

Cours PWA
Complément installation application
Y. Stroppa
V1.0

Installation de l'application Binéro sur un environnement Mac OSX et Windows 10

Objectif de ce document : vous décrire les opérations à exécuter pour récupérer un projet sur github et l'installer sous un environnement Mac OSX ou Windows et le démarrer sous Node.js.

Environnement Mac OS X

Installation des outils git et Node.js à partir des sites de référence

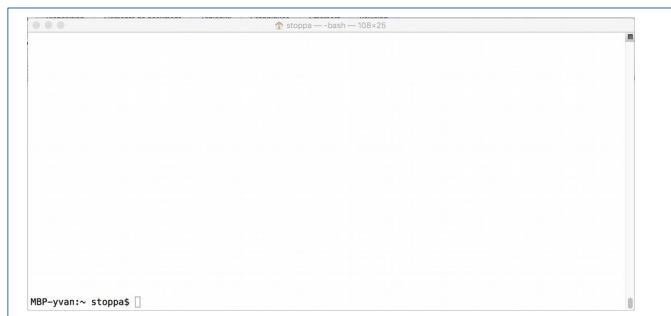
Soit vous téléchargez l'archive pkg du site
<https://nodejs.org/en/download/>

Soit si vous avez brew d'installer exécutez la commande
brew install node

Pour git :
<https://git-scm.com/downloads>

Une fois ces applications installées il faut vérifier qu'elles sont bien opérationnelles :

Pour ce faire ouvrez un terminal



Deux commandes pour vérifier le bon fonctionnement des applications téléchargées :

node -v

```
MBP-yvan:~ stoppa$ node -v  
v13.12.0
```

git

```
MBP-yvan:~ stoppa$ git --version  
git version 2.11.0 (Apple Git-81)  
MBP-yvan:~ stoppa$
```

Pour voir dans quel répertoire vous êtes, il vous suffit de taper la commande

pwd

Si vous voulez créer un dossier pour ce travail, exécutez la commande

mkdir pwa

Ensuite exécutez la commande `cd pwa` pour vous positionnez sous ce répertoire

Récapitulatif
des commandes

```
MBP-yvan:~ stroppa$ pwd
/Users/stroppa
MBP-yvan:~ stroppa$ mkdir pwa
MBP-yvan:~ stroppa$ cd pwa
MBP-yvan:pwa stroppa$ pwd
/Users/stroppa/pwa
MBP-yvan:pwa stroppa$
```

Téléchargement du projet : Exécutez la commande suivante
git clone https://github.com/ystroppa/PSB_M2_projet

```
MBP-yvan:pwa stroppa$ git clone https://github.com/ystroppa/PSB_M2_projet
Cloning into 'PSB_M2_projet'...
remote: Enumerating objects: 30, done.
remote: Counting objects: 100% (30/30), done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 30 (delta 0), reused 30 (delta 0), pack-reused 0
Unpacking objects: 100% (30/30), done.
MBP-yvan:pwa stroppa$
```

Si vous regardez le contenu du répertoire on doit avoir un nouveau répertoire nommé `PSB_M2_projet`

Il faut se mettre sous ce répertoire à l'aide de la commande :
cd PSB_M2_projet

Le contenu de ce répertoire est obtenu à l'aide de la commande `ls -lart` :

```
MBP-yvan:PSB_M2_projet stroppa$ ls -lart
total 24
drwxr-xr-x  3 stroppa  staff  102  6 avr 10:12 ..
drwxr-xr-x  7 stroppa  staff  238  6 avr 10:12 views
-rw-r--r--  1 stroppa  staff 2147  6 avr 10:12 sampleserver.js
drwxr-xr-x  3 stroppa  staff  102  6 avr 10:12 routes
drwxr-xr-x  7 stroppa  staff  238  6 avr 10:12 public
-rw-r--r--  1 stroppa  staff  402  6 avr 10:12 package.json
-rw-r--r--  1 stroppa  staff 1503  6 avr 10:12 app.js
drwxr-xr-x 13 stroppa  staff  442  6 avr 10:12 .git
drwxr-xr-x  9 stroppa  staff  306  6 avr 10:12 .
MBP-yvan:PSB M2 projet stroppa$
```

Il faut installer les modules spécifiques de Node.js à l'aide de la commande npm.

Exécutez maintenant la commande **npm install**

```
MBP-yvan:PSB_M2_projet stroppa$ npm install
npm WARN deprecated jade@1.11.0: Jade has been renamed to pug, please
install the latest version of pug instead of jade
npm WARN deprecated constantinople@3.0.2: Please update to at least
constantinople 3.1.1
npm WARN deprecated transformers@2.1.0: Deprecated, use jstransformer
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN cas1pWA@1.0.0 No description
npm WARN cas1pWA@1.0.0 No repository field.

added 101 packages from 140 contributors and audited 221 packages in 3.726s
found 4 vulnerabilities (3 low, 1 critical)
  run `npm audit fix` to fix them, or `npm audit` for details
```

Quelques warning dûs à des versions de module non réactualisées (pas grave dans notre contexte).

Ensuite pour démarrer le service Node.js, il vous suffit d'exécuter la commande suivante
npm start

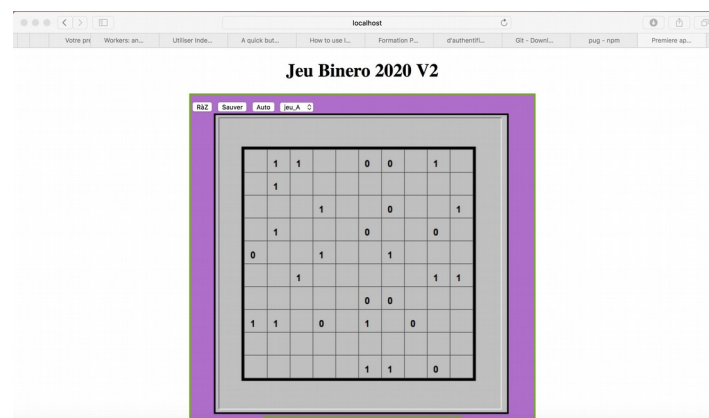
```
MBP-yvan:PSB_M2_projet stroppa$ npm start
```

```
> cas1pWA@1.0.0 start
/Users/stroppa/pwa/PSB_M2_projet
> node ./sampleserver.js
```

```
Express OK
Creation du serveur ... 8124
```

A partir de votre navigateur, vous pouvez vous connecter sur l'url suivante

<http://localhost:8124>



Vous êtes en mesure de travailler sur le projet

Première étape examinez le contenu de l'application

Deux parties :

Une partie serveur

Paramétrage de Node.js

Une partie cliente

index.html

about.html

/scripts

comportements.js

config.js

jqueryjs

main.js

/styles

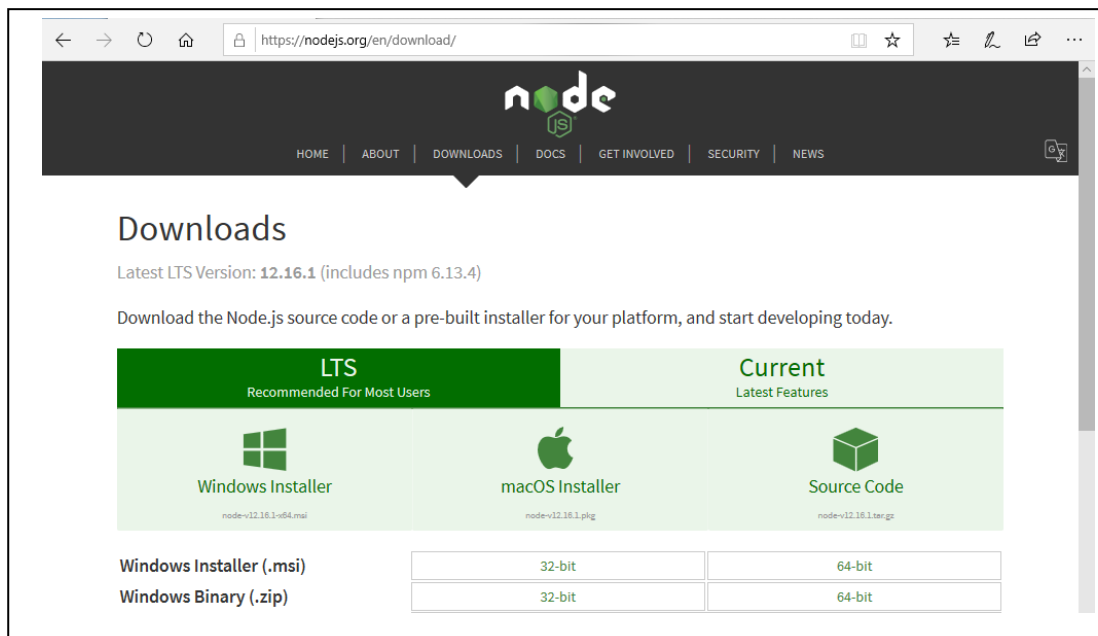
/images

C'est dans ces fichiers qu'il faudra venir insérer les éléments PWA pour rendre cette application Progressive Web en suivant l'énoncé du TP2 déposé sur le githut.

Sous Windows 10

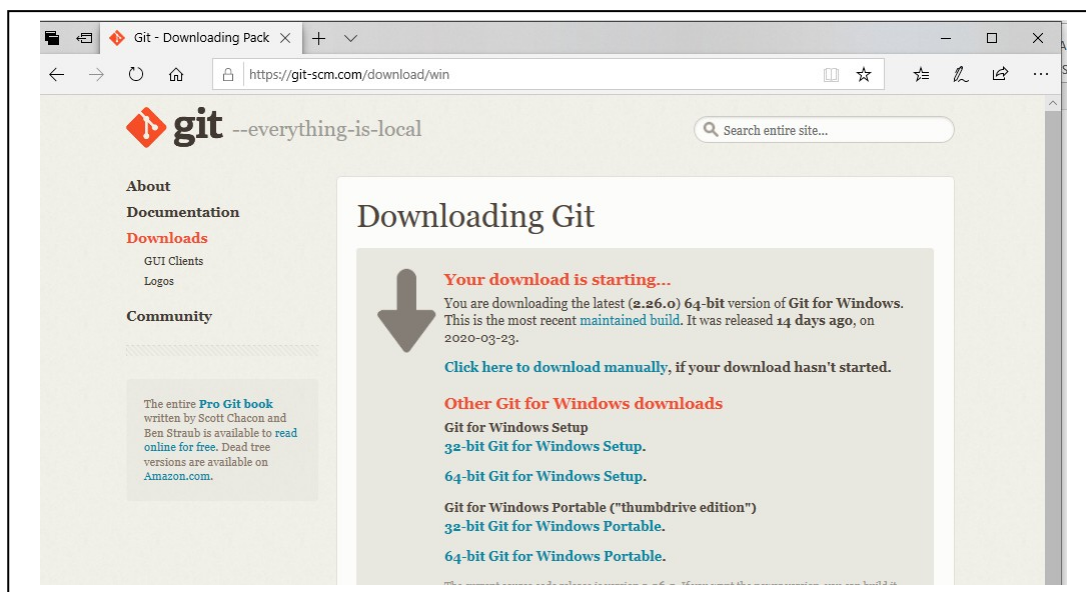
Tuto d'installation de l'application Binero sous Windows 10

Préalable : installation de Node.js et de git sous votre environnement

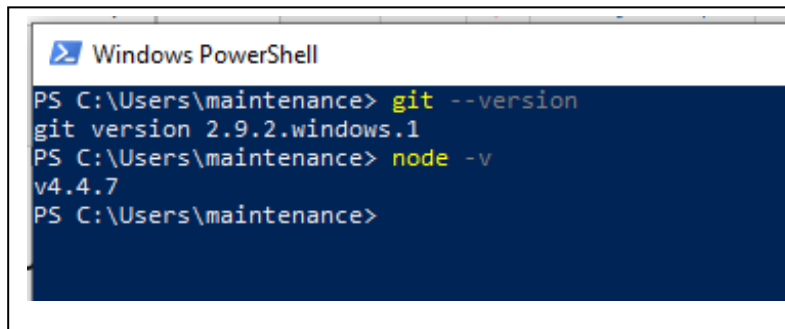


Prendre la version .msi ou zip

Pour l'outil git



Une fois les deux produits installés sur votre machine vous devez retrouver les applications dans vos environnements.
On va exécuter une ouverture de console pour effectuer les différentes commandes d'installation
Vous pouvez utiliser soit une console cmd ou powershell
Dans l'une ou l'autre vous devez avoir accès aux applications installées



```
Windows PowerShell
PS C:\Users\maintenance> git --version
git version 2.9.2.windows.1
PS C:\Users\maintenance> node -v
v4.4.7
PS C:\Users\maintenance>
```

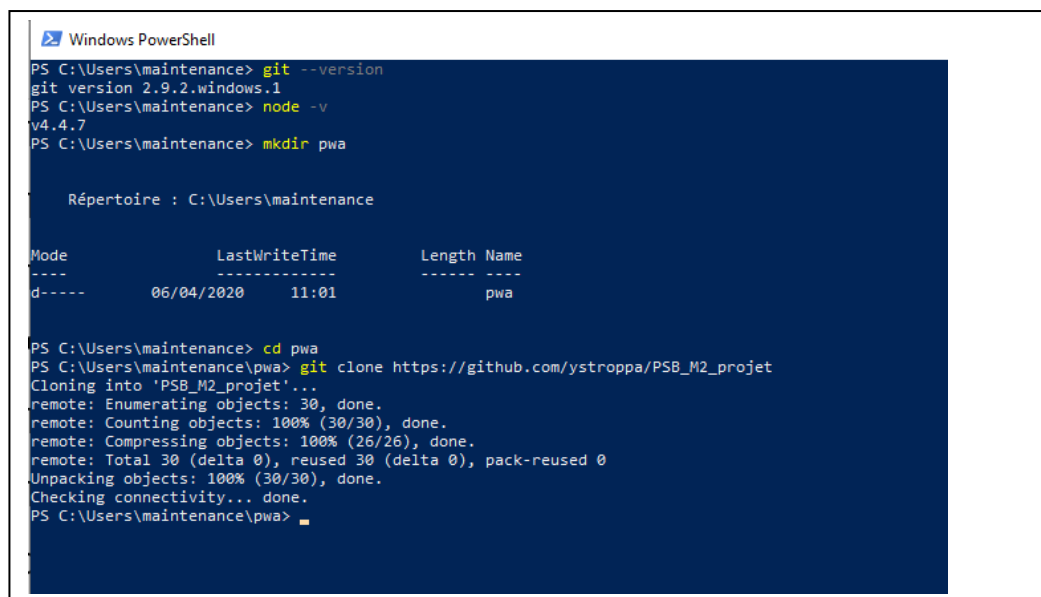
On va tout d'abord créer un nouveau répertoire pour déposer les éléments à l'aide des commandes suivantes :

mkdir pwa

cd pwa

git clone https://github.com/ystroppa/PSB_M2_projet

Ce qui nous donne



```
Windows PowerShell
PS C:\Users\maintenance> git --version
git version 2.9.2.windows.1
PS C:\Users\maintenance> node -v
v4.4.7
PS C:\Users\maintenance> mkdir pwa

Répertoire : C:\Users\maintenance

Mode                LastWriteTime         Length Name
----                -
d-----          06/04/2020   11:01         pwa

PS C:\Users\maintenance> cd pwa
PS C:\Users\maintenance\pwa> git clone https://github.com/ystroppa/PSB_M2_projet
Cloning into 'PSB_M2_projet'...
remote: Enumerating objects: 30, done.
remote: Counting objects: 100% (30/30), done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 30 (delta 0), reused 30 (delta 0), pack-reused 0
Unpacking objects: 100% (30/30), done.
Checking connectivity... done.
PS C:\Users\maintenance\pwa>
```

Ensuite il faut se déplacer sous le répertoire nouvellement créé
cd PSB_M2_projet

Et exécutez la commande **npm install**

```
etag@1.8.1
--
encodeurl@1.0.2
--
path-to-regexp@0.1.7
--
range-parser@1.2.1
--
cookie@0.4.0
--
serve-static@1.14.1
--
safe-buffer@5.1.2
--
cookie-signature@1.0.6
--
merge-descriptors@1.0.1
--
content-disposition@0.5.3
--
setprototypeof@1.1.1
--
statuses@1.5.0
--
depd@1.1.2
--
qs@6.7.0
--
on-finished@2.3.0 (ee-first@1.1.1)
--
finalhandler@1.1.2 (unpipe@1.0.0)
--
proxy-addr@2.0.6 (forwarded@0.1.2, ipaddr.js@1.9.1)
--
type-is@1.6.18 (media-typer@0.3.0, mime-types@2.1.26)
--
send@0.17.1 (ms@2.1.1, destroy@1.0.4, mime@1.6.0, http-errors@1.7.3)
--
accepts@1.3.7 (negotiator@0.6.2, mime-types@2.1.26)
--
body-parser@1.19.0 node_modules\body-parser
--
content-type@1.0.4
--
bytes@3.1.0
--
depd@1.1.2
--
qs@6.7.0
--
on-finished@2.3.0 (ee-first@1.1.1)
--
raw-body@2.4.0 (unpipe@1.0.0)
--
http-errors@1.7.2 (toidentifier@1.0.0, inherits@2.0.3, setprototypeof@1.1.1, statuses@1.5.0)
--
type-is@1.6.18 (media-typer@0.3.0, mime-types@2.1.26)
--
iconv-lite@0.4.24 (safer-buffer@2.1.2)
--
jade@1.11.0 node_modules\jade
--
commander@2.6.0
--
character-parser@1.2.1
--
void-elements@2.0.1
--
mkdirp@0.5.5 (minimist@1.2.5)
--
jstransformer@0.0.2 (is-promise@2.1.0, promise@6.1.0)
--
constantinople@3.0.2 (acorn@2.7.0)
--
with@4.0.3 (acorn@1.2.2, acorn-globals@1.0.9)
--
clean-css@3.4.28 (source-map@0.4.4, commander@2.8.1)
--
transformers@2.1.0 (promise@2.0.0, css@1.0.8, uglify-js@2.2.5)
--
uglify-js@2.8.29 (uglify-to-browserify@1.0.2, source-map@0.5.7, yargs@3.10.0)
PS C:\Users\maintenance\pwa\PSB_M2_projet>
```

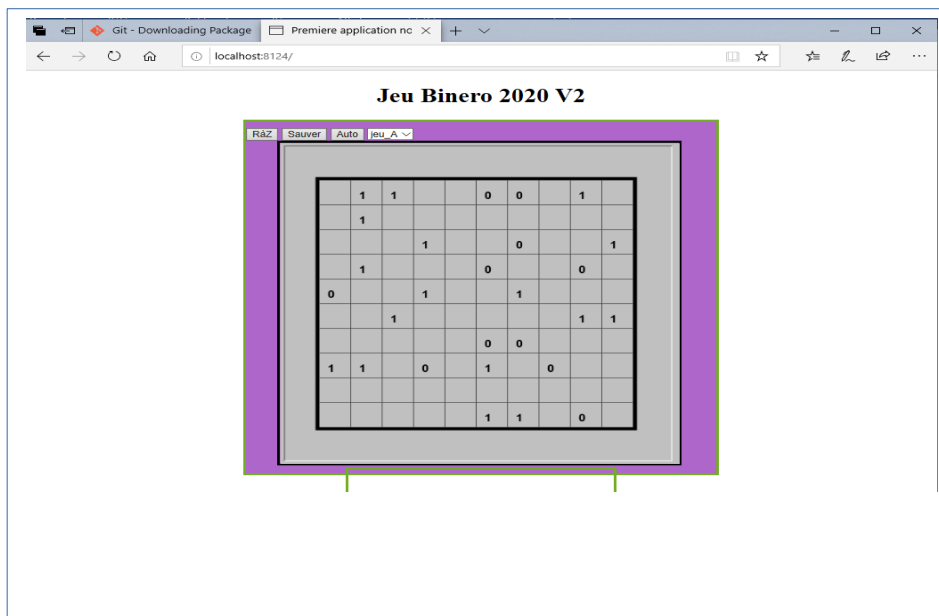
Une fois les modules d'installés, il vous suffit d'exécuter la commande **npm start**

```
PS C:\Users\maintenance\pwa\PSB_M2_projet> npm start

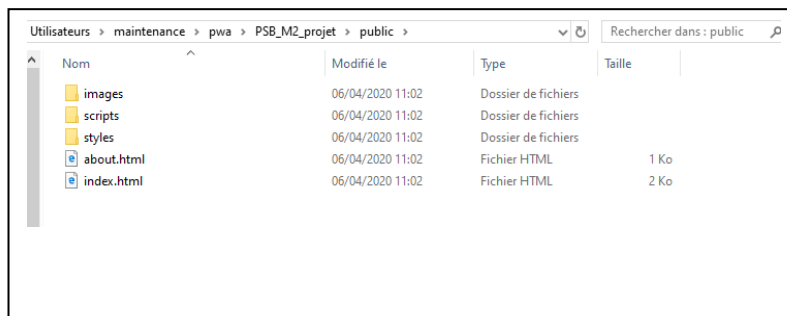
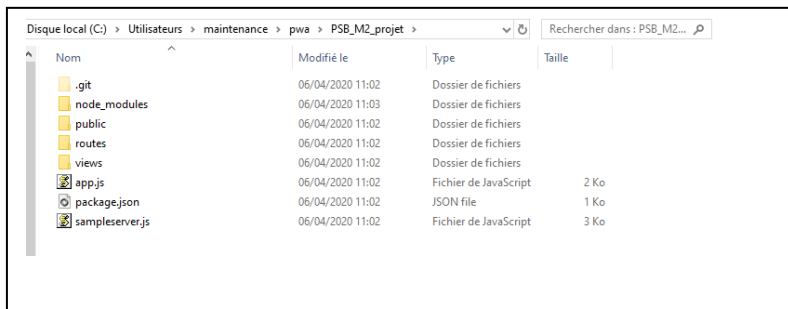
> caslpWA@1.0.0 start C:\Users\maintenance\pwa\PSB_M2_projet
> node ./sampleserver.js

Express OK
Creation du serveur ... 8124
Done
```

A partir d'un navigateur se connecter sur l'url
<http://localhost:8124>



Pour travailler sur l'application, à partir de l'explorateur de fichiers vous pouvez vous déplacer sur le répertoire



Prendre l'énoncé déposé sur github https://github.com/ystroppa/M2_PWA_Travaux_pratique_2.pdf et vous avez la démarche à suivre et à compléter pour aménager cette application en progressive web.