



IA para Project Managers - Tema 5

"De la EDT al diagrama de Gantt"

Dr. Marc Bara

Resumen rápido

Tema 1

La IA conoce la teoría de la gestión de proyectos. Vuestro trabajo consiste en aportar rigor profesional y exigir más que una respuesta de libro de texto.

Tema 2

La forma en que estructuráis y proporcionáis información a la IA cambia fundamentalmente lo que puede hacer por vosotros.

Temas 3 y 4

La IA no reemplaza a vuestro equipo. El contexto local, las normativas y la experiencia de campo son insustituibles. Tampoco gestiona stakeholders.

Ahora: Tema 5

Vamos a intentar **hacer que la IA falle**.

El experimento

Hasta ahora, la IA ha generado texto para nosotros: actas, informes de estado, documentos WBS. Y hace un trabajo razonable. El texto es su zona de confort.

Ahora pediremos algo diferente: generar un archivo técnico en un formato muy específico (Microsoft Project XML), con una lógica interna compleja (dependencias de actividades, ruta crítica, solapamientos realistas) y hacer que el resultado se importe directamente en un software de programación de proyectos.

Esto es un salto. La IA ya no puede salir del paso con una prosa que suena bien. El resultado o se importa en ProjectLibre y funciona, o no lo hace. No hay término medio.

Probablemente veremos algunos modelos tener éxito y otros fallar. Eso es parte del experimento.



Qué estamos pidiendo

Le facilitamos la EDT de Can Rigalt (la versión con el contexto regulatorio de Barcelona de Temas 3 y 4) y le pedimos que genere un cronograma completo:

Secuenciación

Decidir qué actividades se ejecutan en paralelo y cuáles en serie. Ponerlo todo en una sola línea no es aceptable.

Duraciones realistas

Estimaciones basadas en la productividad real de la construcción en Barcelona, no cifras inventadas.

Dependencias lógicas

Toda relación entre actividades debe tener una justificación técnica, normativa o logística.

Formato estricto

Microsoft Project XML con campos específicos, nombres de etiqueta exactos y estructura precisa. Un solo campo mal nombrado y el archivo no se importará.

Ruta crítica

La cadena más larga de actividades con holgura cero debe ser identificable.

Por qué esto es difícil para la IA

El problema de la lógica

La planificación es la primera tarea de este curso en la que la IA no puede esconderse tras una prosa bien redactada.

Un cronograma tiene una lógica interna verificable: si la actividad A precede a la B, y la B precede a la C, las fechas deben ser coherentes. Si indicáis que una actividad dura 5 días y comienza un lunes, debe terminar un viernes. Si dos actividades se ejecutan en paralelo, sus fechas deben solaparse.

El problema del formato

Además de eso, el formato XML de MS Project es inflexible. Dentro de una tarea debéis escribir `<Start>`, no `<StartDate>`. Las duraciones se expresan como `PT40H0M0S`, no como "5 días". Si las asignaciones de recursos están anidadas dentro de las tareas en lugar de en su propia sección, el software las ignorará.

- ❏ La IA tiene que acertar tanto con la lógica COMO con el formato. Veamos si es capaz.

 15 MINUTOS

La tarea

Vuestro entregable

Un cronograma completo en formato XML de Microsoft Project, listo para importar en ProjectLibre. Con duraciones estimadas, dependencias entre actividades, hitos de control y ruta crítica.

Instrucciones

Disponéis de un **megaprompt profesional diseñado para la generación de cronogramas**. Introducid en él la EDT de Can Rigalt y generad el archivo XML.

Cada grupo importará su resultado en ProjectLibre o Microsoft Project. Veremos lo que sucede en directo.

El momento de la verdad

Vamos a importar los XML en ProjectLibre.

Preguntas para cada grupo:

- ¿Se ha importado sin errores? ¿O el software lo ha rechazado?
- ¿Hay flechas de dependencia en el diagrama de Gantt o cada barra flota por sí sola?
- ¿Hay actividades huérfanas (sin predecesora ni sucesora)?
- ¿Las duraciones se muestran correctamente o son todas cero?
- ¿Sois capaces de identificar la ruta crítica?
- ¿El cronograma tiene solapamientos realistas o es una secuencia estrictamente lineal?

Lo que suele fallar

Formato XML incorrecto

La IA inventa etiquetas o utiliza nombres de campo erróneos. `<StartDate>` en lugar de `<Start>` dentro de las tareas. Duraciones como "5d" en lugar de PT40H0M0S. Asignaciones anidadas dentro de las tareas en lugar de en su propia sección. Resultado: el software importa un archivo vacío o muestra todas las duraciones como cero.

Programación lineal sin solapamientos

Todas las actividades en estricta secuencia, una tras otra, sin paralelismo. En la construcción real existen actividades simultáneas. Una programación totalmente lineal es ficticia.

Actividades sin sucesor

Si una actividad no alimenta a ninguna otra, su resultado se pierde en la red de programación. Toda actividad, excepto el hito final, debe tener al menos un sucesor.

Dependencias sin lógica de construcción

La IA conecta las actividades porque aparecen una tras otra en la lista, no porque exista una restricción técnica real entre ellas.

Lo que esto nos indica

Si su archivo XML se importó correctamente

Enhorabuena. Pero verifique la lógica. Una importación limpia no significa que el resultado tenga sentido.

Si no se importó, o se importó con duraciones nulas

Acaba de comprobar el límite actual de la IA. Genera texto con una estructura correcta, pero los formatos técnicos estrictos hacen que falle.

En ambos casos, la lección es la misma: **la IA genera, pero vosotros validáis**. Un cronograma «casi correcto» es un cronograma que no podéis utilizar. Y en un proyecto real, un archivo que no se importa en la herramienta del equipo no sirve de nada, independientemente de lo buena que sea la lógica que hay detrás.

Conclusiones clave

- **Tema 1**

Vuestro criterio profesional importa más que el conocimiento teórico de la IA.

- **Tema 2**

La forma en que suministráis información a la IA determina la calidad del resultado.

- **Temas 3 y 4**

La IA no reemplaza el conocimiento local de vuestro equipo ni gestiona stakeholders

- **Tema 5**

La IA tiene límites técnicos reales. Cuando le pedís algo más allá de la generación de texto (un archivo con un formato estricto, lógica interna verificable y compatibilidad con software externo) descubriréis dónde está el muro.