

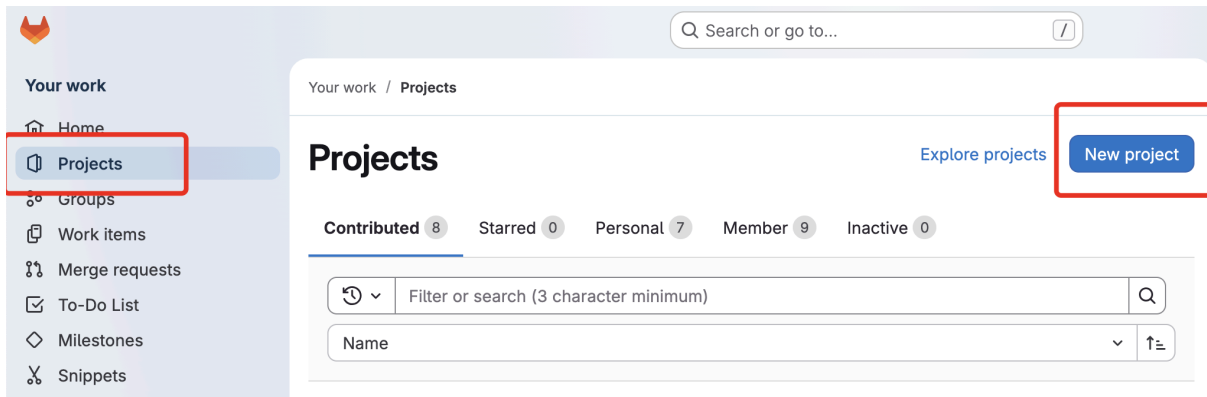
# Repository Gitlab

Pré-requis :

- Avoir une machine virtuel ou un environnement avec git d'installer ainsi que la configuration du compte et de la clé ssh

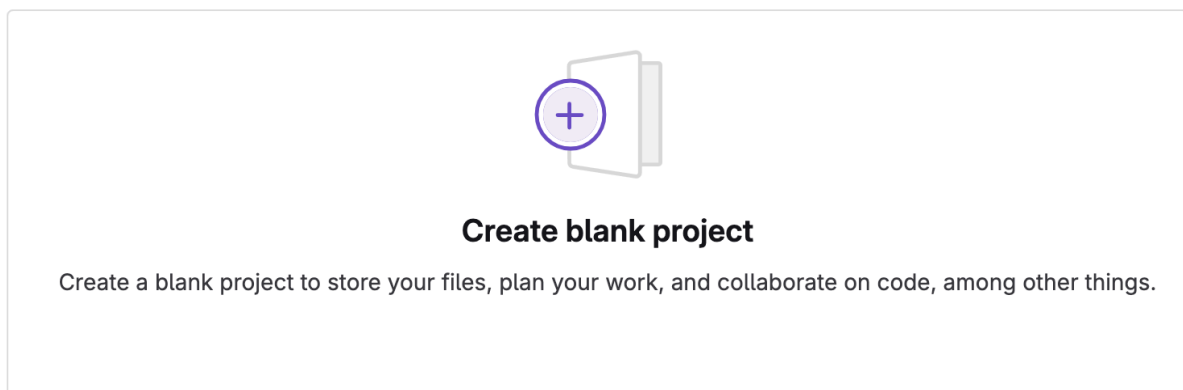
Créer le repository sur Gitlab :

New Project



Créer un projet en partant de zéro

## Create new project



Configurer le projet avec son nom, son URL, s'il est privé ou public (possibilité de changer plus tard). Bien penser à décocher le README puisque nous allons envoyer notre code depuis un projet existant sur notre machine.

# Create blank project

Create a blank project to store your files, plan your work, and collaborate on code, among other things.

## Project name

Must start with a lowercase or uppercase letter, digit, emoji, or underscore. Can also contain dots, pluses, dashes, or spaces.





## Project URL



## Project slug

## Project deployment target (optional)

## Visibility Level

-  Private  
Project access must be granted explicitly to each user. If this project is part of a group, access is granted to members of the group.
-  Internal   
The project can be accessed by any logged in user except external users.
-  Public  
The project can be accessed without any authentication.

## Project Configuration

- Initialize repository with a README  
Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.

## Créer ensuite le projet

[> Experimental settings](#)

On va désormais suivre la procédure que propose Gitlab avec quelques changements :

### Configure the Git repository

```
git init --initial-branch=main --object-format=sha1
git remote add origin git@gitlab.com:Sunliaa/gsb-appli.git
git add .
git commit -m "Initial commit"
git push --set-upstream origin main
```

Aller dans le dossier et le terminal du projet qu'on souhaite envoyer sur le repository et faire `git init --initial-branch=master --object-format=sha1`  
(bien remplacer main par master puisque ce sont le nom des branches sur gitlab, la procédure n'est pas à jour)

```
lea@mac GSB2 % git init --initial-branch=master --object-format=sha1
Dépôt Git vide initialisé dans /Users/lea/Desktop/GSB2/.git/

lea@mac GSB2 % █
```

La commande suivante est identique  
`git remote add origin git@gitlab.com:Sunliaa/gsb-appli.git`

```
lea@mac GSB2 % git remote add origin git@gitlab.com:Sunliaa/gsb-appli.git

lea@mac GSB2 % █
```

De même pour la suivante, on récupère tout avec `git add .`

```
lea@mac GSB2 % git add .

lea@mac GSB2 % █
```

Ensuite on crée son premier commit avec le message voulu (possibilité de laisser celui de gitlab par défaut)

`git commit -m "Initial commit"`

Tous les fichiers sont committés

```
lea@mac GSB2 % git commit -m "Premier commit dépôt repo"
[master (commit racine) 22d9b9e] Premier commit dépôt repo
49 files changed, 1172 insertions(+)
create mode 100644 .gitignore
create mode 100644 .idea/.gitignore
create mode 100644 .idea/.name
create mode 100644 .idea/AndroidProjectSystem.xml
create mode 100644 .idea/compiler.xml
create mode 100644 .idea/deploymentTargetSelector.xml
create mode 100644 .idea/deviceNanager.xml
create mode 100644 .idea/gradle.xml
create mode 100644 .idea/inspectionProfiles/Project_Default.xml
create mode 100644 .idea/migrations.xml
create mode 100644 .idea/misc.xml
create mode 100644 .idea/runConfigurations.xml
create mode 100644 app/.gitignore
create mode 100644 app/build.gradle.kts
create mode 100644 app/proguard-rules.pro
```

On l'envoie sur la branche master (bien penser à modifier main en master comme sur la première commande)

git push --set-upstream origin master

```
Lea@mac GSB2 % git push --set-upstream origin master
Énumération des objets: 83, fait.
Décompte des objets: 100% (83/83), fait.
Compression par delta en utilisant jusqu'à 8 fils d'exécution
Compression des objets: 100% (66/66), fait.
Écriture des objets: 100% (83/83), 90.30 Kio | 12.90 Mio/s, fait.
Total 83 (delta 1), réutilisés 0 (delta 0), réutilisés du paquet 0 (depuis 0)
To gitlab.com:Sunliao/gsb-appli.git
* [new branch]      master -> master
La branche 'master' est paramétrée pour suivre 'origin/master'.
Lea@mac GSB2 % █
```

Le projet est sur gitlab (rafraichir la page gitlab initiale)

Léa PERRET / GSB-Appli

+ ▾Code ▾⋮

**Premier commit dépôt repo**  
Léa authored 1 minute ago

22d9b9e2History

Name	Last commit	Last update
.idea	Premier commit dépôt repo	1 minute ago
app	Premier commit dépôt repo	1 minute ago
gradle	Premier commit dépôt repo	1 minute ago
.gitignore	Premier commit dépôt repo	1 minute ago
build.gradle.kts	Premier commit dépôt repo	1 minute ago
gradle.properties	Premier commit dépôt repo	1 minute ago
gradlew	Premier commit dépôt repo	1 minute ago
gradlew.bat	Premier commit dépôt repo	1 minute ago
settings.gradle.kts	Premier commit dépôt repo	1 minute ago