✉ (mailto:contact@meumobi.com) | 🐦 (http://twitter.com/meumobi) | ⓖ (http://github.com/meumobi)

meumobi Dev Blog (/)

Using Ionic/Cordova app sources as PWA

MARCH 26TH, 2018

Many companies have now started to build PWAs instead of native apps
(https://www.cabotsolutions.com/building-an-app-for-your-business-consider-progressive-web-apps-
instead-of-native-apps), but sometimes you want/need both: PWA **and** native app. One main reason
could be the non-support by iOS of a major PWA feature: Push Notification.

Then the ideal scenario should be to use the same code to build PWA or native app. Ionic
(https://ionicframework.com/) is one of the best Framework to build native app and PWA (see
Ionic PWA toolkit (https://ionicframework.com/pwa)), but out of the box it's necessary to
update/(un)comment source code to switch from one mode to other. On this post we'll suggest small
updates to turn it auto, and you'll be able to choose on commande line to build/serve PWA or native
app, with resp. `ionic serve --mode=pwa` and `ionic serve --mode=native`

Setup Application

If you're not familiar with Ionic already, I'd recommend reading Getting Started with Ionic
(/ionic/2017/08/15/getting-started-ionic.html) to get up and running and understand the basic concepts.

Prerequisites

We need to have Node.js (https://nodejs.org/en/download/) and Git (http://git-scm.com/download)
installed in order to install both Ionic (https://ionicframework.com/) and Cordova
(https://cordova.apache.org/).

```
$ npm install cordova ionic typescript -g
...
$ npm ls -g cordova ionic npm typescript --depth 0
/usr/local/lib
├── ionic@3.20.0
├── npm@5.8.0
├── phonegap@8.0.0
├── typescript@2.7.2
└── typings@2.1.1
```

Starting a project with Ionic CLI

```
$ ionic start meu-starter sidemenu
✓ Creating directory ./meu-starter - done!
✓ Downloading and extracting sidemenu starter - done!

? Would you like to integrate your new app with Cordova to target native iOS and Androi
d? (y/N)
```

Yes we'd like to target native iOS and Android.

Preprocess index.html

PWA and native app declarations

Now that your Ionic app is ready, the next task is to convert it into a Progressive web app. This is done by editing `index.html` to enable **Web Manifest** and **Service Worker**. By default Ionic provide all resources required to run a PWA,… but some of them commented.

I sum up below all lines of index.html exclusively related to PWA

```
<link rel="icon" type="image/x-icon" href="assets/icon/favicon.ico">
<link rel="manifest" href="manifest.json">
<meta name="theme-color" content="#4e8ef7">

<!-- add to homescreen for ios -->
<meta name="apple-mobile-web-app-capable" content="yes">
<meta name="apple-mobile-web-app-status-bar-style" content="black">

<script>
  if ('serviceWorker' in navigator) {
    navigator.serviceWorker.register('service-worker.js')
      .then(() => console.log('service worker installed'))
      .catch(err => console.error('Error', err));
  }
</script>
```

And below lines exclusively related to native app

```
<script src="cordova.js"></script>
```

Gulp preprocess

The idea of the preprocess is to generate a target index.html before the ionic serve/build with the exact declarations required for the mode selected, pwa or native. To achieve it we'll use Gulp (https://gulpjs.com/) and its gulp-preprocess (https://www.npmjs.com/package/gulp-preprocess) plugin.

Prerequisites

```
$ npm install gulp gulp-preprocess gulp-load-plugins gulp-rename --save-dev
...
$ npm ls gulp gulp-preprocess gulp-load-plugins gulp-rename --depth 0
meu-starter@0.0.1 /Users/victor/Dvpt/PROJECTS/meu-starter
├── gulp@3.9.1
├── gulp-load-plugins@1.5.0
├── gulp-preprocess@2.0.0
└── gulp-rename@1.2.2
```

Declare mode and conditions

So, we rename the `index.html` as `index-pp.html`, it will be the source to manipulate in order to build the target `index.html` required by the framework. And we add conditions to use or not declarations, the syntax is self-explanatory:

```html
<!-- @if MODE='native' -->
<script src="cordova.js"></script>
<!-- @endif -->
<!-- @if MODE='pwa' -->
<link rel="icon" type="image/x-icon" href="assets/icon/favicon.ico">
<link rel="manifest" href="manifest.json">
<meta name="theme-color" content="#4e8ef7">

<!-- add to homescreen for ios -->
<meta name="apple-mobile-web-app-capable" content="yes">
<meta name="apple-mobile-web-app-status-bar-style" content="black">

<script>
  if ('serviceWorker' in navigator) {
    navigator.serviceWorker.register('service-worker.js')
      .then(() => console.log('service worker installed'))
      .catch(err => console.error('Error', err));
  }
</script>
<!-- @endif -->
```

And to process it a simple `gulpfiles.js`:

```js
var gulp = require('gulp'),
    args = require('yargs').argv,
    $ = require('gulp-load-plugins')();

var mode = args.mode || 'pwa';

gulp.task('preprocess',function(){
  return gulp.src('./src/**/*-pp.*')
  .pipe($.preprocess({context: { MODE: mode }}))
  .pipe($.rename(function(opt) {
    opt.basename = opt.basename.replace(/-pp/, '');
    return opt;
  }))
  .pipe(gulp.dest('./src/'));
});
```

Then now if you run `gulp preprocess --mode=pwa` you'll see a new `index.html` file to appear as required for a PWA. And for a native app run `gulp preprocess --mode=app`

Ionic cli and Gulp integration

Fine, we did it, but now we need to integrate it with ionic commands. Ionic cli provide hooks (https://ionicframework.com/docs/cli/configuring.html#hooks) to run scripts during CLI events, inclusive gulp scripts

Only add following lines on your gulpfile and `preprocess` will be run before each `ionic build` or `ionic serve`

```
gulp.task('ionic:build:before', ['preprocess']);
gulp.task('ionic:watch:before', ['preprocess']);
```

CLI pass args to gulp script, then you can pass mode as `ionic build --mode=pwa`. Et voila!

Build PWA with npm script

For Ionic/Cordova apps, the CLI will run cordova prepare, which copies the built web assets (www directory) into the Cordova platforms that you've installed. I personaly prefer to skip this useless step when I build PWA, and use instead `npm run ionic:build --prod`. For that purpose I've added a npm hook on my `package.json`

```
"scripts": {
  "clean": "ionic-app-scripts clean",
  "build": "ionic-app-scripts build",
  "lint": "ionic-app-scripts lint",
  "preionic:build": "gulp ionic:build:before --mode=pwa",
  "ionic:build": "ionic-app-scripts build",
  "ionic:serve": "ionic-app-scripts serve"
},
```

📂   Ionic (11) , (/categories.html#Ionic-ref)   PWA (3) (/categories.html#PWA-ref)

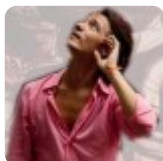🏷️   push (2) , (/tags.html#push-ref)   onesignal (1) (/tags.html#onesignal-ref)

Share Post

🐦 Twitter (http://twitter.com/share?text=Using Ionic/Cordova app sources as PWA&via=meumobi)

f Facebook (https://www.facebook.com/sharer/sharer.php)   G+ Google+

Victor Dias

Sharing mobile Experiences

🐦 Follow me (https://twitter.com/meumobi)

0 Comments   **meumobi-dev**   ① **Login** ▾

♡ **Recommend**   ⬆ **Share**   Sort by Best ▾

Start the discussion…

LOG IN WITH   **OR SIGN UP WITH DISQUS** ?

Name

Be the first to comment.

ALSO ON MEUMOBI-DEV

Get Realtime Stock Quotes using Yahoo Finance API

41 comments • 2 years ago

Hiru — I am using this https://query.yahooapis.com...And getting response as{"error":{"lang":"en-

Install and Configure DKIM with sendmail on Debian

3 comments • 3 years ago

Ant — Hi,I know this is an old article. I am trying to get sendmail to work with gmail and apparently dkim is needed.I am stuck

Alternatives to Launchkit

1 comment • a year ago

Anatoly Sharifulin — Thanks a lot for the AppFollow's mention.We have a lot features for tracking reviews & updates in

Provide an auto task listing for your gulpfile

1 comment • 2 years ago

Mykhailo Stadnyk — As alternative you can also try this https://github.com/Mikhus/g...it gives a possibility to describe tasks just

← Previous (/ionic/2018/02/22/creating-ionic-module.html)

Next → (/ionic/pwa/2018/04/12/using-analytics-ionic-app-pwa-native.html)

© 2018 meumobi with Jekyll. Theme: dbyll (https://github.com/dbtek/dbyll) by dbtek.