

*Beste de savoir*

Vue.js + ASP.NET Core + Webpack

---

21 décembre 2018



# Table des matières

1.	Introduction . . . . .	1
2.	Créer le projet . . . . .	1
3.	Mon projet ne compile pas! . . . . .	2
4.	En conclusion . . . . .	8

## 1. Introduction

Je me suis lancé il y a peu dans un projet pour un ami, afin de l'aider à automatiser certaines tâches. Comme je suis une quiche en frontend, je me suis dit que c'était l'occasion d'essayer un framework JavaScript. J'ai mis de côté les deux poids-lourds Angular et React pour jeter mon dévolu sur Vue.js. Pourtant, j'ai galéré comme un veau à avoir un projet qui compile et qui soit fluide.

Je vais donc vous faire un **retour d'expérience** pour que, dans le cas où vous seriez dans la même situation que moi, vous puissiez moins galérer.

## 2. Créer le projet

Comme je l'ai dit, je voulais utiliser .NET Core. Je suis donc parti sur un back Web API et un front utilisant Vue.js et TypeScript. Pour le back, aucun problème. Seulement voilà, même avec Visual Studio 2017, il n'y a **pas de template pour créer un front utilisant Vue.js**. Bien sûr, il y a Angular et React, mais pas celui que je cherche.

Pas de soucis, on va chercher sur Internet. On tombe sur cette commande pour installer les templates [SPA](#).

```
1 dotnet new --install Microsoft.AspNetCore.SpaTemplates::*
```

Il y a bien un template Vue.js, mais celui-ci est complètement dépassé : Webpack 2 (aujourd'hui 4), jQuery 2 (aujourd'hui 3), TypeScript 2 (aujourd'hui 3), etc. Le projet compile, mais dès que je tente de mettre à jour les dépendances NPM, **c'est la catastrophe**, tout s'effondre. Le code s'exécute, mais je me prends une volée d'erreurs à l'exécution, dont même une recherche Google n'a pas été capable de m'aider.

### 3. Mon projet ne compile pas !

```
1 // Des dépendances pas à jour.
2
3 {
4   "name": "Site",
5   "private": true,
6   "version": "0.0.0",
7   "devDependencies": {
8     "@types/webpack-env": "^1.13.0",
9     "aspnet-webpack": "^2.0.1",
10    "awesome-typescript-loader": "^3.0.0",
11    "bootstrap": "^3.3.6",
12    "css-loader": "^0.25.0",
13    "event-source-polyfill": "^0.0.7",
14    "extract-text-webpack-plugin": "^2.0.0-rc",
15    "file-loader": "^0.9.0",
16    "isomorphic-fetch": "^2.2.1",
17    "jquery": "^3.1.1",
18    "style-loader": "^0.13.1",
19    "typescript": "^2.2.1",
20    "url-loader": "^0.5.7",
21    "vue": "^2.2.2",
22    "vue-loader": "^11.1.4",
23    "vue-property-decorator": "^5.0.1",
24    "vue-router": "^2.3.0",
25    "vue-template-compiler": "^2.2.2",
26    "webpack": "^2.2.0",
27    "webpack-hot-middleware": "^2.12.2"
28  }
29 }
```

Alors on continue et on tombe sur [ce projet](#) . Un nouveau template à installer. Mais celui-ci n'est que du JavaScript pur, ce qui ne m'arrange pas. Et en jetant un œil aux dépendances, elles ne sont pas vraiment plus à jour. Mince.

Je tombe sur [cet article](#) qui va sans doute m'aider. Alors je suis les instructions, mais l'article n'est pas clair, je galère, je n'y arrive pas. Je me tape toujours la même exception.

Puis, après encore d'autres recherches, je tombe sur [ce projet](#) . Une lueur d'espoir se rallume. Je suis les instructions, je crée un projet pour tester, il marche, les dépendances ne sont pas trop vieilles, **je suis sur le bon chemin.**

### 3. Mon projet ne compile pas !

Je vais donc créer mon projet moi-même et m'inspirer de ce qu'a fait l'auteur du template pour compléter ce qui me manque et ainsi avoir un projet fonctionnel et personnalisé. Je crée d'abord un projet ASP.NET Core vide, que je vais remplir au fur et à mesure. Et, évidemment, je rencontre plusieurs erreurs que je ne voyais pas, même avec le template le plus à jour.

### 3. Mon projet ne compile pas !

```
1 You are using the runtime-only build of Vue where the template compiler is not available.
```

Une recherche assez rapide m'indique une ligne à rajouter dans le fichier de configuration de Webpack. Un alias

```
1 module.exports = {
2   ...
3   resolve: {
4     ...
5     alias: {
6       'vue$': 'vue/dist/vue.esm.js'
7     }
8     ...
9   }
10  ...
11 }
```

```
1 TypeScript : Cannot find module './App'.
```

Précisons que cette erreur a lieu alors même que le module est accessible et dans le même dossier que le fichier `.ts` en cours d'édition. Apparemment, TypeScript n'aime pas trop les fichiers `.vue`. On règle ce problème en créant un fichier `vue-shims.d.ts`, à la racine du projet.

```
1 declare module "*.vue" {
2   import Vue from "vue";
3   export default Vue;
4 }
```

Et, comme j'utilise `vue-loader` dans sa dernière version, j'ai eu le droit à une belle erreur due à des modifications dans la v15 de ce package.

```
1 Vue loader was used without the corresponding plugin. Make sure to include VueLoaderPlugin in your webpack config.
```

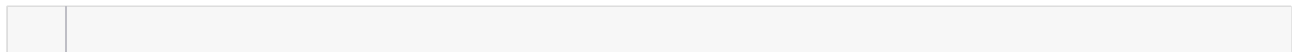
Bon, il faut donc importer ce plugin et puis l'ajouter à la liste correspondante.

### 3. Mon projet ne compile pas !

```
1  const VueLoaderPlugin = require('vue-loader/lib/plugin');
2
3  ...
4
5  module.exports = {
6    ...
7    plugins: [
8      new VueLoaderPlugin(),
9      ...
10   ]
11   ...
12 }
```

Maintenant, le projet compile et s'exécute, mais survient un problème qui m'embête : **le Hot Module Replacement ne fonctionne qu'une seule et unique fois**. Ce module permet de recharger les fichiers `.vue` (entre autres) au moindre `Ctrl+S`, au lieu d'un `F5` ou d'une recompilation. Pourtant, sur l'autre projet, ça marche. Alors, où est le problème ?

Ce n'est pas le fichier `Startup.cs`, puisque le HMR est bien activé.



Peut-être qu'en changeant les ports de l'application ?

```
1  {
2    "iisSettings": {
3      "windowsAuthentication": false,
4      "anonymousAuthentication": true,
5      "iisExpress": {
6        "applicationUrl": "http://localhost:51192/",
7        "sslPort": 0
8      }
9    },
10   "profiles": {
11     "IIS Express": {
12       "commandName": "IISExpress",
13       "launchBrowser": true,
14       "environmentVariables": {
15         "ASPNETCORE_ENVIRONMENT": "Development"
16       }
17     },
18     "vuets": {
19       "commandName": "Project",
20       "launchBrowser": true,
21       "environmentVariables": {
22         "ASPNETCORE_ENVIRONMENT": "Development"
23       }
24     }
25   }
26 }
```

### 3. Mon projet ne compile pas !

```
23     },
24     "applicationUrl": "http://localhost:51193/"
25   }
26 }
27 }
```

Décidément, d'où vient ce problème. Aucune des solutions que j'ai trouvé sur Internet n'a fonctionné pour moi. Suis-je condamné à m'en passer ? En désespoir de cause, je modifie mon fichier `.csproj`. Sait-on jamais...

Voici celui avant.

```
1 <Project Sdk="Microsoft.NET.Sdk.Web">
2
3   <PropertyGroup>
4     <TargetFramework>netcoreapp2.2</TargetFramework>
5     <AspNetCoreHostingModel>InProcess</AspNetCoreHostingModel>
6     <TypeScriptToolsVersion>3.1</TypeScriptToolsVersion>
7   </PropertyGroup>
8
9   <ItemGroup>
10    <PackageReference Include="Microsoft.AspNetCore.All" />
11    <PackageReference Include="Microsoft.AspNetCore.App" />
12    <PackageReference Include="Microsoft.AspNetCore.Razor.Design"
13      Version="2.2.0" PrivateAssets="All" />
14    <PackageReference Include="Microsoft.AspNetCore.SpaServices"
15      Version="2.2.0" />
16    <PackageReference
17      Include="Microsoft.AspNetCore.SpaServices.Extensions"
18      Version="2.2.0" />
19    <PackageReference
20      Include="Microsoft.VisualStudio.Web.CodeGeneration.Design"
21      Version="2.2.0" />
22  </ItemGroup>
23
24  <ItemGroup>
25    <Folder Include="wwwroot\" />
26  </ItemGroup>
27
28  <Target Name="DebugRunWebpack" BeforeTargets="Build"
29    Condition="'$(Configuration)' == 'Debug' And !Exists('wwwroot\dist') ">
30    <!-- Ensure Node.js is installed -->
31    <Exec Command="node --version" ContinueOnError="true">
32      <Output TaskParameter="ExitCode" PropertyName="ErrorCode" />
33    </Exec>
34    <Error Condition="'$(ErrorCode)' != '0'"
35      Text="Node.js is required to build and run this project. To continue, p
36    />
```

### 3. Mon projet ne compile pas !

```
28
29     <!-- In development, the dist files won't exist on the first
30          run or when cloning to
31          a different machine, so rebuild them if not already
32          present. -->
33     <Message Importance="high"
34             Text="Performing first-run Webpack build..." />
35     <Exec
36         Command="node node_modules/webpack/bin/webpack.js --config webpack.conf" />
37     <Exec Command="node node_modules/webpack/bin/webpack.js" />
38 </Target>
39
40 <Target Name="PublishRunWebpack"
41     AfterTargets="ComputeFilesToPublish">
42     <!-- As part of publishing, ensure the JS resources are freshly
43          built in production mode -->
44     <Exec Command="npm install" />
45     <Exec
46         Command="node node_modules/webpack/bin/webpack.js --config webpack.conf" />
47     <Exec
48         Command="node node_modules/webpack/bin/webpack.js --env.prod" />
49
50     <!-- Include the newly-built files in the publish output -->
51     <ItemGroup>
52         <DistFiles Include="wwwroot\dist\*" />
53         <ResolvedFileToPublish Include="@{(DistFiles->'%(FullPath)')"
54             Exclude="@{(ResolvedFileToPublish)}">
55             <RelativePath>%(DistFiles.Identity)</RelativePath>
56
57             <CopyToPublishDirectory>PreserveNewest</CopyToPublishDirectory>
58         </ResolvedFileToPublish>
59     </ItemGroup>
60 </Target>
61 </Project>
```

Et le voici après.

```
1 <Project Sdk="Microsoft.NET.Sdk.Web">
2
3 <PropertyGroup>
4 <TargetFramework>netcoreapp2.2</TargetFramework>
5 <TypeScriptCompileBlocked>>true</TypeScriptCompileBlocked>
6 <TypeScriptToolsVersion>Latest</TypeScriptToolsVersion>
7 <IsPackable>>false</IsPackable>
```



### 3. Mon projet ne compile pas !

```
8 </PropertyGroup>
9
10 <ItemGroup>
11 <PackageReference Include="Microsoft.AspNetCore.App" />
12 </ItemGroup>
13
14
15 <Target Name="DebugRunWebpack" BeforeTargets="Build"
16 <Condition=" '$(Configuration)' == 'Debug' And !Exists('wwwroot\dist') ">
17 <!-- Ensure Node.js is installed -->
18 <Exec Command="node --version" ContinueOnError="true">
19 <Output TaskParameter="ExitCode" PropertyName="ErrorCode" />
20 </Exec>
21 <Error Condition="'$(ErrorCode)' != '0'"
22 <Text="Node.js is required to build and run this project. To continue, p
23 />
24
25 <!-- In development, the dist files won't exist on the first
26 run or when cloning to
27 a different machine, so rebuild them if not already
28 present. -->
29 <Message Importance="high"
30 <Text="Performing first-run Webpack build..." />
31 <Exec
32 <Command="node node_modules/webpack/bin/webpack.js --config webpack.conf
33 />
34 <Exec Command="node node_modules/webpack/bin/webpack.js" />
35 </Target>
36
37 <Target Name="PublishRunWebpack"
38 <AfterTargets="ComputeFilesToPublish">
39 <!-- As part of publishing, ensure the JS resources are freshly
40 built in production mode -->
41 <Exec Command="npm install" />
42 <Exec
43 <Command="node node_modules/webpack/bin/webpack.js --config webpack.conf
44 />
45 <Exec
46 <Command="node node_modules/webpack/bin/webpack.js --env.prod"
47 />
48
49 <!-- Include the newly-built files in the publish output -->
50 <ItemGroup>
51 <DistFiles Include="wwwroot\dist\*" />
52 <ResolvedFileToPublish Include="@((DistFiles->'%(FullPath)')"
53 <Exclude="@((ResolvedFileToPublish))">
54 <RelativePath>%((DistFiles.Identity))</RelativePath>
55
56 <CopyToPublishDirectory>PreserveNewest</CopyToPublishDirectory>
57 </ResolvedFileToPublish>
```

#### 4. En conclusion

```
42     </ItemGroup>
43 </Target>
44
45 </Project>
```

Eh bien, aussi bizarre que ça soit, **ça marche** !

Est-ce un Nuget supprimé ? Est-ce la ligne `<AspNetCoreHostingModel>InProcess</AspNetCoreHostingModel>` qui empêchait le HMR de fonctionner ? J'avoue que je n'en sais pas grand chose. Par contre, maintenant, mon projet compile, s'exécute et le HMR est fonctionnel.

## 4. En conclusion

Arf, j'ai bien galéré. Autant le back, ça a roulé nickel chrome tout seul, autant le front j'en ai douillé comme pas possible. Et je suis entièrement d'accord avec @SpaceFox sur [la folie furieuse](#) de la communauté.

Y'a un moment faut **arrêter de délirer**. Tous les millions de packages pour une appli simple, chaque mise à jour qui peut potentiellement déprécier la moitié de ton code et donc te forcer à pleins de modification sans aucune valeur ajoutée, les erreurs cryptiques, les documentations pas à jour et donc un code marche chez ton voisin mais plus chez toi, etc.

Voilà, j'espère que si, vous aussi, galérez comme pas permis, j'ai pu vous aider un peu. Bonne journée bien zestueuse.

# Liste des abréviations

**SPA** Single Page Application.. 1