
Support de Travaux Pratiques Développement de portails et outil de travail collaboratifs

Niveau 2^{ème} année cycle Ingénieur en Informatique

Etablissement ISSAT Sousse

Année universitaire 2021-2022

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique
Université de Sousse
Institut Supérieur des Sciences Appliquées et de Technologie de Sousse



Support de travaux pratiques

Développement de portails et outil de travail collaboratifs

2021 – 2022

2^{ème} année Ingénieur en Informatique

Enseignantes :

Nessrine ELLOUMI

Aicha BEN MAKHLOUF

Directeur de Département
Informatique
Moeiz ROUIS

Sommaire

TP 1 : Commandes et opérations sur GitHub

TP 2 : L'outil de travail collaboratif « Trello »

TP 3 : Espace de travail collaboratif : Slack

TP 4 : Espace de travail collaboratif : Jira

TP 1 : Commandes et opérations sur GitHub

L'objectif de ce TP est de suivre l'évolution d'un projet, localement et à distance. Généralement, en génie logiciel un programme est développé par un groupe de développeur. Pour que les différents membres d'une équipe communiquent facilement et pour gérer les versions modifiées par chaque membre, plusieurs outils de travail collaboratifs sont utilisés. Dans ce TP, nous allons choisir **GitHub** comme un outil de travail collaboratif.

1. Introduction

Git permet de gérer les versions d'un projet. Ce logiciel libre créé par Linux est le plus populaire dans la gestion des versions des projets. En effet, il était sous la demande des millions de personnes.

L'utilisation de git repose sur le dépôt d'une version d'un projet stockée sur un serveur sur Internet. Les modifications faites sur votre ordinateur sont enregistrées sur un espace distant par git. Git offre aussi la possibilité de partager vos travaux enregistrés sur le dépôt distant avec votre équipe. Aussi, il est possible de récupérer le travail fait par vos coéquipiers enregistré sur le dépôt distant.

2. Principe de fonctionnement

La figure suivante présente le fonctionnement du système de gestion de versions.

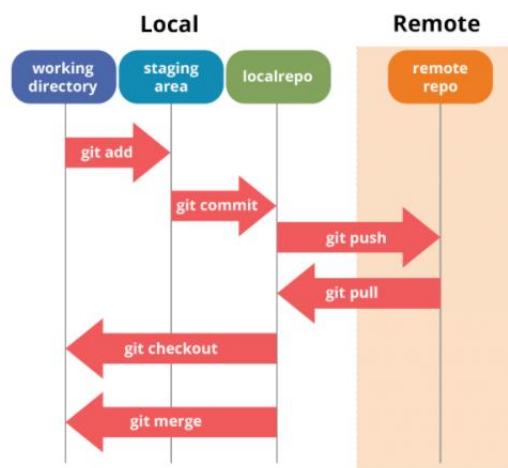


Figure 1. Fonctionnement du logiciel « git »

2.1. Le stockage sur Git

Au début les fichiers de votre projet sont enregistrés sur votre ordinateur dans un dossier local appelé **Working Directory**. Les modifications faites dans ce répertoire sont enregistrées dans un espace temporaire appelé **Staging area** qui contient les modifications prêtes à être « committées »

avant de les enregistrer dans le dépôt local appelé **Local Repository**. Enfin, toutes les modifications sont enregistrées et synchronisées avec la version ancienne dans un dépôt distant appelé **Remote Repository**.

2.2 Les opérations de bases de Git

Clone: `git clone <url_remote_repository>` permet de copier le contenu du dépôt distant dans le dossier courant. Cette commande est utilisée lorsqu'on n'a pas encore de dépôt local du projet.

Initialize: `git init` permet de créer un dépôt local git vide ou réinitialiser un existant.

Add: `git add <file>` permet la mise à jour du contenu de l'arborescence de travail avant l'exécution de la commande commit.

Commit: `git commit -a -m <message>` permet d'enregistrer les modifications effectuées sur un projet dans un dépôt local.

Push: `git push <remote-name> <branch-name>` permet de partager votre projet dans un dépôt distant.

Pull: `git pull origin <branch-name>` Cette commande met à jour le dépôt local à partir du dépôt distant.

3. GitHub

GitHub est un service d'hébergement Web et de gestion du développement logiciel basé sur le logiciel de gestion de versions Git. Le service a été développé par Chris Wanstrath, PJ Hyett et Tom Preston-Werner en utilisant Ruby on Rails et Erlang.

4. Manipulation

Dans ce TP, vous allez installer git sur votre ordinateur, créer un dépôt git distant sur GitHub et apprendre à utiliser ses fonctionnalités.

Après avoir installé Git sur votre machine, l'étape suivante consiste à créer un compte GitHub gratuit.

1. Visiter la page officielle de création de compte : Rejoindre GitHub
2. Choisir un nom d'utilisateur, saisir votre adresse électronique et choisir un mot de passe.
3. Accepter ou refuser de recevoir des mises à jour et des annonces en cochant/décochant la case Préférences de messagerie.
4. Vérification que ce n'est pas un robot.

5. Cliquer sur Créer un compte.
6. GitHub envoie un code de lancement à l'adresse électronique spécifiée. Passer le code pour vérification.

Vous avez maintenant créé avec succès un compte GitHub.

Partie I: Utilisation de l'éditeur « Visual Studio Code »

Pour faciliter l'utilisation de git nous allons utiliser l'éditeur Visual Studio Code (VS Code).

Pour ce faire, vous devez installer VS Code.

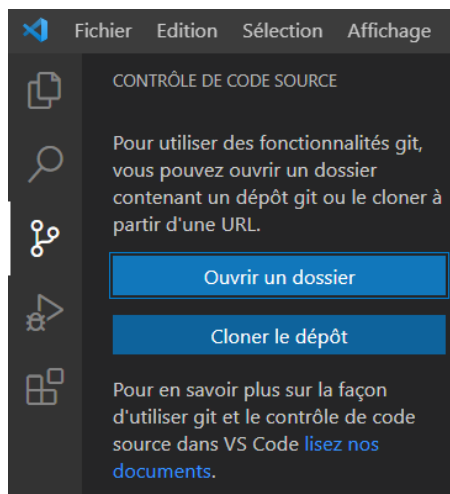
Étape 1: Initialisation du projet en tant que référentiel Git

Afin d'initialiser votre projet, vous devez suivre ces étapes :

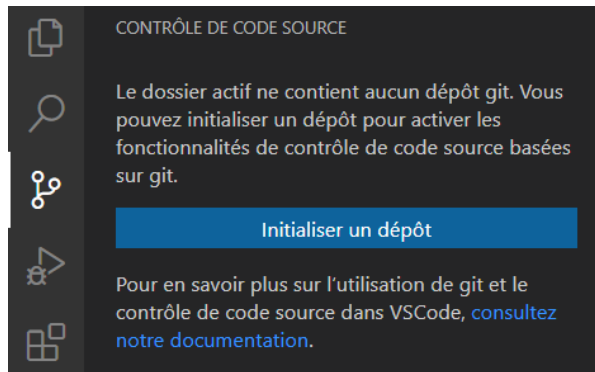
1. Dans le panneau de gauche de VS Code, ouvrir l'onglet **Contrôle de Code Source**:



2. Cliquer sur **Ouvrir un dossier**



3. L'explorateur de fichiers du répertoire actuel est ouvert. Choisir votre répertoire de projets, puis cliquer sur **Ouvrir**.
4. Choisir **Initialiser un dépôt**

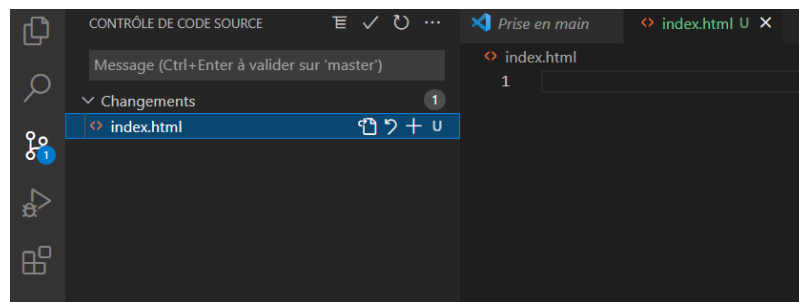


Votre système de fichiers, contient maintenant un répertoire. git.

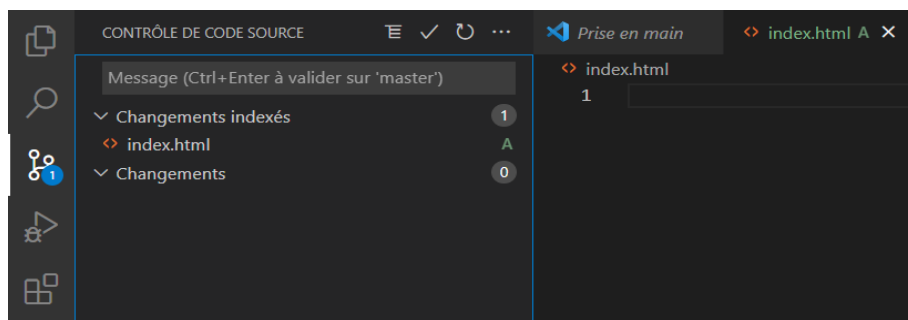
Utiliser le terminal afin d'explorer le répertoire du projet.

5. Créer le fichier « index.html ».

Dans le panneau **Contrôle de Code Source**, ce fichier apparaît avec la lettre **U** (*untracked file*). Cela veut dire que ce fichier est nouveau, modifié ou non traqué, donc n'a pas été ajouté au référentiel.



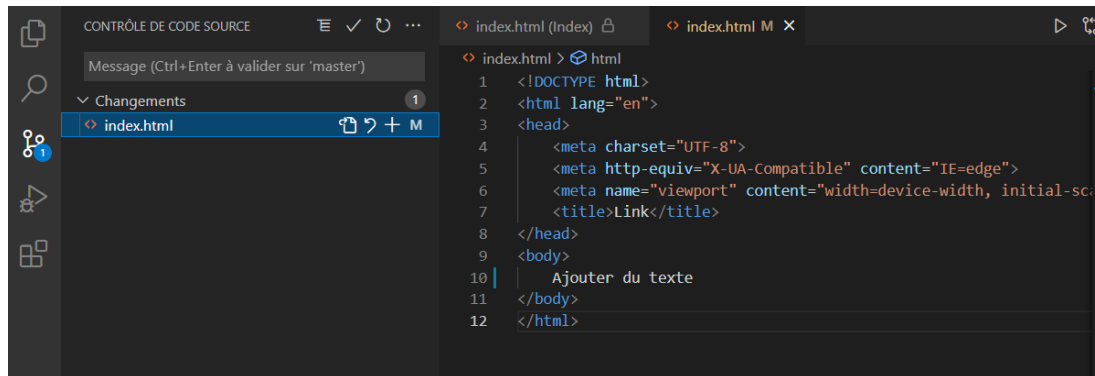
- ➔ En cliquant sur l'icône **plus (+)**, la lettre à côté du fichier se transforme en **A**.
Donc, ce fichier qui a été bien ajouté au référentiel.



6. Pour la validation des modifications, il faut taper un message de commit en cliquant sur l'icône **check** dans la zone de saisie située en haut du panneau **Contrôle de Code Source**.

Étape 2: Modification du projet

1. Ajouter du contenu au fichier index.html.



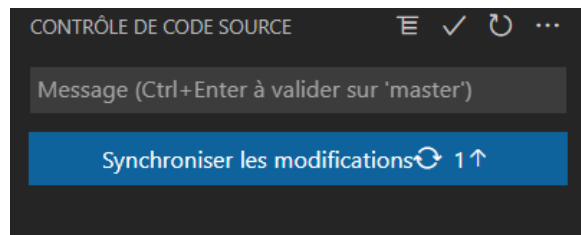
2. Cliquer sur l'icône **plus** (+) pour la modification du fichier.

Une lettre « **M** » apparaîtra qui représente la modification du dossier.

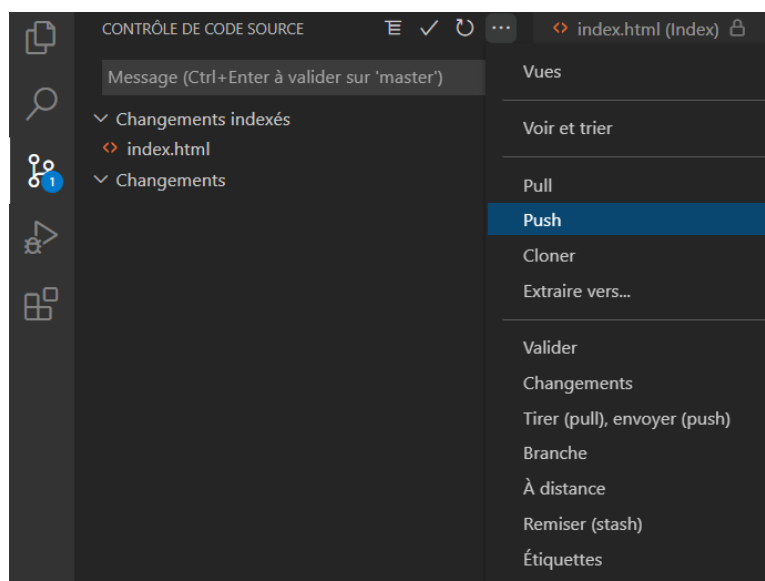
3. Afin de valider ces changements, cliquer sur l'icône **check** (pour créer un commit).

Un bouton pour synchroniser les modifications apparaîtra.

En cliquant sur ce bouton, ces modifications seront enregistrées.



4. Une autre façon pour enregistrer ces modifications dans le dossier distant consiste à choisir la commande « Push » se trouvant dans le panneau **Contrôle de Code Source**

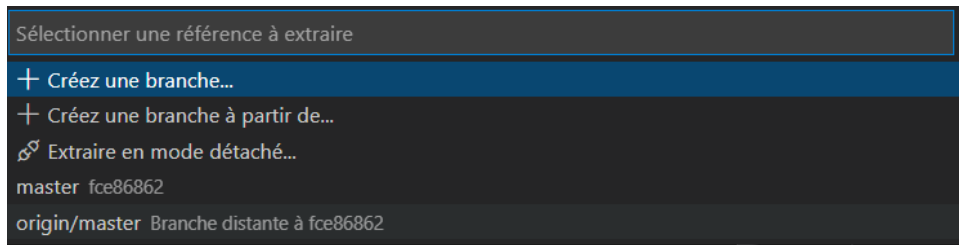


Étape 3 - Création des branches

1. Afin de créer et de changer de branche, cliquer sur l'icône « contrôle de source » :



2. Afin de créer la branche, entrer son nom.



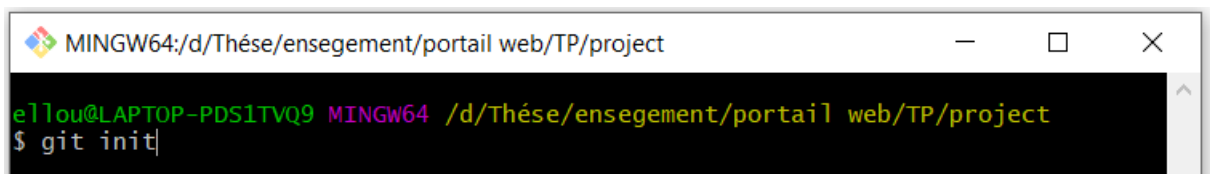
3. Créer la nouvelle branche « test ».
4. Modifier le fichier « index.html ».
5. Enregistrer les modifications et revenir à la branche master.

Méthode 2: Utilisation des commandes Git:

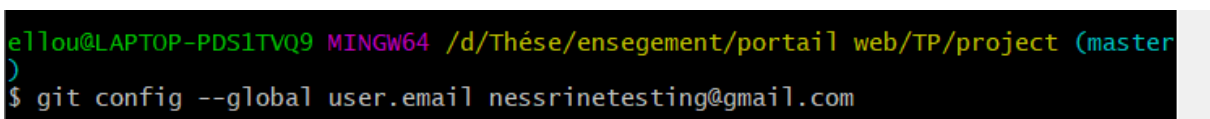
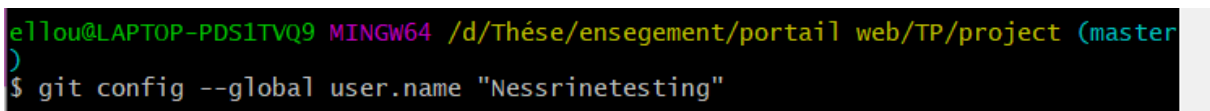
Étape1: Créer un dépôt Git local

Après avoir installé ou mis à jour Git, l'étape suivante consiste à créer un dépôt Git local.

1. Ouvrir un terminal Git Bash et aller dans le répertoire où vous souhaitez conserver le projet sur votre machine locale. Par exemple :
2. Créer un dépôt Git dans le dossier sélectionné en exécutant la commande git init :



3. Configurer votre compte Github avec l'espace de travail local sur votre ordinateur.



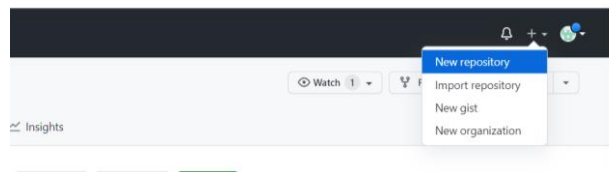
Vous avez maintenant configuré avec succès un dépôt Git local.

Étape 2: créer un nouveau dépôt sur GitHub

GitHub permet la sauvegarde d'une trace du code lorsque vous travailler avec une équipe et vous modifier le code du projet de manière collaborative.

Pour créer un nouveau dépôt sur GitHub :

1. Connecter à la page d'accueil de GitHub sur <https://github.com/>.
2. Cliquer sur l'option Nouveau dépôt sous le signe +, puis cliquer sur New repository.



3. Nommer votre référentiel (repository) appelée TP1, saisir une description significative et choisir un paramètre de confidentialité.


Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner * / Repository name *

Great repository names are short and memorable. Need inspiration? How about [cautious-potato](#)?

Description (optional)

☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.


☐  **Private**
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☒ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

☐ **Add .gitignore**
Choose which files not to track from a list of templates. [Learn more.](#)

☐ **Choose a license**
A license tells others what they can and can't do with your code. [Learn more.](#)

This will set  **main** as the default branch. Change the default name in your [settings](#).

[Create repository](#)

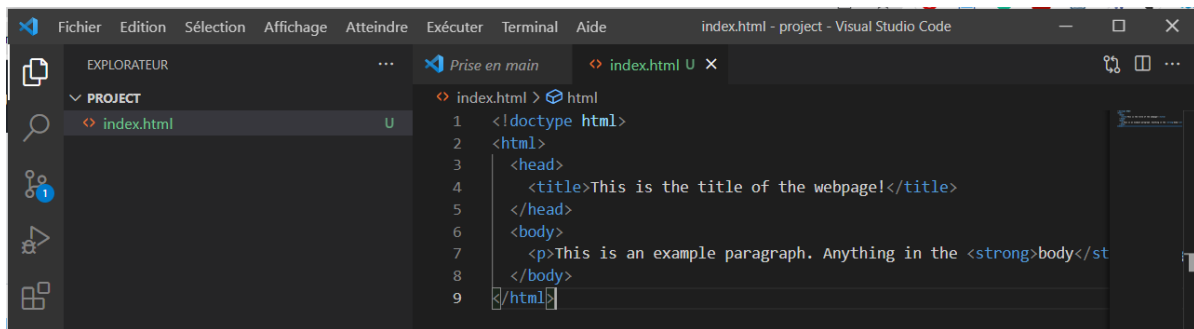
4. Cliquer sur le bouton « Create repository ».

GitHub vous permet d'ajouter un dépôt existant que vous avez créé localement. Pour héberger un dépôt local de votre machine vers GitHub à travers la commande « git remote » :

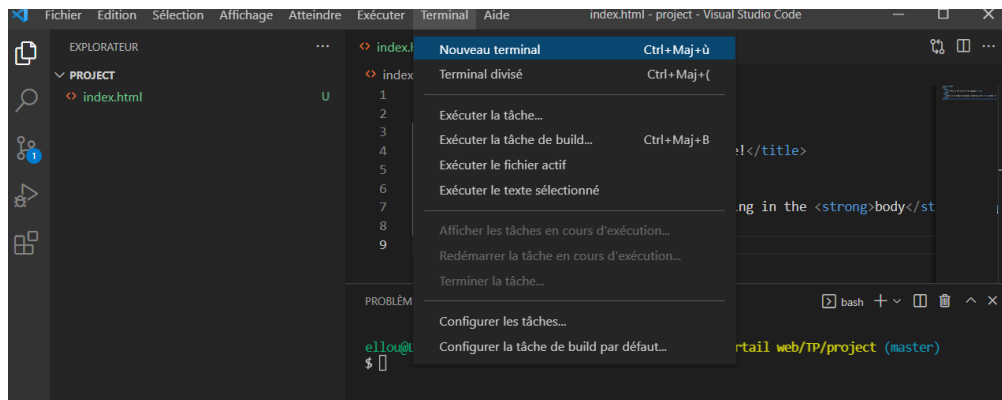
```
ellou@LAPTOP-PDS1TVQ9 MINGW64 /d/Thèse/enseignement/portail web/TP/project (master)
$ git remote add origin https://github.com/Nessrinetesting/TP1.git
```

Étape 3: Ajouter un fichier dans un projet

1. Ouvrir l'éditeur VS Code et créer un nouveau fichier appelé index.html.



2. Ouvrir un nouveau terminal « Git Bash » sous VS Code



3. Vérifier les fichiers suivis par Git en exécutant la commande « git status » :

```
ellou@LAPTOP-PDS1TVQ9 MINGW64 /d/Thèse/enseignement/portail web/TP/project (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    index.html

nothing added to commit but untracked files present (use "git add" to track)
```

Ajouter le fichier index.html en utilisant la commande « git add ».

Étape 4: Créer un commit

Après avoir ajouté les fichiers spécifiés à l'environnement de transit, référencier les fichiers modifier en utilisant la commande « git commit ».

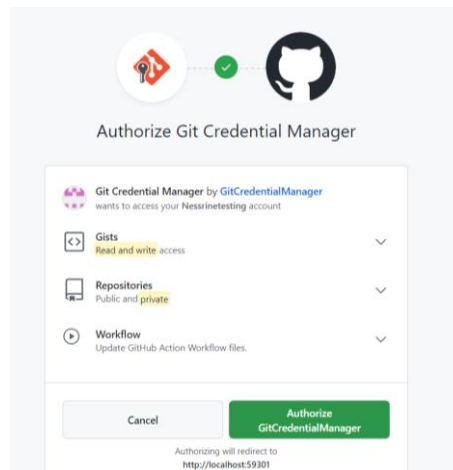
```
ellou@LAPTOP-PDS1TVQ9 MINGW64 /d/Thèse/enseignement/portail web/TP/project (master)
$ git commit -m "version 1"
[master (root-commit) b9e7b59] version 1
1 file changed, 9 insertions(+)
create mode 100644 index.html
```

Étape 5: Ajouter les paramètres d'accès au GitHub sous VS Code

1. Utiliser la commande **git config --global credential.helper store** pour enregistrer les paramètres de connexion sous VS Code.

```
ellou@LAPTOP-PDS1TVQ9 MINGW64 /d/Thèse/enseignement/portail web/TP/project (
master)
$ git config --global credential.helper store
```

2. Cliquer sur Autoriser.



3. Synchroniser le dossier local TP1 (origin) avec le dossier distant TP1 (remote) dans la branche master en utilisant la commande « **git push origin master** ».

```

ellou@LAPTOP-PDS1TVQ9 MINGW64 /d/Thèse/enseignement/portail web/TP/project (
master)
$ git push origin master
info: please complete authentication in your browser...
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 382 bytes | 382.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/Nessrinetesting/TP1/pull/new/master
remote:
To https://github.com/Nessrinetesting/TP1.git
 * [new branch]      master -> master

```

Étape 6: Ajouter une nouvelle branche « groupe1 »

1. Créer une nouvelle branche appelée groupe1

```

ellou@LAPTOP-PDS1TVQ9 MINGW64 /d/Thèse/enseignement/portail web/TP/project (
master)
$ git branch groupe1

```

2. Ajouter la branche groupe 1 dans le dossier remote en utilisant la commande « **git push origin groupe1** ».

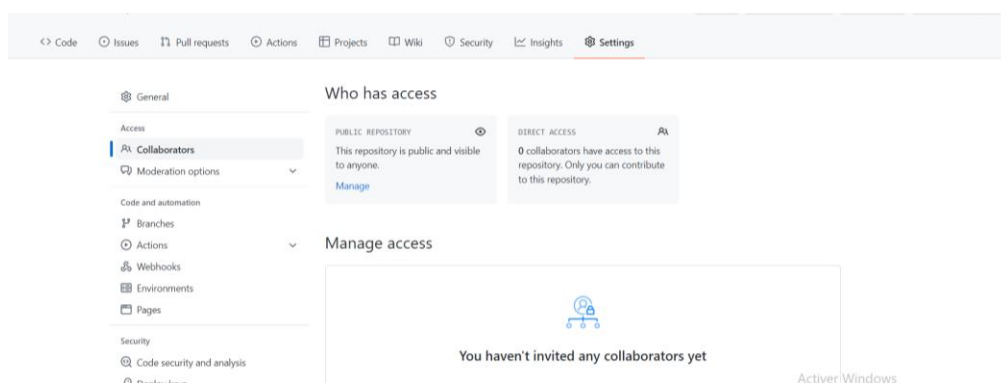
```

ellou@LAPTOP-PDS1TVQ9 MINGW64 /d/Thèse/enseignement/portail web/TP/project (
master)
$ git push origin groupe1
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'groupe1' on GitHub by visiting:
remote:   https://github.com/Nessrinetesting/TP1/pull/new/groupe1
remote:
To https://github.com/Nessrinetesting/TP1.git
 * [new branch]      groupe1 -> groupe1

```

Étape 7: Invitation des collaborateurs sur le projet TP1

Cliquer sur le répertoire **TP1** et sélectionner l'onglet **Settings** puis **collaborators** pour ajouter des Coéditeurs sur ce projet.



TP 2 : L'outil de travail collaboratif « Trello »

Ce TP vise à manipuler un nouvel outil de travail collaboratif « Trello » en apprenant ses notions de base et en explorant les fonctionnalités de base de cet outil.

1. Introduction

Trello est parmi les meilleures applications qui suivent les méthodes agiles en implémentant **le tableau kanban** pour gérer l'avancement d'un tel projet. Avant, Trello a été destiné aux petits projets. En intégrant les nouveaux services de timeline et de tableau de bord de pilotage, il est devenu très populaire et utilisé par des millions de personnes. L'application « Trello » est offerte gratuitement aux équipes de professionnels, d'étudiants et de groupes qui souhaitent collaborer afin de travailler leurs projets.

2. Principe de fonctionnement

Pour utiliser cet outil, il suffit de créer un compte. Une fois ce compte est ouvert, un premier tableau peut être réalisé. On peut après créer des colonnes pour les étapes du projet courant. Un tableau crée contient les tâches simples présentés sous les colonnes suivantes ("A faire", "En cours" et "Fait"), ou des tâches plus difficiles. Prenons l'exemple de création des sprints selon la méthodologie « Scrum » pour obtenir le **Scrumban** au lieu du **Kanban**.

3. Les principaux composants de « Trello »

Cet outil de travail collaboratif est basé sur trois fondements :

- Le tableau : C'est le projet sur lequel les membres d'une équipe travailleront en mode collaboratif.
- La liste : permet de répartir les tâches à accomplir dans le cadre d'un projet.
- La carte : sert à mentionner les tâches qui doivent participer à la réussite du projet. On peut ajouter à une carte des descriptions, des commentaires et des pièces jointes. En ajoutant un deadline à une carte, ses membres recevront des notifications.

4. Manipulation :

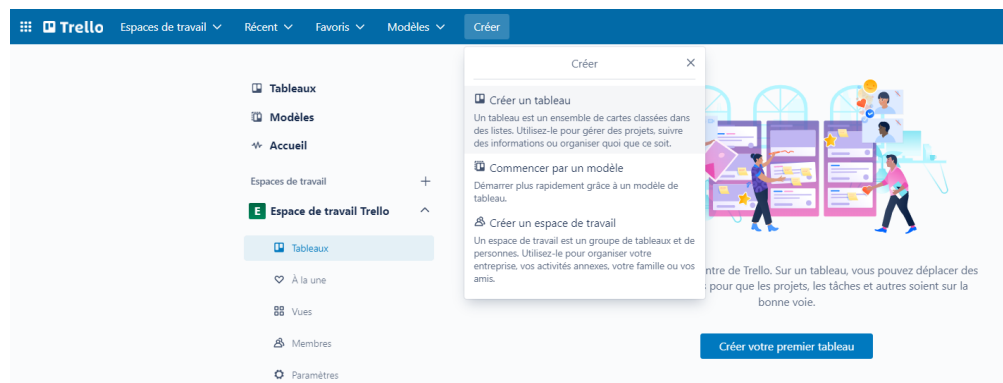
Etape 1: Inscription et personnalisation de votre profil Trello

- Il faut s'inscrire gratuitement sur Trello pour démarrer ([Créer un compte Trello](#)).
- On peut personnaliser ou modifier le profil Trello

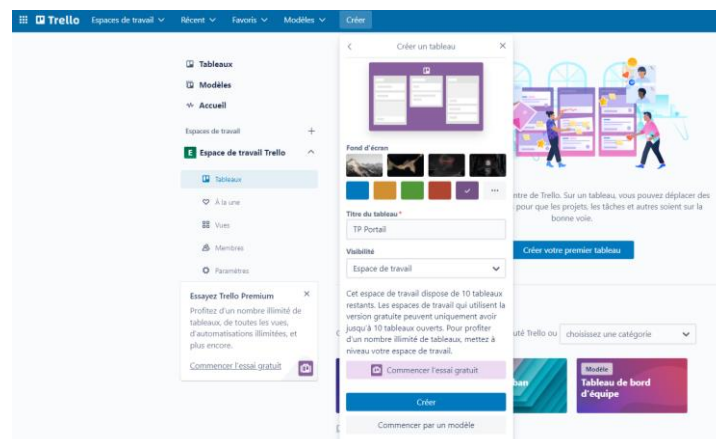


Etape 2: Création d'un tableau

- Créer un tableau en cliquant sur « **Créer un tableau** » en haut à gauche de l'espace (accueil).

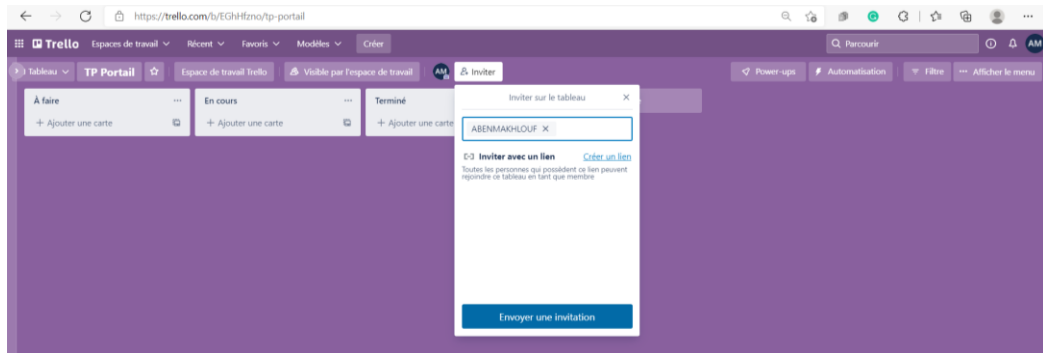


- Il faut maintenant créer un premier tableau en le donnant un nom (Par exemple: *TP Portail*).
- On pourra changer la couleur du fond.

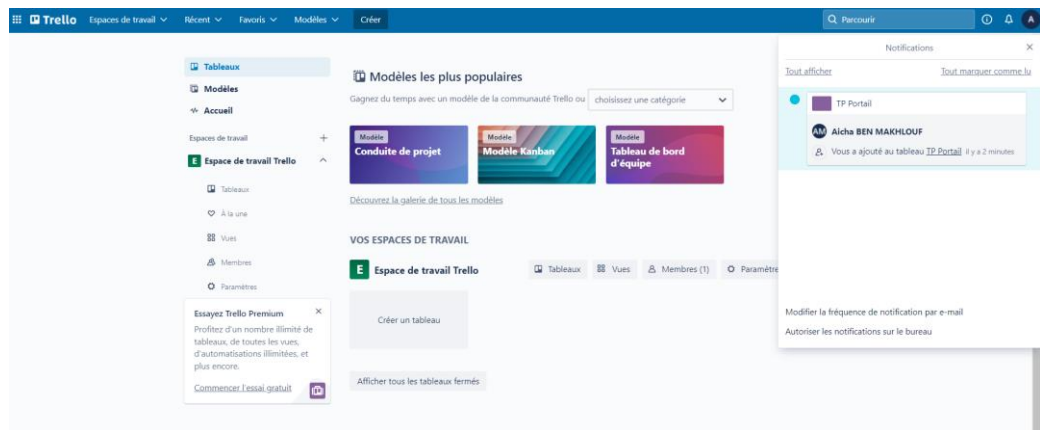


Étape 3: Ajout des membres et attribution des rôles:

- Si le projet est créé, on peut ajouter des membres ou attendre la création des listes et des cartes du projet.

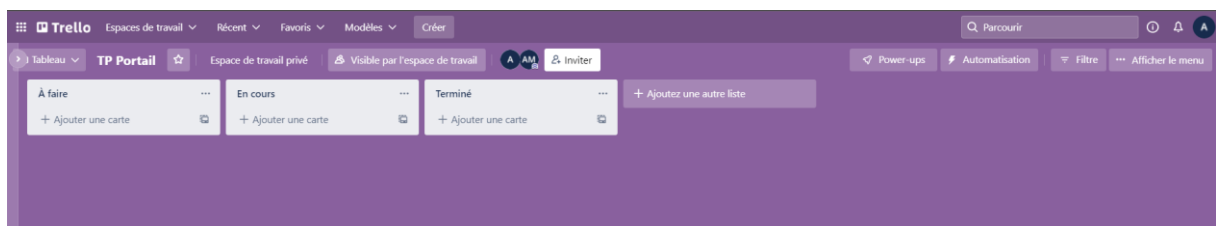


- La personne invitée recevra une notification (en haut à droite)

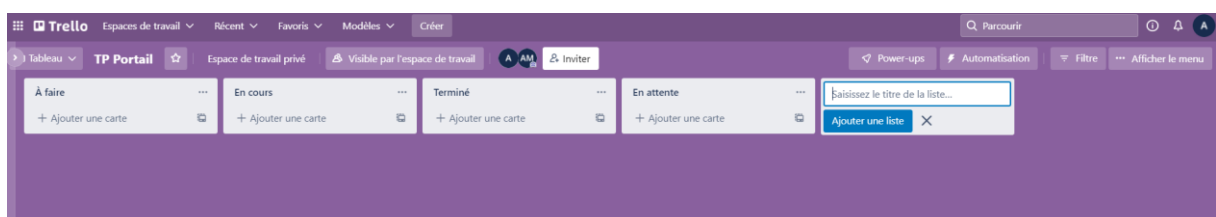


Étape 4: Ajout des listes:

- On peut ajouter maintenant de nouvelles listes en cliquant simplement sur « Ajoutez une autre liste ».

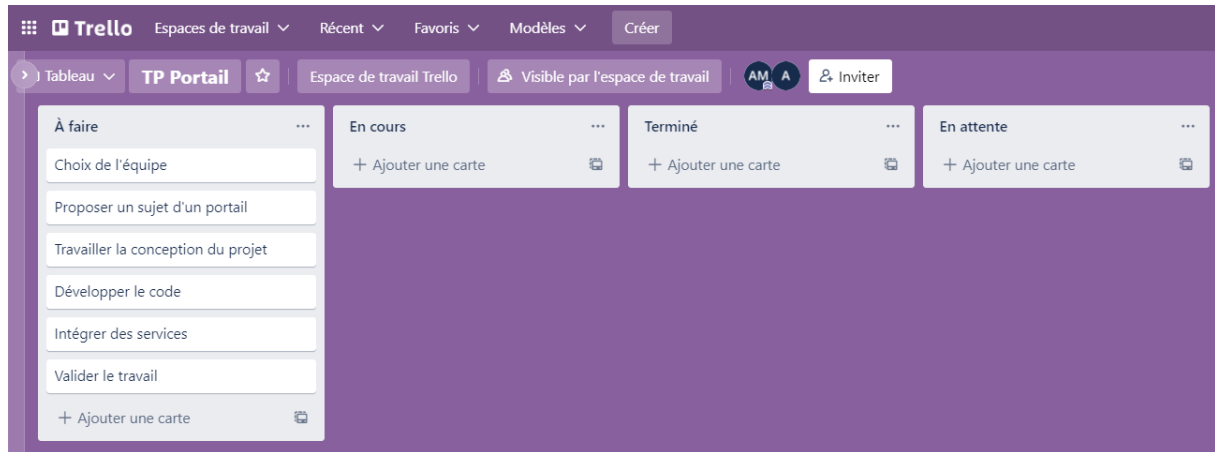


- Ajouter la liste des tâches en attente en la nommant « *En attente* ».



Etape 5: Ajout des cartes:

Maintenant, on peut créer nos cartes en cliquant tout simplement sur « Ajouter une carte » et en lui attribuant un nom.

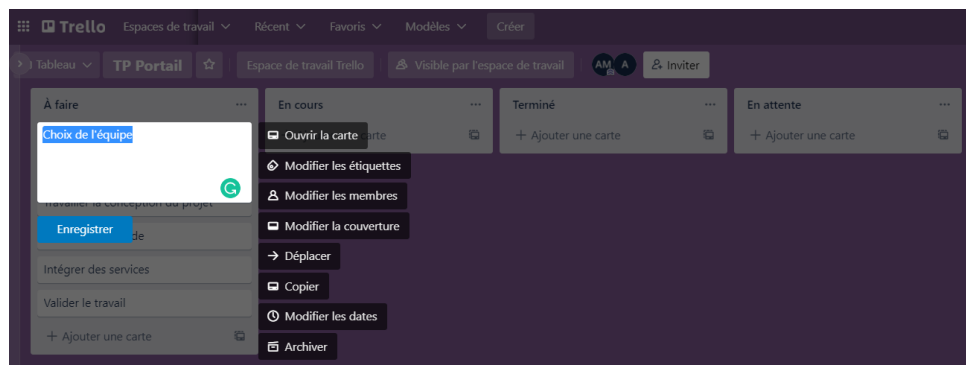


Etape 6: Modification des cartes:

Une fois la carte est créée, il y a deux possibilités afin de la modifier et de la gérer.

- **Modification de la carte**

Cliquer sur le crayon à droite de la carte, on peut accéder au différentes options:



1. Modification des étiquettes.
2. Modification des membres qui accèdent à la carte.
3. Déplacement de la carte.
4. Copiage de la carte vers une liste.
5. Changement du deadline du travail.
6. Archivage (Toutes les tâches sont accomplies).

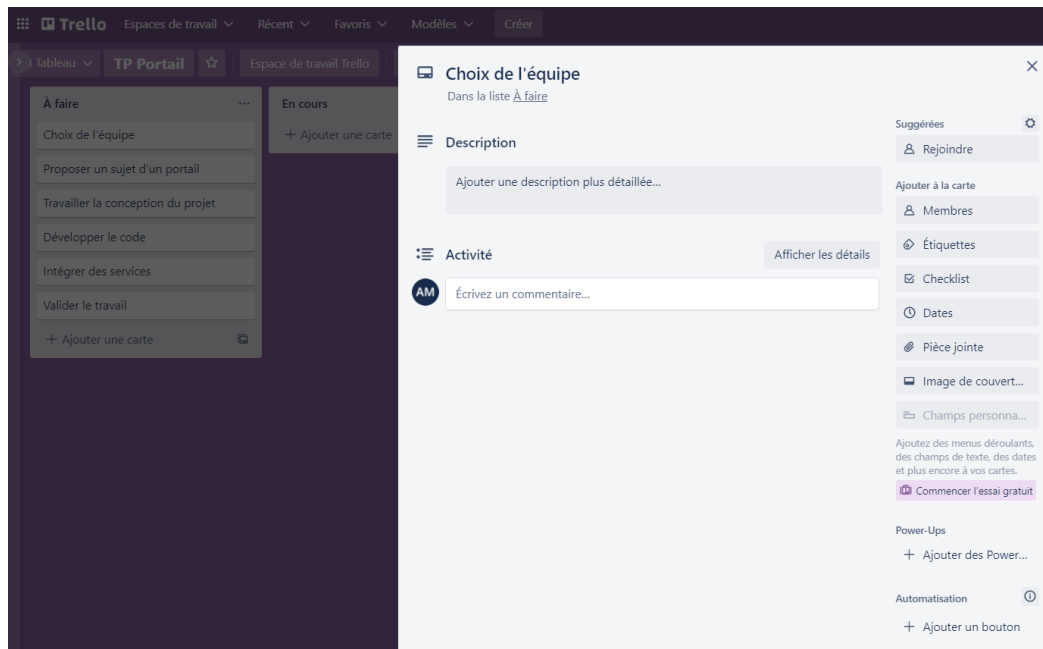
- **Menu Gérer la carte**

Ce menu est accessible directement lors du clic sur le bouton de la carte.

On peut faire l'ajout des éléments et la gestion des actions:

On peut ajouter les éléments suivants:

1. Des membres
2. Des étiquettes
3. Des checklists
4. Des échéances
5. Des pièces jointes



Pour la liste des actions qu'on peut exécuter :

1. Déplacement de la carte sur une liste
2. Copiage de la carte vers une liste
3. S'abonner
4. Archiver la carte

✕

Choix de l'équipe

Dans la liste [À faire](#)

Étiquettes

+

Date limite

lundi à 20:32

▼

☰

Description

Ajouter une description plus détaillée...

☑

Checklist

0%

Ajouter un élément

Supprimer

☰

Activité

AM

Écrivez un commentaire...

Afficher les détails

Suggérées

⚙

Rejoindre

Ajouter à la carte

Membres

Étiquettes

Checklist

Dates

Pièce jointe

Image de couvert...

Champs persona...

Ajoutez des menus déroulants, des champs de texte, des dates et plus encore à vos cartes.

Commencer l'essai gratuit

Power-Ups

+ Ajouter des Power...

Automatisation

ⓘ

+ Ajouter un bouton

Actions

→ Déplacer

📄 Copier

🏗 Créer un modèle

👁 Suivre

📁 Archiver

🔗 Partager

Nessrine ELLOUMI et Aicha ben Makhlouf

16

TP 3 : Espace de travail collaboratif : Slack

L'objectif de ce TP est de maîtriser un espace de travail collaboratif : « Slack ». C'est outil permet de suivre l'évolution d'un projet, localement et à distance. Slack permet aux membres de l'équipe de communiquer facilement en intégrant plusieurs plugins de travail collaboratif. Dans ce TP nous allons travailler avec l'outil de travail collaboratif Slack.

1. Introduction

Slack est une application dédiée pour les entreprises. Slack transforme la communication des organisations en mettant en contact les personnes pour les faire collaborer comme une seule équipe unifiée.

2. Maîtriser l'environnement « Slack »

Slack est un logiciel qui vous permet de travailler d'une façon connecté, flexible et inclusive.

Connecté

C'est outil nous permet de communiquer facilement, échanger des messages avec des membres à l'intérieur ou à l'extérieur de l'organisation.

Flexible

Ce logiciel permet de travailler d'une façon asynchrone.

Inclusif

Dans ce logiciel, tous les membres ont accès aux mêmes informations. Les informations sont partagées d'une façon simultanée avec tout le monde, ce qui facile aux membres de l'équipe de discuter et prendre des décisions rapidement.

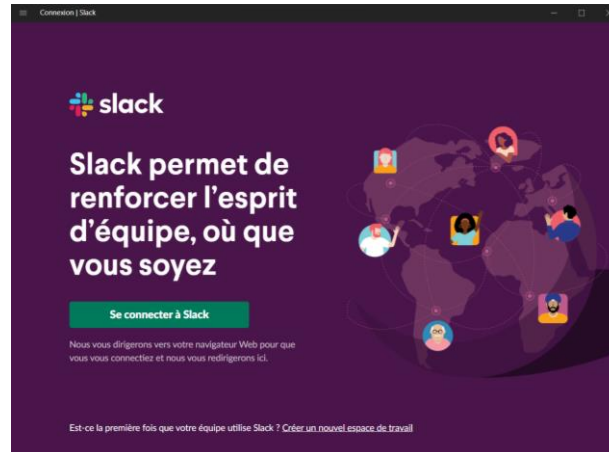
3. Manipulation :

Maintenant, vous allez installer Slack sur votre ordinateur, créer un espace de travail sous Slack et apprendre à utiliser ses fonctionnalités.

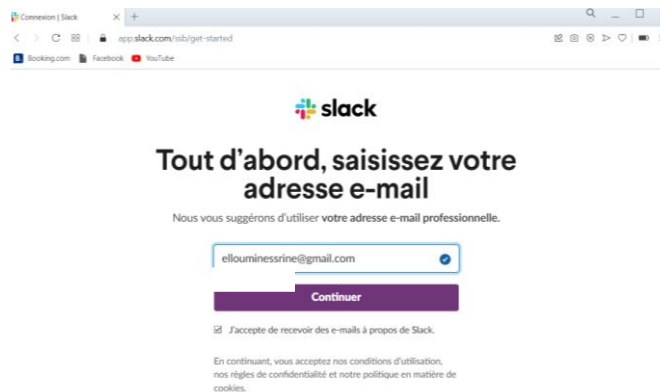
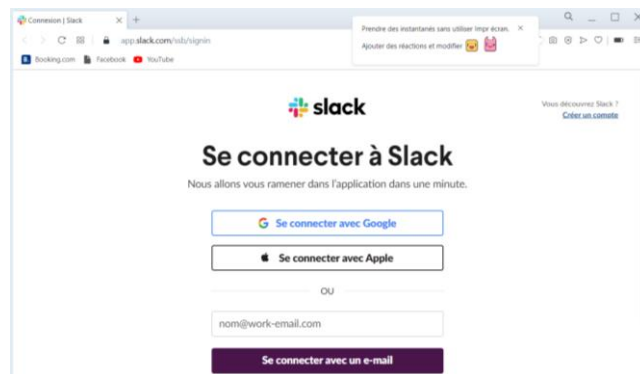
Etape 1 : Téléchargement et installation du logiciel Slack sur vos PC

Pour utiliser Slack il faut créer un compte sur <https://slack.com/> puis configurer votre espace de travail. Voici les étapes nécessaires pour configurer un espace de travail sous Slack.

1. Cliquer sur « Se connecter à Slack »



2. Saisir votre adresse E-mail pour se connecter à Slack.



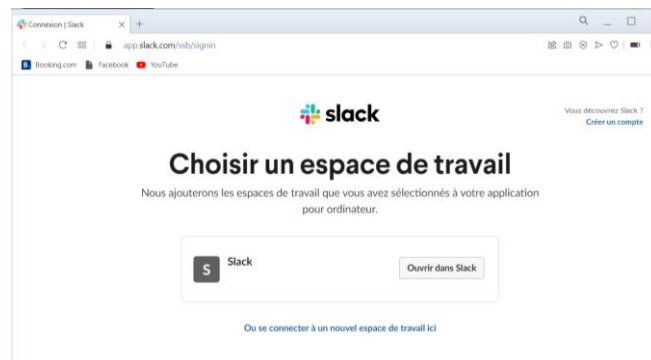
Pour valider la configuration de votre espace de travail un E-mail qui contient un code de validation sera envoyer à votre adresse mail que vous avez saisissez dans l'étape d'inscription.

3. Entrer votre code de validation.

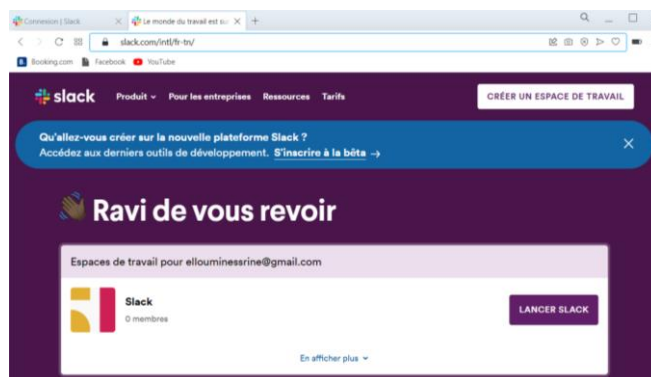


Après l'étape de configuration en ligne, Slack fait une synchronisation automatique avec l'application « Slack » que vous avez déjà installé.

4. Pour faire la synchronisation cliquer sur « Ouvrir dans Slack ».



L'application Slack local (Sur votre PC) va être lancer automatiquement.



Etape 2 : Configuration de votre espace de travail sur l'application Slack local

Pour configurer votre espace de travail, vous devez exécuter les étapes suivantes :

1. Entrer le nom de votre équipe :



ISSAT

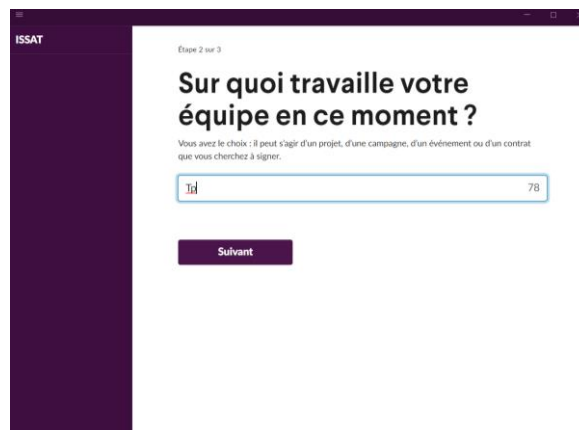
Étape 1 sur 3

Quel est le nom de votre entreprise ou de votre équipe ?

Ceci sera le nom de votre espace de travail Slack. Choisissez quelque chose de reconnaissable pour votre équipe.

 45

2. Définir le sujet sur lequel vous travaillez, exemple : « Tp »
Sous Slack les sujets sont représentés par le label « Canal ».



ISSAT

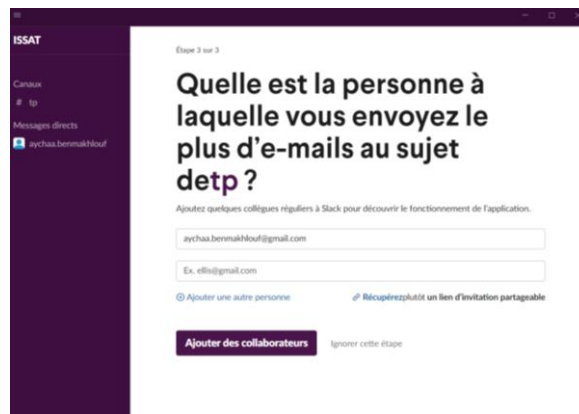
Étape 2 sur 3

Sur quoi travaille votre équipe en ce moment ?

Vous avez le choix : il peut s'agir d'un projet, d'une campagne, d'un événement ou d'un contrat que vous cherchez à signer.

 78

3. Pour ajouter des collaborateurs à votre espace de travail, il faut entrer leurs adresses électroniques.



ISSAT

Étape 3 sur 3

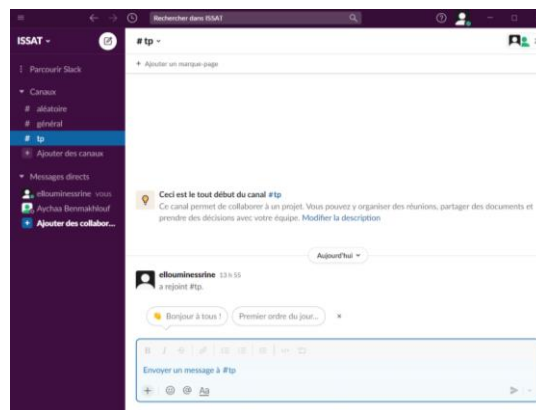
Quelle est la personne à laquelle vous envoyez le plus d'e-mails au sujet detp ?

Ajoutez quelques collègues réguliers à Slack pour découvrir le fonctionnement de l'application.

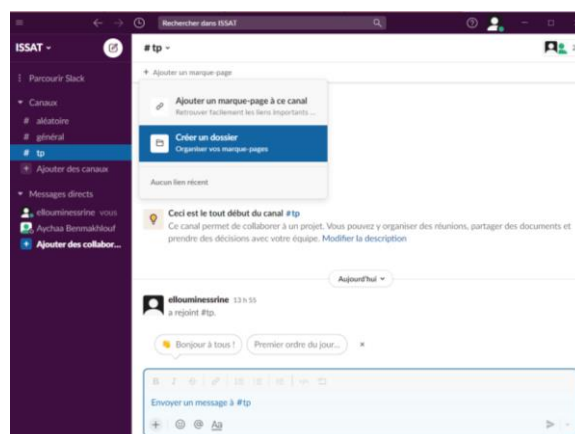
4. Un E-mail d'invitation est envoyé à l'adresse électronique de votre collaborateur que vous avez saisie.



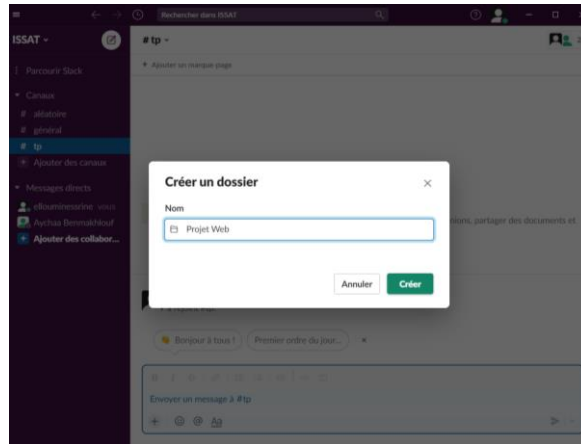
Il suffit que l'invitation soit envoyée, le nom de votre collaborateur s'affiche dans la partie Message direct.



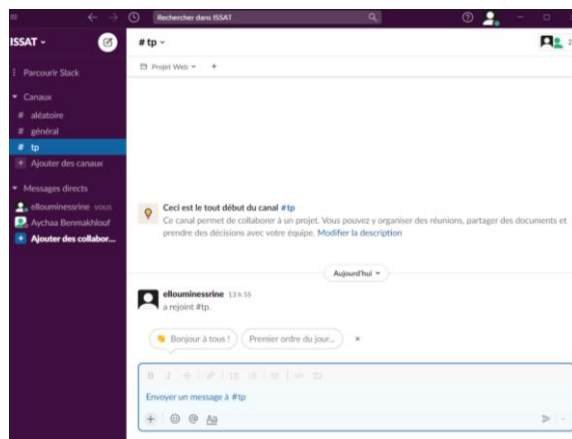
5. Dans l'espace de travail « tp », créer un dossier appeler « Projet Web ».



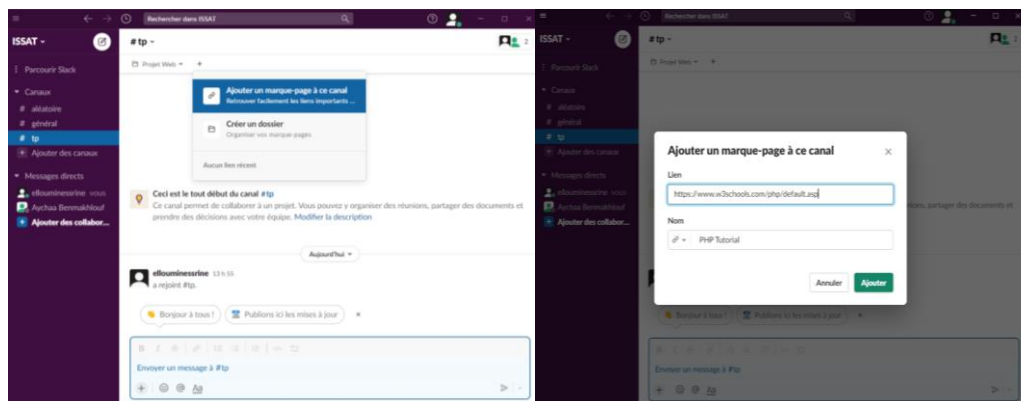
6. Saisir le nom de votre Projet « Projet Web »



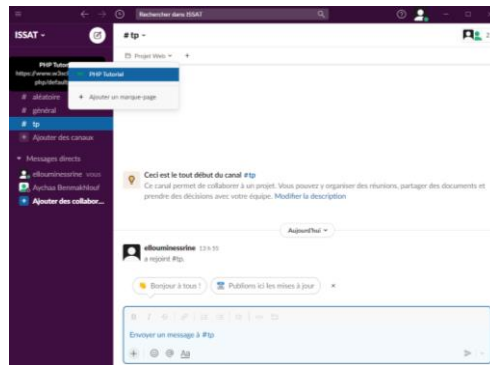
Le dossier « projet web » est créé sous votre projet « tp » s'affiche sous la rubrique « Canaux » qui se trouve sous votre espace de travail ISSAT.



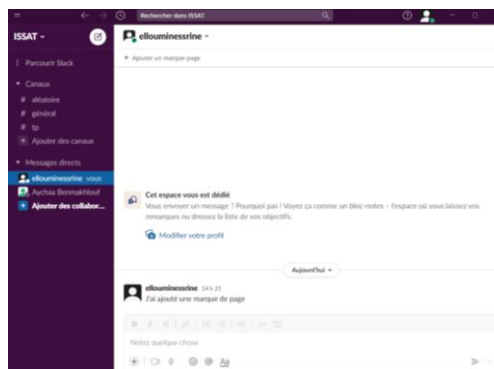
7. Dans le dossier « Projet web », ajouter des ressources par exemple des liens pour apprendre le langage PHP.



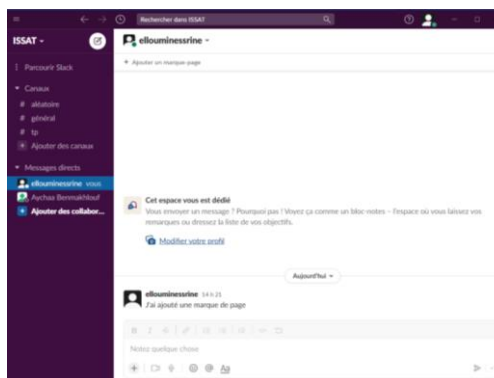
Le lien que vous avez créé est associé au dossier « Projet web » sous le nom « PHP Tutoriel ». Slack définit le nom des liens d'une manière automatique suivant le contenu du lien.



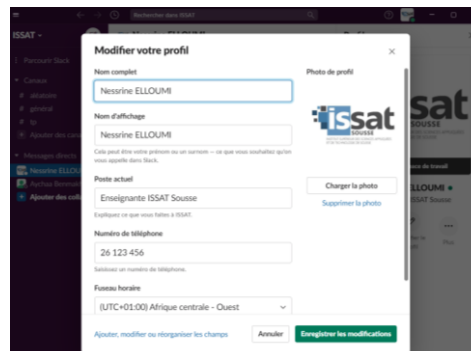
Dans l'espace de travail dans lequel vous êtes connectés, Slack offre un espace d'envoi de message entre les collaborateurs du projet.



8. Slack offre aux utilisateurs de créer un profil.



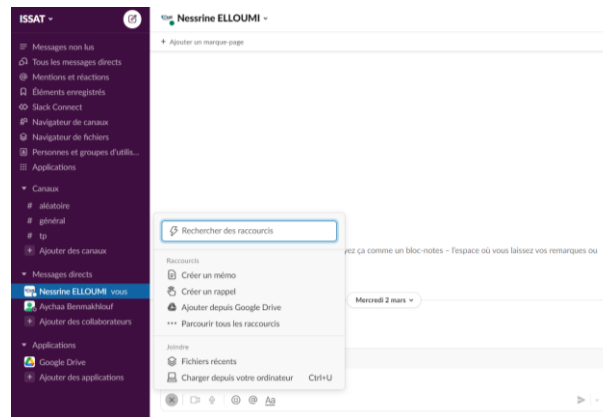
9. Cliquer sur votre nom est modifier les informations de votre profil.



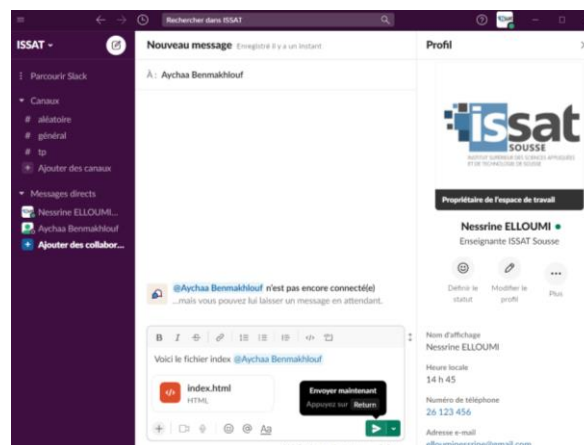
Etape 3 : Dépôt de fichiers

Slack permet à ses utilisateurs de Déposer des ressources « Fichier » exemple texte, image, vidéo etc...

1. Cliquer sur votre nom, sur l'icône (+), puis cliquer sur charger depuis votre ordinateur.



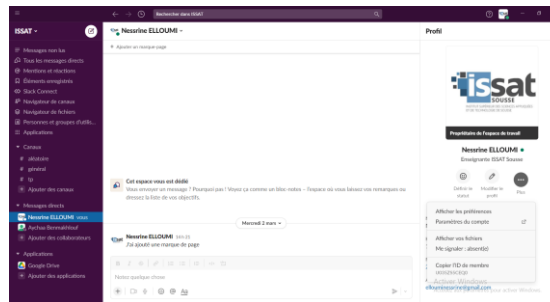
2. Choisir le fichier que vous voulez déposer, puis cliquer sur envoyer.



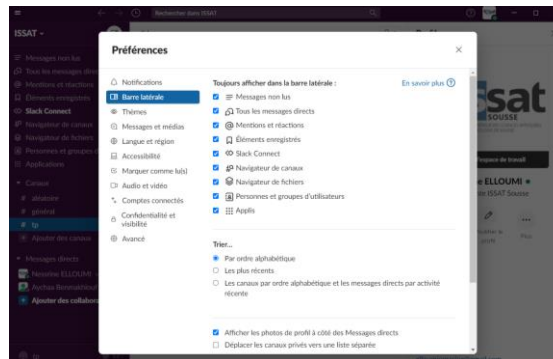
Etape 4 : Ajout des anglets de fonctionnalités sous Slack

Pour ajouter des anglets sous votre espace de travail, il faut travailler les étapes suivantes :

1. Cliquer sur le nom, dans la partie Profil cliquer sur « Plus » puis sur « Afficher les préférences ».



2. Choisir l'onglet barre latérale puis cochez les barres que vous voulez afficher.



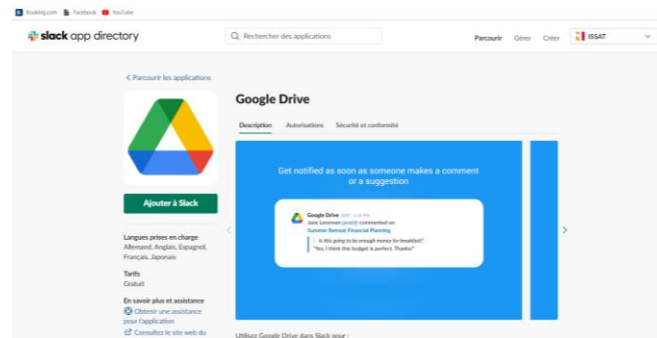
Etape 5 : Ajout des Plugin à Slack.

Slack permet à ses utilisateurs d'ajouter plusieurs applications qui permettent de faciliter le travail collaboratif dans un équipe exemple : Trello, Google Drive, Zoom, Google Calender etc... Dans ce Tp nous allons ajouter le plugin « Google Drive » à Slack.

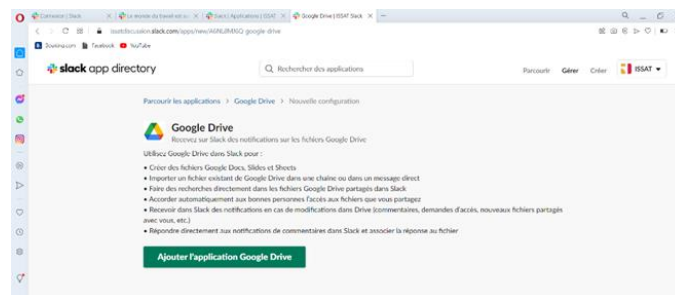
1. Se connecter à votre espace de travail en ligne sur Slack, Choisir l'application « Google Drive ».



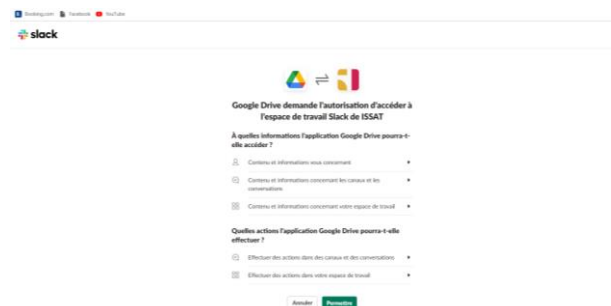
2. Cliquer sur « Ajouter à Slack ».



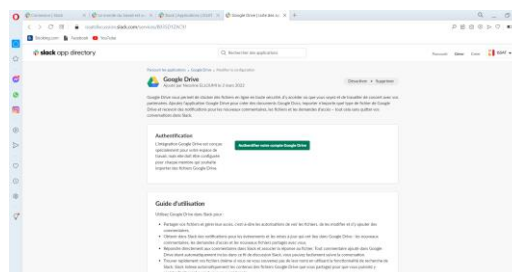
3. Cliquer sur « Ajouter l'application Google Drive »



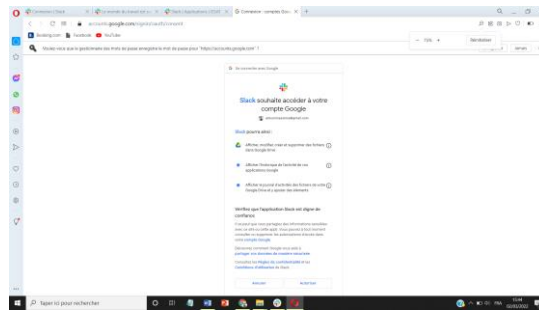
4. Google Drive demande l'autorisation d'accéder à l'espace de travail Slack de L'ISSAT. Pour autoriser l'accès cliquer sur « Permettre ».



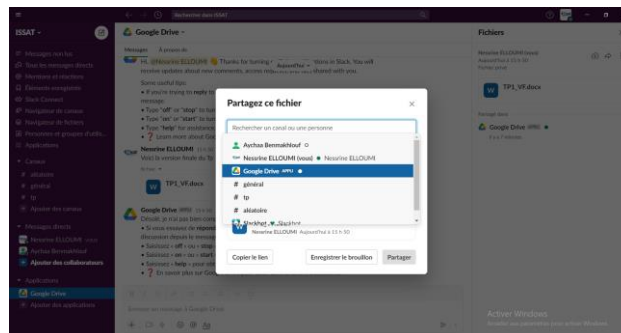
5. Enregistrer l'authentification sur Google Drive.



6. Autoriser l'accès de Slack a votre espace « Google drive ».



7. Pour partager un fichier sous « Google drive » via « Slack » cliquer sur l'onglet application, choisir « Google Drive » puis sélectionner vos collaborateurs.



TP 4 : Espace de travail collaboratif : Jira

Dans ce TP, vous devez maîtriser l'espace de travail collaboratif « Jira ». Pour que les membres d'une équipe communiquent facilement et afin de partager le travail en groupe ainsi que travailler à distance, nous allons aborder dans ce TP l'outil de travail collaboratif « Jira ».

1. Introduction

Jira est un logiciel qui permet de gérer les projets, développé par Atlassian en 2002, répond aux besoins des équipes de développement en mode agile. Ce logiciel permet de planifier, suivre et livrer les projets rapidement et efficacement tout cela dans une logique de gestion de projet agile.

2. Travailler dans l'environnement Jira

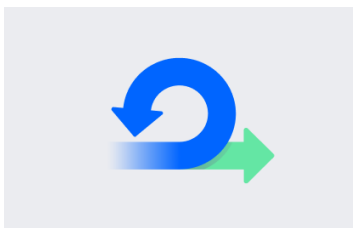
Dans ce TP, vous allez manipuler l'outil de travail collaboratif Jira, créer un projet en choisissant son type, créer des tickets en précisant le type, inviter l'équipe de travail, ajouter des outils et apprendre à utiliser les fonctionnalités de base selon le modèle du projet choisit.

Il existe plusieurs modèles dans la gamme de produits Jira.

Les modèles les plus connus que Jira Software propose sont les suivants :

- **La méthodologie Scrum**

C'est utilisée par les équipes Agile utilisant un backlog, pour planifier et estimer leur travail en sprints.



- **Kanban**

Afin que des équipes suivent leurs travaux en utilisant un backlog Kanban.



- **Suivi des bugs**

Idéal pour les équipes qui n'ont pas besoin d'un tableau Kanban et préfèrent gérer les tâches de développement et les bugs sous forme de liste.



Types de projet

Concernant les modèles **Scrum** et **Kanban**, on doit sélectionner un type de projet. La différence fondamentale entre ces deux types de projets réside dans la manière dont ils sont gérés et si la gestion se fait au niveau de l'équipe ou au niveau de l'entreprise/de l'administrateur Jira.

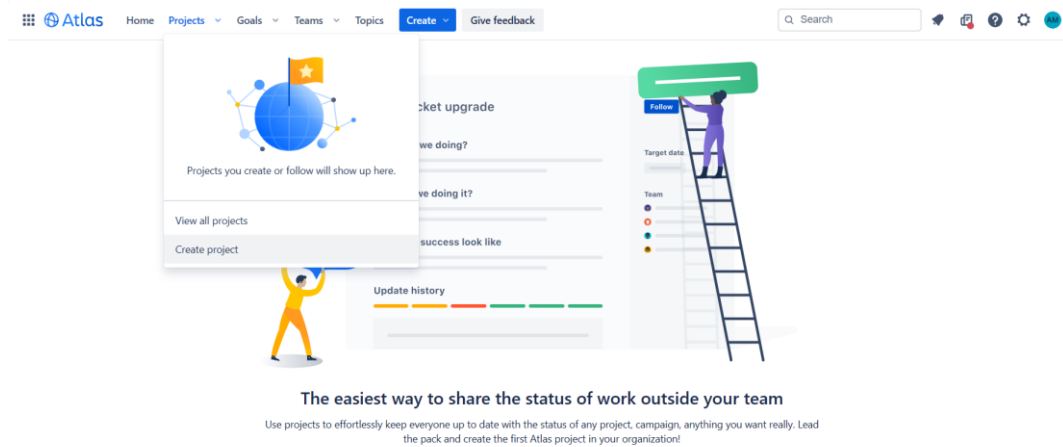
- ✚ Les **projets gérés en équipe** conviennent aux équipes indépendantes souhaitant contrôler leurs propres processus et pratiques de travail dans un espace autonome.
- ✚ Les **projets gérés par l'entreprise** sont configurés et maintenus par les administrateurs Jira. Ils sont conçus pour les entreprises qui souhaitent standardiser leur façon de travailler au sein de plusieurs équipes, comme les flux de travail partagés.

3. Manipulation

Étape 1 – Création d'un projet

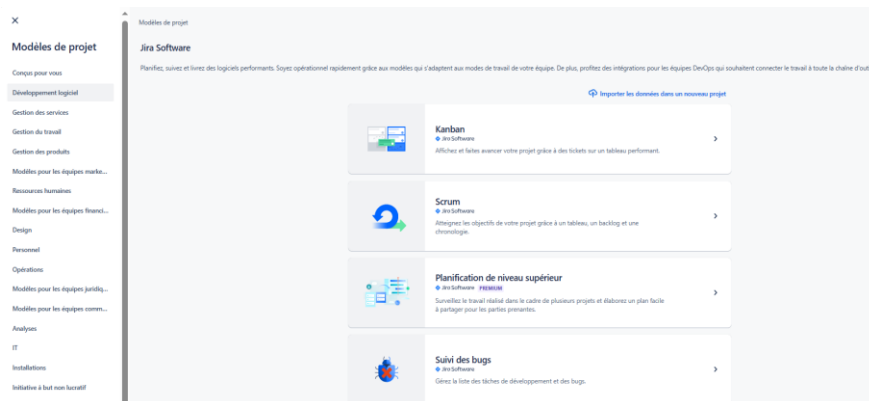
Se connecter à votre compte de Jira Software.

Dans la barre de navigation supérieure, sélectionnez le menu déroulant « Projets », puis « Créer un projet ».

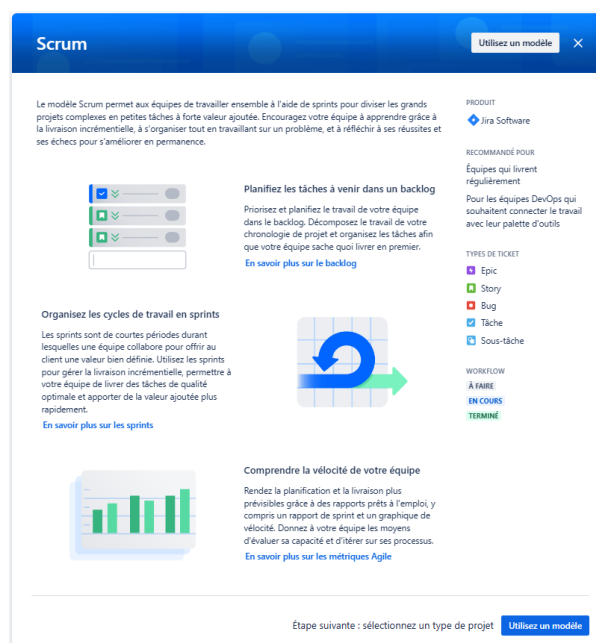


Étape 2 – Sélection d'un modèle

Sélectionnez le modèle qui correspond le mieux aux méthodes de travail uniques de votre équipe.



Par exemple : Choisissez le modèle **Scrum**



➔ Les deux types de projets s'affichent: projet géré par l'équipe / un projet géré par l'entreprise.

1 Modèle de projet

Scrum
Jira Software
Atteignez les objectifs de votre projet grâce à un tableau, un backlog et une chronologie.
Changer de modèle

2 Choisir un type de projet

⚠ Vous devrez créer un projet si vous décidez de changer de type de projet ultérieurement.

Géré par l'équipe

Configuration et gestion par votre équipe.

Pour les équipes qui veulent contrôler leurs propres processus et pratiques de travail dans un espace autonome. Choisissez à la carte des fonctionnalités Agile pour soutenir votre équipe à mesure qu'elle prend de l'ampleur ou gagne en complexité.

Configuration simplifiée

Soyez opérationnel rapidement, grâce à une configuration simplifiée.

Géré par l'entreprise

Configuration et gestion par vos administrateurs Jira.

Pour les équipes qui souhaitent travailler avec d'autres équipes sur plusieurs projets de manière standard. Encouragez et favorisez les bonnes pratiques et les processus organisationnels grâce à une configuration partagée.

Configuration pour les experts

Bénéficiez d'un contrôle complet grâce à une configuration, une personnalisation et une flexibilité assurées pour les experts.

Sélectionner un projet géré par l'équipe

Sélectionner un projet géré par l'entreprise

1. Choisir « Sélectionner un projet géré par l'équipe »

- Ajouter les informations de votre projet

Ajouter les informations du projet

Découvrez le champ des possibles lorsque vous collaborez avec votre équipe. Modifiez les informations du projet à tout moment dans les paramètres le concernant.

Nom *

TP portail

Accès Toutes les personnes ayant accès à aichabenmakhlouf peuvent accéder à ce projet et l'administrer. [Passez à la version supérieure](#) pour personnaliser les autorisations du projet.

Cle *

TP

Modèle

Changer de modèle

Scrum
Jira Software
Atteignez les objectifs de votre projet grâce à un tableau, un backlog et une chronologie.

Type

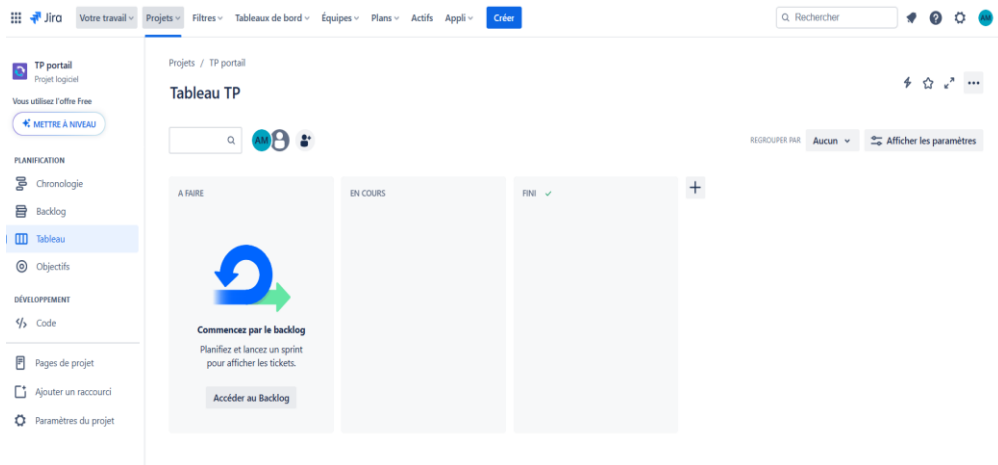
Changer de type

Géré par l'équipe
Contrôlez vos propres processus et pratiques de travail dans un espace autonome.

Annuler

Suivant

➔ Le tableau du projet géré par l'équipe est maintenant créée.



2. Choisir « Sélectionner un projet géré par l'entreprise »

Ajouter les informations du projet

Découvrez le champ des possibles lorsque vous collaborez avec votre équipe. Modifiez les informations du projet à tout moment dans les paramètres le concernant.

Nom *

Clé *

☐ Partager les paramètres avec un projet existant

Modèle

Changer de modèle



Scrum

Jira Software

Atteignez les objectifs de votre projet grâce à un tableau, un backlog et une chronologie.

Type

Changer de type



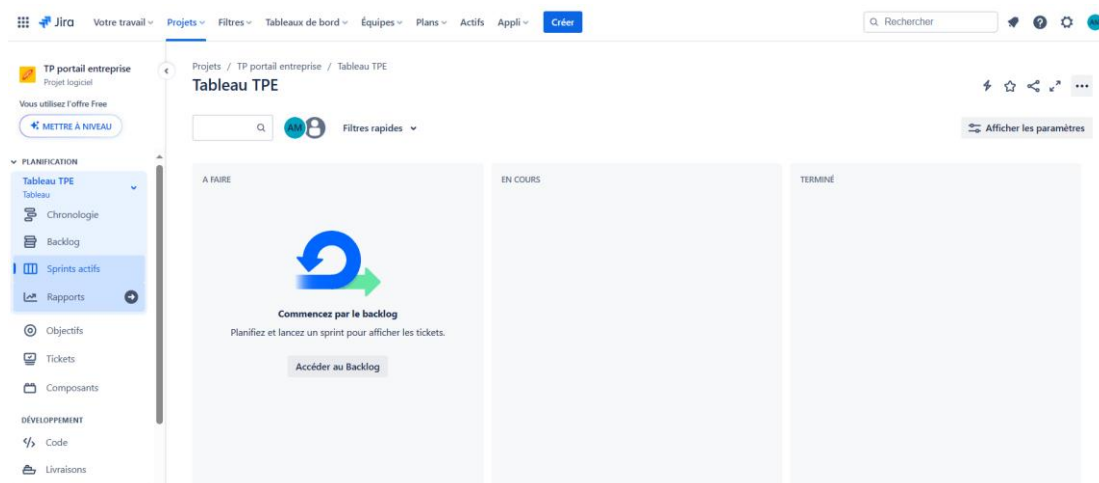
Géré par l'entreprise

Travaillez avec d'autres équipes sur plusieurs projets de manière standardisée.

Annuler

Suivant

➔ Le tableau du projet géré par l'entreprise est maintenant créée.



Étape 3 – Configuration des colonnes

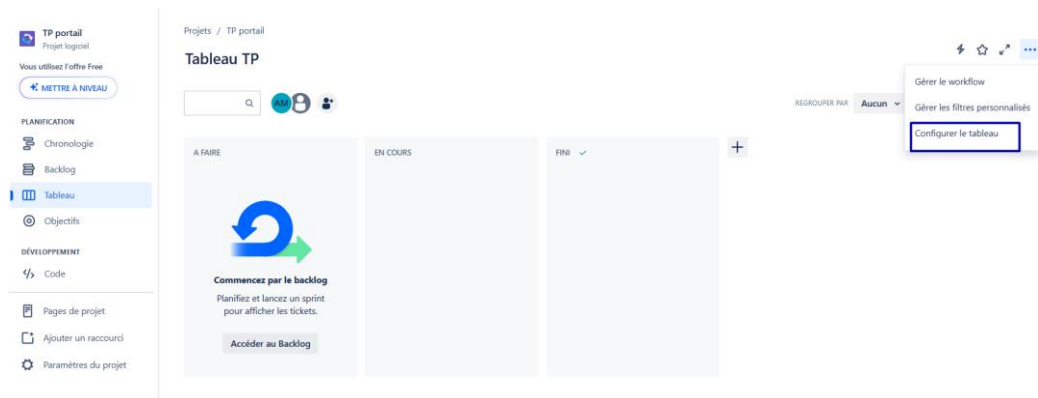
Dans Jira Software, le tableau affiche une sélection de tickets sous forme de colonnes, chaque colonne représentant une étape du workflow de votre équipe jusqu'à l'achèvement du travail.

La configuration des colonnes sur votre tableau selon les méthodes Scrum et Kanban dépend de si votre projet est géré par une équipe (directement sur le tableau) ou par une entreprise (dans les paramètres du tableau).

Dans les projets gérés par l'équipe :

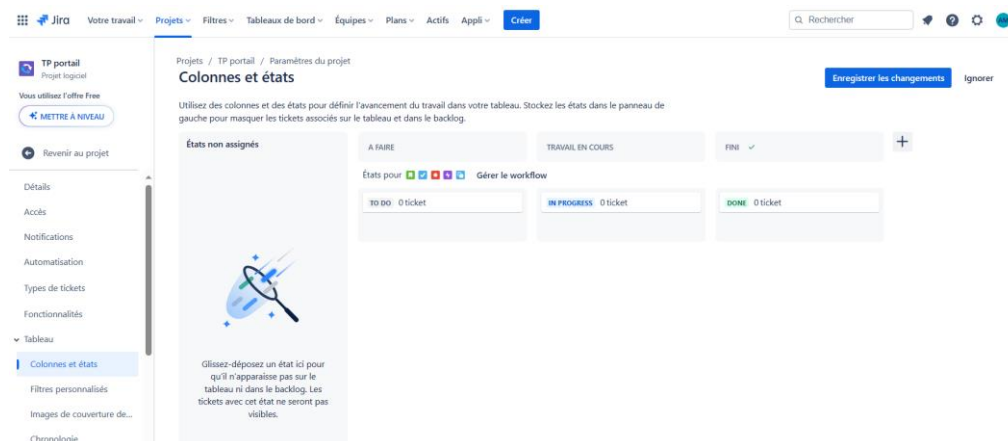
- Accédez à votre tableau.

Sélectionnez (•••) en haut à droite et cliquez sur Configurer le tableau.

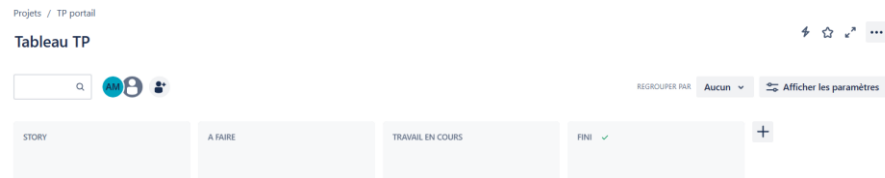


- Ajoutez, supprimez, renommez ou déplacez les colonnes selon vos besoins.

➔ Modification du nom d'une colonne :



➔ Ajout d'une colonne :

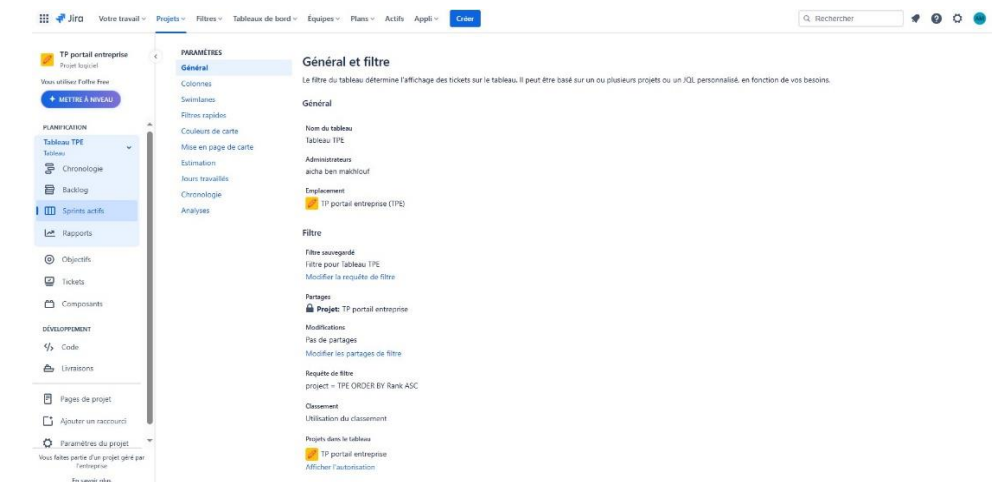


➔ Suppression une colonne :

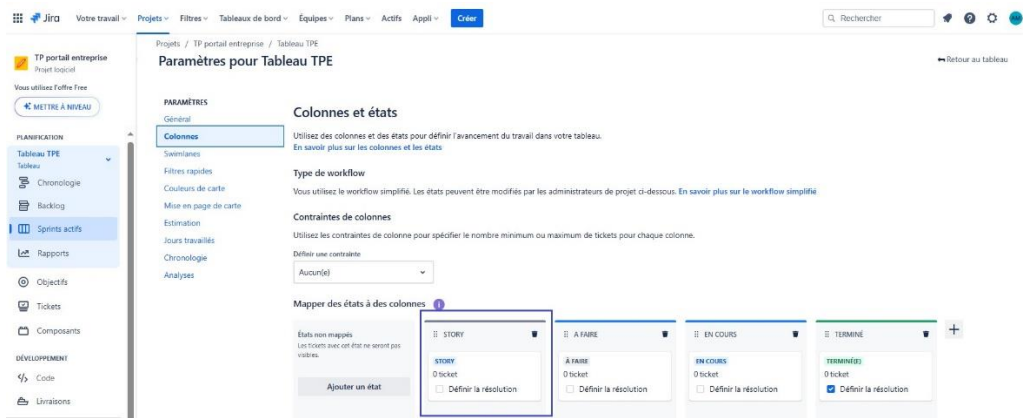


Dans les projets gérés par l'entreprise :

- Accédez à votre tableau. Sélectionnez (••) en haut à droite, puis cliquez sur Paramètres du tableau.



- Cliquez sur « Colonnes ».
- Ajoutez, supprimez, renommez ou déplacez les colonnes selon vos besoins.

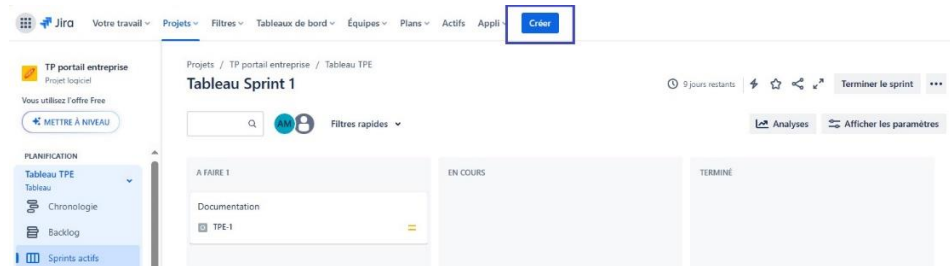


Étape 4 – Création d'un ticket

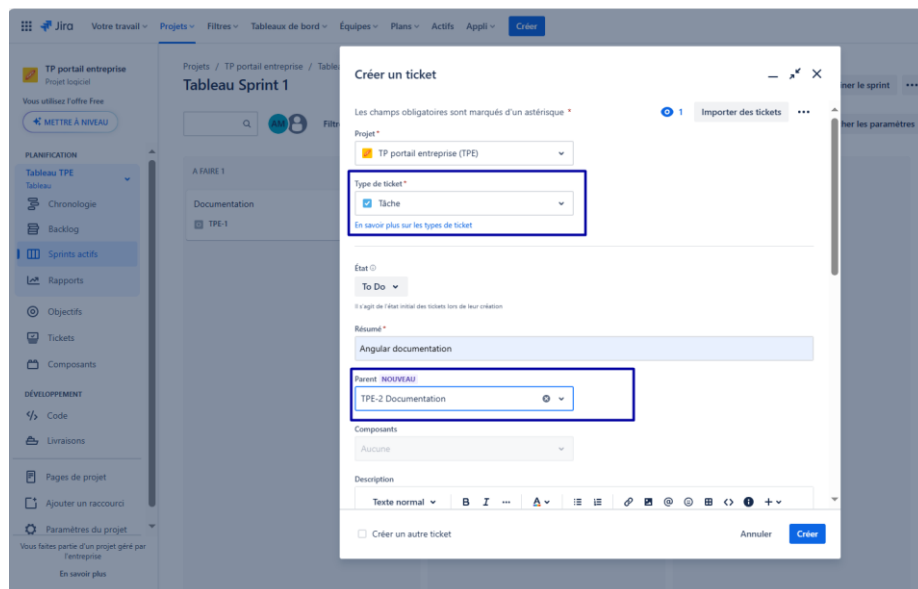
Les tickets constituent la pierre angulaire de votre projet Jira Software.

Un ticket peut représenter une story, une epic, un bug, une fonctionnalité à développer ou toute autre tâche de votre projet.

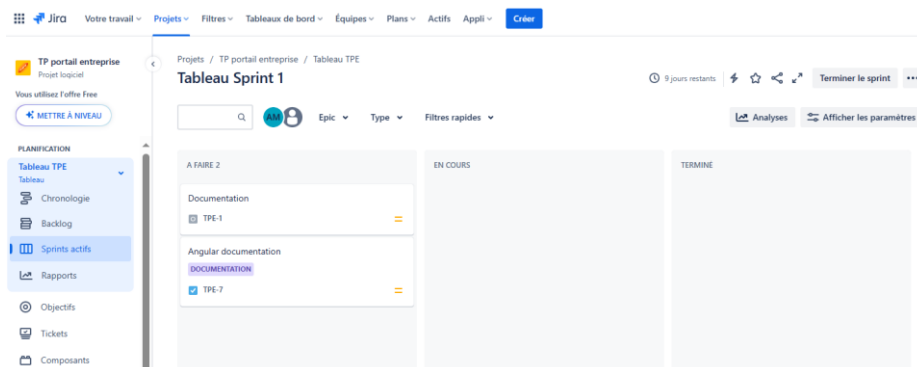
- Sélectionnez « Créer » dans le menu de navigation supérieur.



- Choisir le type de ticket

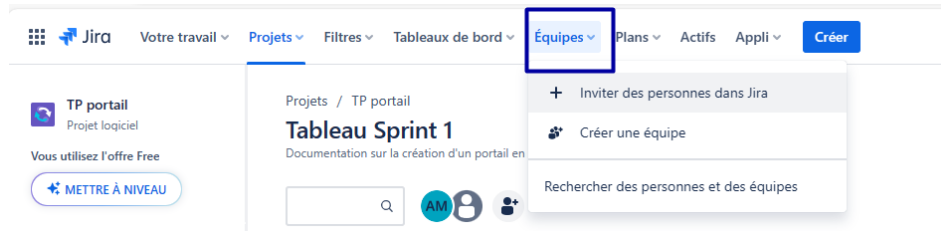


➔ Votre ticket apparaîtra dans le tableau du projet.

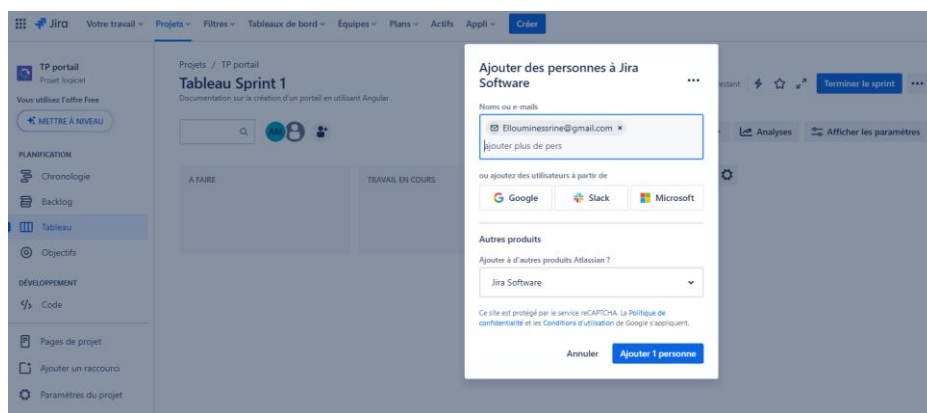


Étape 5 : Invitation de l'équipe

Pour ajouter les membres de l'équipe, cliquez simplement sur le bouton « Équipes » et ajoutez les membres.



➔ Vous pouvez ajouter des membres en entrant les adresses e-mail.



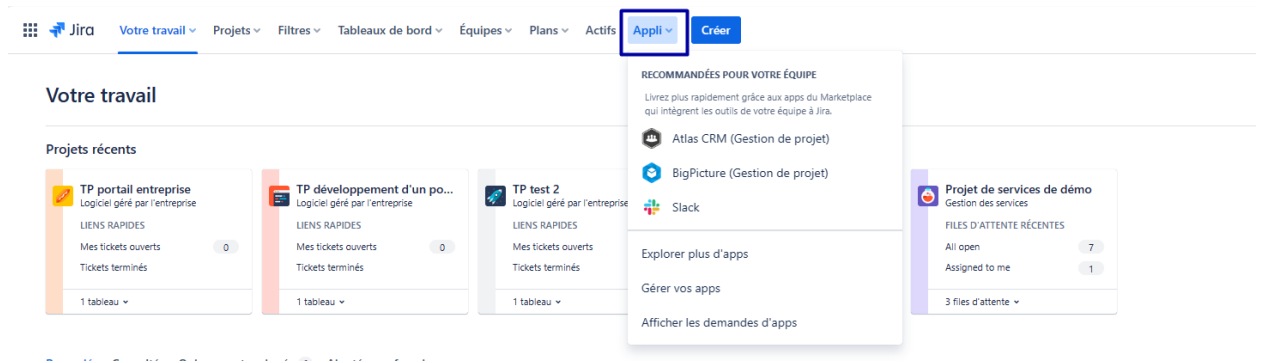
Vous êtes prêt à travailler ensemble maintenant que votre équipe a rejoint votre site Jira Software. Vous devrez créer et lancer un sprint si vous travaillez dans un projet Scrum. Vous pouvez commencer à suivre les principes de base d'un projet Kanban. Déplacez un ticket d'une colonne à l'autre à mesure qu'il progresse dans le processus de votre équipe pour suivre les tâches.

Étape 6 : Connexion des outils

Vos équipes et vous pouvez consacrer moins de temps à la gestion du travail et plus de temps au développement de logiciels exceptionnels grâce à plus de 3 000 apps Jira Software qui peuvent être personnalisées pour s'adapter à tous les cas d'usage.

Créez un espace de travail unifié grâce à plus de 3 000 apps et intégrations de l'Atlassian Marketplace.

- ✓ Sélectionnez la commande « **Appli** » en haut à droite dans le menu de navigation.



- ✓ Cliquez sur **Explorer plus d'apps**.
- ✓ Recherchez le nom d'une app ou choisissez une catégorie.
Exemple : Slack integration for Jira
- ✓ Suivez les instructions pour **installer**, **acheter** ou démarrer un **essai gratuit**.