

TP GitHub – Pull Requests

Objectifs pédagogiques

- Comprendre le principe des Pull Requests
- Travailler avec des branches dans un contexte collaboratif
- Proposer des modifications via GitHub
- Relire et fusionner du code proprement

Astuce : Une Pull Request n'est pas seulement une fusion, c'est aussi un outil de discussion et de validation.

Contexte

Vous travaillez sur un projet hébergé sur GitHub. Pour ajouter une amélioration, vous devez passer par une Pull Request afin que votre travail soit relu avant d'être intégré.

1 Préparation du projet

Utilisez un dépôt existant (par exemple celui du site GitHub Pages).

```
git clone https://github.com/NOM_UTILISATEUR/nom-du-projet.git
cd nom-du-projet
```

Explication : Le clonage permet de récupérer le projet et son historique complet sur votre machine.

2 Créer une branche de travail

```
git checkout -b amelioration-contenu
```

Explication : Chaque Pull Request doit correspondre à une branche dédiée, contenant une modification précise.

3 Apporter une modification

Modifiez le fichier `index.html` (ou `README.md`) en ajoutant une section ou un texte.
Puis :

```
git add .  
git commit -m "Ajout d'une amélioration du contenu"
```

4 Envoyer la branche sur GitHub

```
git push -u origin amelioration-contenu
```

Explication : La branche est maintenant disponible sur GitHub et prête à être proposée via une Pull Request.

5 Créer la Pull Request

Sur GitHub :

- Aller dans l'onglet **Pull Requests**
- Cliquer sur **New Pull Request**
- Sélectionner la branche `amelioration-contenu`
- Ajouter un titre et une description clairs

Explication : La Pull Request permet de comparer les changements et de demander leur intégration dans `main`.

6 Relecture et validation

Relisez les changements proposés :

- Vérifier le contenu
- Ajouter un commentaire si nécessaire
- Valider la Pull Request

7 Fusionner la Pull Request

Une fois validée :

- Cliquer sur **Merge Pull Request**
- Confirmer la fusion
- Supprimer la branche si proposé

Explication : La branche est fusionnée dans `main`. Le projet contient maintenant la nouvelle fonctionnalité.

Conclusion

Vous avez appris à :

- Travailler avec des branches collaboratives
- Proposer des modifications via Pull Request
- Relire et valider du code
- Utiliser GitHub comme outil de collaboration professionnelle

Bonne pratique : Une Pull Request doit être courte, claire et dédiée à une seule amélioration.