



# Git + Bitbucket

## Guión

26



## Git

### Configurar Git

- Ejecutar en la Terminal el comando **git --version**
- Ejecutar los siguientes dos comandos en la Terminal
  - › `git config --global user.name "Nombre de usuario"`
  - › `git config --global user.email "Email de usuario"`

### Clonar esqueleto del proyecto

- Crear una carpeta en /home/alumno (por ejemplo **esqueleto**)
- Ejecutar en la Terminal el comando **git clone** <https://llabor@bitbucket.org/llabor/demo-repo.git>

27



## Bitbucket

### Crear cuenta Bitbucket

- Ir a <https://bitbucket.org/>
- Indicar e-mail y password
- Confirmar cuenta
- Ir a la parte izquierda al botón (+) y elegir **Repository**
- Indicar **Repository name** (por ejemplo esqueleto)
- Marcar **Access level** como público → desmarcar la opción **private**
- Elegir en **Version Control System** → **Git**

28



## Bitbucket

### Vincular repositorio Bitbucket

- En la Terminal ir a la carpeta previamente creada (esqueleto)
- Ejecutar los siguientes comandos:
  - › `git remote add origin <aliasderepositorio ej. mirepo>  
https://<usuario>@bitbucket.org/<usuario>/<nombre del repo creado en el paso anterior ej. Esqueleto>.git`
  - › `git push -u <aliasderepositorio ej. mirepo> master`
- Con esto habremos subido el código del repositorio clonado a nuestro repositorio personal

### Ver repositorios y alias actuales

- En la Terminal ejecutar el siguiente comando:
  - › `git remote -v`

29



## Bitbucket

### Gestionar ciclo de vida de cambios

- Modificar los ficheros con Atom según el ejercicio realizado
- Una vez queramos subir los cambios ejecutar los siguientes comandos:
  - › **git status** (para ver en qué situación estamos)
  - › **git remote -v** (muestra los alias ej. mirepo y repositorio vinculado)
  - › **git add .** (para añadir los archivos al repositorio local)\*
  - › **git commit -m "Versión xxx"** (añade un comentario al commit)
  - › **git push -u mirepo master** (para subir los cambios a BitBucket en mirepo en la rama master)\*\*

30



## Git + Bitbucket

### Ignorar ficheros

- Crear un fichero llamado **.gitignore**
- Añadir las carpetas o ficheros que no queremos subir al repositorio:
  - › **/node\_modules/\*** → No subirá la carpeta y los ficheros incluidos en ella
  - › **\*.log** → No subirá los ficheros .log
- Subir los cambios
  - › **git add .**
  - › **git commit -m "Versión con tests Mocha"**
  - › **git push -u mirepo master**

31