



APPRENDRE À UTILISER GITHUB

Aux Professionnels d'entreprises

Bobet Goualo Victorien

Consultant & Formateur en DATA, IA & Cloud

Formateur à l'ESATIC, l'ENSEA & l'INP-HB

Executive Master en Big Data & Data Science de ISAE-ISM Paris

Certifié Oracle, Centrale supélec, Orange Digital Center (ODC)

bobetvictorien@gmail.com / +225 0748845128

Prérequis au cours

Notion de collaboration

Notion de traçabilité

Plan du cours

- I. Premiers pas avec Git et GitHub
- II. Collaboration et bonnes pratiques
- III. Automatisation et intégration

JOUR 1

Martiné : Présentation Git & GitHub

Soirée : Prise en main de Git & GitHub

I. Premiers pas avec Git et GitHub

La gestion de versions consiste à gérer l'ensemble des versions d'un ou plusieurs fichiers.

Les logiciels évoluant, chaque étape d'avancement est appelée version.

Une modification du code est une évolution entre deux versions.

Il existe de nombreux logiciels capables d'aider le développeur dans cette tâche. Exemple : Git, TFS (Team Foundation Server), etc.

I. Premiers pas avec Git et GitHub

Ces outils (logiciels) permettent :

- ❖ La sauvegarde (avec ou sans serveur distant);
- ❖ La conservation de l'historique des fichiers (l'auteur des modifications, la date, une note explicative de cette modification);
- ❖ Voir les changements;
- ❖ Revenir aux versions précédentes;
- ❖ Collaborer à des projets.

I. Premiers pas avec Git et GitHub

Le couple (Git , GitHub) aide à gérez les projets de développement.

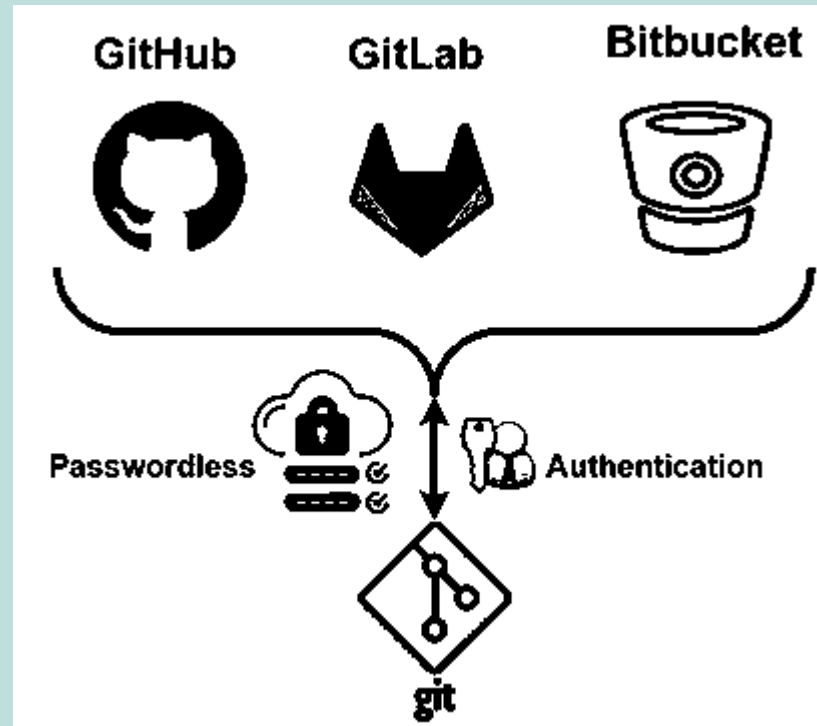
Git est un système de contrôle de version distribué (outil de versionning) pour un dépôt local

GitHub une plateforme web pour le dépôt du projet en ligne (dépôt distant)

Une alternative du moment à GitHub est GitLab ou Bitbucket (utilisateur jira).

I. Premiers pas avec Git et GitHub

Ci-dessous un aperçu de l'interaction entre Git et les plateformes de dépôt distant.



I. Premiers pas avec Git et GitHub

Introduction à Git

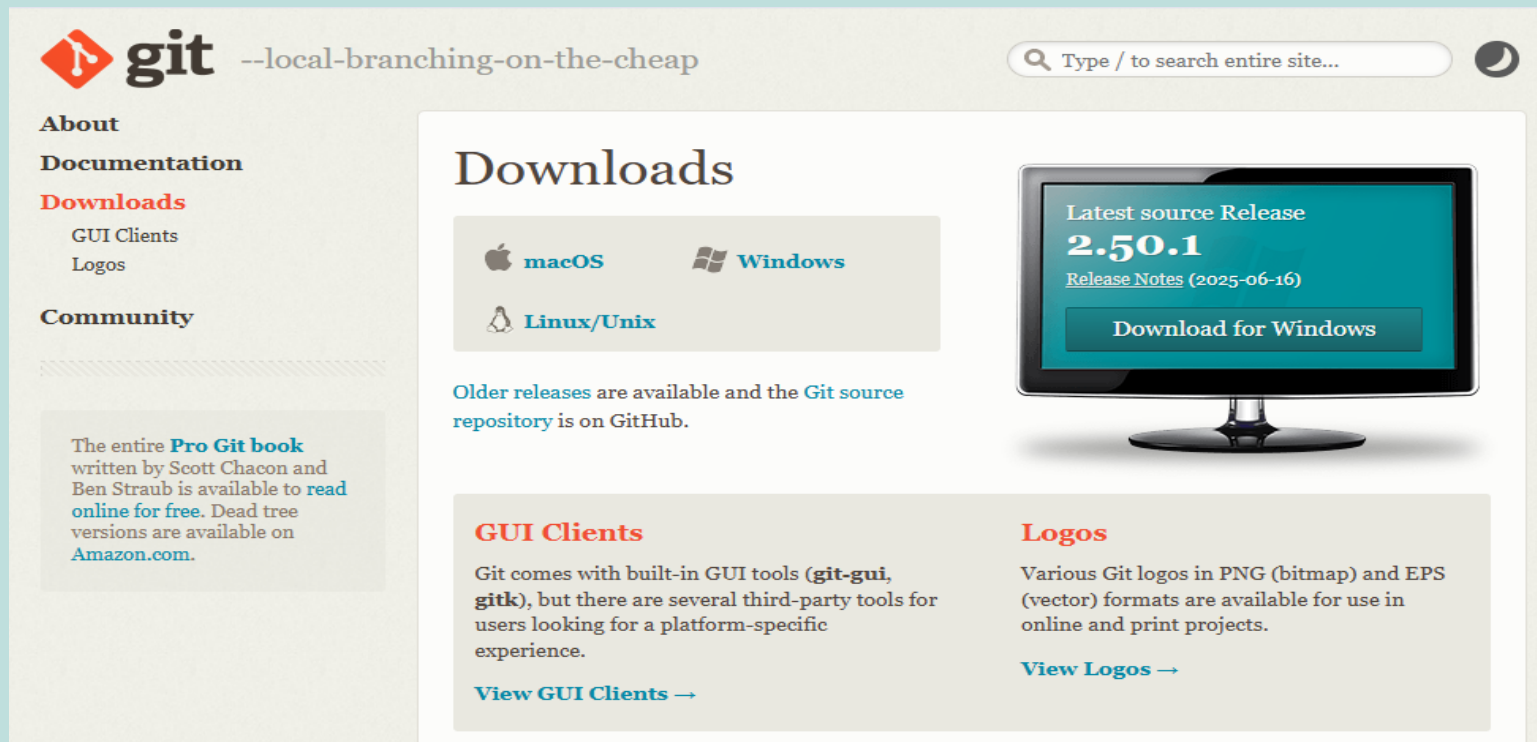
Git est un logiciel libre et open source qui vu le jour en 2005 grâce à Linus Torvald, auteur du noyau linux.

Git un outil de traçabilité, de versionning.

I. Premiers pas avec Git et GitHub

Installation Git

Se rendre sur <https://git-scm.com/downloads> , ensuite faire son choix selon le système d'exploitation à disposition.



I. Premiers pas avec Git et GitHub

Installation Git

Git Bash est l'interface permettant d'utiliser Git en ligne de commande sous Windows.

Git CMD plus compatible aux habitués de Windows. Moins de fonctionnalités que Git Bash.

Git GUI une interface graphique qui facilite les opérations sans ligne de commandes.

I. Premiers pas avec Git et GitHub

Répertoire de travail ou Working directory

On appelle répertoire de travail les fichiers effectivement présents dans le répertoire géré par Git.

Un dépôt

Un dépôt (repository) est un dossier contenant l'ensemble des données associées au projet (les versions et les modifications d'un projet).

I. Premiers pas avec Git et GitHub

git init pour la création d'un dépôt local (initialiser un dépôt Git)

Créer dans un premier temps un dossier : **\$ mkdir crrae_project**

Afficher la configuration : **\$ git config --list**

Configurer son identité

\$ git config --global user.name "Goualo Victorien"

\$ git config --global user.email victorien.bobet@crrae.org

I. Premiers pas avec Git et GitHub

Configurer les couleurs

```
$ git config --global color.diff auto
```

```
$ git config --global color.status auto
```

```
$ git config --global color.branch auto
```

Configurer l'éditeur

```
$ git config --global core.editor notepad++
```

```
$ git config --global merge.tool vimdiff
```

I. Premiers pas avec Git et GitHub

L'outil de merge permet de fusionner deux parties distinctes d'un projet.

Créer un dépôt local

```
$ cd Documents/Git/crrae
```

```
$ git init
```

il est possible de cloner un projet existant depuis GitHub.

I. Premiers pas avec Git et GitHub

Commit

Sauvegarder au sens de git, c'est effectuer un commit.

Etat

La commande **git status** permet de connaître l'état d'un fichier du répertoire de travail vis-à-vis du dernier commit.

Historique git

La commande **git log** permet de voir l'ensemble des commit.

I. Premiers pas avec Git et GitHub

Introduction à GitHub

GitHub, créée en 2008 est un outil de communication et de collaboration entre plusieurs développeurs. C'est une interface web qui interagit avec Git.

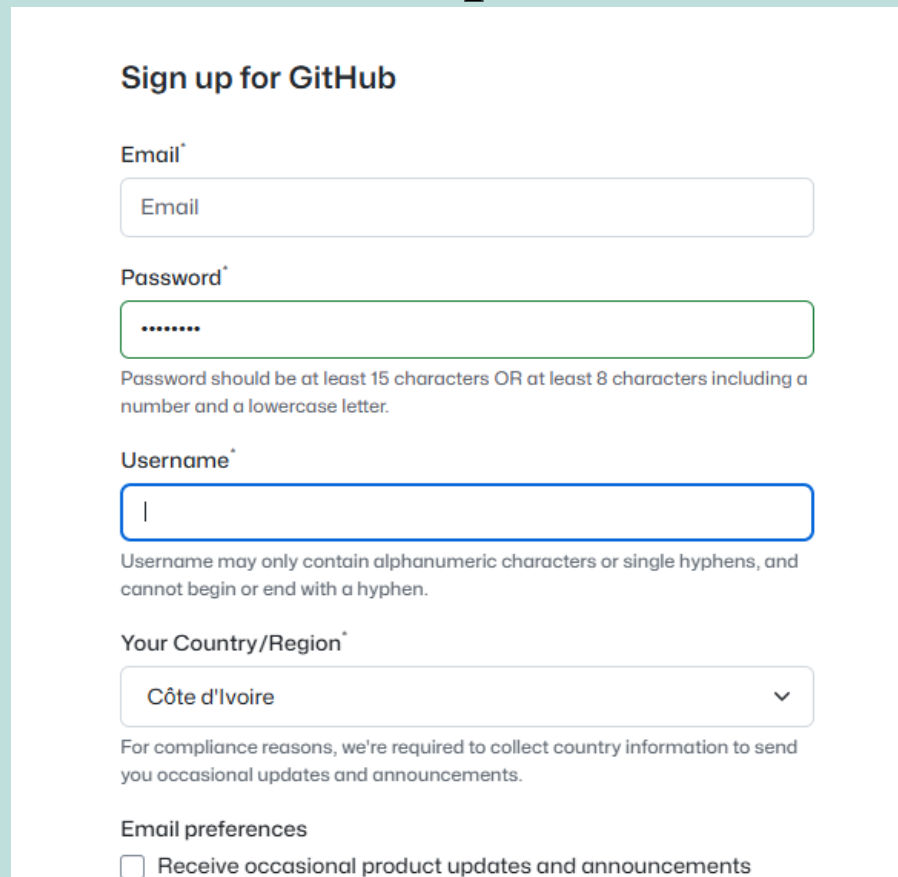
2 types de dépôt existent : le dépôt privé et le dépôt public

Possibilité de créer son lien github

Les avantages de GitHub

I. Premiers pas avec Git et GitHub

Créer un compte GitHub



The screenshot shows the GitHub sign-up page. It has a title 'Sign up for GitHub'. Below it are four input fields: 'Email' (with a placeholder 'Email'), 'Password' (with a placeholder of seven dots and a note below stating it should be at least 15 characters or 8 characters with a number and a lowercase letter), 'Username' (with a placeholder '|'), and 'Your Country/Region' (a dropdown menu showing 'Côte d'Ivoire'). At the bottom, there is a section for 'Email preferences' with a checkbox labeled 'Receive occasional product updates and announcements'.

Une fois votre compte crée, vous avez accès à

- **un tableau de bord**
- **Aux principaux repository de vos équipes**

I. Premiers pas avec Git et GitHub

Les différents plans sur GitHub

The image shows the GitHub pricing page with three plans. The 'Enterprise' plan is highlighted with a blue border and a 'RECOMMENDED' badge. Each plan includes a description, a price per user/month, and a call-to-action button.

Plan	Description	Price	Call to Action
Free	The basics for individuals and organizations	\$0 USD per user/month	Join for free
Team	Advanced collaboration for individuals and organizations	\$4 USD per user/month	Continue with Team
Enterprise	Security, compliance, and flexible deployment <small>Starting at</small>	\$21 USD per user/month	Start a free trial / Contact Sales

I. Premiers pas avec Git et GitHub

Créer un compte GitHub

L'onglet **Pull request** pour les demandes de modification sur le code principal aux autres développeurs.

L'onglet **Explore** pour permet de trouver des projets open source

I. Premiers pas avec Git et GitHub

Créer un repository

La création d'un repository(dépôt) sur GitHub permet de déposer son projet

Le dépôt peut être public (attention à la sécurité) ou privé.

L'activation de Push protection peut aider à contrôler les informations sensibles (mot de passe, identifiants, information carte bancaire, etc.).

I. Premiers pas avec Git et GitHub

Créer un repository

README contient les informations clés d'un projet

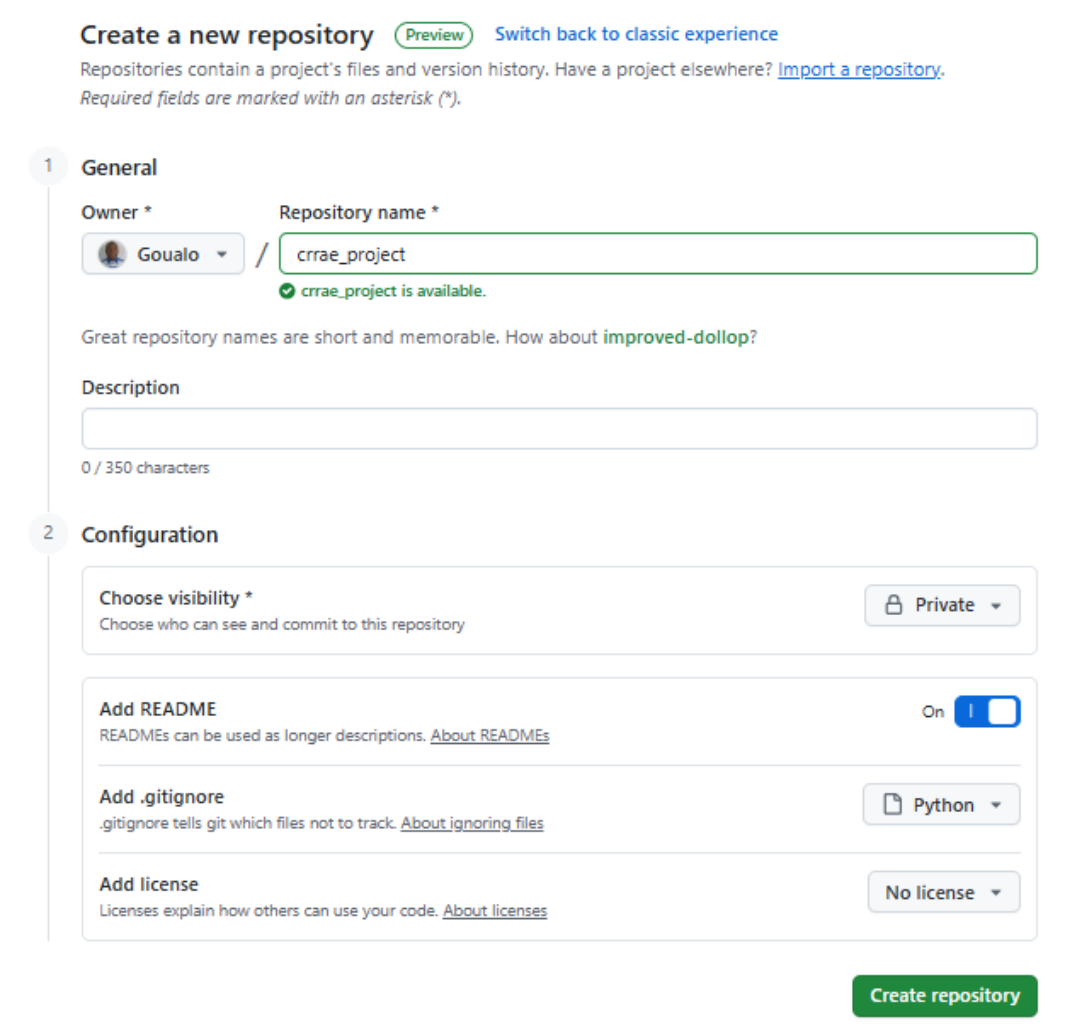
gitignore permet d'ignorer la mise à jour de certains fichiers d'un projet Git.

I. Premiers pas avec Git et GitHub

Créer un repository

Un espace de stockage et de gestion de Code source pour un projet donné.

C'est là que va se trouver notre dépôt distant qui recevra plus tard les différentes modifications des membres équipe projet.



The screenshot shows the 'Create a new repository' page on GitHub. At the top, there are links for 'Preview' and 'Switch back to classic experience'. Below this, a note states: 'Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository.](#) Required fields are marked with an asterisk (*).' The form is divided into two sections: '1 General' and '2 Configuration'. In the 'General' section, the 'Owner' is set to 'Goualo' and the 'Repository name' is 'crrae_project', with a green checkmark indicating it is available. A suggestion for 'improved-dollop' is shown. The 'Description' field is empty, with a character count of '0 / 350 characters'. The '2 Configuration' section includes 'Choose visibility' set to 'Private', 'Add README' with a toggle switch turned 'On', 'Add .gitignore' set to 'Python', and 'Add license' set to 'No license'. A green 'Create repository' button is at the bottom right.

Create a new repository [Preview](#) [Switch back to classic experience](#)

Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository.](#)
Required fields are marked with an asterisk (*).

1 General

Owner * / Repository name *

Goualo / crrae_project

✓ crrae_project is available.

Great repository names are short and memorable. How about improved-dollop?

Description

0 / 350 characters

2 Configuration

Choose visibility * Choose who can see and commit to this repository Private

Add README READMEs can be used as longer descriptions. [About READMEs](#) On

Add .gitignore .gitignore tells git which files not to track. [About ignoring files](#) Python

Add license Licenses explain how others can use your code. [About licenses](#) No license

Create repository

I. Premiers pas avec Git et GitHub

Créer un repository

The screenshot shows a GitHub repository page for 'crrae_project', which is marked as 'Private'. At the top right, there are buttons for 'Watch' (0), 'Fork' (0), and 'Star' (0). Below the repository name, there are tabs for 'main' (selected), '1 Branch', and '0 Tags'. A search bar labeled 'Go to file' and buttons for 'Add file' and '<> Code' are also present. The main content area shows a list of files: '.gitignore' and 'README.md', both from the 'Initial commit' 2 minutes ago. Below this is a 'README' section with the title 'crrae_project' and the description 'ce projet s'effectue dans le cadre de la formation des agents de la CRRAE-UMOA'. On the right side, there is an 'About' section with the same description, and sections for 'Releases' (No releases published, with a link to 'Create a new release') and 'Packages' (No packages published, with a link to 'Publish your first package').

crrae_project Private

Watch 0 Fork 0 Star 0

main 1 Branch 0 Tags

Go to file Add file <> Code

Goualo Initial commit 4e9b711 · 2 minutes ago 1 Commit

.gitignore	Initial commit	2 minutes ago
README.md	Initial commit	2 minutes ago

README

crrae_project

ce projet s'effectue dans le cadre de la formation des agents de la CRRAE-UMOA

About

ce projet s'effectue dans le cadre de la formation des agents de la CRRAE-UMOA

- Readme
- Activity
- 0 stars
- 0 watching
- 0 forks

Releases

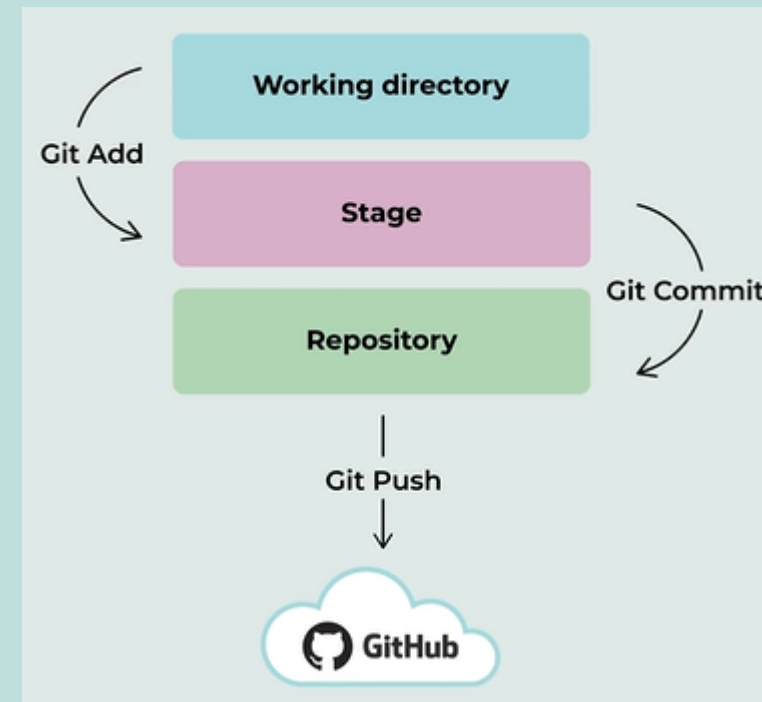
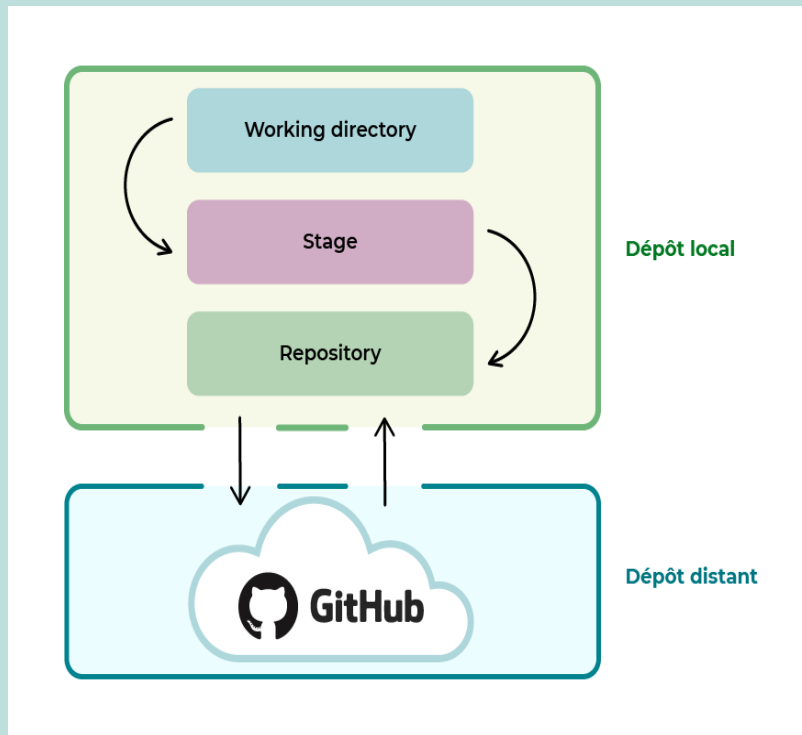
No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

I. Premiers pas avec Git et GitHub

Les étapes d'interaction entre Git et GitHub



I. Premiers pas avec Git et GitHub

- Le protocole SSH permet d'authentifier une personne grâce à une clé unique et personnelle.
- Vous pouvez ajouter une clé SSH sur votre compte GitHub afin de vous authentifier plus rapidement.
- `git add` permet d'ajouter des fichiers dans l'index, qui est une zone intermédiaire dans laquelle stocker les fichiers modifiés.
- `git commit` permet de créer une nouvelle version avec les fichiers situés dans l'index.
- `git commit -m` permet de créer une nouvelle version et de préciser le message rattaché au commit.
- `git push` permet d'envoyer les modifications faites en local vers un dépôt distant.