

The background features a dark blue gradient with a starry pattern. On the left side, there are several circular elements: a large scale with numbers from 140 to 260, and several smaller circles with arrows indicating clockwise or counter-clockwise rotation. The scale has tick marks and numbers at intervals of 10, with some numbers like 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, and 260 visible. The circles are composed of solid and dashed lines, some with arrows pointing in different directions.

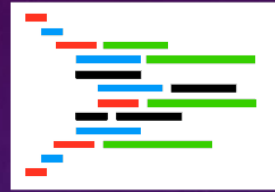
# CÓDIGO LIMPIO

LUCAS CASTRO ANTUÑA - UO289769

JESÚS GARCÍA RUIZ - UO285427

ABEL MENÉNDEZ HERNÁNDEZ – UO283274

# ¿QUÉ ES EL CÓDIGO LIMPIO?



Organización y  
legibilidad



Mantenimiento y  
escalabilidad



Facilita la detección de  
errores

# IMPORTANCIA DEL RENDIMIENTO EN SOFTWARE



Alta demanda



Recursos limitados



Satisfacción del cliente

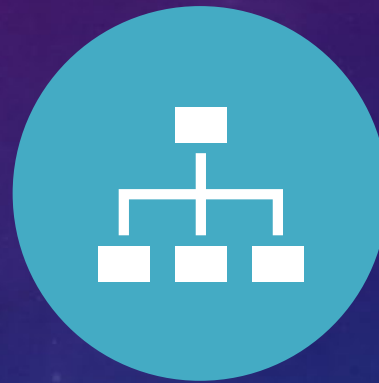


Competitividad en el mercado

# CÓDIGO LIMPIO VS RENDIMIENTO EFICIENTE



OCULTAR DETALLES DE  
IMPLEMENTACIÓN



NO USAR IFs, NO USAR SWITCHES,  
USAR JERARQUÍAS DE CLASES



LEGIBILIDAD Y COGNICIÓN

# CÓDIGO LIMPIO Y BUEN CÓDIGO



CLARIDAD Y PRECISIÓN



EFFECTOS COLATERALES



ABIERTO A  
OPTIMIZACIONES

# CÓDIGO LIMPIO Y BUEN CÓDIGO



Ley de Conway y Microservicios

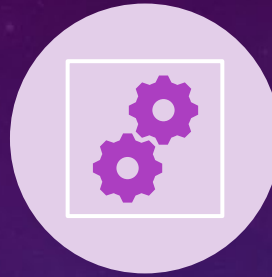


Lenguajes de programación

# ESTRATEGIAS PARA OPTIMIZAR EL RENDIMIENTO DEL SOFTWARE

The background is a dark blue gradient with a starry or particle effect. It features several faint, light blue technical graphics: a large circular gauge with numerical markings (100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 210) and an arrow pointing upwards, located in the top right; a smaller circular gauge with an arrow pointing downwards, located in the bottom right; and a circular gauge with an arrow pointing to the left, located in the bottom left.

# TÉCNICAS DE OPTIMIZACIÓN



**PERFILADO Y  
OPTIMIZACIÓN SELECTIVA**



**ALGORITMOS Y  
ESTRUCTURAS DE DATOS  
EFICIENTES**



**CONTEXTO Y REQUISITOS  
DEL PROYECTO**



**PRUEBAS EXHAUSTIVAS Y  
VALIDACIÓN DEL  
RENDIMIENTO**



# PERFILADO Y OPTIMIZACIÓN SELECTIVA

1. Información sobre tiempo de ejecución, frecuencia de llamadas, consumo de memoria, etc
2. Identificar puntos críticos de rendimiento
3. Optimizar estas zonas del código sin tocar el resto

# ALGORITMOS Y ESTRUCTURAS DE DATOS EFICIENTES



Los desarrolladores deben conocer varios algoritmos y estructuras de datos



Utilizarlos según el contexto



El uso de estructuras de datos según el tamaño y cantidad de los datos importa



Los algoritmos más eficientes reducen costes de tiempo y consumo de recursos

# CONTEXTO Y REQUISITOS DEL PROYECTO

---

Establecer el contexto de la aplicación

---

Aclarar los requisitos

---

Definir los objetivos

---

Priorizar aspectos como el rendimiento,  
o la mantenibilidad y legibilidad.

# PRUEBAS EXHAUSTIVAS Y VALIDACIÓN DEL RENDIMIENTO

- Validar las mejoras introducidas
- Las pruebas deben abarcar la funcionalidad, la estabilidad y el rendimiento
- Comprobar que no se producen efectos secundarios no deseados
- Garantizar que el rendimiento cumple con las expectativas



“

*Encontrar el equilibrio entre código limpio y eficiencia es fundamental para garantizar una experiencia satisfactoria y, a la vez, mantener la competitividad en el mercado*

”

The background is a dark blue gradient with a pattern of question marks inside speech bubbles. The speech bubbles are in various colors including purple, red, and blue. There are also faint technical graphics like a circular scale with numbers (140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260) and dashed lines.

PREGUNTAS