

Ingeniería de Software: de asignatura imposible a desarrollo práctico mediante LLMs

Pablo Gervás

Universidad Complutense de Madrid

VIII Jornada de Innovación Docente
Facultad de Informática, Madrid,
9 de junio de 2025

Experiencias con IA Generativa en la enseñanza

ChatGPT como programador en equipo de desarrollo, UCM

Ingeniería de Software 1, 2023-2024

Desarrollo de software (IES Abroad) 2023-2024

Ingeniería de Software 1 2024-2025

Desarrollo de software (IES Abroad) 2024-2025

Contenido

Dos problemas:

La llegada de la Inteligencia Artificial

Desafíos de enseñar Ingeniería del Software

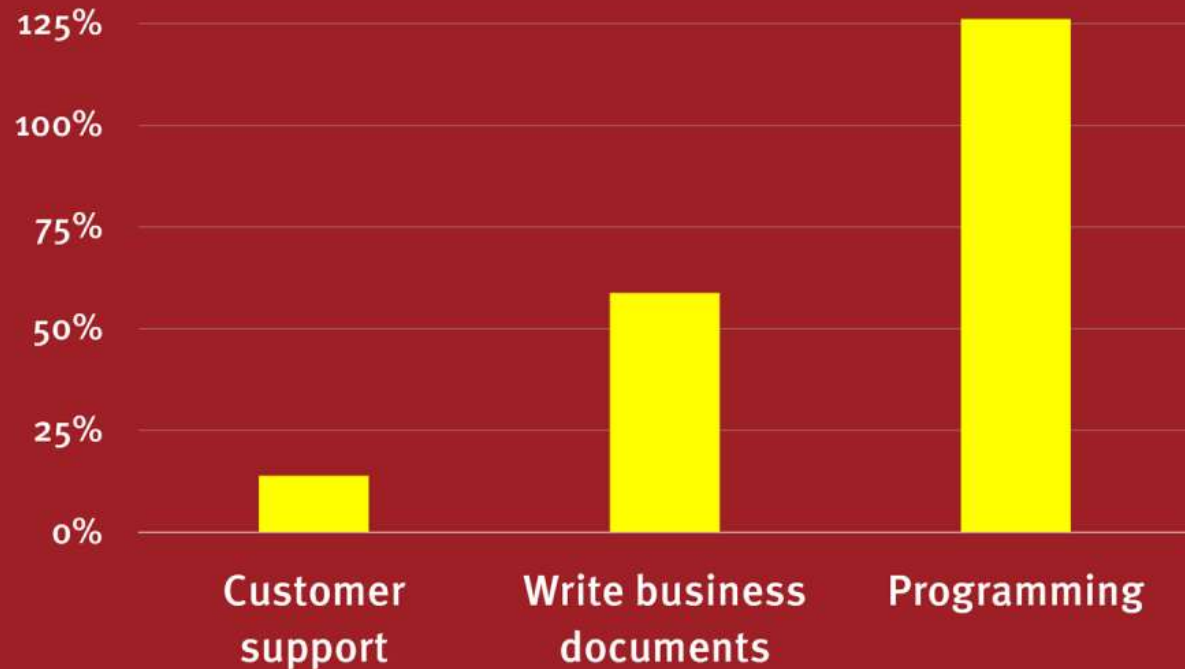
Una posible solución: ingenieros de software para una IA que programa

Conclusiones

AI Improves Employee Productivity by 66%

Productivity Increases with AI

Increase in users' task performance when using **generative AI** tools



nngroup.com

NN/g



MORE SCHOOLS BAN A.I. PROGRAM CHATGPT

VARNEY & CO.

BITCOIN 17,366.00 ▼ 72.00 -0.41% | **ETHEREUM** 1,326.70 ▼ 11.00 -0.82%

LAST TRADES ► **FARGO (WFC)** 42.58 ▲ 0.22 | **EXXONMOBIL (XOM)** 110

DOW
64
+0.24%



Contenido

Dos problemas:

La llegada de la Inteligencia Artificial

Desafíos de enseñar Ingeniería del Software

Una posible solución: ingenieros de software para una IA que programa

Conclusiones

Qué incluye la ingeniería de software

1. Modelos de desarrollo de software
2. Requisitos
3. Gestión de riesgos
4. Gestión de la configuración
5. Planificación
6. Modelado de software (UML)
7. Diseño de software - Patrones y arquitecturas
8. Calidad del software
9. Pruebas de software
10. Mantenimiento del software



Requisitos
Gestión de riesgos
Diseño de software
Planificación



Modelado de Software
Gestión de Configuración

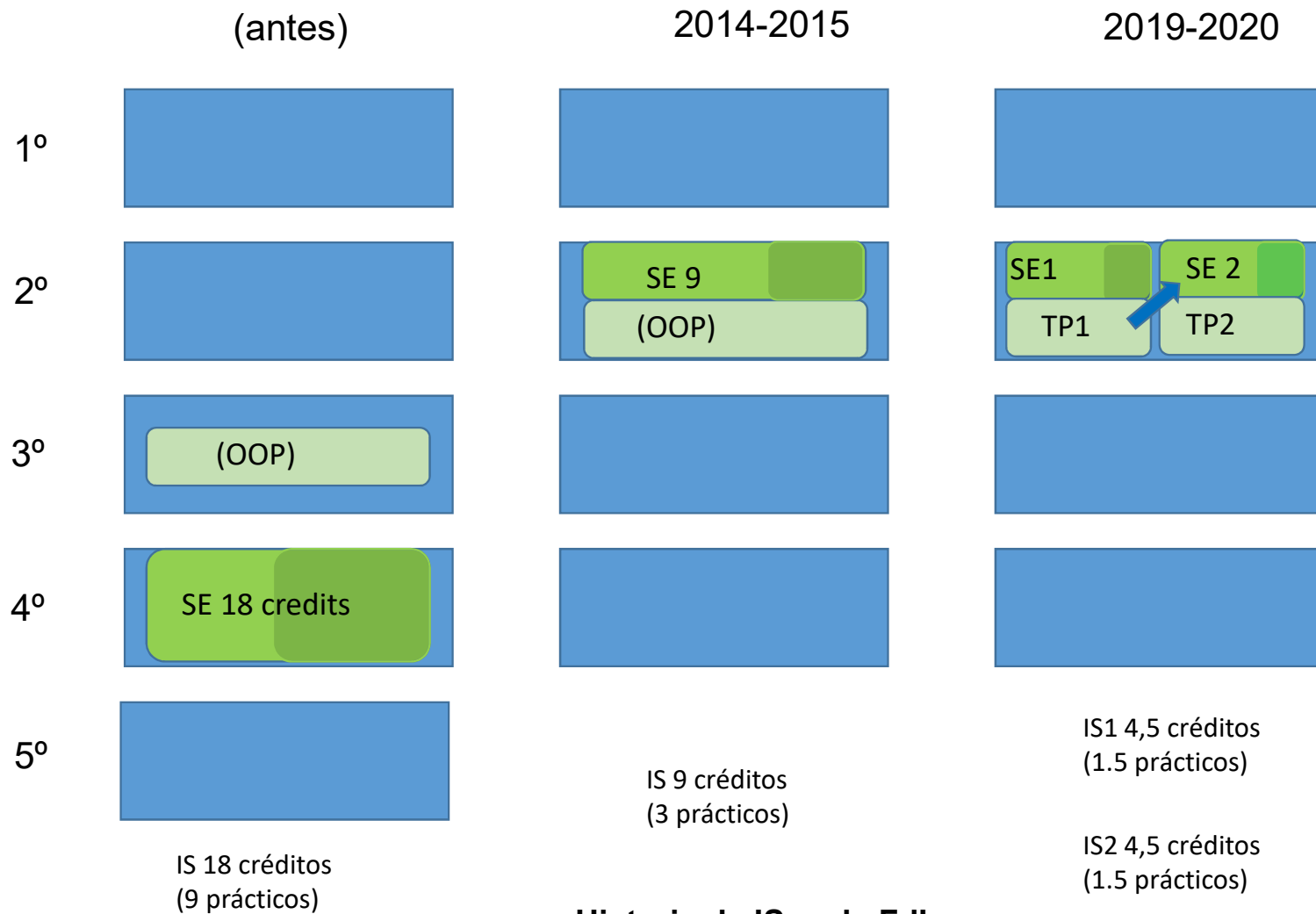
Modelos de Desarrollo de Software



Calidad de Software
Prueba de Software
Mantenimiento de Software



[Programación]



Historia de IS en la Fdi

Cómo está distribuido entre IS1 e IS2

Ingeniería del Software I

1. Introducción a la Ingeniería del Software.
2. Lenguajes de Modelado de Software.*
3. Proceso de desarrollo de software: Modelado de flujos de trabajo.
4. Planificación y gestión de proyectos.
5. Análisis y especificación de requisitos: Modelado de requisitos de software..

Ingeniería del Software II

1. Diseño de software: Modelado estructural y de comportamiento.
2. Implementación y validación.
3. Mantenimiento del software.
4. Ingeniería del Software en la Práctica.

* (Aplicado en ambas asignaturas)

Objetivo: aprender a desarrollar software en equipo y en proyectos grandes

La asignatura trata realmente sobre cómo gestionar equipos de programadores PERO

la única manera de comprobar si se hace bien es desarrollar código

que requiere que los estudiantes programen



Requisitos
Gestión de riesgos
Diseño de software
Planificación



Modelado de Software
Gestión de Configuración

Modelos de Desarrollo de Software



Calidad de Software
Prueba de Software
Mantenimiento de Software



[Programación]

Desafíos

El nivel de conocimientos de programación de los estudiantes es muy escaso a estas alturas de la carrera

El tiempo disponible para esta asignatura no permite desarrollar proyectos grandes

El trabajo en equipos grandes dificulta la evaluación objetiva del trabajo individual de cada estudiante

Contenido

Dos problemas:

La llegada de la Inteligencia Artificial

Desafíos de enseñar Ingeniería del Software

Una posible solución: ingenieros de software para una IA que programa

Conclusiones

Requisitos

Gestión de riesgos

Diseño

Planificación

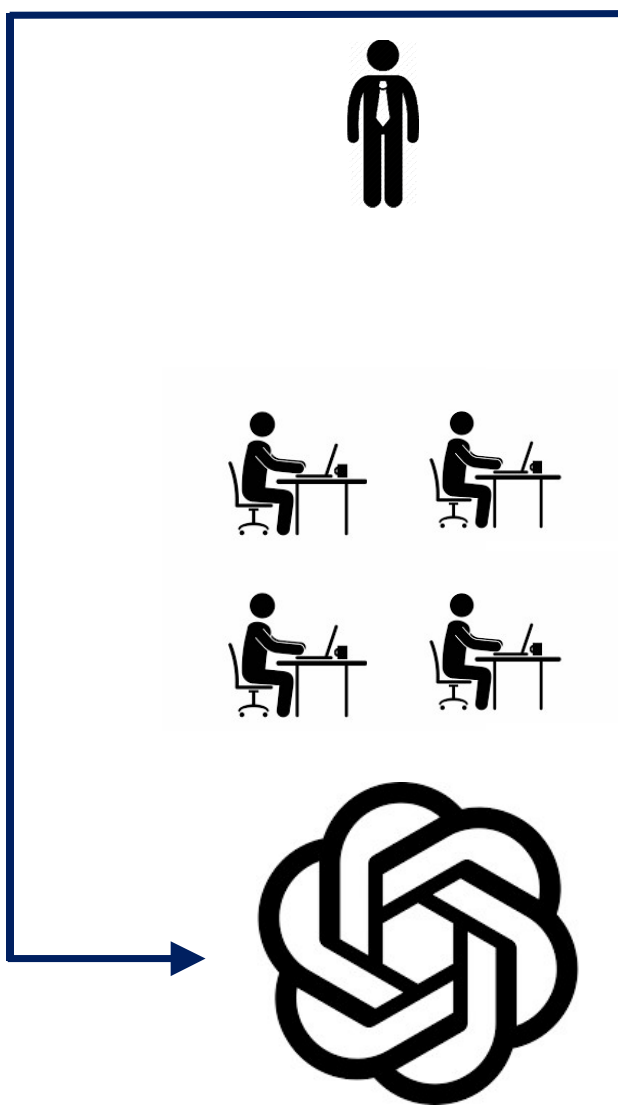
Modelos de
Desarrollo de
Software

Pruebas de Software

Mantenimiento de Software

[Programación]





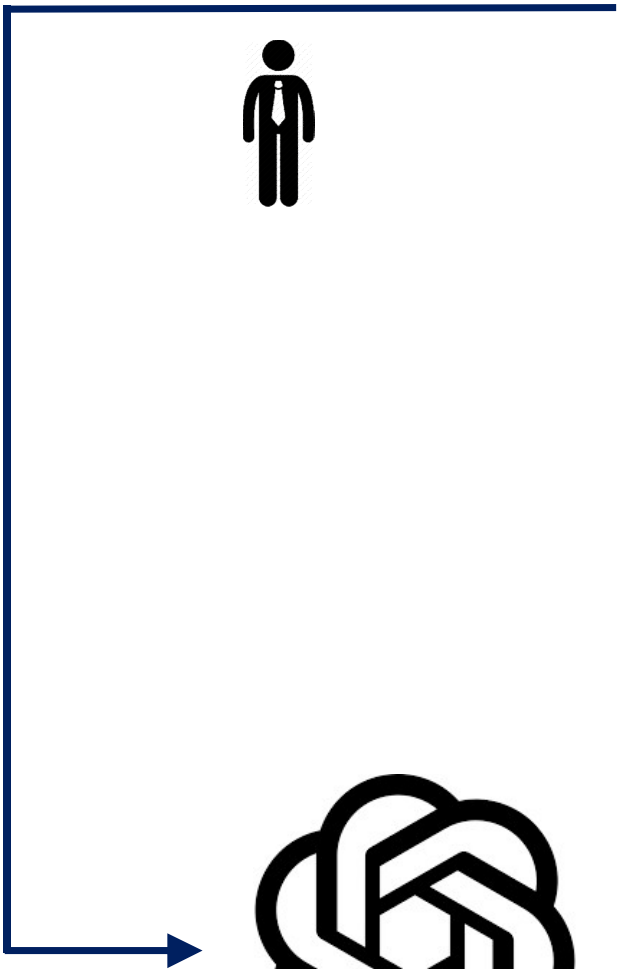
Requisitos
Gestión de riesgos
Diseño de software
Planificación

Modelado de Software
Gestión de Configuración

Modelos de Desarrollo de Software

Calidad de Software
Prueba de Software
Mantenimiento de Software

[Programación]

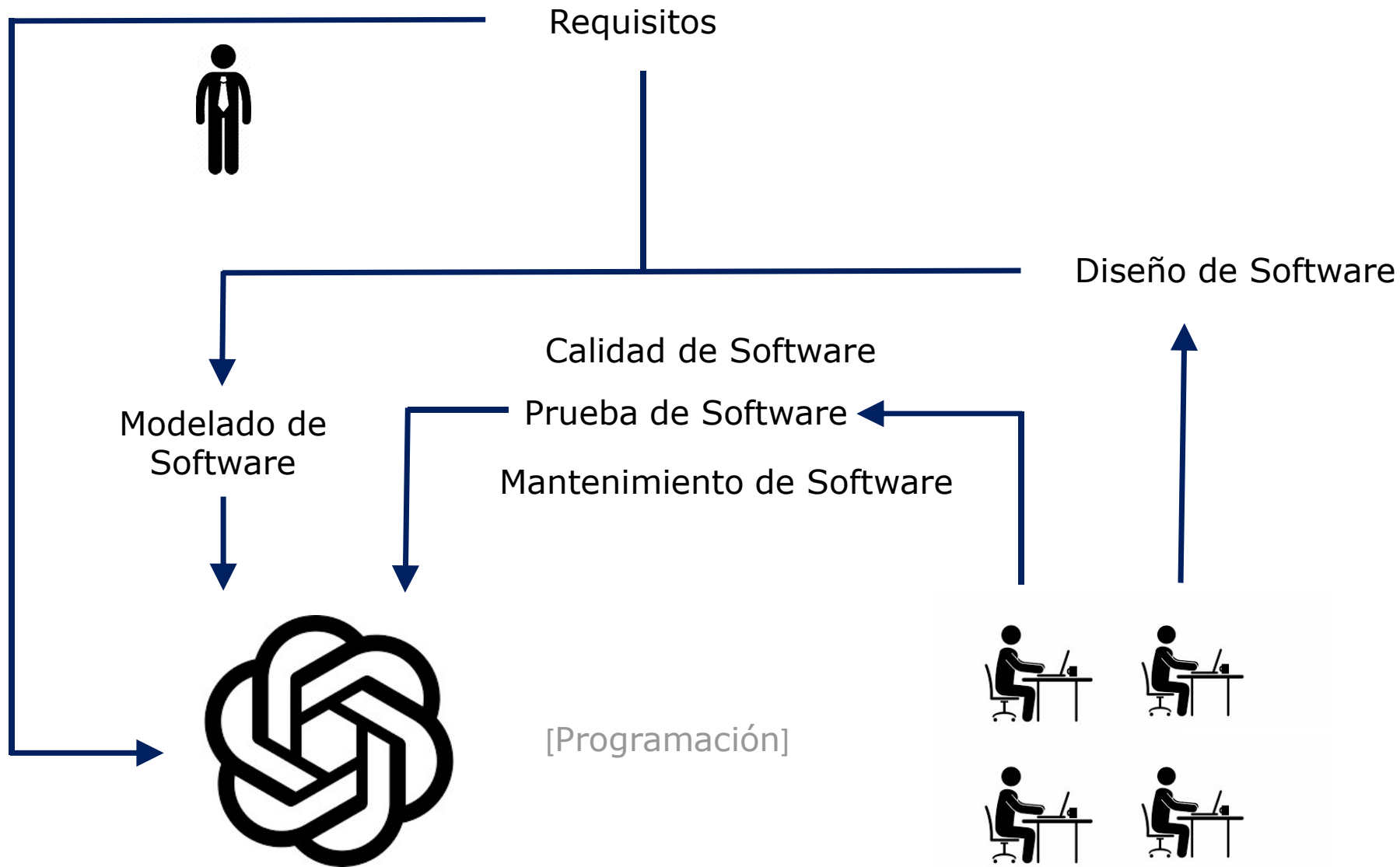


Requisitos



[Programación]

??



Contenido

Dos problemas:

La llegada de la Inteligencia Artificial

Desafíos de enseñar Ingeniería del Software

Una posible solución: ingenieros de software para una IA que programa

Conclusiones

Reflexiones tras las primeras experiencias

Potencial para hacer proyectos de complejidad antes imposible

Los estudiantes se sienten capaces de afrontar cualquier reto: tamaño del proyecto, nueva tecnología...

Necesidad de imponer límites artificiales

Manejar bien la IA sin saber programar no es fácil

Muy difícil integrar cada nueva contribución en el proyecto en curso

Todos los temas de la asignatura cobran una importancia real para conseguir buenos resultados que no tenían en versiones anteriores

(!)

Solve each equation.

$$b + 7 = 12$$

$$5. \quad 20 = 24 - n$$

$$9. \quad j \div 3 = 2$$

$$13. \quad c \div 10 = 8$$

$$17. \quad \frac{f}{2} = 12$$

$$21. \quad 66 = 22m$$

$$25. \quad b = \frac{72}{6}$$

$$29. \quad 4w = 52$$

$$33. \quad 48 \div n = 8$$

$$2. \quad a + 3 = 1$$

$$6. \quad 4x = 36$$

$$10. \quad 14 = n$$

$$14. \quad 6 =$$

$$18. \quad 1$$

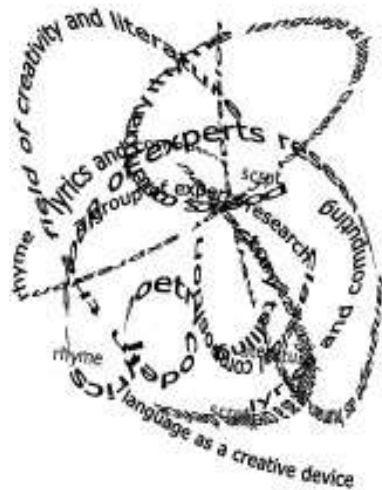
$$27$$





¡Muchas gracias!

<http://nil.fdi.ucm.es>



natural interaction based on language
nil