

## TP Création de Portfolio

## Sommaire

|                                       |   |
|---------------------------------------|---|
| 1) Problématique.....                 | 1 |
| 2) Analyse et préparation .....       | 1 |
| 3) Réalisation .....                  | 1 |
| 4) Mise en ligne .....                | 3 |
| 5) Modification à l'aide de Git ..... | 4 |
| 6) Conclusion.....                    | 6 |

### 1) Problématique

Nous souhaitons savoir comment créer un portfolio en partant de zéro, pour cela on va utiliser un guide, l'IA et plusieurs logiciels afin de pouvoir y parvenir.

### 2) Analyse et préparation

Tout d'abord, pour mener le projet à bien on commence par créer un dossier, dans lequel on regroupera tous les fichiers nécessaires à la création du portfolio.

Dans mon cas je l'ai appelé :

 Portfolio SIO1

Le guide nous demande de mettre un CV et une photo de profil sur notre portfolio, donc il va d'abord falloir créer un CV à l'aide d'un site, comme Canva par exemple, puis ajouter un fichier jpg d'une photo de profil sérieuse de nous.

### 3) Réalisation

Ensuite, on nous dit qu'il faut utiliser le prompt suivant :

« Je joins mon CV ici ; utilisez-le comme référence complète. J'ai une photo enregistrée localement et un fichier CV dans mon dossier, et je vous demande de créer pour moi un site web de portfolio personnel de haute qualité et remarquable, utilisant HTML, CSS, JavaScript et Bootstrap 5. Le site doit être entièrement autonome dans un seul fichier HTML (avec tous les CSS et JS intégrés) et inclure des sections essentielles telles que Accueil, À propos de moi, Compétences (présentées visuellement), Projets (avec images et fenêtres contextuelles pour plus de détails), Expérience, Contact, des liens vers les réseaux sociaux et un bouton de téléchargement de CV qui déclenche un téléchargement immédiat sans être bloqué par le navigateur. Utilisez des images de substitution de haute qualité provenant d'Unsplash pour les projets, assurez-vous que tout est bien formulé dans un français correct et que le design général soit professionnel, soigné et visuellement attrayant pour me démarquer. ».

Il faut l'ajouter sur une IA pour pouvoir créer la page html, nous allons utiliser Deepseek car cette IA n'a pas de limite de caractère lorsqu'elle répond.

Après avoir donné le prompt à Deepseek, il va comment à créer le code permettant d'afficher le portfolio :

```
html

<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Alexandre Ewan - Développeur Web</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.10.0/font/bootstrap-icons.css">
  <style>
    :root {
      --primary-color: #4e54c8;
      --secondary-color: #8f94fb;
      --dark-color: #1a1a2e;
      --light-color: #f8f9fa;
      --accent-color: #ff6b6b;
    }

    body {
```

A partir de là il faut télécharger le fichier html et l'appeler « index.html » puis l'ajouter au dossier.



Maintenant il suffit simplement de modifier la page HTML avec Notepad++ pour changer la localisation du répertoire de la photo de profil :

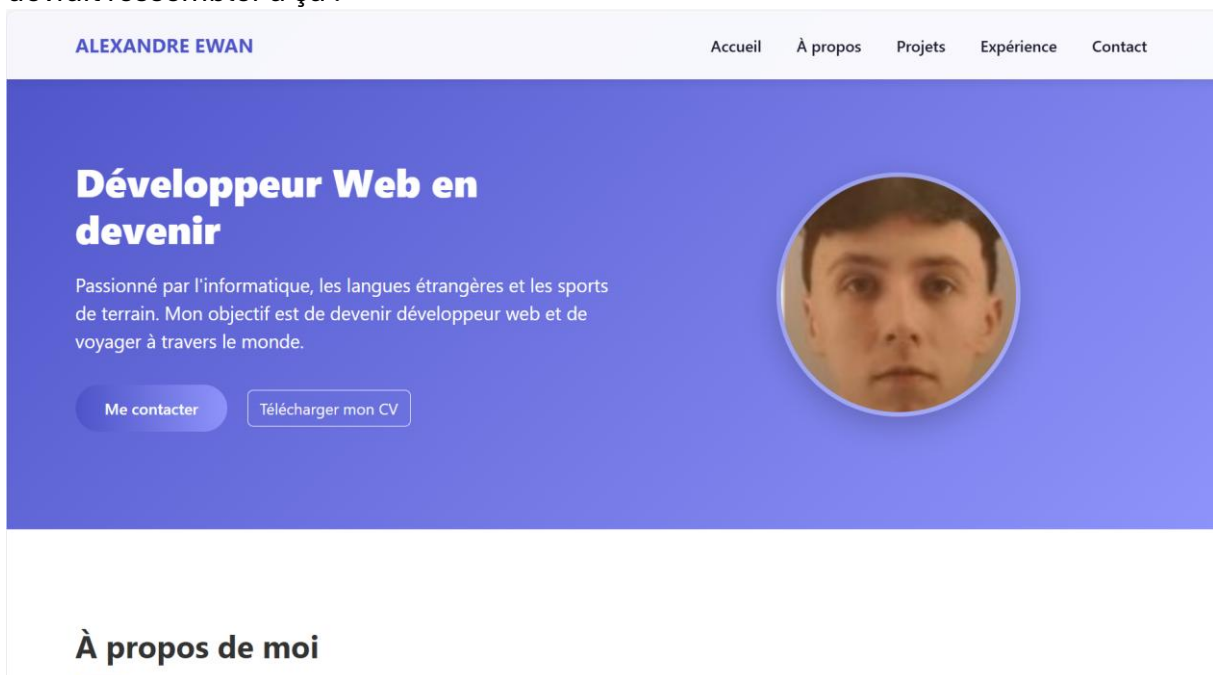
```

```

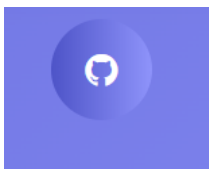
Et du CV :

```
const projects = [
  {
    name: "TP01 - BEAUVALLET",
    description: "Travail pratique : résolution de labyrinthes avec des blocs de code (style Scratch).",
    image: "https://studio.code.org/script_assets/k_1_images/maze_images/maze_WS.png",
    file: "TP01_BEAUVALLET.pdf",
    type: "pdf"
  },
];
```

puis possiblement de modifier les détails erronés donnés par Deepseek. Le résultat devrait ressembler à ça :



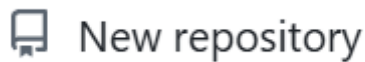
Deepseek a créé un bouton qui permet de télécharger le CV, il y a une barre permanente en haut de la page qui nous amène sur la partie du site en rapport avec le bouton, par exemple quand je clique sur Contact, cela va m'amener tout en bas du site sur lequel figure un bouton qui ramène sur mon profil GitHub :



#### 4) Mise en ligne

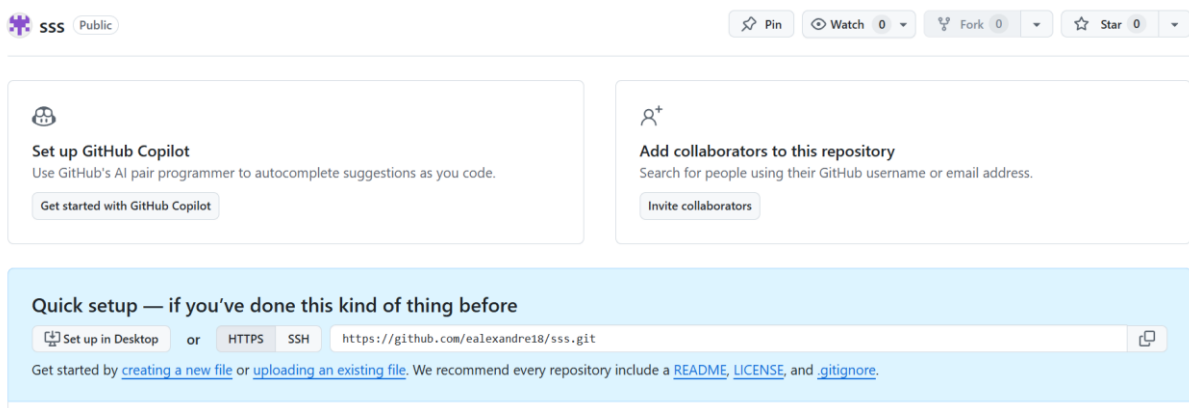
Lorsque le site est bien complet, on peut désormais passer à sa mise en ligne, il faut d'abord créer un compte GitHub, puis un compte Netlify sur lequel on se connectera en utilisant « Connexion avec Github ».

Lorsque cela est fait, on se connecte à Github, ensuite il faut créer un dépôt (repository) :



On y ajoute un titre, une description (possiblement) et on laisse la visibilité en publique par défaut.

Le dépôt devrait ressembler à ça :




Ensuite on clique sur « uploading an existing file » et on téléverse le cv, la photo, l'index, et dans mon cas aussi les projets en .pdf. Cela permettra à Netlify de pouvoir accéder à ces fichiers pour les afficher, comme en local.

Ensuite on se connecte simplement sur Netlify avec le compte github, ensuite on clique sur « add new site », puis on importe notre projet et on sélectionne notre dépôt, cela va automatiquement créer un site en .netlify.app, et on peut ensuite changer dans les paramètres le nom de domaine afin de mettre le nom qu'on veut.

Dans mon cas, je l'ai changé portfolioewanalexandre.netlify.app

## 5) Modification à l'aide de Git

A partir de maintenant, si l'on veut pouvoir modifier rapidement son dépôt GitHub, il va falloir installer le logiciel Git  qui va permettre de pouvoir enregistrer toutes les modifications apportées au dossier du portfolio.

Tout d'abord on va créer un nouveau dépôt appelé mon-portfolio-git-commands :

## ealexandre18 / mon-portfolio-git-commands

Ensuite, pour pouvoir enregistrer les modifications du dossier portfolio, nous allons utiliser cmd, mais cmd ne va pas pouvoir modifier le dossier si on ne lui indique pas la localisation, il va donc falloir utiliser la commande cd + localisation du répertoire du portfolio, dans notre cas ça sera ceci :

```
C:\Windows\System32>cd C:\Users\Ewan Alexandre\Documents\Portfolio SI01
```

Et cela va donc nous laisser avec ceci :

```
C:\Users\Ewan Alexandre\Documents\Portfolio SI01>
```

Maintenant il va falloir initialiser Git dans notre dossier, donc dans notre cas nous allons simplement écrire la ligne de commande « git init » ce qui va créer un dépôt Git et un fichier caché .git dans notre dossier.

Ensuite nous allons donc pouvoir effectuer les commandes git sur le dossier portfolio.

Pour pouvoir continuer il va falloir créer un sorte de compte pour pouvoir ajouter enregistrer des fichiers sur le dépôt avec ces commandes :

```
C:\Users\Ewan Alexandre\Documents\Portfolio SI01>git config --global user.email "ealexandre2804@gmail.com"
C:\Users\Ewan Alexandre\Documents\Portfolio SI01>git config user.name "Ewan Alexandre"
```

Maintenant, nous n'avons besoin que de deux commandes :

git add . et git commit -m.

git add . va ajouter tous les fichiers présents dans le dossier dans le dépôt git, il va falloir le faire à chaque fois que l'on modifie un fichier ou que l'on en ajoute un nouveau dans le dossier.

Git commit -m va enregistrer les modifications du dépôt git :

```
C:\Users\Ewan Alexandre\Documents\Portfolio SI01>git commit -m "ajout de fichiers dans le dépôt"
[master (root-commit) 00fb89a] ajout de fichiers dans le dépôt
 8 files changed, 718 insertions(+)
 create mode 100644 CVSIO.pdf
 create mode 100644 PHOTO.png
 create mode 100644 TP Portfolio.docx
 create mode 100644 TP01_ATELIER.pdf
 create mode 100644 TP01_BEAUVALLET.pdf
 create mode 100644 TP01_SOTOCA.docx
 create mode 100644 TP01_SOTOCA.pdf
 create mode 100644 index.html
```

Et finalement pour faire le lien entre le dépôt Git et le dépôt GitHub, nous allons utiliser les commandes suivantes :

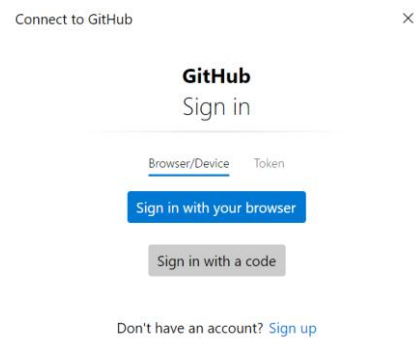
git branch -M main

puis git remote add origin liendudépôtgithub/nomdufichier.git, dans notre cas ça sera :

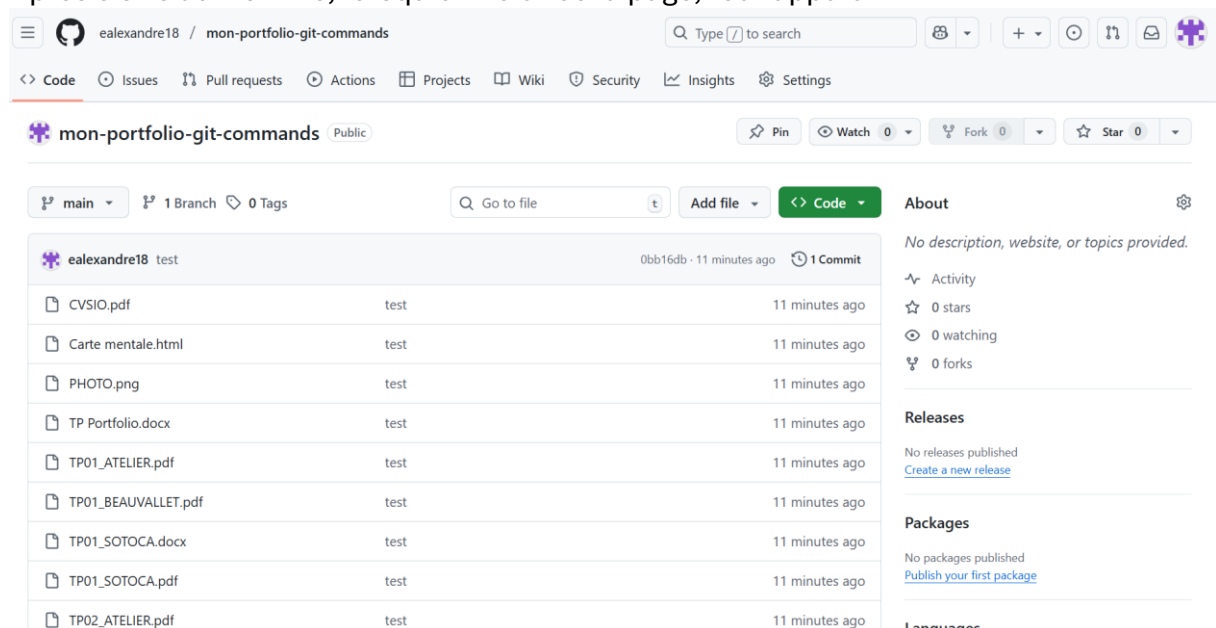
```
C:\Users\Ewan Alexandre\Documents\Portfolio SI01>git remote add origin https://github.com/ealexandre18/mon-portfolio-git-commands.git
```

et enfin git push -u origin main

ensuite il va falloir s'identifier auprès de GitHub :



Après s'être authentifié, lorsqu'on relance la page, tout apparaît :



Et voilà, à partir de maintenant je peux rendre plus rapide la modification du HTML grâce à Git.

## 6) Conclusion

En conclusion, j'ai appris de façon autonome durant ce TP comment créer un portfolio à partir de 0, j'ai appris comment modifier le code pour pouvoir rendre mon site plus pratique, j'ai aussi appris comment créer un site web grâce à GitHub et Netlify et finalement j'ai appris comment rendre plus rapide la modification de mon portfolio grâce aux commandes git.