

## Pour proposer des modifications sur le Wiki RealT (page 1) ou YAM (page 6)

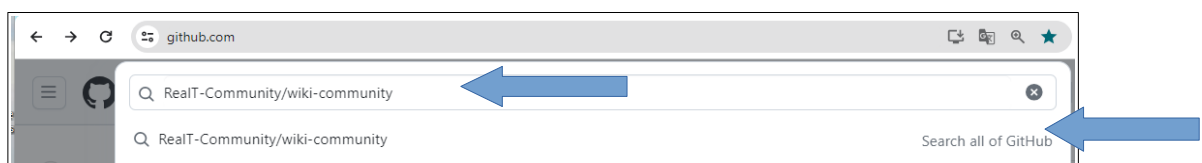
### 1 - Se créer un compte Github :

C'est gratuit sur <https://github.com/> via Sign up :

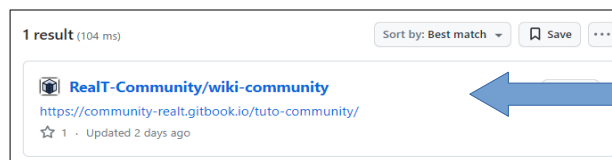


### 2 – Créer une copie (fork) du Github « RealT-Community/wiki\_community » sur votre Github

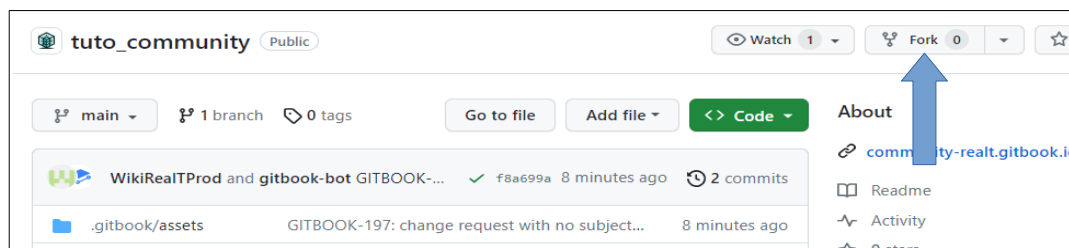
- Dans la partie recherche en haut, mettre le nom : RealT-Community/wiki\_community, puis faire search all



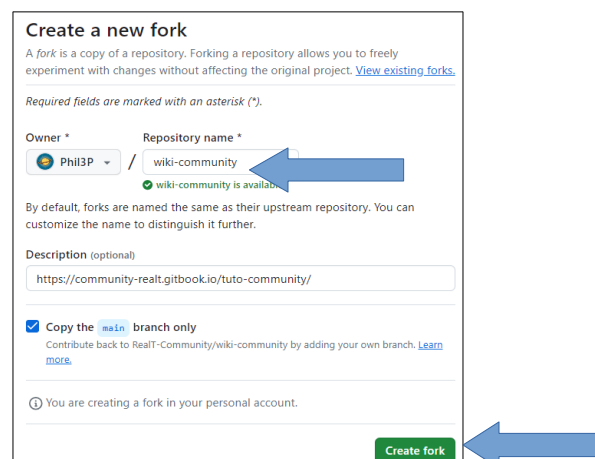
puis cliquer sur le lien trouvé :



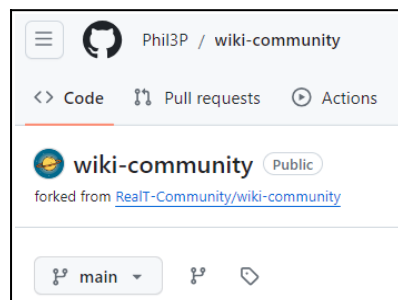
- Une fois sur le Github (/repository) cible, copiez (« forkez ») le sur votre Github:



- Le Owner sera le nom de votre compte Github, vous pourrez si vous le souhaitez modifier le nom du repository, puis créer le fork :

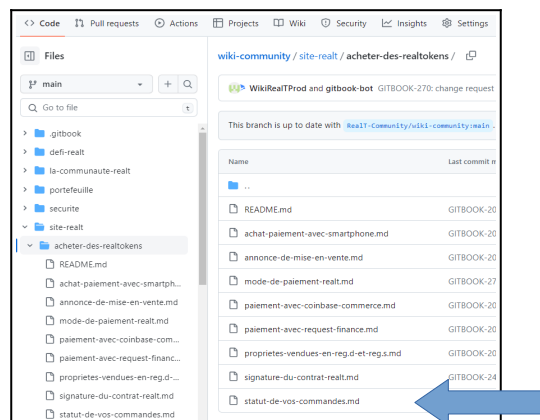


vous voyez en haut à gauche, le nom de votre nouveau repository (et le repository dont il est forké)



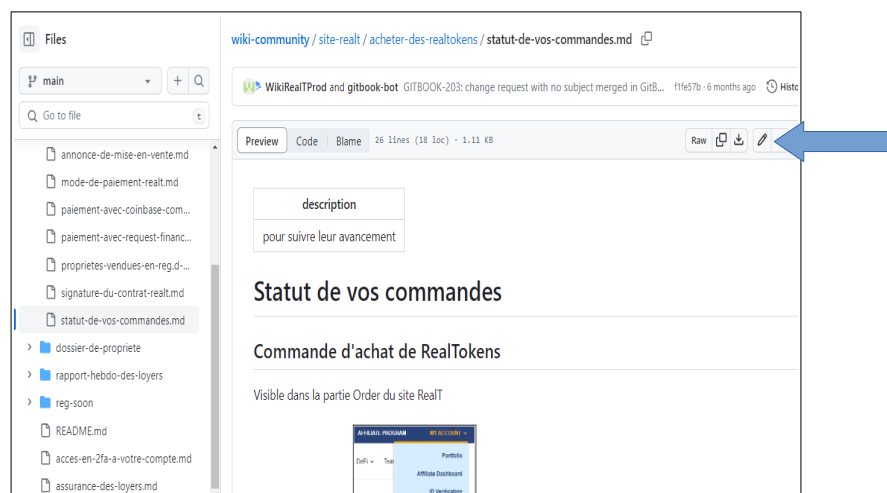
### 3 – Modifier votre version :

Vous accéder à tout l'arborescence des pages du Wiki et vous pouvez aller sur celle que vous souhaitez modifier :

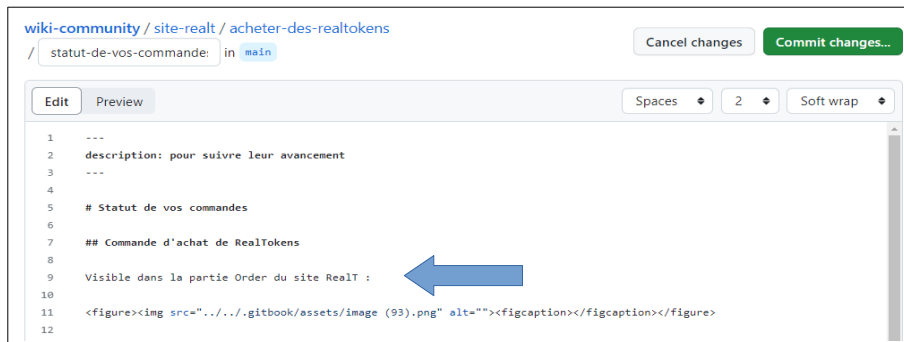


Prenons un exemple (ci-dessus) :

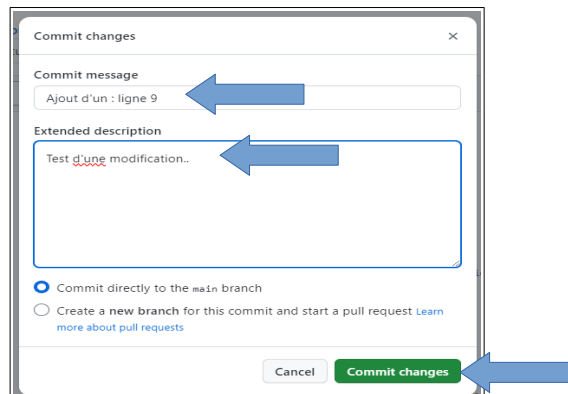
- vous souhaitez modifier la page « Statut de vos commandes »,
- vous vous rendez, dans l'arborescence, sur la page correspondante (soit vous chercher la page avec le titre de la page),
- vous passez en mode éditeur,



- vous faites votre modification : par exemple ajout d'un « : » après RealT sur ligne 9

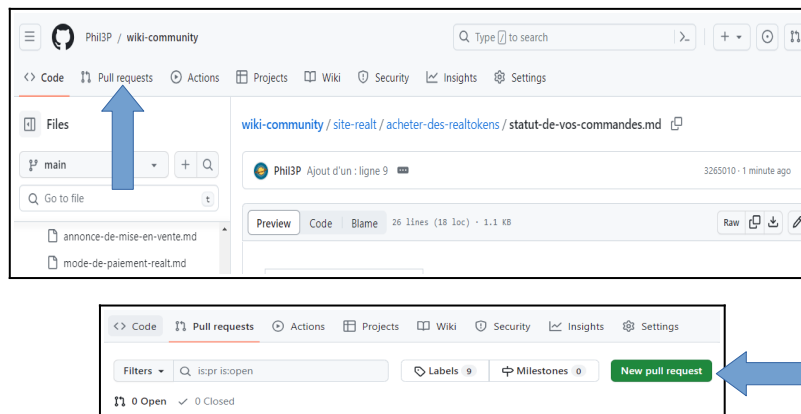


- puis vous « commitez » votre modification, et compléter le message de commit :

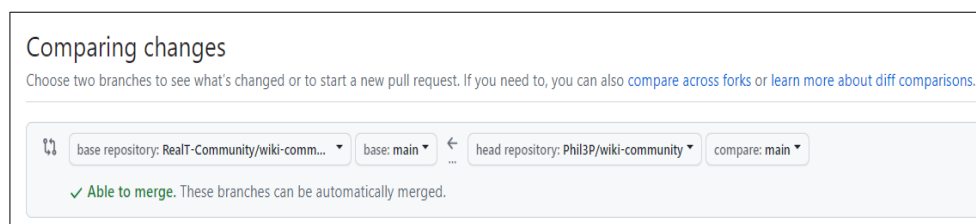


#### 4 – Créer une demande de mise à jour (Pull Request)

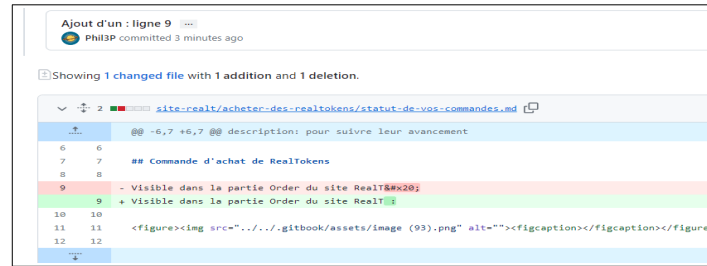
Une fois que vous avez fait toutes vos modifications (vos commits), vous pouvez créer une demande de mise à jour (Pull Request)



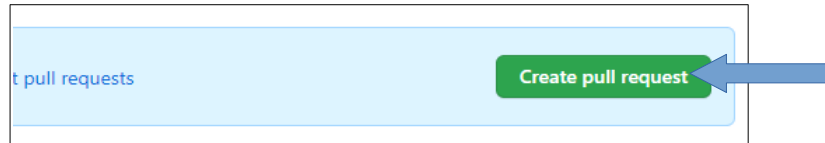
- Vous pouvez vérifier la source et la cible de la modification proposée :  
(de votre repository vers celui du Wiki)



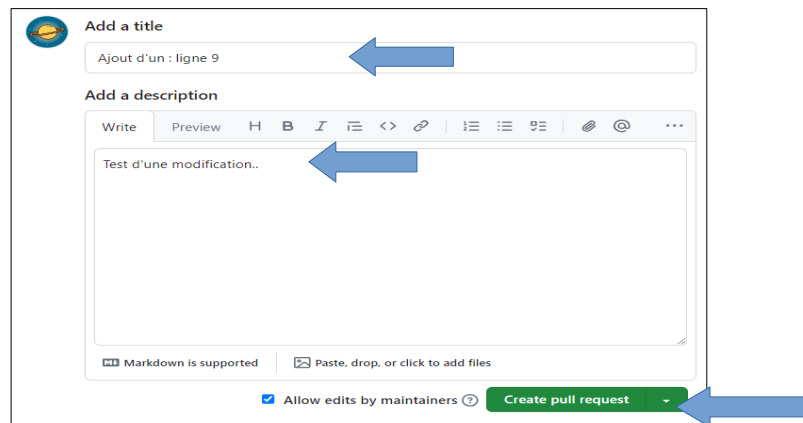
- Vous pouvez vérifier les modifications, que vous proposez :



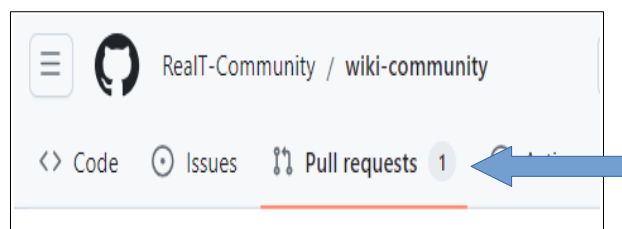
- après ces vérifications, vous pouvez créer la pull request :



- en mettant un commentaire, si vous ne l'avez pas fait précédemment (ce qui facilitera la validation de la modification)



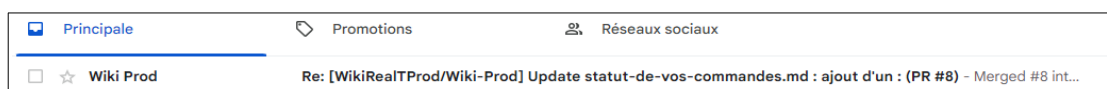
- Vous avez rebasculé vers le Repository cible (celui du Wiki) et vous voyez apparaître votre PR



Et voila, c'est terminé.. pas trop difficile !.. ;-)

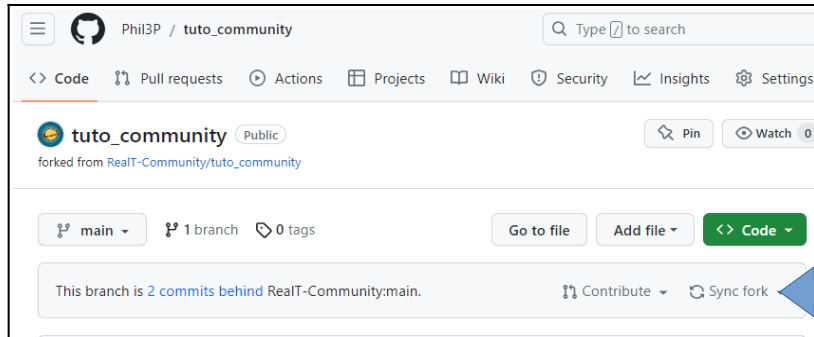
Enfin, juste encore un détail :

Quand le validateur aura traité votre PR, vous recevrez un message dans la messagerie que vous avez mentionnée lors de la création de votre compte Github



Ah oui, un autre détail..

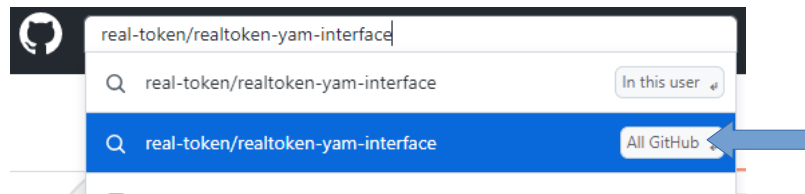
Si après avoir proposé des modifications, vous y prenez goût ;-)) et souhaitez en refaire :  
il faut absolument avant toute modification de votre repository le **resynchroniser** avec la  
version en production (des modifications ont pu avoir lieu entre temps)



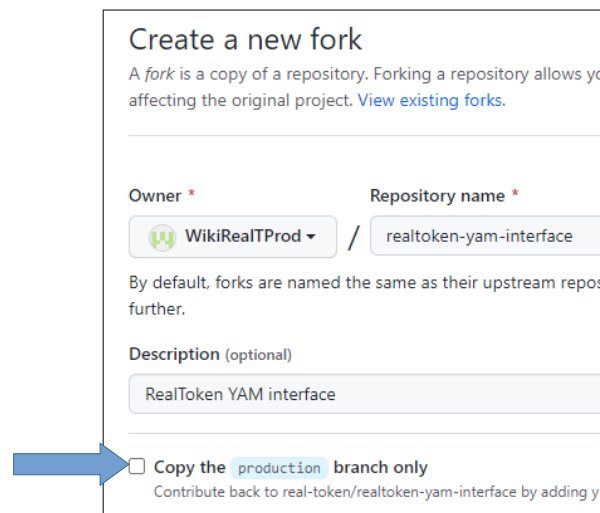
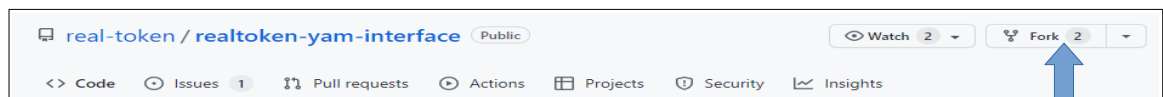
## Pour proposer des modifications sur YAM

La procédure est semblable, à quelques détails près :

- Le repository pour l'interface YAM, a forké est : real-token/realtoken-yam-interface



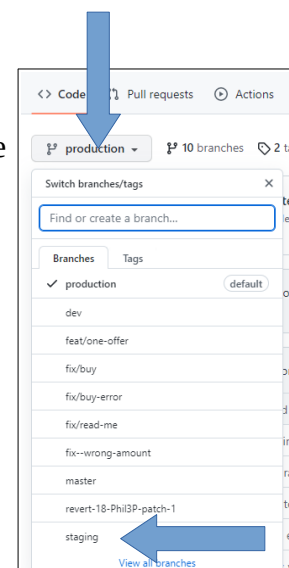
- Lorsque vous allez forké ce repository, il faut décocher « copy the production branch only » :



- Comme vous allez le voir, ce repository contient de nombreuses branches (10) :

Et c'est sur **staging** qu'il faut faire vos modifications

sinon, elles ne seront pas prise en compte



- Cf extrait du Read.me :

## Contributing

The community is welcome to participate in the development of the YAM.

### Introduction

All commit's name must follows the [conventional commit](#) convention.

⚠ ALL PR need to be created from `staging` branch and should requested to be merge into `staging` branch. Otherwise PR will be refused.

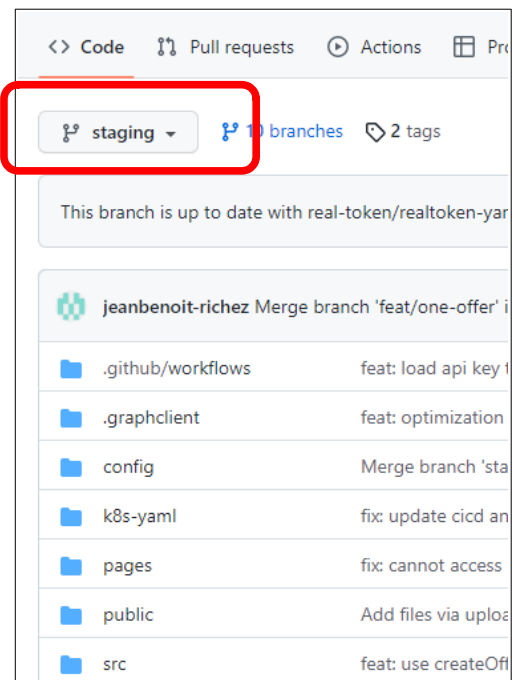
### Create a PR (Pull request)

If you have a suggestion that would make this better, please fork the repo and create a pull request. You can also simply open an issue with the tag "enhancement".

Don't forget to give the project a star! Thanks again!

1. Fork the Project
2. Move to `staging` branch ( `git checkout staging` )
3. Create your Feature Branch ( `git checkout -b feature/AmazingFeature` )
4. Commit your Changes ( `git commit -m 'Add some AmazingFeature'` )
5. Push to the Branch ( `git push origin feature/AmazingFeature` )
6. Open a Pull Request

- Donc bien vérifier que vous avez sélectionné la bonne branche avant toute modification de votre Repository



- De même lors de la création de votre PR, la branche par défaut sera celle de la production et vous devrez la changer pour passer sur le staging pour la source et la cible :

The screenshot shows the GitHub 'Comparing changes' interface. At the top, there's a navigation bar with links for Code, Issues (1), Pull requests, Actions, Projects, Security, and Insights. Below this, the title 'Comparing changes' is followed by a subtitle: 'Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).' The main area contains two dropdown menus. The first dropdown is labeled 'base repository: real-token/realtoken-yam-inter...' and has a 'base: production' option selected, which is highlighted with a red box. The second dropdown is labeled 'head repository: Phil3P/realtoken-yam-interface' and has a 'compare: production' option selected, also highlighted with a red box.

The screenshot shows the GitHub 'Comparing changes' interface. At the top, there's a navigation bar with links for Code, Issues (1), Pull requests, Actions, Projects, Security, and Insights. Below this, the title 'Comparing changes' is followed by a subtitle: 'Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).' The main area contains two dropdown menus. The first dropdown is labeled 'base repository: real-token/realtoken-yam-inter...' and has a 'base: staging' option selected, which is highlighted with a red box. The second dropdown is labeled 'head repository: Phil3P/realtoken-yam-interface' and has a 'compare: staging' option selected, also highlighted with a red box.

- Les noms de vos commits doivent respecter un certain format :  
<https://www.conventionalcommits.org/en/v1.0.0/>