

# PLANTILLA DE PROMPTS



- Prompts para consultar código en IA's

## Opción 1: Plantilla General para Consultas de Código

**Rol:** Eres un desarrollador full stack con experiencia en principios SOLID y Clean Code.

**Tarea:** Escribe una función en [lenguaje o framework, por ejemplo, TypeScript, React] que resuelva el siguiente problema:

[Descripción del problema: por ejemplo, Crear un endpoint REST API para gestionar horarios de empleados, asegurando validación de entradas y siguiendo prácticas de Clean Code].

**Contexto:**

- Esta función debe seguir los principios SOLID, particularmente [principio específico, por ejemplo, Responsabilidad Única].
- Incluye comentarios relevantes para mayor claridad.
- Usa buenas prácticas para un código mantenible y escalable.

**Intención:** Proveer un código fácil de entender, extender y adaptar.

## Opción 2: Prompt Específico para el Front-End

**Rol:** Eres un desarrollador de React especializado en crear interfaces amigables y responsivas. Con experiencia en principios SOLID y Clean Code

**Tarea:** Construye un componente de React para [tarea específica, por ejemplo, mostrar y filtrar una lista de horarios de empleados].

**Contexto:**

- Usa componentes funcionales y hooks.
- Asegúrate de que el componente sea reutilizable y respete la separación de responsabilidades.
- Incluye estilos escritos de manera modular y escalable, usando Tailwind css

- Incluye comentarios relevantes para mayor claridad.

**Intención:** Crear un componente limpio, eficiente y reutilizable siguiendo las prácticas de Clean Code.

### Opción 3: Prompt Específico para el Back-End

**Rol:** Eres un desarrollador back-end con experiencia en Node.js y Express.

**Tarea:** Implementa una ruta de Express para [funcionalidad específica, por ejemplo, gestionar solicitudes de vacaciones de empleados].

**Contexto:**

- Asegúrate de que la ruta siga el diseño RESTful.
- Valida las entradas usando una librería como Joi o express-validator.
- Usa principios SOLID para separar responsabilidades (por ejemplo, capas de controlador, servicio y modelo).
- Asegura un manejo adecuado de errores y devuelve códigos de estado HTTP significativos.

**Intención:** Crear un manejador de rutas robusto, modular y mantenible.

### Opción 4: Consulta para Flujo Completo (Full-Stack)

**Rol:** Eres un desarrollador full stack con experiencia en construir sistemas integrados de front-end y back-end.

**Tarea:** Crea una funcionalidad completa para [funcionalidad específica, por ejemplo, permitir que los empleados vean y soliciten días de vacaciones].

**Contexto:**

- **Front-End:** Usa React para crear un formulario y un componente de visualización.
- **Back-End:** Usa Node.js, Express y MySQL para implementar el endpoint de la API y la lógica de la base de datos.
- Sigue los principios SOLID y asegúrate de que el front-end y el back-end estén desacoplados.
- Usa prácticas de Clean Code para nombres de variables y modularidad.

**Intención:** Entregar una funcionalidad cohesiva que demuestre las mejores prácticas en desarrollo full stack.

- Plantilla para crear commits

**Prompt:**

*"He realizado los siguientes cambios: [descripción breve de los cambios]. ¿Me puedes dar un mensaje de commit en inglés siguiendo las convenciones de prefijo y scope?"*

**Ejemplo de uso:**

*"He realizado los siguientes cambios: he añadido validación de formularios en la página de registro de empleados para asegurarnos de que todos los campos obligatorios están completos antes de enviar. ¿Me puedes dar un mensaje de commit en inglés siguiendo las convenciones de prefijo y scope?"*