



Organización del proyecto

- Creación del repositorio en GitHub
- Definición de roles: Admin, colaboradores, reviewers
- Rama master protegida
- Uso de README.md para documentar el proyecto

1



Creación y gestión de ramas

- Crear ramas desde master
- Una rama por programador o funcionalidad
- Tipos de ramas:
 - feature (nuevas funciones)
 - fix (correcciones)
- Asignación de ramas por el líder del proyecto

2



Clonado y sincronización

- Clonar el repositorio: git clone
- Descargar cambios iniciales: git pull origin master
- Cambio a la rama asignada: git checkout nombre_rama
- Mantener sincronización constante

3



Desarrollo y commits

- Trabajo individual en la rama
- Commits frecuentes y con mensajes claros
- Control del historial de cambios
- Evitar trabajar directamente en master

4

TRABAJANDO CON REPOSITORIOS REMOTOS EN GITHUB

5

Subida y actualización de ramas

- Verificar cambios antes de subir
- Actualizar rama: git pull origin nombre_rama
- Subir trabajo: git push origin nombre_rama
- Resolver conflictos si existen



6

Pull Request (PR)

- Solicitud para unir código a master
- Comparación de ramas en GitHub
- Generación de conversación entre el equipo
- Posibilidad de validaciones automáticas (builds)



7

Revisión de código

- Revisores analizan el código
- Opciones:
 - Aprobar
 - Comentar
 - Pedir cambios
- Correcciones sobre la misma rama
- El PR se actualiza automáticamente



8

Merge, pruebas y producción

- Aprobación final del Pull Request
- Merge a master desde GitHub
- Pruebas en servidores staging / develop
- Despliegue final a producción

