Ajout d’un nouveau fichier de traduction

1. Création du fichier JSON

Le système de traduction de l’application est basé sur l’utilisation de fichiers au format JSON. Ces fichiers contiennent des chaines de caractères en anglais, qu’il faut ensuite traduire dans la langue souhaitée.

Voici une capture d’écran du fichier de traduction « fr.json » permettant la traduction de l’application web en français :

Une image contenant texte

Description générée automatiquement

Comme affiché, le fichier débute par une accolade ouvrant « { »  et se termine par une accolade fermante « } ». Vous retrouverez sur chaque ligne à gauche, la chaine de caractère à traduire, puis sa traduction dans la langue souhaitée à droite. A noter qu’il peut s’agire de mots ou de suites de mots.

Chaque ligne se termine par une virgule **sauf la dernière** **qui ne se termine par rien**.

Le respect de la syntaxe est primordial au bon fonctionnement de l’application. Veillez donc à relire le nouveau fichier pour éviter d’éventuelles fautes de frappes.

Pour créer votre nouveau fichier de traduction, il suffit donc de dupliquer une traduction existante et de remplacer les chaines de caractères traduites (situées à droite des « : » ) par de nouvelles dans la langue souhaitée.

Les traductions existantes sont situées dans le dossier **FINAL\_UI/src/locales**

Vous retrouverez ici la liste des langues acceptées : <http://www.iana.org/assignments/language-subtag-registry/language-subtag-registry>

Le nouveau fichier devra porter le nom correspondant au « Subtag » de la langue choisie comme par exemple « fr.json » pour le français ou « en.json » pour l’anglais.

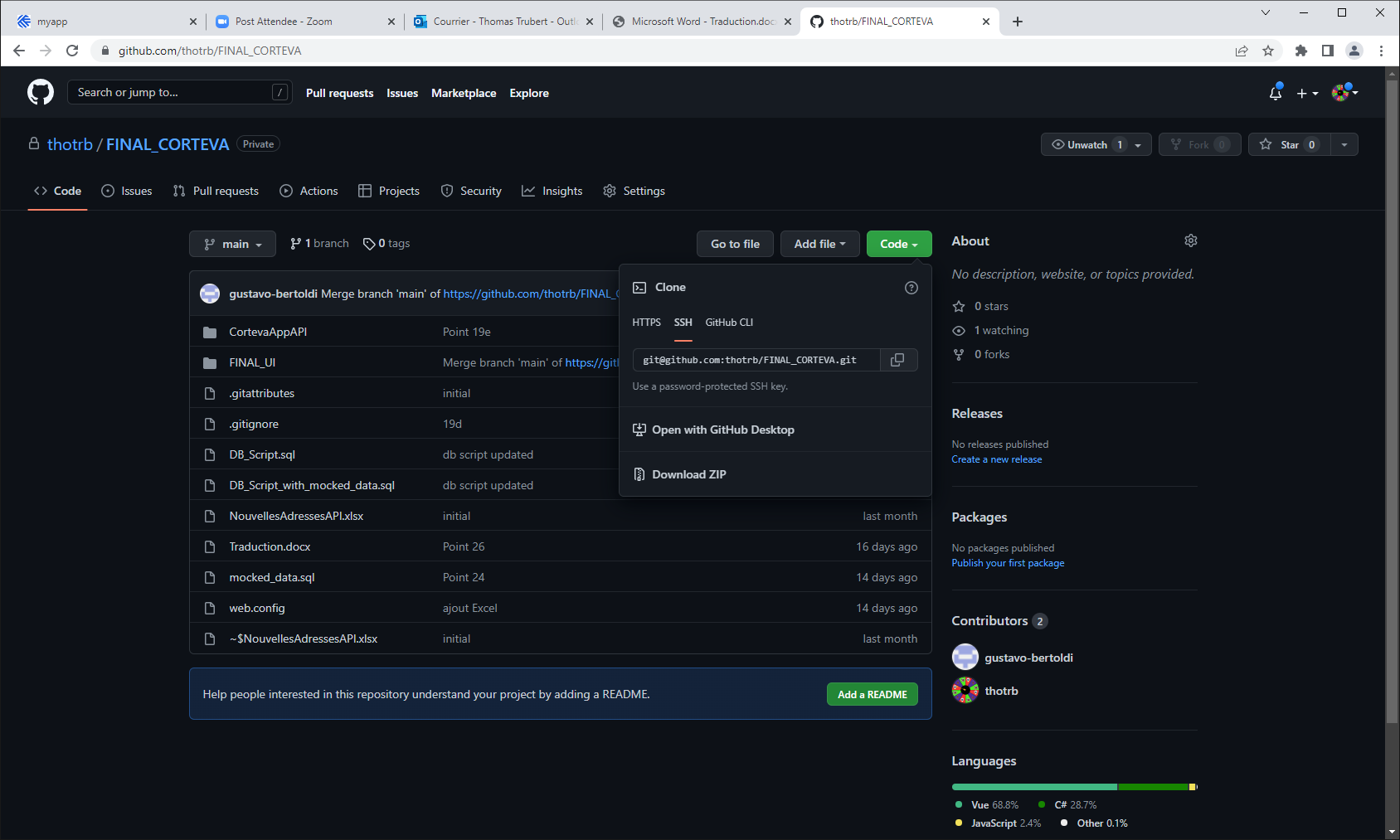
Il vous suffira donc de renommer le nouveau fichier.

Pour éditer ou créer des fichiers JSON, il est conseillé d’utiliser un éditeur de texte comme Bloc-notes, Notepad++ ou Visual Studio Code par exemple et non la suite Office ou Libre Office, qui ne sont pas adaptés.

1. Déploiement du fichier dans l’application

Pour récupérer le code source du projet vous pouvez accéder au dépôt Git du projet sur le lien suivant et il faut cliquer sur « Download ZIP » :

<https://github.com/thotrb/FINAL_CORTEVA>



Une fois le fichier créé JSON, il faut ensuite le déployer dans l’application. Le dossier prévu à cet effet est le suivant **FINAL\_UI/src/locales**. Vous y retrouverez déjà au moins deux fichiers : fr.json et en.json correspondants respectivement aux traductions françaises et anglaises.

Vous n’avez donc qu’à copier votre nouveau fichier JSON de traduction à cet endroit.

Ensuite, dans les fichiers **FINAL\_UI/src/components/UserInputComponents/NavbarSaisie.vue** et **FINAL \_UI/src/components/DashboardComponents/Navbar.vue**, une modification sera nécessaire.   
Voici une capture d’écran de ces fichiers :

Une image contenant texte

Description générée automatiquement

Il faudra donc insérer dans la zone entourée en rouge une option supplémentaire pour permettre à l’utilisateur de sélectionner le nouveau langage ajouté en respectant le modèle suivant :

Une image contenant texte

Description générée automatiquement



<option value="<Subtag de la nouvelle langue>" >  
 <Subtag de la nouvelle langue>  
</option>

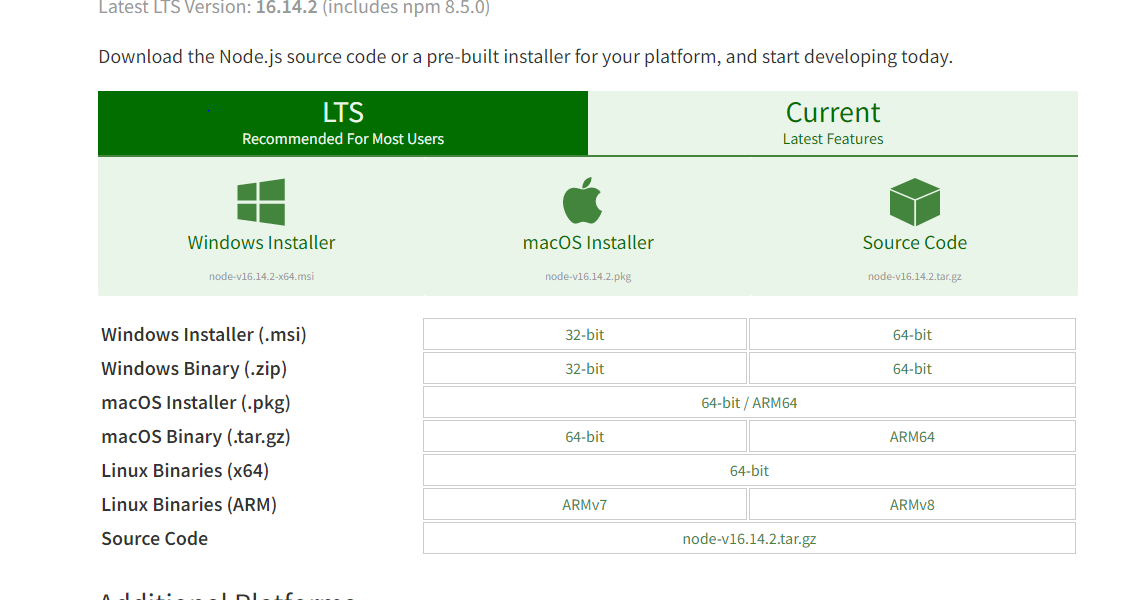
Il suffira ensuite de recompiler le projet pour que les modifications soient prises en compte.

Si NodeJs n’est pas installé sur votre machine, il faudra l’installer.

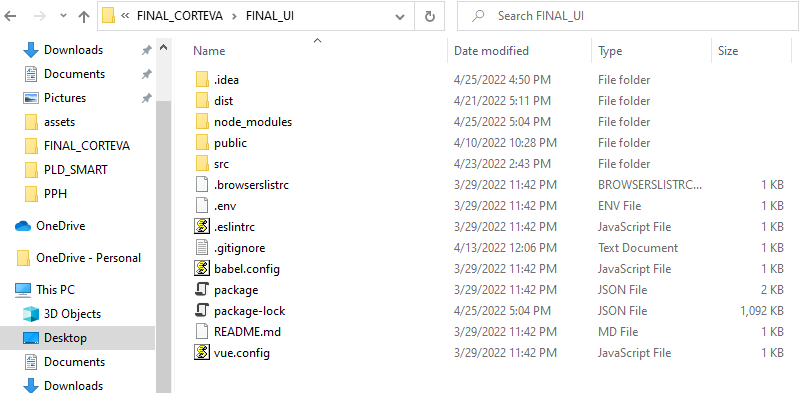
Pour cela, il suffit de ce rendre sur ce lien :

<https://nodejs.org/en/download/>

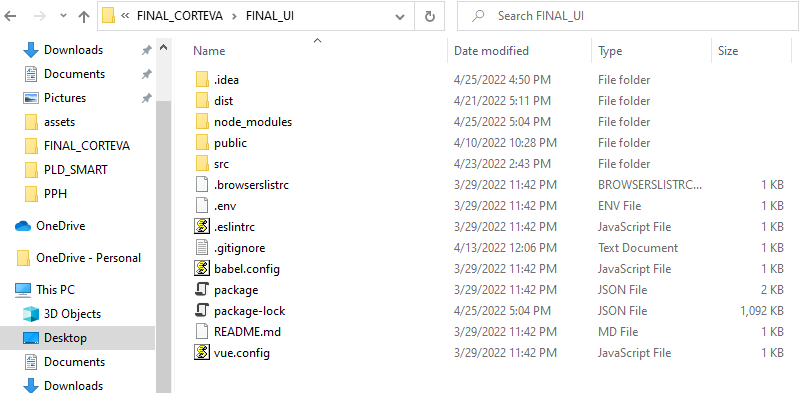
Il faut ensuite sélectionner la version adaptée à votre système d’exploitation (Windows, MacOS, ...) en cliquant sur le logo de votre choix comme indiqué.



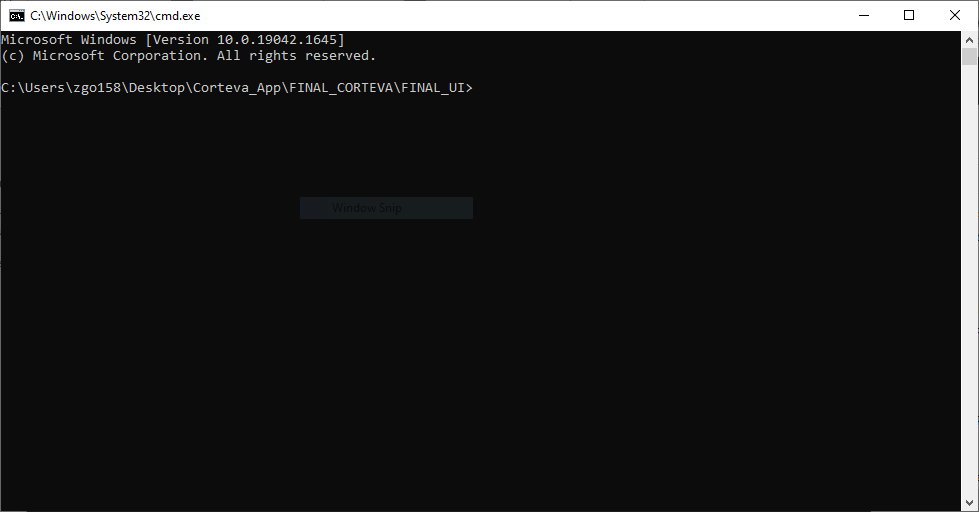
Une fois l’installation passée, il faut ouvrir le projet dans votre explorateur de fichiers en vous rendant dans le dossier **FINAL\_UI**



Ensuite, vous devez ouvrir votre terminal à cet endroit en remplaçant la zone entourée en rouge par « cmd » :



Un terminal devrait alors s’ouvrir :

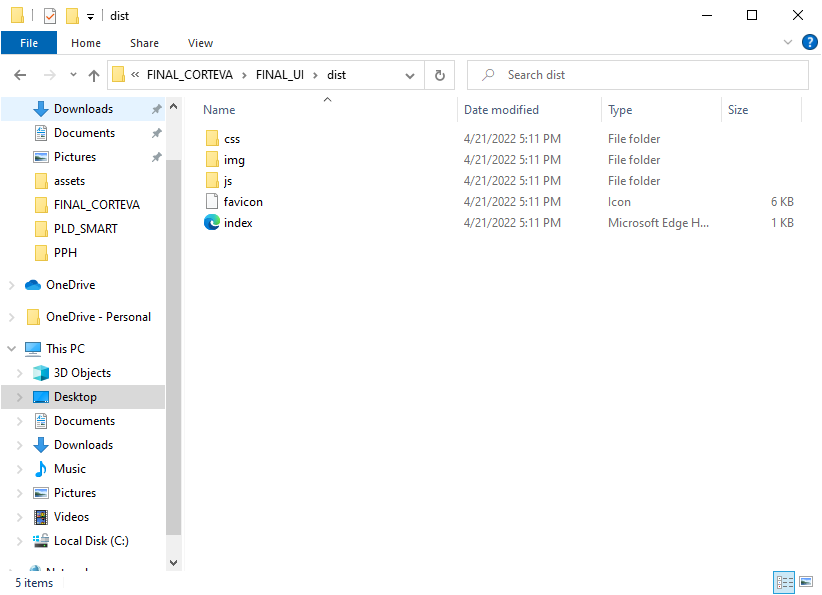


Il suffit alors de taper les commandes suivantes et attendre qu’elles soient achevées :

« npm install » puis « npm run build »

Le projet est alors recompilé et prêt à être exporté sur le serveur.

Il faut copier les dossiers « **css** », « **img** », « **js** » et les fichiers « **favicon** » et « **index** » contenus dans le dossier **FINAL\_UI/dist**



En suite il faut se connecter à distance au serveur et se rendre dans le dossier : **F:\wwwroot\myvueapp** sur le serveur. Vous pouvez alors remplacer les dossiers et fichiers « **css** », « **img** », « **js** », « **favicon** » et « **index** » déjà présents par ceux nouvellement créés.

**Attention à ne pas supprimer le fichier web.config présent sur le serveur !**

Les modifications sont désormais prises en compte sur le serveur et le nouveau langage est alors disponible.