



# ***El arte de los commits***

**Guía completa para escribir mensajes  
útiles y claros.**



**PABLO DEL ÁLAMO**

# Introducción



Los commits son como los ladrillos de tu código.

Vamos a ver cómo escribir mensajes útiles y claros para que tu código sea más comprensible.

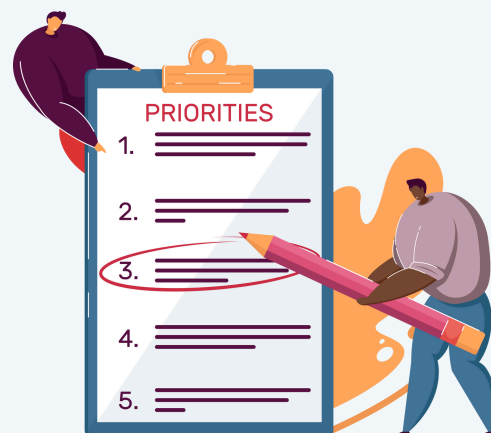


PABLO DEL ÁLAMO



# ¿Por qué son importantes los buenos mensajes de commit?

Los mensajes de commit claros facilitan la colaboración, ayudan a rastrear cambios, y son esenciales para entender el historial del proyecto. ¡No subestimes su poder!



PABLO DEL ÁLAMO



# Errores comunes en los mensajes de commit

Muchos cometen errores como ser demasiado generales ("cambios"), o demasiado extensos. Vamos a explorar cómo evitar estos errores.



PABLO DEL ÁLAMO



# La importancia de ser breve pero preciso

Imagínalo como un tweet: debe ser conciso, pero cargado de valor. Debe explicar qué cambio has hecho, no solo que has hecho "algo".



PABLO DEL ÁLAMO



# Usa un estilo consistente

Mantén un formato consistente en tu equipo.  
Ejemplo: [Tipo de cambio]: [Descripción breve].



PABLO DEL ÁLAMO



# Usa verbos en imperativo

Escribe como si dieras una orden, por ejemplo:  
"Añadir función de búsqueda" en lugar de "Añadida función de búsqueda".



PABLO DEL ÁLAMO



# Contexto es clave

Tu mensaje debe responder al menos a estas preguntas: ¿Qué has cambiado? ¿Por qué lo has cambiado?



PABLO DEL ÁLAMO





# Ejemplo de un buen mensaje de commit

Mejora de rendimiento: Refactorizar la función de búsqueda para reducir el tiempo de respuesta.



PABLO DEL ÁLAMO



# La anatomía del mensaje perfecto

1. Breve título descriptivo
2. Cuerpo opcional (detalles técnicos, si es necesario)
3. Referencia a tickets o problemas relacionados



PABLO DEL ÁLAMO



# Evita hacer commits monstruosos

Intenta hacer commits pequeños y frecuentes.

Cambios gigantescos son difíciles de revisar y comprender.



PABLO DEL ÁLAMO



# Beneficios de commits pequeños

Permiten un feedback más rápido y son mejores para pruebas y deshacer cambios si es necesario.



PABLO DEL ÁLAMO



# Ejemplo de commit detallado

Título: Corregir error de inicio de sesión

Cuerpo: Se ha resuelto el problema de inicio debido a la función incorrecta X.

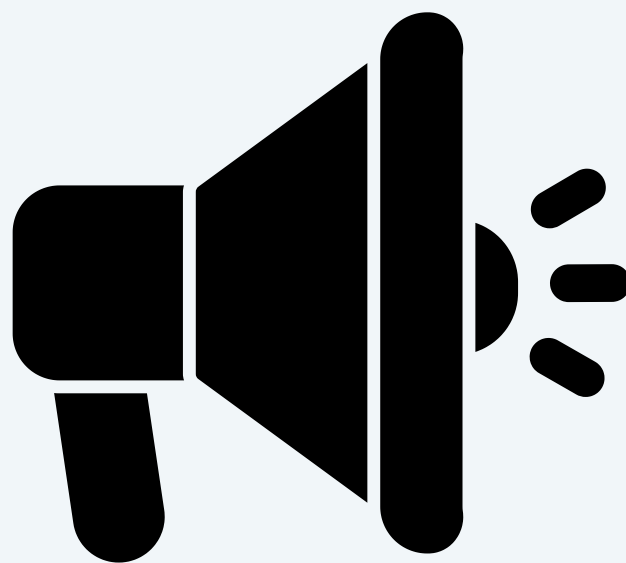


PABLO DEL ÁLAMO



# Cómo manejar errores anteriores en commits

Si corriges un error, menciona el ID del commit previo o la cuestión que resuelve para un seguimiento más fácil.

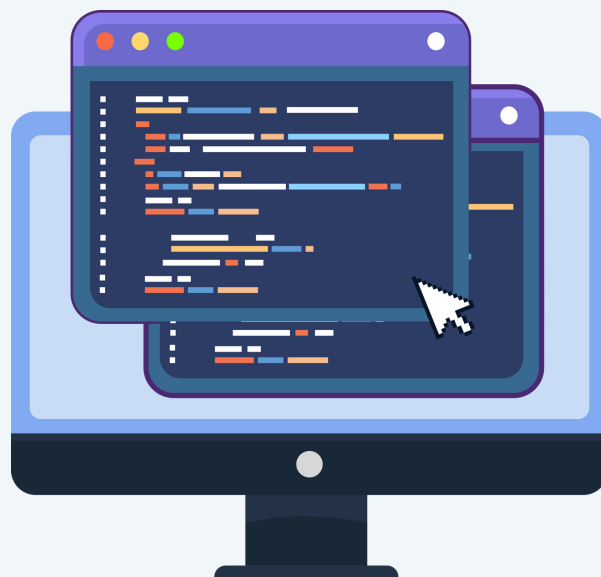


PABLO DEL ÁLAMO



# Herramientas de ayuda para escribir mejores commits

Plugins de IDE y herramientas de línea de comandos pueden ayudar a estructurar mejor los mensajes.



PABLO DEL ÁLAMO



# La regla de los 50 caracteres

El título debe ser breve, idealmente menos de 50 caracteres.

Deja los detalles para el cuerpo del mensaje.



PABLO DEL ÁLAMO





# Léelo de nuevo

Revisa tu mensaje antes de hacer commit, asegúrate de que quede claro incluso para alguien que no esté inmerso en el proyecto.

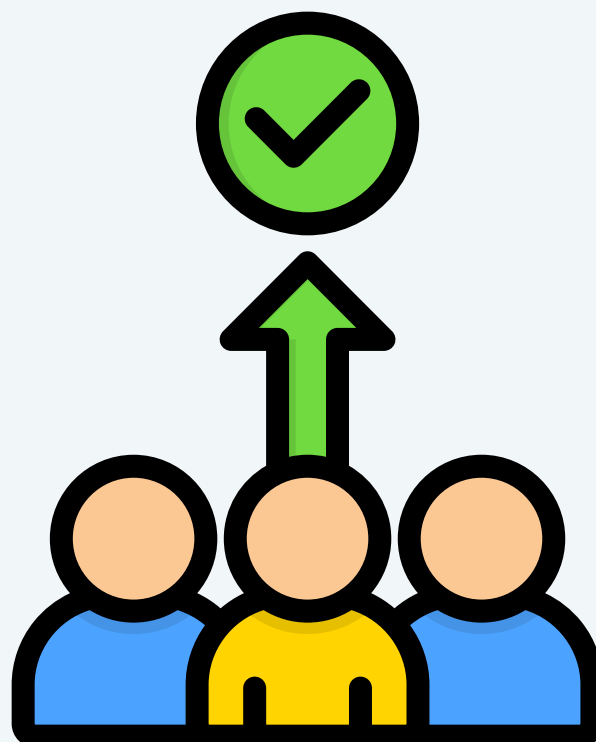


PABLO DEL ÁLAMO



# Comunicación dentro del equipo

Asegúrate de que todos en tu equipo estén alineados con los mismos estándares para escribir mensajes de commit.



PABLO DEL ÁLAMO

# Conclusión



Conseguir que tus mensajes de commit sean claros y útiles es una habilidad esencial en el desarrollo de software colaborativo.

Recuerda ser conciso/a pero detallado/a, usa verbos en imperativo y proporciona contexto.

Evita los commits gigantescos y siempre revisa tu mensaje antes de hacer el commit.

La consistencia dentro del equipo es clave para una mejor colaboración.

Al practicar estos consejos, mejorarás la legibilidad y el seguimiento de tu historial del proyecto, facilitando el trabajo futuro.



PABLO DEL ÁLAMO



# ¿Te ha resultado útil?



- Comparte esta guía con tu equipo o amigos desarrolladores.
- Guárdala para tenerla siempre a mano.
- ¡Dale un like o comenta si tienes preguntas!

