

# Contribuer au projet

Esup-multi est un projet communautaire et open-source. Chacun peut y apporter ses contributions, qu'il s'agisse du code ou de la documentation.

## Avant de commencer...

Faites-vous connaître en échangeant sur les canaux suivants

- Liste de diffusion : [esup-multi@esup-portail.org](mailto:esup-multi@esup-portail.org)
- Rocket Chat : <https://rocket.esup-portail.org/channel/esup-multi>

Il se peut que vos projets de contribution soient déjà en cours dans d'autres établissements.

## Contribuer à la documentation

Pour contribuer à la documentation, il suffit de demander les droits d'écriture sur l'espace wiki du projet.

## Contribuer au code

Le projet est hébergé sur gitHub : <https://github.com/univlorraine/Esup-Pod>

Dans un premier temps l'Université de Lorraine validera chaque contribution et se chargera de l'intégration du code dans les sources du projet.

Le projet devra donc être forké (au sens gitHub), la contribution devra faire l'objet d'une branche dédiée et sera remontée par le mécanisme des pull request (PR) de gitHub dans le dépôt de l'Université de Lorraine afin de pouvoir profiter à la communauté.

### Corriger un bug

Pour chaque bug avéré il conviendra de faire une Issue décrivant précisément le problème afin de permettre les échanges et le suivi : <https://github.com/univlorraine/esup-multi/issues>

1 Issue = 1 branche = 1 PR

### Ajouter une fonctionnalité

L'idéal est d'éviter à un établissement de diverger des sources sur Esup-Multi et devoir maintenir des différences de code s'il souhaite ajouter un service ou une fonctionnalité. Il est donc nécessaire de faire remonter au groupe de projet Esup-Multi (à l'Université de Lorraine dans un premier temps) tout besoin d'une nouvelle fonctionnalité. Cela permettra d'en évaluer la pertinence, d'identifier les éventuels projets similaires et d'étudier au sein d'un GT sa généralisation pour l'ensemble des établissements intéressés.

## Exigences dans le code



Ce paragraphe s'étoffera en fonction des contributions qui seront apportées.

Chaque contribution doit être unitaire et se matérialise par une branche Git

## Règles de nommage GIT

### Nom de branche

- *feat/MULTI-XX-trois-quatre-mots-description*
- *fix/MULTI-XX-trois-quatre-mots-description*

### Titre de la PR

- *feat: MULTI-XX Trois quatre mots de description* (pour les nouvelles feature)
- *fix: MULTI-XX Trois quatre mots de description* (pour les bug)

Respecter les [règles des commit](https://github.com/googleapis/release-please) car ils permettent de générer automatiquement le ChangeLog de l'application grâce à l'outil **release-please** (<https://github.com/googleapis/release-please>)

Respecter les [Lignes directrices pour le développement](#).

TO BE CONTINUED

## Comment faire une Pull Request (PR)

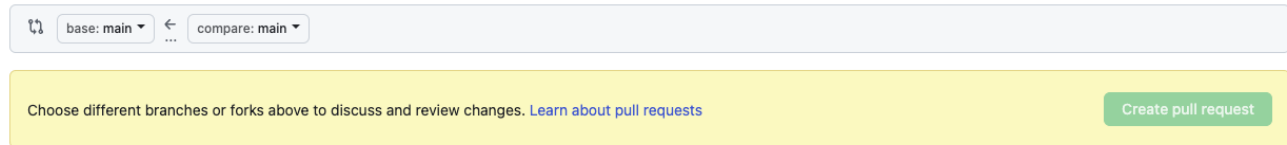
Se rendre sur l'onglet Pull Request du projet : <https://github.com/univlorraine/esup-multi/pulls>

Cliquer sur le bouton

Puis sur "compare across forks"

## Compare changes

Compare changes across branches, commits, tags, and more below. If you need to, you can also [compare across forks](#).



base: main ... compare: main

Choose different branches or forks above to discuss and review changes. [Learn about pull requests](#)

Create pull request

Choisir la branche de développement du dépôt de départ et la branche dédiée à la contribution du dépôt forké

Enfin, indiquer le titre et compléter la description de la PR. En cas de correction de d'un bug, il sera intéressant d'indiquer le numéro de l'issue dans la PR.

Et soumettre la PR

L'outil va vérifier qu'un éventuel merge n'entraînera pas de conflits et le processus d'intégration continue de gitHub va alors se déclencher.

## Développer un nouveau module

Côté backend => Détaillé dans le README de /dev/user-backend-nest

Côté frontend => Détaillé dans le README de /dev/user-frontend-ionic