



Guía de migración de aplicaciones angular de 3.x.x a 4.x.x

16 de noviembre de 2018

Índice de contenidos

1	INTRODUCCIÓN	3
2	ACTUALIZAR LA VERSIÓN DE NODE.JS	4
3	ACTUALIZAR PACKAGE.JSON	4
4	ACTUALIZAR ANGULAR-CLI	5
5	ACTUALIZAR EL CÓDIGO DE LA APLICACIÓN	6
6	ACTUALIZAR FICHEROS DE CONFIGURACIÓN	7

1 Introducción

En este documento se explica paso a paso cómo actualizar una aplicación web realizada con la versión 3.x.x del framework *OntimizeWeb* a las versiones 4.x.x.

1. Actualizar la version de Node.js
2. Actualizar dependencias en el fichero *package.json*
3. Actualizar angular-cli
4. Actualizar el código de la aplicación
5. Actualizar ficheros de configuración

2 Actualizar la versión de Node.js

Para utilizar las versiones 4.x.x de Ontimize Web hay que actualizar Node.js a la versión 8 (o posteriores). Enlace de descarga: <https://nodejs.org/en/download>

3 Actualizar package.json

Modificar las siguientes dependencias en el fichero package.json

- Actualizar *dependencies*:
 - @angular/material: 6.4.7
 - @angular/material-moment-adapter: 6.4.7
 - @angular/cdk: 6.4.7
 - @angular/flex-layout: 6.0.0-beta.18
 - El resto de los @angular: 6.1.10
 - @ngx-translate/core: 10.0.2
 - @ngx-translate/http-loader: 3.0.1
 - core-js: 2.5.7
 - rxjs: 6.3.3
 - zone.js: 0.8.26
- Actualizar *devDependencies*:
 - Añadir @angular-devkit/build-angular: ~0.8.0
 - Añadir copy-webpack-plugin: 4.5.4
 - Añadir webpack-cli: 3.1.0
 - @angular/cli: 6.2.6
 - @angular/compiler-cli: 6.1.10
 - compression-webpack-plugin: 1.1.12
 - html-webpack-plugin: 4.0.0-beta.2
 - node-sass: 4.9.4
 - typescript: 2.9.2
- Subir versión de ontimize-web-ngx a 4.x.x
- Subir versión de ontimize-web-ngx-theming a 4.x.x

Para realizar las actualizaciones de versiones se recomienda ejecutar la consola con permisos de administrador y previamente borrar las carpetas de las versiones antiguas contenidas en *node_modules*.

4 Actualizar angular-cli

Para actualizar el fichero *.angular-cli.json* y convertirlo en el nuevo *angular.json* hay que ejecutar el siguiente comando (donde xxx es la anterior versión de @angular/cli instalada en la aplicación):

```
ng update @angular/cli --from=xxx --migrate-only
```

5 Actualizar el código de la aplicación

Existen una serie de cambios en los componentes que pueden afectar al funcionamiento de la aplicación:

1. Angular

Por causa de la actualización de los diferentes módulos de @angular puede que haya que actualizar varios puntos de la aplicación. En el siguiente link se pueden ver todos los cambios para esta versión:

<https://github.com/angular/angular/blob/6.1.x/CHANGELOG.md>

(únicamente los relacionados con las versiones 6.x.x).

2. Angular Material

Debido a la actualización de @angular/material se deben realizar una serie de cambios en los componentes. Todos estos cambios se pueden ver en el siguiente link: *<https://github.com/angular/material2/blob/6.4.x/CHANGELOG.md>*.

3. OntimizeWeb

De igual manera que en los puntos anteriores, al actualizar la versión de ontimize-web-ngx hay que hacer una serie de cambios expuestos en el siguiente link: <https://github.com/OntimizeWeb/ontimize-web-ngx/blob/4.0.0-rc.0/CHANGELOG.md>.

6 Actualizar ficheros de configuración

Para poder crear la versión de producción se necesitan modificar el fichero *webpack-aot.config.js* situado en la carpeta *aot-config* (Este fichero se encuentra disponible en el apartado de recursos de la documentación). Las modificaciones realizadas sobre el anterior *webpack-aot.config.js* deben ser aplicadas también a este nuevo fichero.

Además, hace falta añadir un parámetro nuevo al script *production-aot*. Para generar la versión de producción hay que añadir el parámetro *--project-name XXX*, donde *XXX* es el nombre del proyecto en el fichero *angular.json* (lo más probable es que sea el mismo valor que el atributo *name* del fichero *package.json*).