

Github et IntelliJ

Ce document résume comment utiliser **Github** à partir d'**IntelliJ** pour gérer ses projets.

Contenu

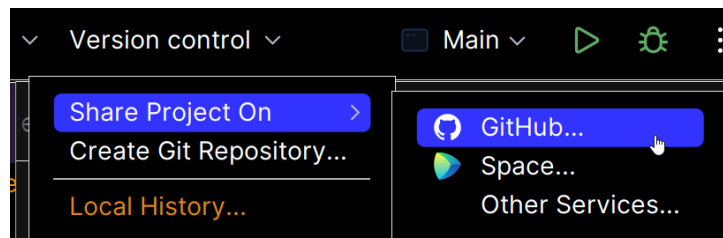
Partager un projet sur Github.....	1
Autoriser l'intégration de JetBrains avec Github.....	2
Importer un projet à partir de Github.....	4
.gitignore : Liste d'exclusion.....	6

Notez bien :

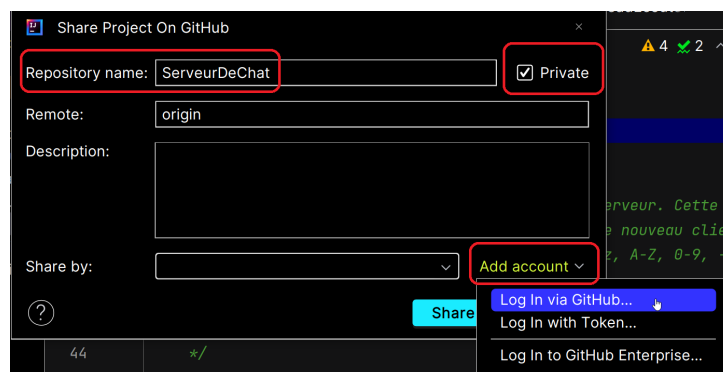
- ❖ Il faut avoir un compte *Github*;
- ❖ Il faut être connecté au compte;
- ❖ La première fois qu'on accède à **Github** à partir d'**IntelliJ**, on doit autoriser l'intégration de **JetBrains** (l'éditeur d'**IntelliJ**) et de *Github*. Cette autorisation est faite **une seule fois**.

Partager un projet sur Github

1. Créer ou ouvrir un projet dans *IntelliJ*;
2. À partir du menu **Version control**, choisir l'item **Share Project On | Github** :



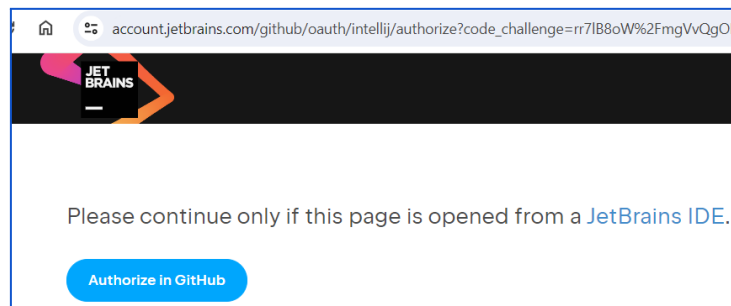
3. Spécifier un nom au référentiel ou dépôt (*repository*), décocher la case **Private** (les projets réalisés dans le cadre du cours seront tous publics) et indiquer le compte *Github* à utiliser :



Autoriser l'intégration de JetBrains avec Github

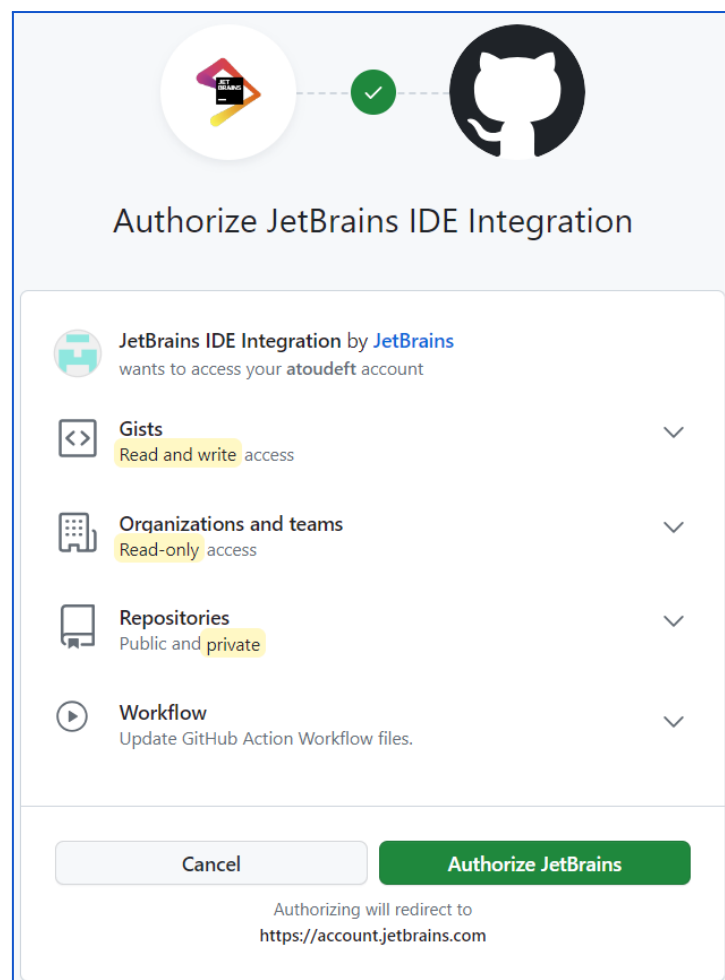
Cette étape est faite **une seule fois**.

4. En cliquant sur **Log In via Github...**, une page web apparaît pour autoriser *JetBrains* à interagir avec *Github* :

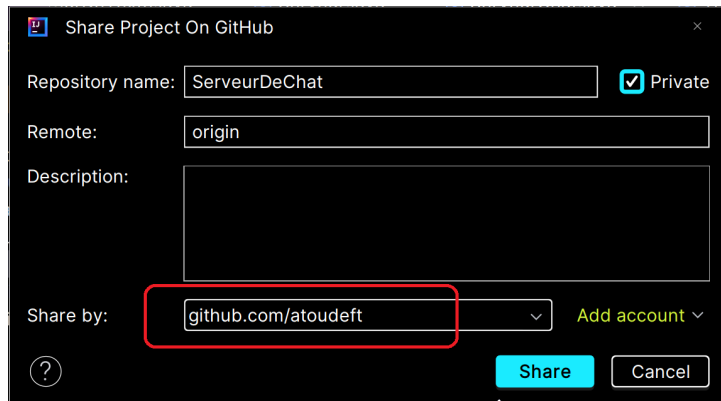


Cliquer sur le bouton **Authorize in Github**.

5. Une page demande d'autoriser l'intégration :



Cliquer sur **Authorize JetBrains** pour autoriser *JetBrains* à accéder au compte *Github*. Après l'autorisation, le fenêtre dans *IntelliJ* indique le compte *Github* où sera mis le projet :

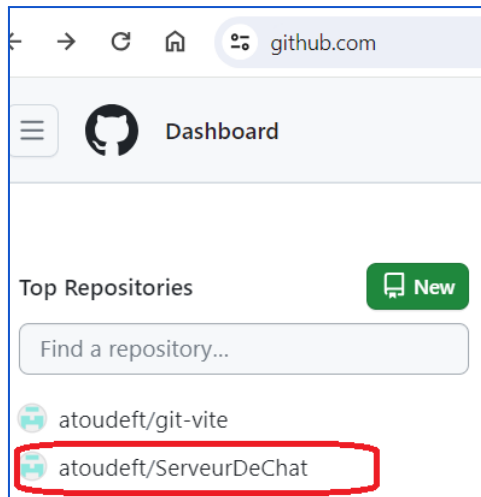


6. Cliquer sur **Share** puis sélectionner les fichiers à ajouter au référentiel. **Décocher tout sauf les *packages*, *classes* et *interfaces*** du projet (donc tout ce qui est dans le dossier **src**) :

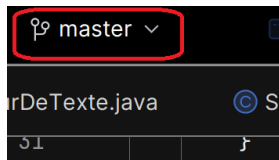


Le dossier **src** contient le code du projet qui doit être partagé entre les membres d'une équipe de développement. Le reste peut dépendre des configurations locales de chacun des membres et **ne doit pas être partagé**.

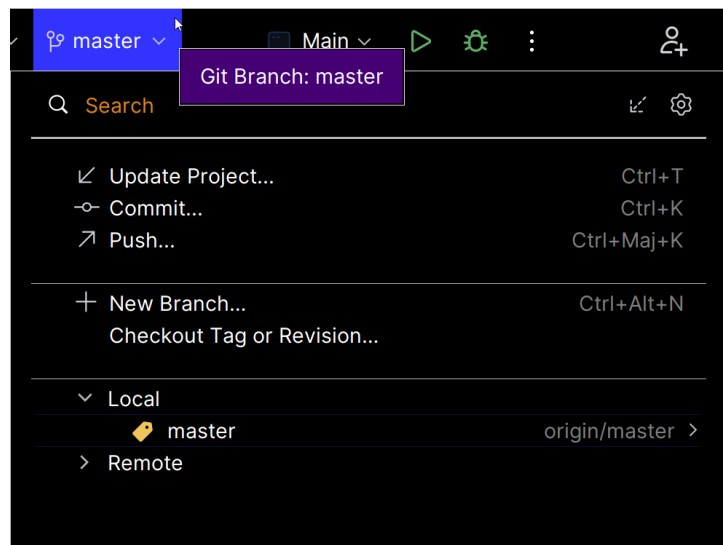
Cliquer sur **Add**. Le projet est ajouté sur *Github* :



Et dans *IntelliJ*, la branche actuelle (**master**) apparaît :



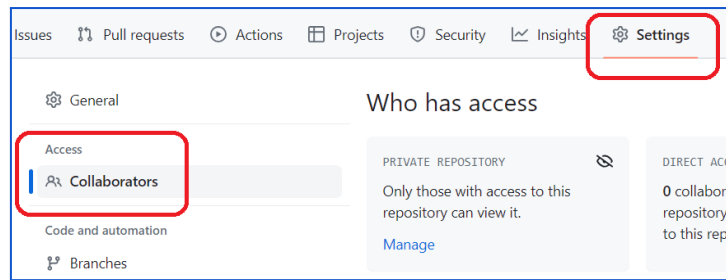
Elle donne accès aux principales fonctions pour utiliser *Git* avec *Github* :



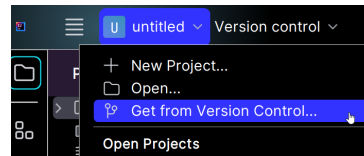
Importer un projet à partir de Github

Pour pouvoir importer un projet il faut :

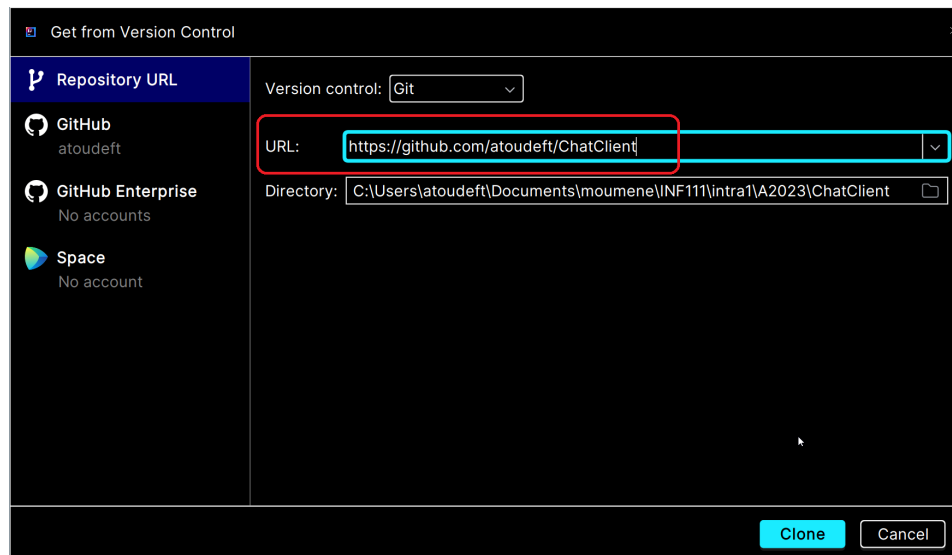
- que le **projet soit public**;
- **ou être membre du projet**, c'est-à-dire être ajouté comme collaborateur par le propriétaire du projet (*Collaborators* dans les *Settings* du projet) :



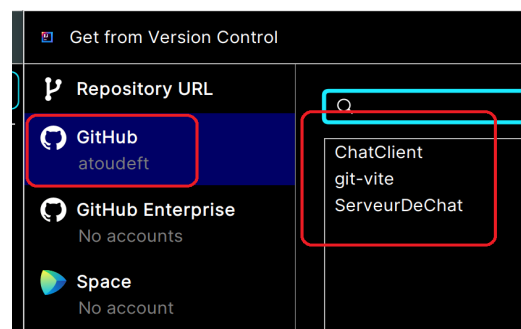
7. Dans *IntelliJ*, lancer le menu **Get from Version Control...** ;



8. Saisir l'**URL du projet** (disponible sur *Github* ou fournie par le propriétaire du projet) :



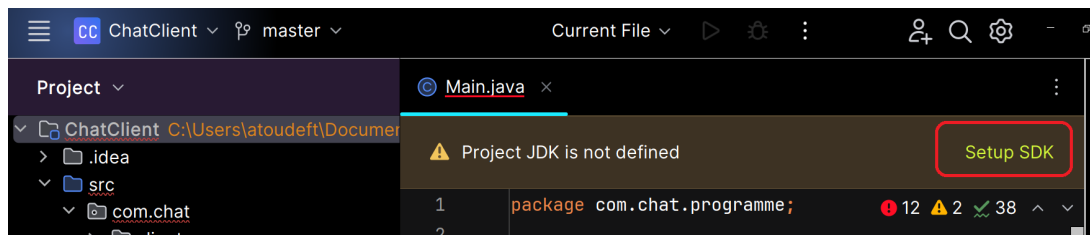
Si on est propriétaire du projet, on peut voir et sélectionner le projet dans la liste dans l'onglet *Github* :



9. Cliquer sur le bouton **Clone**. Le projet est importé dans *IntelliJ*.

10. Normalement, le projet sur *Github* ne contient pas la configuration de

l'environnement de développement. Il faut donc configurer le *JDK* :



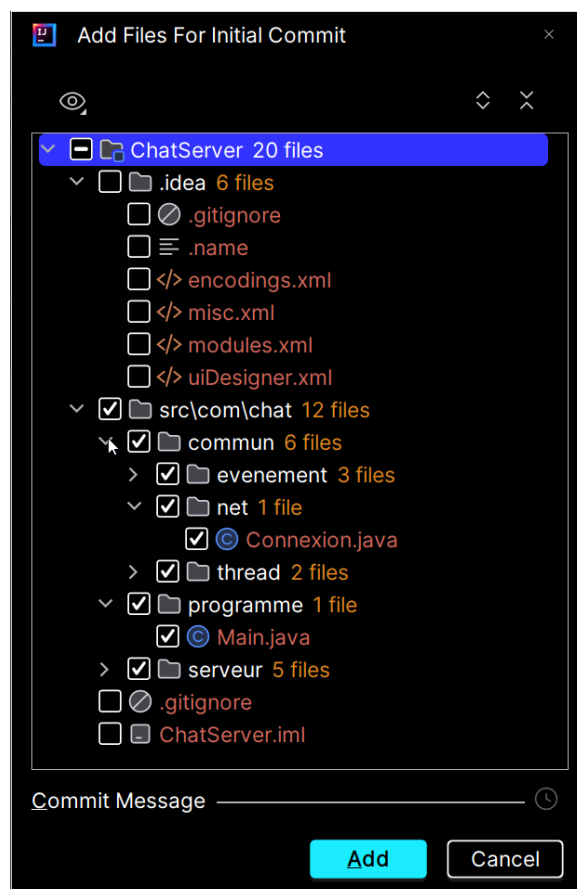
11. Cliquer sur **Setup JDK** pour choisir le *JDK* à utiliser.

Le projet est prêt.

.gitignore : Liste d'exclusion

Le fichier **.gitignore** contient la liste des fichiers du projet qui ne sont pas pris en charge par *Git*. Tous les fichiers spécifiques à l'environnement de développement devraient être dans **.gitignore**. Typiquement, seuls les fichiers et dossiers (packages) qui contiennent le code du programme doivent être partagés (ne seraient donc pas dans **.gitignore**).

On peut sélectionner les fichiers et dossiers à partager lors du transfert initial du projet sur *Github* :



On peut ajouter ultérieurement des fichiers et dossiers à **.gitignore** en utilisant le clic-droit ensuite **Git | Add to .gitignore | .gitignore**.

