

Suivi de projet dev - Ynov Campus Rennes - Évaluation 1/3 : Git et gitflow

Objectif

Utiliser le workflow [Gitflow](#) pour couvrir le développement d'une simple application web de calculatrice.

Prérequis

- git installé sur votre machine
- accès internet et à un dépôt central sur GitHub, GitLab, etc. (au choix)

Livrables

Déposer votre travail **sur un dépôt git public** (GitHub, Gitlab, BitKeeper, etc.) et fournir **le lien du dépôt**. Le dépôt contiendra votre code source et un fichier `README.md` donnant les instructions pour installer et lancer le projet. La branche qui sera inspectée pour l'évaluation sera la branche `main`, qui doit contenir la version la plus à jour du projet prête à la mise en production.

Penser à utiliser un `.gitignore` pour éviter de pousser sur votre dépôt le contenu de `node_modules` !

Envoyer l'url du dépôt à l'adresse mail suivante: paul.schuhmacher@ynov.com, avec le sujet suivant `rendu - evaluation gitflow - x_abc` où `x` est la première lettre de votre nom et abc votre prénom (je soumettrai donc le sujet `rendu - evaluation gitflow - s_paul`).

Merci de respecter les consignes sur les livrables (où soumettre votre travail et le format de l'email), au risque d'encourir une pénalité.

Date limite

Vous avez jusqu'au **29/06/23 23h59 (UTC+2)** pour me rendre votre travail. Le travail publié au-delà de cette date ne sera pas pris en compte.

Énoncé

On souhaiterait développer une application web de calculatrice. Le code doit fonctionner, mais l'historique de votre développement doit être *propre* et respecter les principes du workflow *Gitflow*. Une *attention particulière* sera portée aux commentaires des commits.

1. **Cloner le dépôt suivant** sur votre machine qui contient le projet dans un état initial non fonctionnel.
2. **Créer** le patch (un ensemble de commits) pour corriger l'application en utilisant le pattern *Gitflow*.
3. **Développer** les fonctionnalités définies dans le cahier des charges mis à votre disposition en respectant le workflow **Gitflow** et en gardant *un historique propre*.

Vous êtes libre d'utiliser des librairies ou des frameworks JS/CSS si vous le souhaitez

Cahier des charges

Voici la liste des exigences fonctionnelles de l'application :

- L'utilisateur·ice doit pouvoir choisir réaliser les opérations standards et bien définies suivantes: l'addition, la soustraction, la multiplication et la division
- La calculatrice doit pouvoir manipuler des nombres entiers et décimaux
- La calculatrice doit être simple à utiliser (un effort sur la mise en page sera apprécié)
- La calculatrice doit implémenter les fonctionnalités CE (Clear Error) et C (Clear)
- La calculatrice conserve un historique des opérations réalisées durant la session
- La calculatrice doit être documentée (manuel) pour que n'importe qui puisse l'utiliser facilement

Bonus

Vous ne sera pas pénalisés si vous n'implémentez pas ces spécifications

- Export CSV de l'historique du calcul (**2pts**)
- Manipuler la calculatrice avec les touches du clavier (**1pt**)

Modalités d'évaluation

- Respect du modèle Gitflow (penser à pousser toutes vos branches locales sur le dépôt central) (**8pts**)
- Historique des commits propre (messages des commits, corps de message éventuels justifiant un choix éventuel, etc.) (**6pts**)
- Un fichier **README** qui précise comment installer et exécuter votre application (**2pts**)
- La qualité du code (lisibilité, maintenabilité) (**2pts**)
- Implémentation des fonctionnalités du cahier des charges (**2pts**)

Utiliser le rebase interactif et le squash pour refactor votre historique de commits !

Ressources

- [Le dépôt à cloner pour démarrer le projet](#)
- [Une bonne application de calculatrice en ligne documentée](#), un exemple duquel s'inspirer