

Guía paso a paso — Empaquetar a .IPK para LG webOS

¿Qué vas a obtener?

Un archivo .ipk listo para instalar en TVs LG webOS y enviar a la LG Content Store.

Usaremos la webOS TV CLI (ARES). Esta guía cubre: instalación de la CLI, preparación del proyecto, empaquetado, instalación en TV (Developer Mode), ejecución e inspección.

Requisitos previos

- Node.js LTS (v16+ recomendado) y NPM.
- TV LG con la app “Developer Mode” (desde la LG Content Store) o Emulador webOS TV.
- PC y TV en la misma red local.
- Tu proyecto con appinfo.json en la raíz.

Revisa/ajusta appinfo.json (mínimos)

Ejemplo (según tu proyecto actual):

```
{
  "id": "com.flama.app",
  "version": "6.6.0",
  "title": "Tu título",
  "vendor": "Tu compañía",
  "main": "index.html",
  "icon": "icon.png",
  "bgColor": "#000000"
}
```

Notas:

- id único (com.empresa.app) y version incrementada en cada build.
- main apunta al HTML de inicio y icon debe existir (PNG 512×512 o 1024×1024).

Instala la CLI de webOS (ARES)

Windows/macOS/Linux:

```
npm i -g @webosose/ares-cli
```

Verifica:

```
ares --version
```

(El binario se llama “ares” tanto en CLI oficial como en OSE).

Empaqueta a .ipk

1) Abre una terminal en la raíz del proyecto (donde está appinfo.json).

2) Ejecuta:

```
ares-package .
```

o, para dejarlo en una carpeta específica:

```
ares-package . -o dist
```

3) Salida esperada:

```
<appId>_<version>_all.ipk
```

(Ej.: com.empresa.app_1.0.0_all.ipk)

Activa Developer Mode en la TV (una vez)

- 1) En la TV: abre "Developer Mode" y pulsa Enable (inicia sesión si lo pide).
- 2) Verás una passphrase y la IP de la TV (la sesión dura ~50 h).

Registra la TV en la CLI

En tu PC:

```
ares-setup-device --list
```

Si no aparece tu TV, añádela:

```
ares-setup-device --add
```

```
name: LGTV
```

```
host: <IP_de_tu_TV>
```

```
port: 9922
```

```
username: developer
```

```
password/passphrase: <passphrase que muestra la TV>
```

Verifica:

```
ares-setup-device --list # Debería mostrar "LGTV (device)"
```

Instala el .ipk en la TV

Si guardaste en dist/:

```
ares-install -d LGTV dist/com.flama.app_6.6.0_all.ipk
```

(Ajusta el nombre real del archivo según tu salida).

Lanza, inspecciona y logs

Lanzar por appld:

```
ares-launch -d LGTV com.flama.app
```

Abrir DevTools/inspector (si aplica):

```
ares-inspect -d LGTV com.flama.app --open
```

Ver logs en consola:

```
ares-inspect -d LGTV com.flama.app
```

Detener la app:

```
ares-launch -d LGTV -r com.flama.app
```

Checklist antes de enviar a la tienda

- ☐ id y version correctos en appinfo.json (y version incrementada).
- ☐ Icono presente y en tamaño correcto.
- ☐ Rutas relativas (./assets/...) y nombres con mayúsc/minúsc exactos.
- ☐ App probada en TV/emulador sin errores de CORS/HTTPS.
- ☐ Streaming con certificados válidos si usas video.
- ☐ Capturas e iconos listos para el listing en Seller Lounge.

Problemas comunes y soluciones

- "ares-package no se reconoce" → reinstala Node/NPM y corre: `npm i -g @webosose/ares-cli`
- No conecta con la TV → misma red, IP correcta, Developer Mode activo y passphrase vigente.
- "appld mismatch" → el id de appinfo.json debe coincidir con el usado para lanzar/instalar.
- Recursos no cargan → revisa rutas relativas y mayúsculas/minúsculas.
- Video no reproduce → revisa CORS, codecs soportados y HTTPS en el servidor.

Anexo — Flujo rápido (copiar/pegar)

1) CLI

```
npm i -g @webosose/ares-cli
```

```
ares --version
```

2) Empaquetar

```
cd /ruta/a/tu/proyecto
```

```
ares-package . -o dist
```

3) Registrar TV (una vez)

```
ares-setup-device --add # name=LGTV host=<IP> port=9922 user=developer pass=<passphrase>
```

4) Instalar IPK

```
ares-install -d LGTV dist/com.flama.app_6.6.0_all.ipk
```

5) Lanzar/Inspeccionar

```
ares-launch -d LGTV com.flama.app
```

```
ares-inspect -d LGTV com.flama.app --open
```