

Justification des Choix Techniques

1. Organisation du Projet Git

J'ai utilisé Git pour gérer le projet de manière efficace en suivant une structure de branches bien définie. L'organisation repose sur les principes suivants :

- **Branche main** : Branche principale contenant la version stable du projet.
- **Branche feature-1** : Dédiée à l'implémentation de la Hero Section.
- **Branche feature-2** : Dédiée à l'ajout du bouton stylisé sous la phrase.

Cette séparation permet un développement modulaire et facilite l'intégration des fonctionnalités sans impacter directement la branche principale.

2. Workflow Git Adopté

J'ai suivi un workflow basé sur les bonnes pratiques Git :

1. Création du dépôt Git :

```
PS C:\Users\emman\Documents\Keyce\Eval> git init
Initialized empty Git repository in C:/Users/emman/Documents/Keyce/Eval/.git/
PS C:\Users\emman\Documents\Keyce\Eval> git add .
```

```
PS C:\Users\emman\Documents\Keyce\Eval> git commit -m "Initialisation du projet avec structure HTML et fichier CSS vide"
[master (root-commit) 6b77617] Initialisation du projet avec structure HTML et fichier CSS vide
2 files changed, 15 insertions(+)
create mode 100644 Index.html
create mode 100644 styles.css
```

2. Création et bascule vers la branche feature -1:

```
PS C:\Users\emman\Documents\Keyce\Eval> git checkout -b feature-1
Switched to a new branch 'feature-1'
```

3. Ajout et commit des modifications :

```
PS C:\Users\emman\Documents\Keyce\Eval> git add .
PS C:\Users\emman\Documents\Keyce\Eval> git commit -m "Ajout du style de fond darkgoldenrod et ajout du style de titre"
[feature-1 d7e5556] Ajout du style de fond darkgoldenrod et ajout du style de titre
3 files changed, 18 insertions(+), 2 deletions(-)
create mode 100644 .vscode/settings.json
```

4. Fusion de la branche feature-1 dans master :

```
PS C:\Users\emman\Documents\Keyce\Eval> git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
PS C:\Users\emman\Documents\Keyce\Eval> git merge feature-1
```

5. Push de la branche sur le dépôt distant :

```
git push origin feature-1
```

6. Création de la deuxième branche et implémentation du bouton stylisé :

```
PS C:\Users\emman\Documents\Keyce\Eval> git checkout -b feature-2
Switched to a new branch 'feature-2'
PS C:\Users\emman\Documents\Keyce\Eval> git add .
PS C:\Users\emman\Documents\Keyce\Eval> git commit -m "Ajout du bouton de connexion et de son style"
[feature-2 c27b0a1] Ajout du bouton de connexion et de son style
2 files changed, 10 insertions(+)
```

git push origin feature-2

7. Fusion de la branche feature-2 dans master :

```
PS C:\Users\emman\Documents\Keyce\Eval> git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
PS C:\Users\emman\Documents\Keyce\Eval> git merge feature-2
Updating 175556...c27b0a1
```

3. Justification des Choix de Développement

3.1. Hero Section

J'ai opté pour un fond uni avec une phrase centrée pour assurer une mise en page simple et efficace:

```
.Hero{
  border: solid black 1px;
  background-color: darkgoldenrod;
  justify-content: center;
  align-items: center;
  width: 50%;
  margin-left: 25%;
}

.Titre{
  font: 4rem "Fira Sans", sans-serif;
  justify-content: center;
}
```

3.2. Bouton Stylisé

Le bouton ajoute une interactivité à la page. Il a été conçu avec des effets CSS pour une meilleure expérience utilisateur :

```
.Bouton{  
  margin: 5% 40%;  
  padding: 10px 20px;  
  font-size: 16px;  
}
```

4. Conclusion

Ce projet a été géré efficacement grâce à Git en adoptant une stratégie de branches permettant un développement modulaire et structuré. Chaque fonctionnalité a été implémentée sur une branche distincte avant d'être fusionnée dans master, garantissant un contrôle de version clair et une meilleure collaboration.