

# Jugo de Grunt

Manual de Usuario

Por: Pablo Andueza Munduate



**GRUNT**  
JS Task Runner

# Instalación

*Preparación del entorno si es la primera vez que vas a usar Jugo de Grunt (JdG) en el equipo.*

A modo general, sea el que sea el uso que le vayas a dar al entorno JdG el equipo en el que se vaya a usar tiene que tener instalado npm, npm se suele instalar cuando se instala nodejs. Lo puedes descargar para instalar en:

<https://nodejs.org/en/>

Además, teniendo ya npm, hay que instalar el *Grunt command line interface* en la máquina globalmente, para ello abres la consola de comandos y desde cualquier directorio ejecutas.

```
npm install -g grunt-cli
```

Ahora ya se podría empezar a utilizar algunas de las características de JdG, para ello simplemente se descomprime el zip con el entorno en una carpeta donde vayas a desarrollar un proyecto (también puedes usar el comando `init.bat`). Acciones como hacer backups o minificar automáticamente ficheros (ver manual más adelante) se podrían ya realizar siempre que se lancen ejecutando los diferentes ficheros `.bat` contruidos a tal efecto.

Sin embargo si se quiere hacer uso del menú frontal de lanzamiento de los diferentes servicios, desarrollado en formato página web, se requiere tener instalado algún tipo de servidor php, igualmente si se quiere lanzar servidores para desarrollo o producción a través del entorno requiere tener instalado un programa servidor php.

Dos de los programas servidores más utilizados y que se pueden usar para ello indistintamente son

Xampp → <https://www.apachefriends.org/es/index.html>

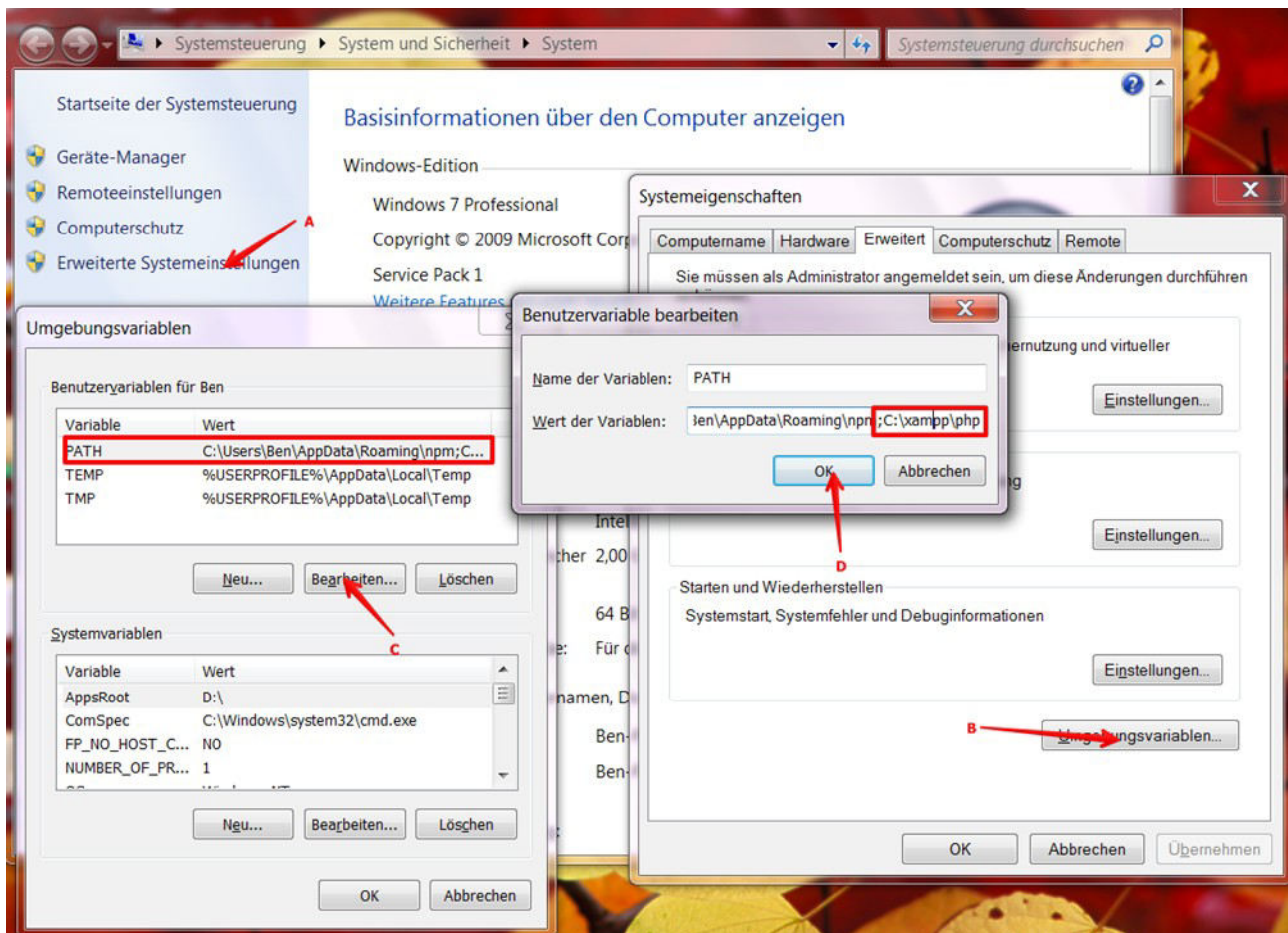
Wamp → <http://www.wampserver.com/en/>

Según el equipo en el que se haya instalado es posible además que sea necesario añadirle el path o ruta al php en las variables del sistema; Ir a

Equipo->propiedades->Configuración avanzada del sistema->Variables de entorno..

Y en PATH hay que añadirle (después de ; ) la dirección donde esté instalado el servidor php (donde este `php.exe`), en Xampp por defecto está en `C:\xampp\php`

Esto se puede ver de manera gráfica en la siguiente imagen:



Si el servidor php instalado es el de Wamp la dirección es algo más larga, del tipo:

C:\wamp64\bin\php\php7.0.4

Y por supuesto varía un poco según la versión instalada.

Ahora ya estarían casi todas las funcionalidades del entorno JdG disponibles. Solo queda por añadir Git para las acciones que usan este sistema, que si no ha sido ya previamente instalado se puede descargar para instalar en

<https://git-scm.com/download/win>

Y eso es todo ya están todas las características de JdG disponibles.

Aviso: No modificar archivos .bat directamente.

Si por alguna razón quisieras modificar los archivos .bat no modifies los que están en raíz del proyecto, es muy probable que luego no funcionen por tema de codificación (por acentos y caracteres especiales). Si puedes, en cambio, modificar los archivos .bat homólogos con en añadido “(modificar\_unicode)” que se encuentran en /\_jugodegrunt/sources, después en esa misma capeta deberías ejecutar:

```
type archivo(modificar unicode) > archivo.bat
```

Y ya ese archivo copiado en raíz de proyecto será totalmente funcional.

## Inicio

Jugo de Grunt (JdG) tiene dos maneras de funcionar, ejecutando archivos .bat creados específicamente para cada acción desde la consola de comandos (o con doble-click en el explorador), o lanzando esos mismos archivos desde un entorno gráfico que se inicializa ejecutando:

000-Start!.bat

Ejecutar los archivos bat independientemente tiene la ventaja de la agilidad (sobre todo si vas a realizar una sola acción en concreto, por ejemplo un backup), el entorno gráfico tiene la ventaja de una visualización más clara además de que se muestran en todo momento las direcciones a las que se apuntan.

Aquí se irán describiendo las diferentes acciones según cómo aparecen en el entorno gráfico pero se adjuntará también el nombre del fichero .bat asociado a la acción. Notar que los archivos bat suelen incluir también pequeños textos descriptivos o de aviso, y pueden realizar preguntas al usuario.

## Acciones



### Vigilar / Minificar

1escuchar.bat

Al ejecutarse queda en segundo plano y, mientras se esta trabajando en un proyecto en el directorio de Desarrollo ('de'), este proceso se encarga de copiar y en los casos de archivos .html, .css y .js además minificar (eliminando espacios y comentarios) el contenido en el directorio de Producción ('pr').

Este proceso es útil porque se reducen los tamaños de los archivos y por lo tanto los tiempos de carga.

Es bastante habitual que se inicie un proyecto no empezando de cero sino teniendo ya cierto contenido en Desarrollo. Si se inicia la vigilancia antes de copiar ese contenido a Desarrollo el programa se enterará de los cambios. Si, por el contrario, se tiene ya cierto contenido y después se empieza a vigilar el programa no se enterará hasta que se haga algún cambio (el que sea).

En este sentido hay una acción relacionada con esto que es "Desarrollo a Producción" y cuya única función es hacer lo mismo que hace "Vigilar / Minificar" solo que en ese caso no espera a que haya un cambio, sino que realiza una ejecución sin más.



## Servidor Producción

1servidor-produccion.bat

Este servicio arrancará un servidor php con el contenido que haya en la carpeta de Producción. Deberá haber algún fichero index (ya sea .html o .php) para que sea lanzado cuando arranque.

Cuando se ejecute esta acción se abrirá una ventana de comandos que irá mostrando los mensajes del servidor y que no se deberá cerrar hasta que queramos paralizar el servicio.



## Servidor Desarrollo

1servidor-desarrollo.bat

Este servicio arrancará un servidor php con el contenido que haya en la carpeta de Desarrollo. Deberá haber algún fichero index (ya sea .html o .php) para que sea lanzado cuando arranque.

Cuando se ejecute esta acción se abrirá una ventana de comandos que irá mostrando los mensajes del servidor y que no se deberá cerrar hasta que queramos paralizar el servicio.



## Desarrollo → Producción

1des-a-prod.bat

Esta acción hace casi lo mismo que Vigilar / Minimificar solo que no se queda vigilando en segundo plano a la espera de un cambio sino que lanza una ejecución sin más.

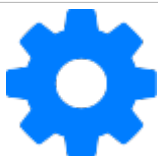
Puede ser útil si, por ejemplo, no queremos tener en segundo plano “Vigilar/Minificar”, podemos ejecutar esta acción cuando se quiera y ya esta.

Sin embargo hay una pequeña diferencia (que no afecta sin embargo a los resultados) y es que si bien el resto de los ficheros solo copia o cambia lo que este modificado en los archivos html, css y js los vuelve a minimizar y copiar todos. De todas formas la ejecución de la minificación es en extremo rápida y, muy gigantesca tiene que ser la web para que afecte a la productividad. (Se intentará solucionarlo en futuras versiones).

Nota: Si al minimizar los archivos html no se ven correctamente (es decir igual que sin minificar) es muy probable que la razón sea que haya algún fallo en el código (del tipo cerrar alguna etiqueta) que si bien los navegadores transigen (porque solucionan por si mismos el fallo) cuando están sin comprimir no lo hacen cuando están comprimidos.

La solución pasa por corregir esos errores y la forma más sencilla es pasarle algún sistema de validación o chequeo de código, la w3c tiene un validador online aquí:

<https://validator.w3.org/>



## Configurar FTP

2ftp-configurar.bat

Esta acción sirve para configurar el acceso a un servidor FTP al que vayamos luego a subir nuestros contenidos.

Durante el proceso de configuración se nos hará una serie de preguntas donde podemos o bien introducir los nuevos datos o bien pulsar enter con lo que se mantendrá la última configuración conocida para cada caso.

Se nos preguntará en el siguiente orden; dirección del servidor, puerto (muy habitualmente 21), usuario de acceso, password de dicho usuario, directorio de destino (que puede ser 'public\_html', simplemente '/' o algún otro dependiendo del servidor; hay que conocer donde nos sitúa el FTP por defecto y donde está en relación con la dirección http del hosting web). por último, además, se nos preguntará si lo que queremos subir es el contenido de Desarrollo o de Producción.

En el menú gráfico se puede ver encima del botón cuál es el sitio, puerto y el directorio al que se subirá y que es lo que se subirá.



## Subir FTP

2ftp-subir.bat

Esta acción subirá por FTP aquello que se haya modificado o añadido en el directorio designado (Desarrollo o Producción) desde la última subida a ese mismo servidor. Si es la primera vez que se sube subirá todo.

También, si hay archivos en el servidor que ya no están en el directorio origen serán eliminados.

El acceso servidor FTP destino se configura mediante la acción independiente "Configurar FTP".



## Generar Sitemap

1sitemap.bat

Se creará un archivo sitemap.xml que contendrá referencias a todos los archivos que haya en el directorio de Desarrollo (ficheros html, imágenes, pdfs, y todo, absolutamente todo, lo demás) que se guardará en la raíz de ese mismo directorio.

Se preguntará cuál será la dirección web raíz a la que estarán asociados los archivos ("homepage") de tal forma que aunque el rastreo se haga en la máquina local las direcciones que se incluirán en el archivo sitemap tendrán como inicio la dirección indicada.

El archivo sitemap.xml es utilizado por los buscadores (una vez subido al directorio raíz de nuestra web) para agilizar y facilitar que se rastreen todos los archivos contenidos en nuestra web y de esa forma puedan salir en las búsquedas.

Una peculiaridad de este modo de crear el sitemap es que no lo hace siguiendo todos los posibles enlaces empezando por la pagina de inicio, sino que sin más añade todos los archivos que encuentra en el directorio. Esto implica que este método no sirve para páginas dinámicas donde el contenido se extrae de bases de datos, por otro lado tiene la propiedad de que añade páginas o contenido que no necesariamente esta enlazado desde nuestro sitio principal, lo cual puede resultar útil (o no) en algunas ocasiones.

A cada enlace añadido se le adjuntará automáticamente la fecha de modificación.

En principio cada enlace añadido contaba además con otro par de parámetros que se añadían automáticamente el "chagefreq" y el "priority" y que en este caso tenían el mismo valor para todos los archivos, los cuales se ha decidido eliminar del sitemap.xml ya que, por un lado, en el caso de changefreq (cuyo valor puede ser "always", "hourly", "daily", "weekly", "monthly", "yearly" o "never") al aplicarle el mismo valor a todos los archivos estaríamos simplemente mintiéndole a los robots de los buscadores, porque evidentemente no todos los archivos iban a cambiarse con tal o cual asiduidad, y ante esta situación es mejor no poner nada y ya esta, el buscador seguirá intentando actualizar la indexación de nuestra web rastreando los archivos de forma equitativa, lo mismo se aplicó al parámetro priority que suele tener un valor de 0 a 1 y en este caso se añadía automáticamente el valor 0.5 a todos los archivos (de tal forma que ninguno tenía prioridad y todos serían equitativamente revisados cada cierto tiempo), que se ha decidido eliminar porque su utilidad era nula y lo único que hacía era añadirle más peso al archivo sitemap.xml

Al eliminar ambos parámetros el archivo sitemap.xml pesa menos y los buscadores podrán dedicar más de su cupo diario de datos con respecto a nuestra web para rastrear las páginas.



## Comprobar Links

1linkcheck.bat

Comprobará si el proyecto de la carpeta Desarrollo tiene enlaces internos (enlaces que apuntan a páginas de la misma web) rotos.

Para ejecutarlo es necesario que previamente este lanzado el servicio “Servidor Desarrollo” pues hace las comprobaciones a partir de este.

Siempre que haya enlaces rotos nos mostrará, al final, tras haber pasado por todos los links las siguientes 2 líneas:

```
Warning: Task "linkChecker:dev" failed. Used --force, continuing  
  
Done, but with warnings.
```

Si no encuentra enlaces rotos nos devolverá Done.



## Configurar Git

3git-configurar.bat

Esta acción servirá para introducir todos los datos necesarios para poder acceder a y, si fuera el caso también escribir en, un repositorio git.

Primero de todo se nos preguntará por la dirección del repositorio git, después se nos preguntará si vamos a definir una rama en concreto o si lo dejamos en blanco y no definimos ninguna. Si no definimos ninguna rama el sistema cuando se vaya a hacer la clonación automáticamente intentará hacerla desde la rama master o principal.

Luego se nos preguntará por el usuario (el usuario solo sería necesario si fuéramos a subir algo al repositorio), este, si queremos que nos sirva, deberá tener permiso de escritura en el repositorio. Si la pregunta se deja en blanco el usuario será el último usuario git conocido, en principio que haya usado “Jugo de Grunt” (JdG), si no hay nadie el usuario git ya configurado en el sistema Windows (y si no hay nadie saldrá una ventana cuando se trate de subir algo pidiéndonos usuario y contraseña). En la pregunta aparece el último usuario conocido por JdG.

Si introducimos un nuevo nombre de usuario de seguido nos preguntará por la contraseña. Y eso es todo.





## Git Clone

3git-clone.bat

Mediante este método haremos una clonación inicial de lo que haya en el repositorio previamente definido con “Configurar Git” en la carpeta de Desarrollo ‘de’.

Esta carpeta deberá estar vacía, y por motivos de seguridad se ha evitado automatizar esta tarea; el usuario deberá ser consciente del contenido que hubiera en esa carpeta antes de borrarlo todo.



## Git Commit

3git-commit.bat

Cuando se hayan hecho uno o varios cambios aparte de guardar de la manera habitual hay que ir haciendo commits que son como instantáneas que quedan guardadas en el registro de git en local y que luego se pueden subir al repositorio (una o varias a la vez).

Al ejecutar esta acción nos preguntará si queremos añadirle algún comentario personalizado al commit o si le añade el comentario automático, el comentario automático tendría el formato:

Repository Updated on fecha/hora.



## Git Pull

3git-pull.bat

Con esta acción se descargará lo que haya sido modificado en el repositorio desde la última vez que se descargó (ya sea con otro pull o con clone). Pero más allá de descargar ficheros o de borrar ficheros en local si no están en el repositorio, lo que hace es integrar el código que ha cambiado en el repositorio con el código que nosotros tenemos y que a su vez hayamos podido cambiar, línea a línea.

La acción incluye el fetch y el merge.

Un caso típico de la necesidad de usar “Git Pull” es que hayas intentado subir y git nos haya dado un error avisándonos de que había modificaciones en el repositorio, en ese caso debemos ejecutar esta acción.

Si después de ejecutarla no da mensaje de que tenemos que solucionar un conflicto a mano, (ver siguientes párrafos) volveremos a hacer un commit y después un push.

Hay conflictos en los que no tenemos que actuar; cuando los archivos modificados en el repositorio son distintos de los modificados en local, o cuando aún siendo el mismo fichero las líneas modificadas son distintas, el sistema se encargará de entramar (o mezclar) las líneas. Sin embargo el caso en el que SI debemos actuar de manera manual es cuando las mismas líneas están modificadas de distinta manera en el repositorio y en local.

En ese caso hay que volver a ejecutar un commit y después tras hacer el pull, el o los ficheros en cuestión incluirán todas las líneas en conflicto, junto con unas marcas llamativas para que las veamos de un simple vistazo y podamos elegir las que queremos que se mantengan. Tras solucionar y elegir las líneas que queremos volvemos a ejecutar un commit y después ya podremos subir mediante el push.



### Git Push

3git-push.bat

Esta acción sirve para subir el/los commit que hayamos hecho en local pero que todavía no se hayan subido al repositorio.

El usuario de acceso, que se puede especificar usando el “Configurar Git” deberá tener permisos de acceso a dicho repositorio.

Si en el repositorio no ha habido cambios con respecto a la copia que teníamos nos subirá sin problemas, si hubiera habido cambios (quizás realizados por otros usuarios) tendremos que hacer primero un “Git Pull”, actuar si fuera menester (ver instrucciones de la acción), volver a hacer un commit y después volver a hacer el push.



### Configurar Dropbox

4dropbox-configurar.bat

Con esta acción se configuraran los parámetros de acceso a un repositorio Dropbox al que podremos luego subir tanto el directorio de Producción como el de Desarrollo.

Al ejecutarlo lo primero que se nos pedirá sera el Token de acceso de la aplicación, tienes las instrucciones de cómo conseguir uno abajo. Con el token ya le estamos dando la dirección donde tiene que subir junto con el permiso.

Después nos dirá que le señalemos si queremos subir el directorio de Producción ('pr') o de Desarrollo ('de').

Se nos pregunta por último cuál queremos que sea el directorio del destino (dentro del espacio al que se accede con el token mediante la aplicación o con la cuenta asociada como usuario).


#### *Cómo conseguir un token de acceso:*

Si no tienes cuenta Dropbox crea una cuenta (no hace falta que ejecutar el programa que se empieza a bajar automáticamente, simplemente mira el correo con el que te registraste y te habrán enviado un correo para que cheques y confirmes, y ya tendrás la cuenta activada)

Después accede a <https://www.dropbox.com/developers/apps>

Crea una aplicación, te saldrá una ventana, entre las opciones a elegir escoge "Dropbox Api", el tipo de aplicación: "App folder– Access to a single folder created specifically for your app.", y le pones el nombre que quieras.

En el menú de la izquierda pulsa "API Explorer" y después "token/revoke" y ya te aparece un botón que pone "Get token" lo pulsas y el string que aparece es el token que se puede usar para la configuración.

	<h2>Subir Dropbox</h2> <p>4dropbox-subir.bat</p>
<p>Con esta acción subiremos un contenido a un espacio en Dropbox, definidos ambos en "Configurar Dropbox".</p> <p>En esta acción se nos preguntará si queremos comprimir el directorio que subamos o lo subiremos tal cual (si dejamos en blanco no comprimirá, si ponemos un nombre lo comprimirá en un zip con ese nombre).</p> <p>Si hemos señalado que si queríamos que lo comprimiera nos preguntará si lo queremos proteger con contraseña, en este caso es importante señalar que si bien la codificación puede ser bastante segura sobre todo si ponemos una contraseña un poco larga, no permitiéndonos descomprimir a no ser que la sepamos (independientemente del programa con el que tratemos de descomprimir), si se nos permite en principio navegar por la estructura de directorios de los archivos contenidos en este, permitiendo ver los nombres de archivos. Por ello aquí se nos preguntará si queremos además ocultar la estructura de archivos, que lo que hará sería una doble compresión (un fichero zip normal con los archivos dentro, a su vez dentro de un archivo zip protegido).</p>	



## Backup

5backup.bat

Con esta acción se realizará una copia integral del proyecto (directorio de Desarrollo) más el entorno (Grunt+JdG) en un directorio que se podrá especificar justo después de iniciar la acción.

Las copias serán de tipo espejo, en las que solo se copiarán los archivos que se hayan modificado o añadido desde la última vez que se hizo la copia en un directorio dado. Si además hay archivos en el destino que ya no se encuentran en origen estos serán eliminados.

## Reinicia Valores

5resetear.bat

Mediante esta acción se borrarán todas las direcciones, usuarios y contraseñas que se han guardado en esta instancia de “Jugo de Grunt”

