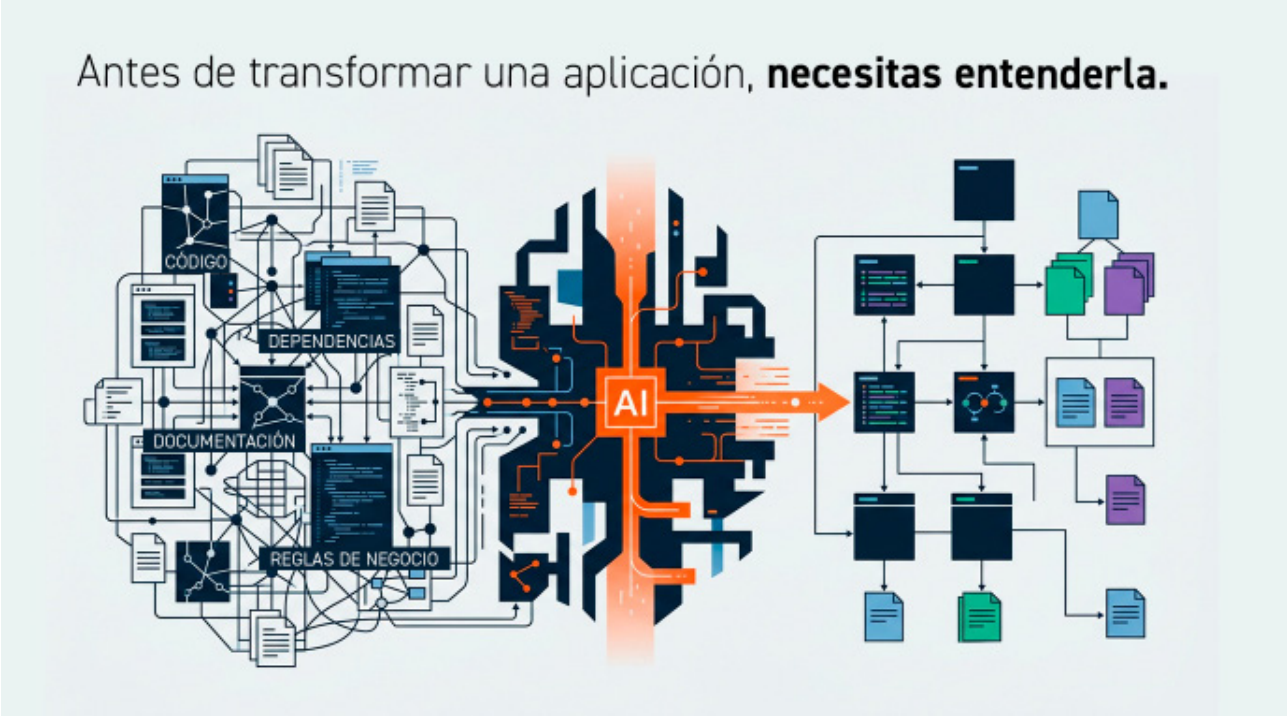
Aplicar IA únicamente a la programación puede hacer más productivo a un desarrollador. Aplicarla al conjunto del ciclo de vida puede cambiar una ecuación mucho mayor.

En cada una de estas fases existe esfuerzo susceptible de ser reducido, asistido o automatizado. Por eso, la oportunidad no debería medirse únicamente en código generado o tareas completadas, sino en horas de trabajo que dejan de ser necesarias para llevar una necesidad desde su definición hasta producción y mantenerla después.

Antes de preguntarte qué herramienta incorporar, hazte otra pregunta: ¿dónde consume hoy más recursos tu ciclo de desarrollo? Ahí debería comenzar tu estrategia de IA.

Si solo aplicas IA al código, solo estás actuando sobre una parte del coste.

Tu legacy no solo acumula deuda técnica. También contiene conocimiento de negocio. **La IA puede ayudarte a recuperarlo.**



¿Cuánto te cuesta entender lo que ya tienes?

Imagina una aplicación que lleva diez o quince años funcionando. Ha pasado por distintos equipos, tecnologías y evoluciones. Parte de su documentación está desactualizada, algunas decisiones solo están reflejadas en el código y determinadas reglas de negocio siguen dependiendo del conocimiento de personas concretas. La aplicación funciona, pero cada vez resulta más difícil y costoso modificarla.

Ahora necesitas modernizarla. ¿Por dónde empiezas?

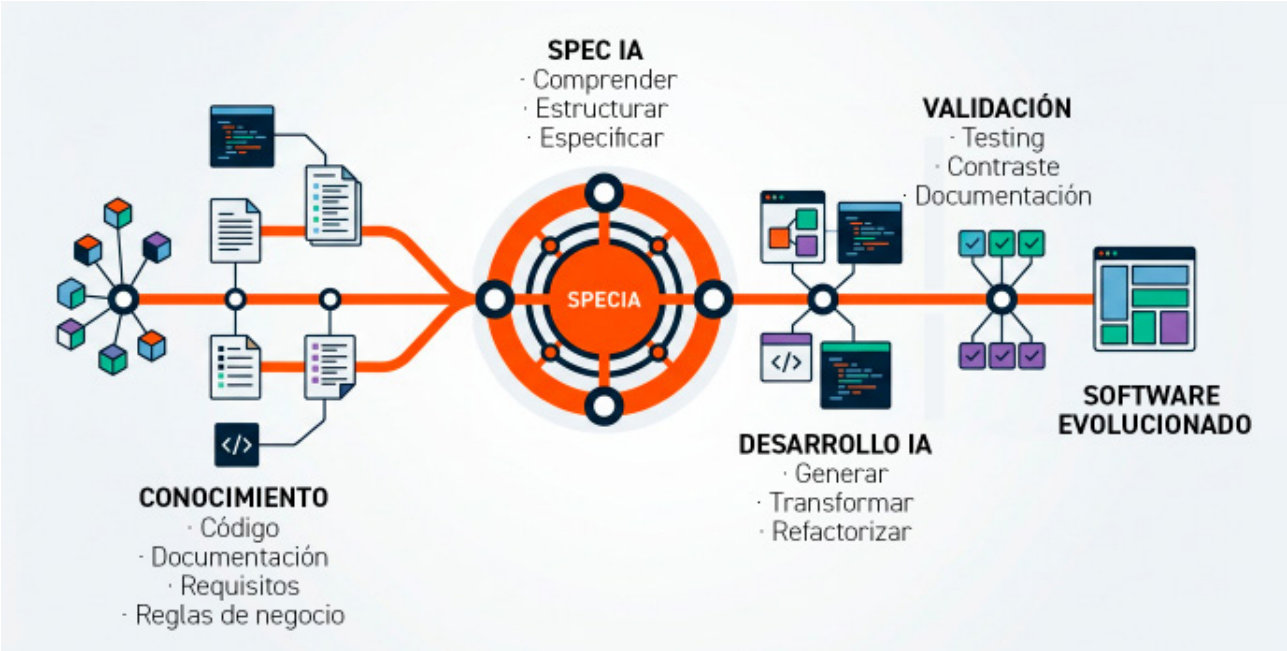
Antes de transformar una aplicación necesitas entenderla: conocer su arquitectura, identificar dependencias, reconstruir reglas de negocio, localizar componentes críticos y determinar qué debes conservar, qué puedes transformar y qué ya no tiene sentido mantener. En entornos legacy, esta fase puede convertirse en una barrera para la modernización.

¿Y si pudieras acelerar también esa parte?

La IA puede ayudar a analizar código existente, explicar componentes, detectar relaciones, facilitar procesos de ingeniería inversa y generar o actualizar documentación. El objetivo no es sustituir el criterio de arquitectos y desarrolladores, sino proporcionarles una base de conocimiento que reduzca el tiempo que necesitan para comprender el sistema.

Esto introduce una variable económica especialmente relevante. Si necesitas menos esfuerzo para conocer tu punto de partida, también necesitas menos esfuerzo para empezar a transformarlo. La modernización con IA comienza, por tanto, antes de generar la primera línea de nuevo código. Comienza utilizando la tecnología para recuperar y estructurar el conocimiento que ya existe dentro de tus aplicaciones

Si generar código es cada vez más fácil, ¿qué se convierte en crítico?



Saber exactamente qué quieres construir.

Cuanto mayor sea la capacidad de la IA para generar software, más importante será la calidad del contexto y de las especificaciones que recibe. Generar código rápidamente aporta poco valor si los requisitos son incompletos, si desconoces determinadas reglas de negocio o si el resultado no responde a lo que realmente necesita la aplicación.

Aquí entra **Spec IA.**

Spec IA lleva la inteligencia artificial a una fase anterior al desarrollo: la comprensión y estructuración de lo que necesitas construir. La IA puede ayudar a trabajar sobre código, documentación, requisitos y conocimiento funcional para convertir información dispersa en especificaciones más estructuradas que sirvan de referencia durante el desarrollo.

¿Por qué importa?

Piensa de nuevo en tu aplicación. ¿Está completamente documentada? ¿Conoces todas sus reglas de negocio? ¿Sabes qué debe mantenerse después de modernizarla? ¿Podrías proporcionar a una IA todo el contexto necesario para generar correctamente una nueva funcionalidad? Si alguna respuesta es negativa, el problema no está todavía en el código.

Está en la especificación.

Con Spec IA, el objetivo es construir una continuidad entre lo que ya sabes de la aplicación, lo que necesitas que haga y lo que finalmente desarrollas y validas. Así, la IA deja de ser únicamente un asistente del programador y empieza a participar en un proceso mucho más amplio.

Cuanto más fácil sea generar código, más importante será especificar correctamente qué debe hacer ese código.

¿Dónde merece realmente la pena aplicar IA?

Imagina una aplicación que lleva diez o quince años funcionando. Ha pasado por distintos equipos, tecnologías y evoluciones. Parte de su documentación está desactualizada, algunas decisiones solo están reflejadas en el código y determinadas reglas de negocio siguen dependiendo del conocimiento de personas concretas. La aplicación funciona, pero cada vez resulta más difícil y costoso modificarla.

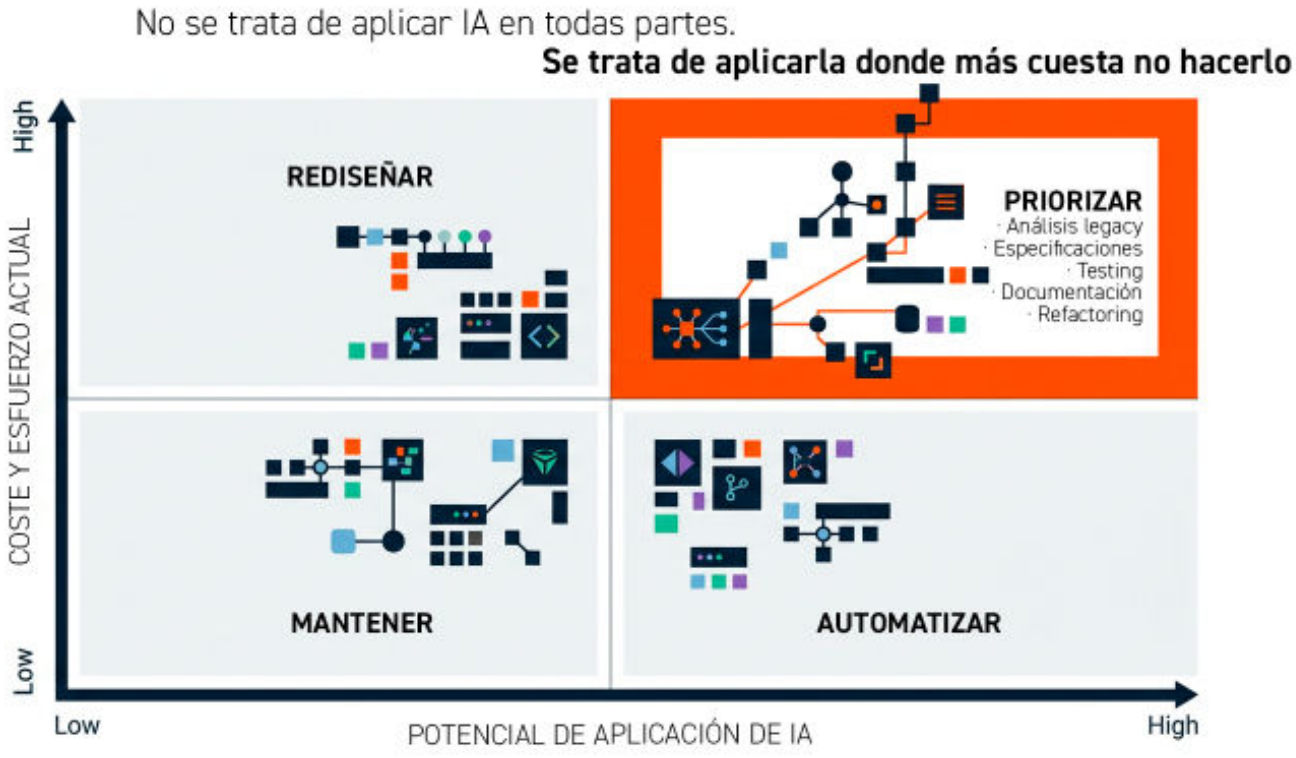
Ahora necesitas modernizarla. ¿Por dónde empiezas?

Antes de transformar una aplicación necesitas entenderla: conocer su arquitectura, identificar dependencias, reconstruir reglas de negocio, localizar componentes críticos y determinar qué debes conservar, qué puedes transformar y qué ya no tiene sentido mantener. En entornos legacy, esta fase puede convertirse en una barrera para la modernización.

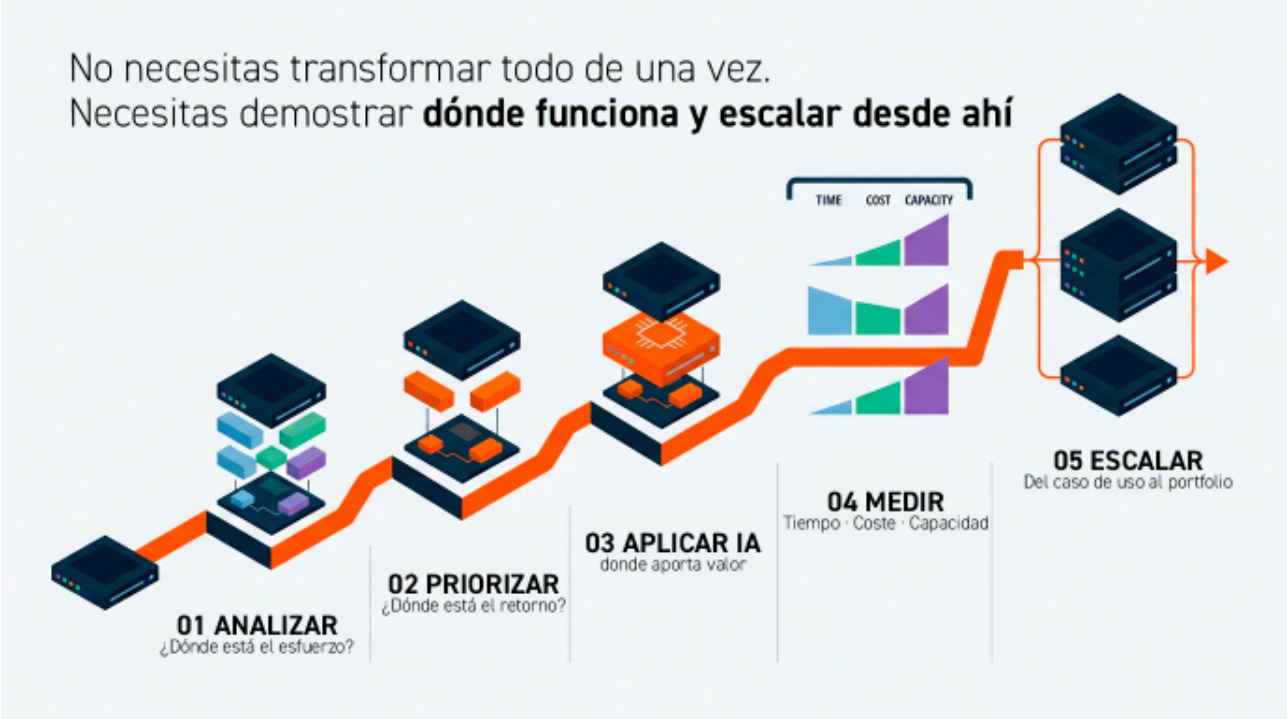
¿Y si pudieras acelerar también esa parte?

La IA puede ayudar a analizar código existente, explicar componentes, detectar relaciones, facilitar procesos de ingeniería inversa y generar o actualizar documentación. El objetivo no es sustituir el criterio de arquitectos y desarrolladores, sino proporcionarles una base de conocimiento que reduzca el tiempo que necesitan para comprender el sistema.

Esto introduce una variable económica especialmente relevante. Si necesitas menos esfuerzo para conocer tu punto de partida, también necesitas menos esfuerzo para empezar a transformarlo. La modernización con IA comienza, por tanto, antes de generar la primera línea de nuevo código. Comienza utilizando la tecnología para recuperar y estructurar el conocimiento que ya existe dentro de tus aplicaciones



¿Cuántodeberíacostartemañana evolucionar tu software?



El impacto dependerá de tus aplicaciones, arquitectura, deuda técnica, documentación, procesos y modelo de desarrollo. Por eso, el punto de partida debería ser tu coste actual. ¿Cuánto esfuerzo necesitas para entender una aplicación? ¿Cuánto para especificar un cambio? ¿Cuánto para desarrollarlo, probarlo y documentarlo? ¿Y cuánto para mantenerlo después?

A partir de ahí puedes identificar qué parte de ese esfuerzo puede reducirse con IA y construir un caso económico basado en tu realidad.

Del desarrollo con IA al desarrollo optimizado con IA

En Entelgy abordamos este proceso desde el conocimiento de software, la modernización de aplicaciones y la inteligencia artificial. El objetivo es identificar dónde existe una oportunidad real de eficiencia y trasladarla al ciclo de desarrollo: desde la comprensión de aplicaciones existentes y Spec IA hasta desarrollo, refactoring, testing, documentación y evolución.

No necesitas transformar todo tu portfolio de una vez. Puedes comenzar con una aplicación, un proceso o un proyecto en el que exista un problema suficientemente relevante, establecer el punto de partida, introducir IA donde pueda generar mayor impacto y medir el resultado.

Así, la decisión de escalar no depende de una promesa genérica sobre IA, sino del impacto que hayas podido comprobar en tiempo, coste y capacidad de evolución.

La pregunta final

La IA ya puede ayudarte a generar más código. Pero ¿es eso lo que realmente necesitas?

Quizá el objetivo no sea producir más. Quizá sea entender antes, especificar mejor, automatizar más y dedicar el conocimiento de tus equipos a aquello donde realmente aporta valor.

¿Estás preparado para desarrollar software de otra manera?

5 preguntas para detectar dónde la IA puede tener impacto real:

1 **¿Entender una aplicación antes de modificarla lleva demasiado tiempo?**

→ SÍ: Tienes una oportunidad en análisis y modernización con IA.

2 **¿Dependes de documentación incompleta o del conocimiento de personas concretas?**

→ SÍ: Tienes una oportunidad para recuperar y estructurar conocimiento con IA.

3 **¿Los requisitos cambian, son ambiguos o provocan retrabajo durante el desarrollo?**

→ SÍ: Tienes una oportunidad en Spec IA.

4 **¿Tus equipos siguen dedicando muchas horas a código repetitivo, pruebas, documentación o mantenimiento?**

→ SÍ: Tienes una oportunidad para automatizar parte del ciclo de desarrollo con IA.

5 **¿Tus equipos siguen dedicando muchas horas a código repetitivo, pruebas, documentación o mantenimiento?**

→ SÍ: Tienes una oportunidad para replantear la modernización con IA.

¿Cuántos "sí" has marcado?

0-1
Tu oportunidad puede estar en casos de uso muy concretos

2–3
La IA puede mejorar varias fases de tu ciclo de desarrollo.

4-5
No tienes un caso de uso de IA. Tienes una oportunidad de transformar tu modelo de desarrollo.

La clave no está en utilizar IA en más sitios, sino en identificar cuáles de estos problemas consumen hoy más tiempo y presupuesto. Ahí está tu punto de partida.

EN PROFUNDIDAD

Si has marcado la 1 o la 2, tu principal fricción aparece antes de desarrollar. La IA puede ayudarte a comprender aplicaciones existentes, recuperar conocimiento y reducir el esfuerzo necesario para abordar su evolución.

Si has marcado la 3, el problema está entre lo que necesitas y lo que finalmente desarrollas. Spec IA permite trabajar sobre requisitos, contexto y conocimiento para construir especificaciones más precisas antes y durante el desarrollo.

Si has marcado la 4, tienes una oportunidad dentro del propio ciclo. Desarrollo, testing, documentación, refactoring y mantenimiento contienen tareas donde la IA puede reducir trabajo manual y liberar capacidad de tus equipos.

Si has marcado la 5, el impacto puede ser más estratégico. La IA puede reducir parte del esfuerzo que hace que determinados proyectos de modernización sean demasiado largos, complejos o costosos.

Entelgy

Human driven technology



Construimos una realidad mejorada, impulsada por la armonía entre personas y tecnología

Argentina Brasil Chile Colombia España México Perú USA

entelgy.com

©Entelgy 2026

Documento asistido por IA