

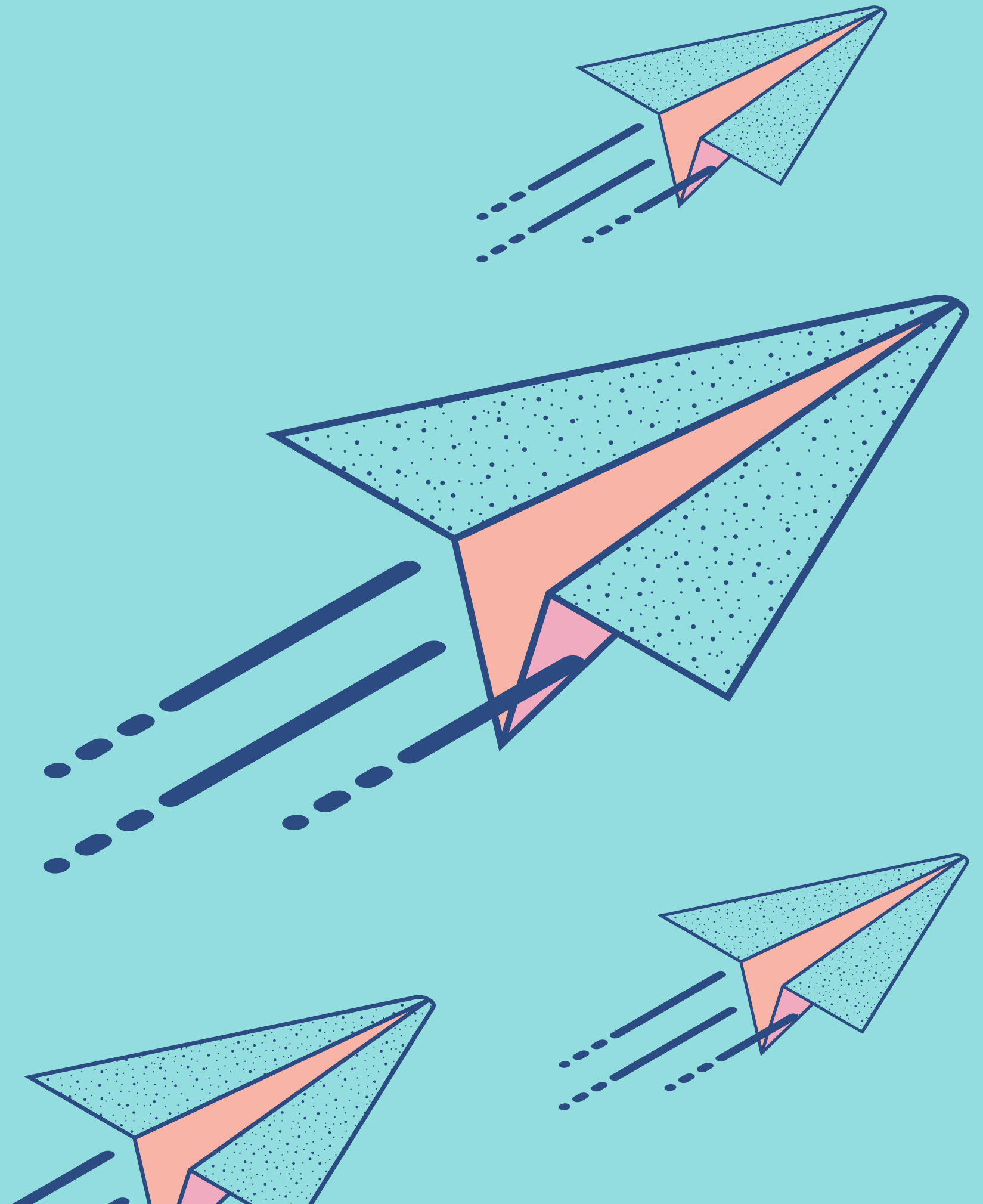


Buenas prácticas en Git



¿Para qué sirven las buenas prácticas?

- Es un estándar manejado en la mayoría de equipos de desarrollo.
- Resolver conflictos o problemas durante el desarrollo es más fácil.
- Tu historial de commits es más legible



1. ¿Cada cuánto debería hacer un commit?

A menudo.

Es mejor hacer commits pequeños, agrupando pequeñas mejoras o acciones, que un commit con todo lo que se quiere hacer.

Hacer commit a menudo no significa que debas hacer commits sin sentido.





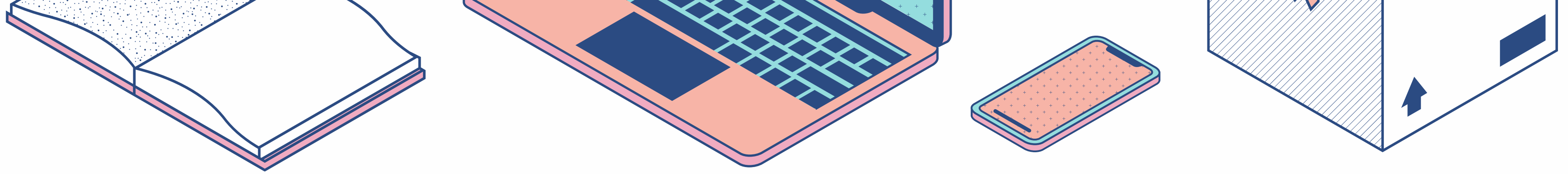
Escribir buenos commits

- Usar el verbo imperativo (Add, Change, Fix, Remove)
- No uses punto final ni puntos suspensivos en tus mensajes (a lo más usa la coma)
- Usa como máximo 50 caracteres para tu mensaje de commit
- Añade todo el contexto que se necesario en el cuerpo del commit (con reglas de puntuación)
- Considera usar utilidades para hacer commit
- Usa un prefijo para tus commits para hacerlos más semánticos

<tipo-de-commit>[scope]: <descripcion>

Por ejemplo:

```
feat: add new search feature
^--^  ^-----^
|      |
|      |L--> # Descripción de los cambios
|      |
|      |L-----> # Tipo del cambio
```

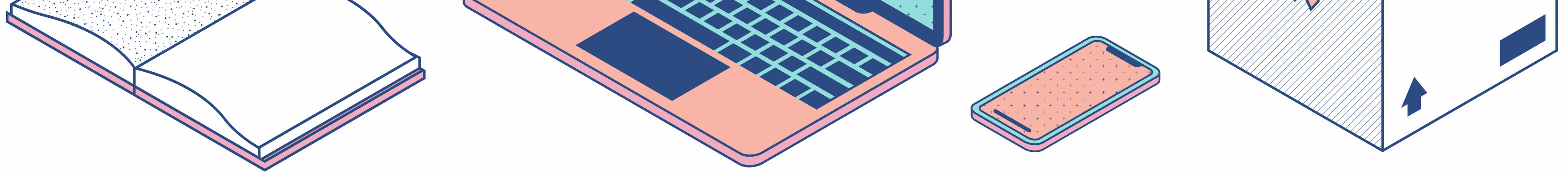


Prefijos para commits

- **feat:** para una nueva característica para el usuario.
- **fix:** para un bug que afecta al usuario.
- **perf:** para cambios que mejoran el rendimiento del sitio.
- **build:** para cambios en el sistema de build, tareas de despliegue o instalación.
- **ci:** para cambios en la integración continua.
- **docs:** para cambios en la documentación.
- **refactor:** para refactorización del código como cambios de nombre de variables o funciones.

- **style:** para cambios de formato, tabulaciones, espacios o puntos y coma, etc; no afectan al usuario.
- **test:** para tests o refactorización de uno ya existente.

```
feat(backend): add filter for cars  
fix(web): remove wrong color
```



Escribir un buen nombre de rama

- Sé consistente al nombrar tus ramas
- Usa el nombre de la acción que se realiza en la rama

```
bug/avoid-creating-lead-twice  
feature/add-new-user-form  
experiment/try-new-ui-design  
hotfix/fix-typo-in-name
```

- Usa los IDs de JIRA o el sistema de tickets que uses

Repository
CodeMasterChefs/Backend-Ruta_del_Programador

Branch from
main

Branch name
RDP-143-Visualizar-cursos-playlists

Create branch