

Nota de clonación de Github

Si acabas de clonar el repositorio « mypelican_static_sites », debes seguir algunos pasos antes de generar los sitios.

Requisitos previos

Nuestras compilaciones para los sitios que se mantienen aquí dependen del proyecto « flitprojects », que también está bajo control de versiones en nuestra cuenta de Github. Ese proyecto utiliza entornos virtuales de Python. Por lo tanto, asumimos que hay un entorno virtual adecuado configurado, y que hemos ejecutado flit build y flit install en él para el paquete « flitprojects/articleware ».

Pasos de configuración tras clonar « mypelican_static_sites » por primera vez

1. Instalar los scripts para generar las versiones PDF de todos los artículos. Estos scripts se generan automáticamente y, por lo tanto, no se mantienen bajo control de versiones. Usamos un script bash para identificar la lista de subdirectorios de sitios y generar nuestros scripts dentro de cada uno de ellos.

```
allputs.sh
# esto genera un par de scripts en cada subdirectorio
# que contiene archivos .rst de artículos.
```

2. Una vez generados los scripts, necesitamos producir todos los archivos .pdf. Para hacerlo, accedemos a cualquiera de los subdirectorios del sitio y usamos make para generar los archivos .pdf.

```
cd www.bernatchez.net
make pdfs
```

3. Genera las versiones locales de tus sitios para verificar que estén listos para publicar.

```
cd www.bernatchez.net
make html
cd ..

cd blog.bernatchez.net
make html
cd ..

cd www.ogopogo.biz
make html
cd ..

cd docs.pelican.bernatchez.net
make html
cd ..
```

4. Genera las versiones publicables de tus sitios y publícalos.

```
cd www.bernatchez.net
make pubhtml
cd ..
```

```
cd blog.bernatchez.net
make pubhtml
cd ..

cd www.ogopogo.biz
make pubhtml
cd ..

# Usar solo la versión del sistema de archivos para este, no generar.
# No mantenemos el sitio con el CDS bunny.
cd docs.pelican.bernatchez.net site
echo 'No mantenemos el sitio "docs.pelican.bernatchez.net" con el CDS bunny.'
cd ..
```

5. El uso de make con « pubhtml » genera URLs en línea dentro de muchos de los enlaces del contenido. Esto es incompatible con el contenido generado por make « html », cuyos enlaces deben ser todos relativos al sistema de archivos. Esto significa que el contenido en el sistema de archivos local deja de ser válido inmediatamente después de generar el contenido « pubhtml ». Por lo tanto, debemos volver a ejecutar make « html » una vez publicado el contenido para revertir el contenido local a su estado correcto.

```
cd www.bernatchez.net
make html
cd ..

cd blog.bernatchez.net
make html
cd ..

cd www.ogopogo.biz
make html
cd ..
```