3/1/2024

Pacyna, Atilla

Collège bois-de-boulogne

Journal

Applications Web – TP1 partie 1

# Tables des matières

[Tables des matières 1](#_Toc160195214)

[Prises d’écrans 2](#_Toc160195215)

[Commandes GIT 3](#_Toc160195216)

[Défis rencontrés 4](#_Toc160195217)

# Prises d’écrans

A screen shot of a computer program

Description automatically generated

# Commandes GIT

Git init (Initialiser le répertoire local)

Git clone (Création d'une copie d'un dépôt distant)

Git branch + Nom de branche (Feature) (Créer une branche pour travailler localement)

Git branch (Pour vérifier dans quelle branche je me trouve)

Git status (Affiche l'état actuel du répertoire de travail et de la zone de staging)

Git add ./ Nom du fichier (Pour ajouter les modifications du répertoire de travail à la zone de staging)

Git commit -m message (Pour enregistrer les modifications qui ont été mises en scène (à l'aide de "git add") dans le dépôt local)

Git push origin feature (Pour téléverser le contenu du dépôt local vers un dépôt distant)

Git log (Git est utilisée pour afficher l'historique des commits d'un dépôt)

- La sortie de "git log" inclut généralement les informations suivantes pour chaque 🡺 commit : d

- Le hash du commit (Un identifiant unique pour le commit)

- Le nom et l'e-mail de l'auteur

- La date et l'heure du commit

- Le message du commit (Une description des modifications apportées dans le commit)

Git pull (Utilisée pour récupérer les dernières modifications depuis un dépôt distant et fusionner ces changements dans votre branche de travail locale.)

Git merge (Utilisée pour fusionner les modifications d'une branche dans une autre) 🡺 Fusionner la branche de fonctionnalité avec la branche principale

# Défis rencontrés

Au niveau de la gestion de versions avec Git, les défis étaient nombreux et instructifs. La commande git init a été le point de départ, établissant un nouveau dépôt local. Ensuite, l'utilisation de git clone a permis de créer une copie de travail d'un dépôt distant, essentielle pour commencer à travailler sur le projet. La création et la gestion des branches avec git branch ont permis de travailler de manière organisée et isolée sur différentes fonctionnalités. Cela a été crucial pour maintenir l'ordre et la clarté dans le projet. La commande git status a joué un rôle clé dans le suivi des modifications et la préparation des commits. Quant à git add et git commit, elles ont été utilisées pour enregistrer les modifications dans le dépôt local, nécessitant des messages de commit clairs et significatifs pour une bonne traçabilité du travail. Enfin, git push a servi à téléverser le contenu du dépôt local vers un dépôt distant, permettant le partage et la collaboration.

Parallèlement, la création du site web a nécessité une maîtrise approfondie du HTML et du CSS. La structure du site, comprenant plusieurs pages HTML et un fichier CSS commun, a exigé une compréhension approfondie de ces technologies pour structurer efficacement le contenu et le styliser de manière attrayante. Un défi majeur a été d'assurer que toutes les pages étaient correctement liées avec des chemins relatifs pour une navigation fluide. De plus, la gestion efficace des images dans un répertoire séparé et leur exclusion des commits via un fichier .gitignore a été une étape clé pour comprendre la gestion des fichiers dans un projet de développement web.

La collaboration en équipe a été un aspect crucial de ce projet. La nécessité de modifications cohérentes sur une branche spécifique et la fusion soignée de ces changements dans la branche principale via git merge ont mis en évidence l'importance de la communication et de la coordination au sein d'une équipe, en particulier pour maintenir la cohérence du design et du fonctionnement du site web.

En résumé, ce travail pratique a été une expérience d'apprentissage inestimable, combinant les défis du développement web avec ceux de la gestion de version avec Git. Il a contribué significativement à l'acquisition de compétences pratiques en HTML, CSS, et Git, ainsi qu'au développement de compétences essentielles en collaboration, communication, et résolution de problèmes dans un cadre de développement de logiciels.