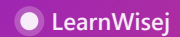


Ejercicio de programación con IA

Construye el ejemplo FieldOps de este módulo con ChatGPT o Claude a partir de un prompt listo para usar que dirige al modelo a la especificación del módulo, y después revisa, ejecuta y amplía lo que te devuelva.



No vas a teclear este ejemplo. Vas a dirigirlo.

Ya casi nadie escribe código Wisej.NET partiendo de un archivo vacío — un asistente escribe el primer borrador y el desarrollador decide si es correcto. Eso solo funciona si dominas el tema, y eso es lo que te dio el resto de este módulo. Aquí le pasas ese conocimiento a ChatGPT o a Claude, recibes el ejemplo de este módulo como una solución de Visual Studio que funciona y lo revisas como revisarías el pull request de un compañero.

Qué estás pidiendo

Completa el proyecto final de FieldOps: firma el instalador de Windows y los paquetes de Android e iOS, prepara una distribución empresarial para los shells móviles y un borrador de ficha para las tiendas, define una regla de compatibilidad entre las versiones del servidor y las del shell con una comprobación de versión mínima al iniciar, añade informes de fallos y un evento de uso para cada función del dispositivo, y entrega una lista de comprobación de publicación que cubra los cuatro modelos de distribución.

El módulo pide estos entregables, y el prompt de abajo le pide al asistente exactamente esos:

- ✓ Paquetes firmados para Windows, Android e iOS
- ✓ Configuración de la distribución empresarial y borrador de ficha para las tiendas
- ✓ Comprobación de compatibilidad de versiones entre servidor y shell
- ✓ Telemetría de fallos y de uso
- ✓ Lista de comprobación de publicación para todos los modelos de distribución

Por qué tiene esta forma

FieldOps termina este curso como un servidor y cuatro formas de llegar a él: una pestaña del navegador, una PWA instalada, un shell de escritorio para Windows y un shell móvil en Android e iOS. Los tres empaquetados son archivos que un sistema operativo se niega a instalar a menos que los haya firmado una identidad de su confianza, y cada uno viaja por un canal distinto con su propia cola de revisión y su propia velocidad de despliegue. Puedes volver a desplegar el servidor antes de comer; la compilación de iOS que tiene un técnico en la mano puede ir tres semanas por detrás, y aun así tiene que funcionar.

Estos son los términos que el asistente tiene que dominar. Si lo que te devuelve los usa a la ligera, no ha entendido el módulo:

Término	Qué significa aquí
firma Authenticode	La firma que aplica un certificado de firma de código a un instalador de Windows o a un MSIX, que demuestra que el archivo no se modificó después de la firma y nombra a un editor verificado.
marca de tiempo de la contrafirma	Una marca de tiempo de confianza registrada en el momento de la firma, para que el instalador siga validándose después de que caduque el propio certificado de firma.
keystore de Android y clave de subida	La clave que firma una compilación de Android; con Play App Signing firmas con una clave de subida y Google vuelve a firmar con la clave de firma de la aplicación que custodia para ese paquete.
certificado de distribución y perfil de aprovisionamiento de Apple	El certificado identifica a tu equipo; el perfil vincula un App ID, ese certificado, los entitlements y, fuera de la App Store, los dispositivos permitidos.
distribución empresarial	Entregar una compilación privada a los dispositivos de una organización (managed Google Play, apps personalizadas de Apple Business Manager, un MSIX distribuido por Intune) en lugar de publicarla públicamente.
versión mínima del shell	La compilación de shell más antigua, por plataforma, que el servidor actual sigue admitiendo; todo lo que esté por debajo se detiene al conectarse, antes de llegar a ningún dato.
evento de uso de una función del dispositivo	Un registro escrito cada vez que se ejecuta una capacidad del shell (cámara, firma, ubicación, notificación push, envío de datos sin conexión) con el resultado, la plataforma y la versión del shell.

Constrúyelo — elige tu asistente

Los dos botones abren una pestaña nueva con el **mismo prompt listo para usar** ya escrito. El prompt dirige al modelo a la especificación de este módulo — un único archivo Markdown con la lección, el enunciado del laboratorio y los entregables — para que trabaje con el material real en lugar de adivinar de qué trata el Módulo 6. Puedes abrir tú mismo [module-06.md](#) para ver exactamente lo que leerá.



Usar ChatGPT

Abre [chatgpt.com](#) con el prompt de este módulo ya cargado. Funciona con GPT-5 o Codex; cualquier plan que pueda navegar por la web descargará la especificación.



Usar Claude

Abre [claude.ai](#) con el mismo prompt ya cargado. Si usas Claude Code, puedes pegarlo en una terminal abierta en una carpeta vacía.



Envíalo tal cual y obtendrás un proyecto, no un fragmento

El prompt termina pidiendo toda la solución como un único `.zip` con los pasos para compilarla y ejecutarla. Descárgalo, abre la solución en Visual Studio, pulsa F5 y FieldOps se abrirá en `localhost`. Necesitas el SDK de .NET y el paquete NuGet `Wisej-4`; todo lo demás, el prompt le pide al asistente que lo escriba él mismo.

Estoy siguiendo el Módulo 6 – «Firma, distribución, actualizaciones y el proyecto final FieldOps» – del curso de LearnWisej «Empaquetado híbrido y nativo», y quiero construir contigo el ejemplo de este módulo. Respóndeme en español.

Empieza leyendo la especificación del módulo. Es un único archivo Markdown (en inglés) que contiene la lección completa, el enunciado del laboratorio y los entregables:

<https://learnwisej.com/assets/courses/hybrid-native-packaging/specs/module-06.md>

Después construye un proyecto Wisej.NET 4 completo que pueda abrir en Visual Studio:

1. Proyecto «FieldOps», C#, net8.0 o posterior, paquete NuGet Wisej-4 4.1.x, host Kestrel que llame a `app.UseWisej()`.
2. Construye lo que pide el laboratorio de la especificación: Completa el proyecto final de FieldOps: firma el instalador de Windows y los paquetes de Android e iOS, prepara una distribución empresarial para los shells móviles y un borrador de ficha para las tiendas, define una regla de compatibilidad entre las versiones del servidor y las del shell con una comprobación de versión mínima al iniciar, añade informes de fallos y un evento de uso para cada función del dispositivo, y entrega una lista de comprobación de publicación que cubra los cuatro modelos de distribución.
3. Genera estos entregables e indícame dónde está cada uno dentro de la solución:
 - Paquetes firmados para Windows, Android e iOS
 - Configuración de la distribución empresarial y borrador de ficha para las tiendas
 - Comprobación de compatibilidad de versiones entre servidor y shell
 - Telemetría de fallos y de uso
 - Lista de comprobación de publicación para todos los modelos de distribución
4. Sigue la especificación siempre que nombre una clase, propiedad o evento de Wisej.NET – usa exactamente ese. Donde deje la elección abierta, dime qué API elegiste y por qué.
5. Mantén la autoridad en el servidor: el estado y la validación viven en .NET, y el navegador muestra e informa. Nunca traslades una decisión a JavaScript porque allí resulte más fácil.
6. Haz que el resultado se pueda demostrar: debo poder ejecutarlo y ver el comportamiento del que trata este módulo, incluido lo que ocurre cuando algo sale mal.

Antes de escribir una sola línea de código, explica con tus propias palabras qué vas a construir y dónde vivirá cada pieza, para que pueda corregir el diseño mientras todavía es barato.

Cuando el código esté terminado, empaqueta toda la solución en un único archivo .zip descargable y dame los pasos exactos para compilarla y ejecutarla.

Por último, responde a estas preguntas de la especificación usando tu propia implementación como prueba:

- El shell de Android de un técnico tiene seis semanas y el servidor se ha publicado cuatro veces desde entonces. ¿Cuáles de esos cuatro cambios del servidor deberían haber subido la versión mínima del shell, y cómo lo habrías sabido en su momento?
- El instalador de Windows se firmó pero sin marca de tiempo, y el certificado caducó el mes pasado. ¿Qué hacen las instalaciones existentes, qué hacen las descargas nuevas y cuál es el camino más corto para volver a tener un instalador de confianza?
- El evento de uso de la cámara muestra un fuerte aumento de la alternativa del navegador solo en iOS. Nombra tres explicaciones compatibles con los datos y di qué campo adicional del evento permitiría distinguirlas.

Pregúntame por cualquier cosa ambigua antes de empezar.

Después revisa lo que te devuelve

Que el código compile no significa que sea correcto. Las preguntas interesantes son de diseño: ¿es este el control de Wisej.NET que la tarea realmente necesita?, ¿sigue el ciclo de vida que te ofrece el framework?, ¿el estado de referencia está en el servidor o se quedó en el navegador porque era más cómodo? Lee el resultado como revisor, no como cliente.

Este módulo te da la lista de comprobación. Responde a cada punto basándote en el código que acabas de recibir:

- ✓ El shell de Android de un técnico tiene seis semanas y el servidor se ha publicado cuatro veces desde entonces. ¿Cuáles de esos cuatro cambios del servidor deberían haber subido la versión mínima del shell, y cómo lo habrías sabido en su momento?
- ✓ El instalador de Windows se firmó pero sin marca de tiempo, y el certificado caducó el mes pasado. ¿Qué hacen las instalaciones existentes, qué hacen las descargas nuevas y cuál es el camino más corto para volver a tener un instalador de confianza?
- ✓ El evento de uso de la cámara muestra un fuerte aumento de la alternativa del navegador solo en iOS. Nombra tres explicaciones compatibles con los datos y di qué campo adicional del evento permitiría distinguirlas.

Todo lo que hayas tenido que corregir merece quedar por escrito. Una nota breve con las convenciones y decisiones que acordaste, pegada en el primer mensaje de tu próxima sesión, vale más que cualquier prompt ingenioso.



Todavía no hay un ejemplo de referencia para este curso — el tuyo es el primero

El [repositorio de ejemplos de LearnWisej](#) tiene soluciones completas para la mayoría de los cursos, pero todavía no para este. Eso significa que no hay nada con qué compararte ni ningún atajo: la especificación y tu propia revisión son toda tu red de seguridad. Guarda lo que construyas — y si sale bien, nos encantaría verlo.

Tres prompts para ir más allá

Dirige tu asistente al proyecto que acabas de construir y prueba estos. Cada uno es un ejercicio de revisión y ampliación, no una reescritura.

- ✓ **Cambia una pieza sin mover las juntas.** *"Elige la parte de este proyecto FieldOps que sea más claramente un sustituto de algo real — una fuente de datos falsa, un almacén en memoria, un servicio simplificado — y reemplázala por una implementación con forma de producción. El contrato a ambos lados no debe cambiar: los mismos tipos, las mismas llamadas, el mismo comportamiento ante fallos. Dime qué te obligó a gestionar la implementación real que el sustituto no gestionaba."*
- ✓ **Haz que aguante un mal día.** *"Revisa este proyecto FieldOps en busca de los modos de fallo que todavía no gestiona — una entrada incorrecta o ausente, una dependencia lenta o no disponible,*

trabajo que llega después de que el usuario haya salido de la pantalla y dos acciones que compiten entre sí. Para cada uno, dime primero si es un riesgo real en Wisej.NET 4 y por qué, y solo después escribe la corrección. Si decides que algo no necesita gestión, dilo y explica el razonamiento en lugar de añadir código defensivo."

- ✓ **Añade el segundo caso.** *"Amplía este proyecto FieldOps con un segundo caso que siga el mismo patrón que el primero — otra entidad, pantalla o control con la misma forma, conectado de la misma manera. Extrae a una clase base o a un servicio lo que de verdad sea común, pero solo donde se justifique; dos cosas parecidas no siempre piden una abstracción. Muéstrame el diff y justifica cada cosa que hayas extraído."*



Llévalo al laboratorio

El laboratorio práctico que cierra este módulo pide los mismos entregables que acabas de hacer que generara el asistente. Usa lo que tienes aquí como forma de tu respuesta y como prueba que la respalda — no como la respuesta en sí. El laboratorio se califica por tu razonamiento, y esa parte tiene que ser tuya.