Git et GitHub

Une image contenant texte, clipart

Description générée automatiquementGit

# Caractéristiques

Git est un logiciel de gestion de versions décentralisé qui utilise un système de connexion pair à pair (chaque entité est à la fois client et serveur). Un logiciel de gestion de versions (*version control system*) est un logiciel qui permet de stocker un ensemble de fichiers en conservant la chronologie de toutes les modifications qui ont été effectuées dessus. Cette gestion des versions permet de gérer ainsi l’ensemble des versions d’un ou plusieurs fichiers. Git fonctionne au travers d’un système d’un tronc dont émane des branches. Ces branches permettent la modification de documents, et, s’il n’y a pas de conflits, ces dernières pourront être fusionnées. Son utilisation est très pratique notamment grâce à toutes les commandes que propose cet environnement.

# Intérêts

L’intérêt d’utiliser un logiciel tel que Git se situe au niveau du rôle et des fonctionnalités des branches. En effet, elles permettent notamment la maintenance d’anciennes versions du logiciel (sur les branches) tout en continuant le développement des futures versions (sur le tronc), mais aussi le développement parallèle de plusieurs fonctionnalités volumineuses sans bloquer le travail quotidien sur les autres fonctionnalités.



GitHub

# Caractéristiques

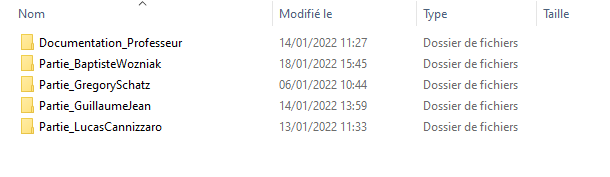
GitHub est une plateforme de gestion et d’organisation de projets qui intègre les fonctions de contrôle de version de Git. En d’autres termes, tous les utilisateurs de GitHub peuvent suivre et gérer les modifications apportées au code source en temps réel tout en ayant accès à toutes les autres fonctions de Git disponibles au même endroit.

# Intérêts

L’intérêt de GitHub est, au travers d’un service qui héberge un système de contrôle de version, de permettre aux différents membres d’une équipe de travail de collaborer et d’apporter des modifications à des projets partagés tout en gardant un suivi détaillé de leur progression. GitHub permet donc pour chaque participant d’une équipe de projet de travailler chacun à son rythme et de manière désynchronisée.

Gestion des branches dans notre projet

Pour notre projet, nous avons décidé d’utiliser des dossiers/répertoires qui font office de branches. Chaque dossier est la partie de travail d’un étudiant dans le projet. Il y a également un répertoire correspondant à la documentation fournie par notre professeur référent. Certes ces répertoires n’ont pas les principaux avantages des branches, notamment dans la gestion des versions qui n’est pas optimale, mais ils sont plus simples d’utilisation.

Voici comment notre projet est structuré :

L’utilisation de Git et de GitHub a pour conséquence de faciliter le développement de notre projet.

Glossaire des principales commandes de Git

Recevoir un nouveau document ou une nouvelle version d’un document

Ajouter un nouveau fichier dans un répertoire



Modifier un fichier



Envoyer un nouveau document ou une nouvelle version d’un document



Lister toutes les branches



Se déplacer dans une branche

Rapatrier une branche à une autre

Supprimer une branche que l’on utilise plus

Notre lien de projet GitHub : https://github.com/lucascanni/Barrieres\_Laser