

0. Entorno de producción (DB, dominio y document root)

Crear el dominio, importante que tenga soporte para DNS.

Base de datos remota, en sql, se debería crear desde el mismo usuario que creó el dominio (para que se encuentren bajo el mismo host).

Crear una custom document root, debe ser el directorio que contiene las carpetas de laravel, node y vue. (Sólo se podrá crear el document root cuando ya exista el directorio, después del primer despliegue).

1. Órdenes de terminal para el despliegue de los archivos:

Abrir el terminal en vscode en la raíz del proyecto y ejecutar los siguientes comandos para preparar laravel o vue para su despliegue, respectivamente. Las instrucciones para node se encuentran en el apartado de despliegue de sockets.

1.1 LARAVEL

```
cd ./laravel/  
composer install  
cp .env.exemple .env
```

(Modificar .env para incluir las credenciales de la BD a utilizar, **sólo la primera vez**)

```
php artisan key:generate
```

1.2 VUE

```
cd ../vue_project/  
  
npm install
```

Cambios de rutas para deploy (explicadas abajo)

```
npm run build
```

2. CAMBIOS DE RUTAS

Cambiar rutas en: `tr2-MultipliCAT-G7/index.html`, `src/components/TaulerView.vue`, `src/components/PaginaPrincipal.vue`, `src/components/PartidaFinalitzada.vue` i `src/utills/socket.js`, les diferents rutes están explicades als propis arxius.

CAMBIAR EL PUERTO EN `node/index.js` Y EN `src/utills/socket.js` EN FUNCIÓN DE SI SE DESPLIEGA EN PROD O PREPROD

3. Credenciales FTP tr2

Las credenciales para conectarse con filezilla al labs varían en función de si queréis acceder a producción o preproducción.

CREDENCIALES EN PREPROD:

Servidor: `preprod.tr2g724.daw.inspedralbes.cat`

Usuario : **a21marbenalc_tr2_{vuestro nombre en minúscula}**Preprod

ejemplo: `a21marbenalc_tr2_mario`Preprod

Contraseña: **Passwo{vuestro nombre en minúscula}**

ejemplo: `Passwomario`

CREDENCIALES EN PROD:

Servidor: `tr2g724.daw.inspedralbes.cat`

Usuario : **a21marbenalc_tr2_{vuestro nombre en minúscula}**

ejemplo: `a21marbenalc_tr2_mario`

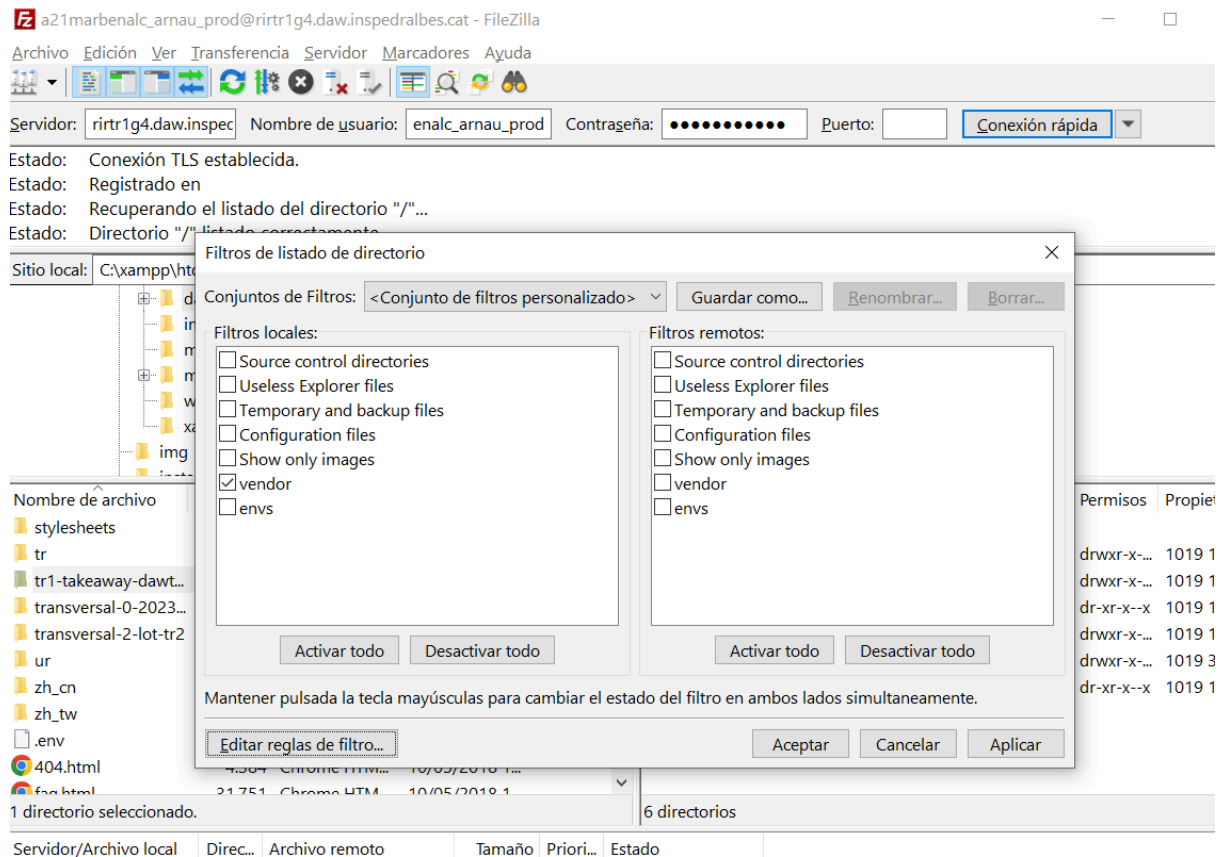
Contraseña: **Passwo{vuestro nombre en minúscula}**

ejemplo: `Passwomario`

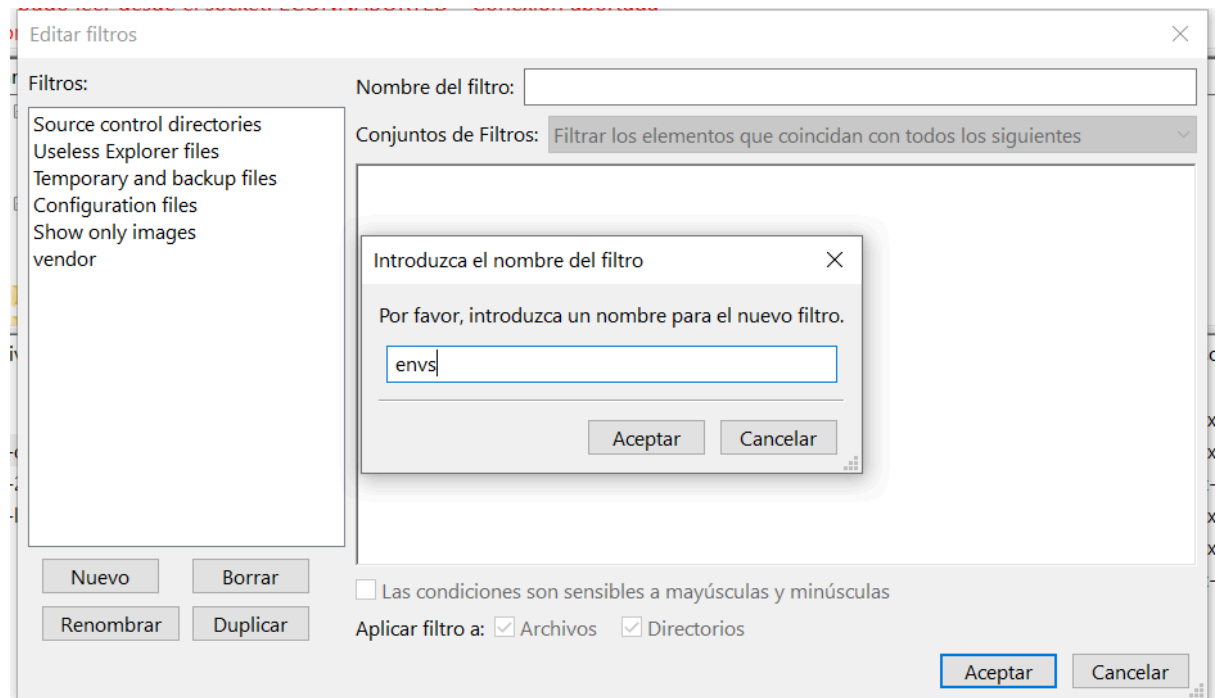
4. Filtros para subir a filezilla

Para mejorar el tiempo de subida y evitar sobreescribir archivos incorrectos, es importante establecer filtros de archivos de subida para filezilla:

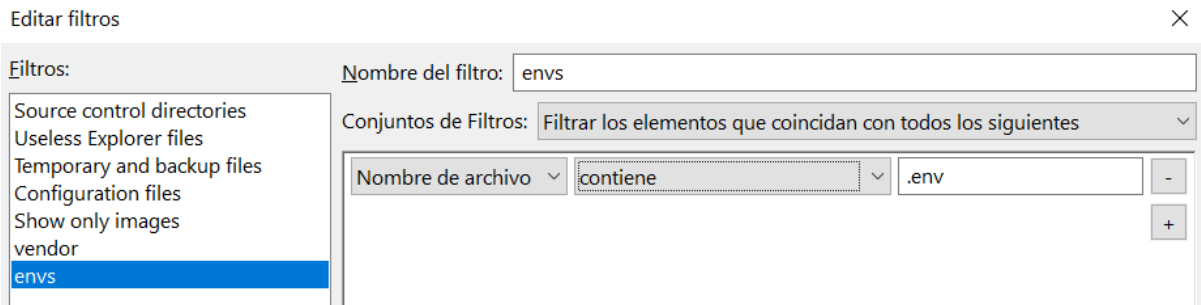
1. Pulsando ctrl+i podemos abrir la lista de filtros activos, debemos seleccionar “editar reglas del filtro”



2. Seleccionamos “Nuevo” y lo nombramos (recomendado nombrarlo cómo el archivo que debe filtrar, para evitar confusiones)



- Determina el nombre de los archivos que filezilla ignorará al transferir archivos, y selecciona aceptar.



- Hacer un filtro para los .env, otro para el directorio vendor y un tercero para el directorio node_modules siguiendo éstos pasos.
- IMPORTANTE** los filtros al ser creados están desactivados por defecto, se deben activar clickando en sus checks respectivos en la pantalla de visualización de filtros explicada en el paso 1



5. Despliegue de archivos

5.1 Primera vez

- Activar SOLAMENTE el filtro de node_modules en filezilla.
- Crear una carpeta con el mismo nombre que la carpeta raíz del proyecto (tr2-MultipliCAT-G7)
- Subir las carpetas de laravel, node y el index.html de la raíz a esa carpeta
- Crear una carpeta de nombre 'vue', subir los contenidos del directorio *dist* (creado con la orden run build) ahí.

5.2 Despliegue continuado

- Activar TODOS los filtros, a menos que se quieran hacer cambios en las dependencias del composer o en los módulos utilizados.
- Subir el directorio de laravel tal cual, sobrescribiendo el directorio anterior.

- Subir el archivo /node/index.js a la carpeta de node, sobrescribiendo el archivo index.js remoto, **ES NECESARIO VOLVER A DESPLEGAR SOCKETS TRAS ÉSTE PASO.**
- Borrar TODO el contenido del directorio 'vue' remoto, y después subir de nuevo los contenidos de *dist*.

6. Despliegue de Sockets

Conectarse al dominio con putty, y hacerle cd a node (consultar la ruta de archivos desde filezilla) hacer install y screen index.js.

- Conectarse con putty al dominio dónde lo quieres desplegar.
- Loguearse con los credenciales pertinentes (ftp o los tuyos, dependiendo del dominio)
- Moverse con cd hasta el directorio de node (lo puedes consultar desde filezilla o navegar por el terminal de putty hasta encontrarlo).
- Ejecutar:

```
npm install
```

```
screen node index.js
```
- Pulsar **Ctrl A** y posteriormente **D** para dejar sockets ejecutándose de forma continua hasta que se reinicie el servidor, haya cambios en el directorio de node o el dominio caiga por otros motivos.
- Puedes consultar la documentación de la orden `screen` [aquí](#) para poder volver a acceder si quieres detener los sockets manualmente.

7. Deploy de Docker (IMPLEMENTACIÓN PENDIENTE)

Crear cuenta en [Docker Hub](#)

Instalar docker desktop (seleccionar WSL2 y la instalación recomendada en lugar de la custom)

Abrir un terminal en la carpeta con el dockerfile y ejecutar:

```
docker build -t nombreDeLaImagen .
```

Ir a Docker Desktop, Images, y seleccionar 'run' en la imagen seleccionada

En los settings opcionales, especificar el puerto que abre el Dockerfile en el apartado de Host Port, y darle a run.

Si la imagen se comporta correctamente en localhost:HostPort, la imagen es apta para despliegue.