

AdminFlix

Parámetros de configuración.

Índice

- 1. Servidor de usuarios con acceso a gestión web.....3
- 2. Consultas a servidor de Youtube.....4
- 3. Tareas en segundo plano en Linux.....5

1. Servidor de usuarios con acceso a gestión web.

Servidor OpenLDAP.

En el archivo `src/.env` establecer los siguientes valores:

```
LDAP_SCHEMA = OpenLDAP
LDAP_HOSTS = ldap.ejemplo.com          # Servidor LDAP
LDAP_BASE_DN = dc=example,dc=com      # base de "nombre distinguido" para usuario
LDAP_USER_SEARCH_ATTRIBUTE = uid      # campo por el que se identifican los usuarios en el servidor
LDAP_USER_BIND_ATTRIBUTE = uid        # campo que se recupera del servidor
LDAP_USER_FULL_DN_FMT = ${LDAP_USER_BIND_ATTRIBUTE}=%s,{LDAP_BASE_DN}
LDAP_CONNECTION = default             # configuración de config/ldap.php
AUTH_USER_KEY_FIELD = usuario         # campo en formulario para usuario
```

Servidor ActiveDirectory.

En el archivo `src/.env` establecer los siguientes valores:

```
LDAP_SCHEMA = ActiveDirectory
LDAP_HOSTS = 192.168.1.1              # servidor AD
LDAP_BASE_DN = dc=ejemplo,dc=com      # dominio
LDAP_USER_SEARCH_ATTRIBUTE = samaccountname # campo de búsqueda
LDAP_USER_BIND_ATTRIBUTE = cn         # valor a recuperar de servidor (nombre del usuario)
LDAP_BASE_DN_PUNTO = ejemplo.com      # dominio en formato con punto
LDAP_USER_FULL_DN_FMT = %s@${LDAP_BASE_DN_PUNTO}
LDAP_CONNECTION = default             # configuración de config/ldap.php
AUTH_USER_KEY_FIELD = usuario         # campo en formulario para usuario
```

2. Consultas a servidor de Youtube.

En el archivo *src/.env* establecer los siguientes valores:

`YOUTUBE_APP_NAME = Innovanda` # nombre de la aplicación en Youtube

`YOUTUBE_KEY = AIzaSyDpBZ4Zmdfjhm4SS_acR32pHEYw0MFXcxs` # clave para acceder a los servicios de Youtube

`YOUTUBE_HORAS_CANAL = 5` # cada cuantas horas se comprueba los datos del canal

`YOUTUBE_HORAS_LISTA = 5` # cada cuantas horas se comprueba el listado de vídeos del canal

`YOUTUBE_HORAS_VIDEO = 24` # cada cuantas horas se comprueba si se modificaron los valores del vídeo

`YOUTUBE_MAXIMO_CONSULTAS = 9900` # número máximo de consultas que la aplicación puede realizar automáticamente. Siempre hay que dejar un margen para poder seguir añadiendo canales si agotar las máximas consultas diarias permitidas para el usuario de Youtube.

3. Tareas en segundo plano en Linux.

Proceso de creación de tareas.

Hay que ejecutar cada cierto tiempo el comando que se encarga de crear las tareas que se han de ejecutar en segundo plano (actualización de datos de canales y vídeos), creando un alista de trabajo.

Para ello añadimos la siguiente línea en el archivo *etc/crontab*.

```
* * * * *      root    cd /carpeta_aplicación/src && docker-compose run --rm artisan schedule:run  
>> dev/null 2>&1
```

Proceso de ejecución de lista de trabajo.

Para ejecutar las tareas que están en lista de trabajo hay que ejecutar el siguiente comando:

```
docker-compose run --rm artisan queue:work
```

Siempre existe la posibilidad de este proceso tenga problemas y deje de ejecutarse. Para evitarlo utilizaremos la aplicación supervisor, que reinicia una aplicación si deja de funcionar.

Creamos el siguiente archivo de configuración para supervisor, */etc/supervisor/conf.d/laravel-worker.conf*, con el siguiente contenido:

```
[program:laravel-worker]  
process_name=%(program_name)s_%(process_num)02d  
command=docker-compose run --rm artisan queue:work --sleep=3 --tries=3 --max-time=3600  
directory=/carpeta_aplicación  
autostart=true  
autorestart=true  
stopasgroup=true  
killasgroup=true  
user=root  
numprocs=2  
redirect=stderr=true  
stdout_logfile=/carpeta_aplicación/src/storage/logs/worker.log  
stopwaitsecs=3600
```