



ORGANISER UN PROJET GITHUB

► **Organiser un projet avec GitHub**

Gérer les équipes



Gestion de l'équipe sur Github

- ≥ Ces paramètres sont seulement accessibles aux « supers admins »
- ≥ Sur GitHub vous pouvez ajouter des développeurs dans un projet et gérer les droits

The screenshot shows the GitHub repository settings page for 'jdev-org / tiriad'. The 'Settings' tab is selected, and the 'Who has access' section is visible. The repository is public, and access is managed through a 'BASE ROLE' of 'Read' and 'DIRECT ACCESS' for 1 member. The 'Manage access' section shows a list of team members, with 'Gaetanbrl' listed as an Admin.

Options

Manage access

Security & analysis

Branches

Webhooks

Notifications

Integrations

Deploy keys

Actions

Secrets

Moderation settings

Who has access

PUBLIC REPOSITORY

This repository is public and visible to anyone.
[Manage](#)

BASE ROLE Read

All 3 members can access this repository.
[Manage](#)

DIRECT ACCESS

1 has access to this repository. 1 member.

Manage access

[Create team](#) [Invite teams or people](#)

Select all Type ▾ Role ▾

Gaetanbrl Role: Admin ▾

[Previous](#) [Next](#)

Niveaux d'autorisation

≥ Les droits sont de 4 niveaux

The screenshot shows the 'Member repository permissions' interface. At the top, there's a title 'Member repository permissions'. Below it, a section titled 'Base permissions' explains that these apply to all members and excludes outside collaborators. A dropdown menu is currently set to 'Read'. Below the dropdown is a list of 'Organization member permissions' options: 'None', 'Read' (which is selected with a checkmark), 'Write', and 'Admin'. Each option has a brief description of the permissions granted. At the bottom of the list is a 'Save' button.

Member repository permissions

Base permissions

Base permissions to the organization's repositories apply to all members and excludes outside collaborators. Since organization members can have permissions from multiple sources, members and collaborators who have been granted a higher level of access than the base permissions will retain their higher permission privileges.

Read ▾

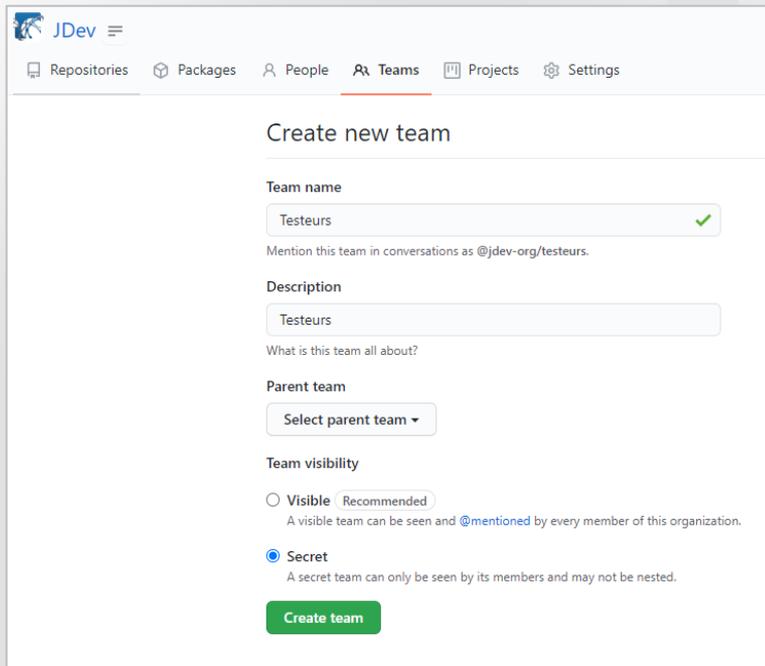
Organization member permissions

- None**
Members will only be able to clone and pull public repositories. To give a member additional access, you'll need to add them to teams or make them collaborators on individual repositories.
- ✓ **Read**
Members will be able to clone and pull all repositories.
- Write**
Members will be able to clone, pull, and push all repositories.
- Admin**
Members will be able to clone, pull, push, and add new collaborators to all repositories.

Save

Création d'une « Team »

- ≥ Au sein d'un projet, vous pouvez regrouper les utilisateurs en équipe (team)



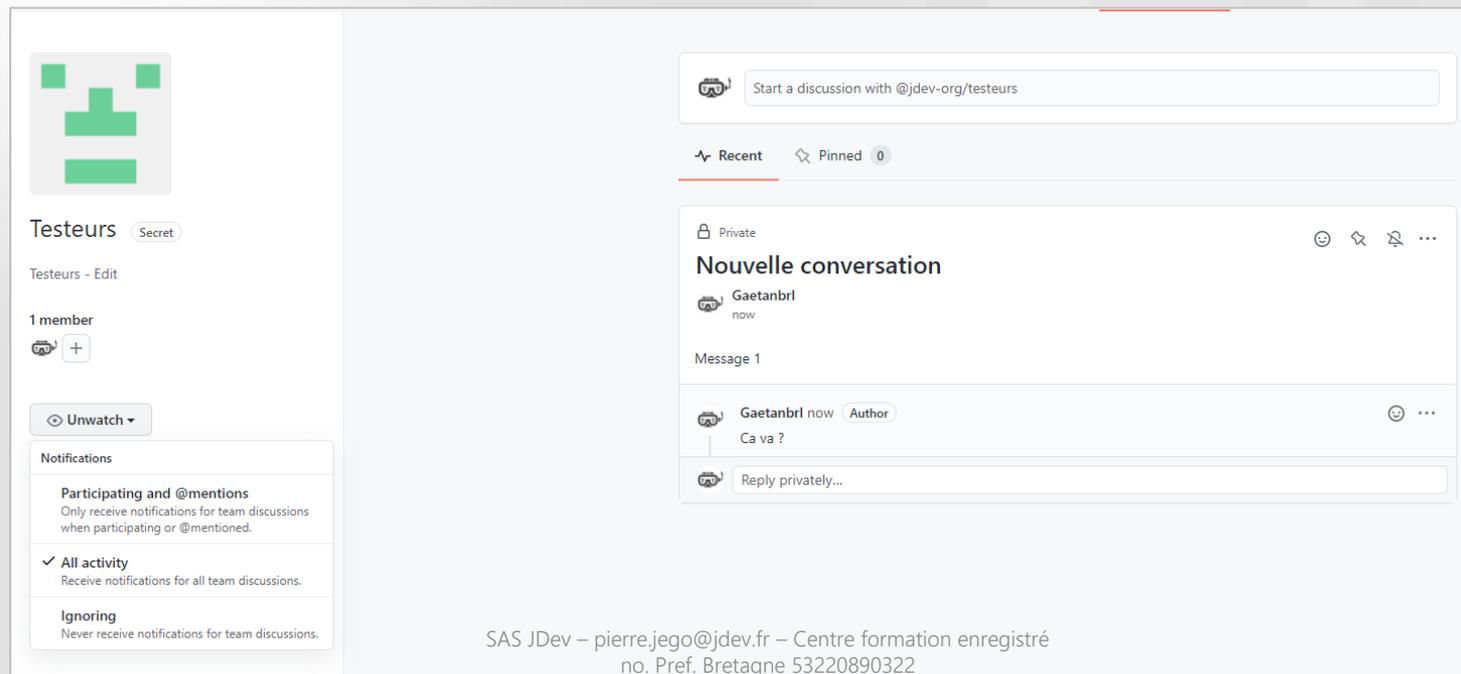
The screenshot shows the GitHub interface for creating a new team. At the top, the user profile 'JDev' is visible, along with navigation links for Repositories, Packages, People, Teams (highlighted), Projects, and Settings. The main heading is 'Create new team'. The form includes the following fields and options:

- Team name:** A text input field containing 'Testeurs' with a green checkmark to its right.
- Description:** A text input field containing 'Testeurs'.
- Parent team:** A dropdown menu with the text 'Select parent team' and a downward arrow.
- Team visibility:** Two radio button options: 'Visible' (with a sub-label 'Recommended') and 'Secret' (which is selected).

Below the 'Visible' option, there is explanatory text: 'A visible team can be seen and @mentioned by every member of this organization.' Below the 'Secret' option, there is explanatory text: 'A secret team can only be seen by its members and may not be nested.' At the bottom of the form is a green 'Create team' button.

Création d'une « Team »

- ≥ Les équipes sont indépendantes et peuvent avoir une discussion générale indépendante, des notifications spécifiques, des repositories propres, des projets propres, etc...



The screenshot displays the GitHub interface for a team named 'Testeurs'. On the left, the team's profile is shown with a green logo, the name 'Testeurs' (marked as 'Secret'), and a notification preference section. The notification section includes three options: 'Participating and @mentions' (selected), 'All activity', and 'Ignoring'. The main content area shows a private conversation interface with a search bar, tabs for 'Recent' and 'Pinned', and a message from 'Gaetanbrl' saying 'Ca va ?'.

Testeurs Secret

Testeurs - Edit

1 member

Unwatch

Notifications

- Participating and @mentions**
Only receive notifications for team discussions when participating or @mentioned.
- All activity**
Receive notifications for all team discussions.
- Ignoring**
Never receive notifications for team discussions.

Start a discussion with @jdev-org/testeurs

Recent Pinned 0

Private

Nouvelle conversation

Gaetanbrl now

Message 1

Gaetanbrl now Author

Ca va ?

Reply privately...

► **Organiser un projet avec GitHub**

Les plugins GitHub



Permettent de rajouter des fonctionnalités, des interactions

- ≥ Marketplace GitHub
- ≥ Libres ou payants (selon conditions)

The screenshot shows the GitHub Marketplace search results for 'Code quality'. The interface includes a search bar with the text 'Search for apps and actions', a left sidebar with navigation options like 'Types', 'Categories', and 'Code quality' (which is selected), and a main content area displaying search results. The results are filtered by 'Code quality' and show four items: CodeFactor, HashiCorp's Link Checker, Codecov | Code Coverage Reports, and Lucidchart Connector. Each item includes a logo, the name, a brief description, and a star count for some items.

Marketplace / Search results

Types

Search for apps and actions

Apps

Actions

Categories

API management

Chat

Code quality

Code review

Continuous integration

Dependency management

Code quality

Automate your code review with style, quality, security, and test-coverage checks when you need them.

719 results filtered by Code quality

- CodeFactor**  Automated code review for GitHub
- HashiCorp's Link Checker**  By hashicorp 
Report broken links found in a given set of files
17 stars
- Codecov | Code Coverage Reports**  Automatic test report merging for all CI and languages into a single code coverage report directly into your pull request
- Lucidchart Connector**  Insert a public link to a Lucidchart diagram so team members can quickly understand an issue or pull request

► **Organiser un projet avec GitHub**

La vue projet



Project dans GitHub

- ≥ Kanban : le plus courant en mode Agile
- ≥ Lié aux issues, aux équipes
- ≥ Possibilité d'un projet par Milestone, par release, etc.

The screenshot shows a GitHub Project Kanban board for the repository 'sigrennesmetropole / cartoHoraires'. The board is titled 'Carto participative des horaires' and was last updated on 29 Sep. It features three columns: 'BACKLOG' (2 items), 'TO DO (5max.)' (1 item), and 'IN PROGRESS (3max.)' (0 items). Each item is a card representing an issue, with a title, issue number, creator, and associated labels.

Repository: sigrennesmetropole / cartoHoraires

Unwatch 3 | Star 1 | Fork 1

Code | Issues 6 | Pull requests | ZenHub | Actions | **Projects 1** | Wiki | Security | ...

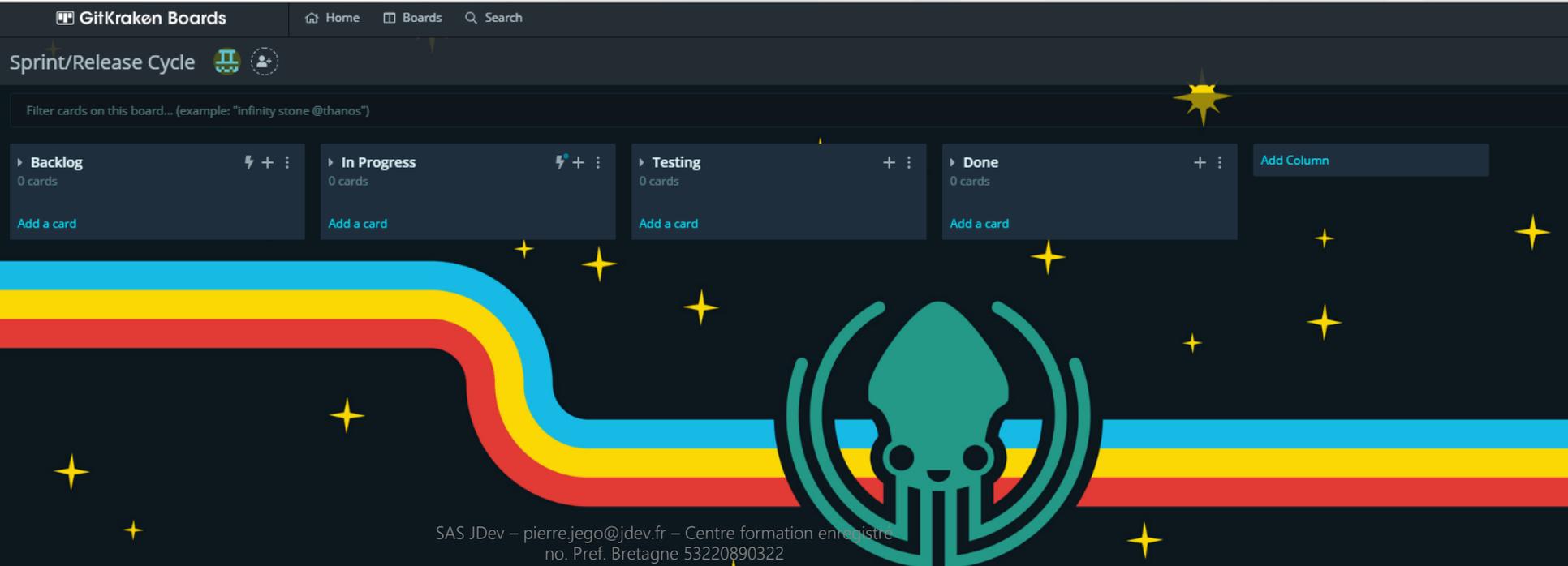
Carto participative des horaires
Updated on 29 Sep

Filter cards | + Add cards(1 new) | Fullscreen | Menu

- 2 BACKLOG**
 - Bancarisation des données ?** #28 opened by cbriand
Labels: Evolution, recueil des données
 - Animation automatique (en attente V2)** #25 opened by cbriand
Labels: Evolution, visualisation
- 1 TO DO (5max.)**
 - Mécanique de détection de l'envoi du mail de rappel et envoi du mail** #8 opened by cbriand
Label: socle technique
- 0 IN PROGRESS (3max.)**

Quelles alternatives ?

- ≥ Des addons (ZenHub free pour open source)
- ≥ Les outils Desktop libres (GitKraken) ou non (Jira)



► **Organiser un projet avec GitHub**

Les Actions



C'est quoi ?

- ≥ Automatisation du workflow (release, gestion dépendances, revue de code, etc.)
- ≥ Fichier YAML
- ≥ Localisées dans `.github/workflow`

```
YAML 📄

name: Super-Linter

# Run this workflow every time a new commit pushed to your repository
on: push

jobs:
  # Set the job key. The key is displayed as the job name
  # when a job name is not provided
  super-lint:
    # Name the Job
    name: Lint code base
    # Set the type of machine to run on
    runs-on: ubuntu-latest

    steps:
      # Checks out a copy of your repository on the ubuntu-latest machine
      - name: Checkout code
        uses: actions/checkout@v2

      # Runs the Super-Linter action
      - name: Run Super-Linter
        uses: github/super-linter@v3
        env:
          DEFAULT_BRANCH: main
          GITHUB_TOKEN: ${ secrets.GITHUB_TOKEN }
```

[Tutorial en ligne](#)



C'est quoi ?

≥ Exemple avec le plugin Qgis Cadastre pour la revue de code

```
23 lines (21 sloc) | 594 Bytes Raw Blame   
```

```
1 name: Flake8
2
3 on: [push, pull_request] Déclenchement sur un event (hook) github
4
5 jobs:
6   flake8_py3:
7     runs-on: ubuntu-latest Lancé avec OS Ubuntu
8     steps:
9       - name: Setup Python
10        uses: actions/setup-python@v2 Appel une action github – Install python
11        with:
12          python-version: 3.7
13          architecture: x64
14       - name: Checkout
15        uses: actions/checkout@master
16       - name: Install flake8
17        run: pip install flake8
18       - name: Run flake8 Install et exécute flake8 pour contrôler la qualité du code python
19        uses: suo/flake8-github-action@releases/v1
20        with:
21          checkName: 'flake8_py3' # NOTE: this needs to be the same as the job name
22
23 env:
24   GITHUB_TOKEN: ${ secrets.GH_TOKEN }
```

SAS JDev – pierre.jego@idev.fr – Centre formation enregistré
no. Pref. Bretagne 53220890322 <https://github.com/3liz/QgisCadastrePlugin/blob/master/.github/workflows/flake8.yml>

► **Organiser un projet avec GitHub**

Documentation



Intégrée

- ≥ La section wiki GitHub permet de créer rapidement une documentation (Markdown)

The screenshot shows a GitHub Wiki page titled 'Cadastrapp'. The page content includes a description of the tool and a diagram of its architecture. The diagram illustrates the data flow and components involved in the system.

Home
Gaetanbri edited this page on 6 Nov 2019 · 8 revisions

Cadastrapp

Cadastrapp est l'outil de consultation des données cadastrales fournies par la Direction Générale des Finances Publiques (DGFiP) dans le visualiseur de geOrchestra. Il est composé d'un module serveur qui est une API d'accès aux données de la matrice foncière et d'un *add-on* au visualiseur de geOrchestra. L'API peut être utilisée par n'importe quelle application client sous réserve de passer par le système d'authentification de geOrchestra.

Pages 19

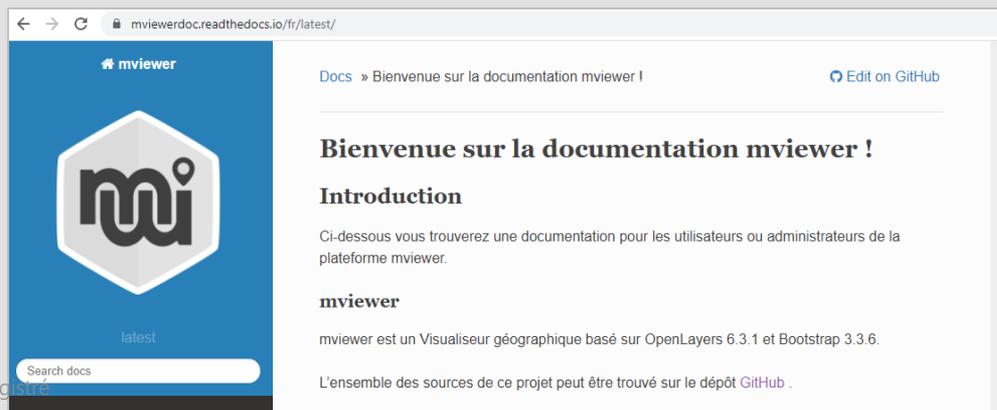
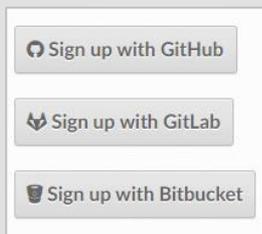
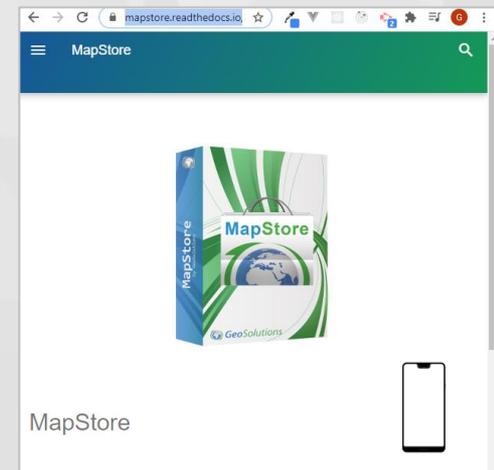
- Accueil
- Versions
- Guide Administrateur
 - Prérequis
 - Installation
 - Configuration
 - Modèle de données Cadastrapp
- Guide Utilisateur
 - Se connecter
 - Charger l'outil
 - Se déplacer / Zoomer sur la carte
 - Outils de sélection de parcelles
 - Affichage des données cadastrales et/ou foncières
 - Info-bulle
 - Fiche de données cadastrales
 - Fiche de données foncières
 - Faire une recherche
 - Recherche de parcelle
 - Recherche de propriétaire

Diagram Description:

- Top Left:** Data sources 'EDIGEO, MAJIC, ...' feed into 'QGIS'.
- Top Middle:** 'QGIS' feeds into a 'Base de données Qgis' (PostgreSQL).
- Top Right:** 'Base de données Qgis' feeds into a 'Base de données Cadastrapp' (PostgreSQL).
- Center:** A script 'createDBUsingQgisModel.sh' with files 'qgisCommunes.sql' and 'qgisParcelles.sql' connects the two databases.
- Bottom Left:** 'GeoServer' (OGC) feeds into 'Cadastrapp'.
- Bottom Right:** 'Cadastrapp' feeds into 'geOrchestra'.
- Bottom Center:** 'Cadastrapp' feeds into 'Fond de plan' and 'Génération de Documents, ...', which in turn feeds back into 'GeoServer'.

Externalisée

- ≥ Solution courante communautaire : [readthedocs](#)
- ≥ Peut-être liée à une release (versionnement)
- ≥ Dans le projet ou via un submodule
- ≥ Hébergement gratuit et illimité de la doc sur
- ≥ Une doc Responsive exportable en PDF !



Exercice : Créer une documentation avec readthedocs

- ≥ Objectif :
 - ≥ Partager la recette des galettes bretonnes

- ≥ Prérequis:
 - Droits d'administration
 - Python et pip doivent être installés
 - Observer la documentation de départ : <https://www.mkdocs.org/#getting-started>

- 1. Sur Github, créez un nouveau dépôt « galettes » avec un README et une branche master principale
- 2. Sur votre machine, récupérez ce dépôt via un clone
- 3. Ajouter un répertoire « docs » et vous y placer
`cd galettes/docs`
- 4. Ouvrez un terminal et via pip installez ces packages :
`pip install sphinx`
`pip install sphinx-quickstart`
`pip install recommonmark`
`pip install sphinx-rtd-theme`
- 5. Placez vous dans le dossier de votre dépôt et saisissez ces commandes
`sphinx-quickstart`

Organiser un projet

Documentation

Exercice : Créer une documentation

- ≥ Objectif :
 - ≥ Partager la recette
- ≥ Prérequis:
 - Droits d'administration
 - Python et pip doivent être installés
 - Observer la documentation

1. Sur Github, créez un nouveau dépôt
2. Sur votre machine, récupérez le dépôt
3. Ajouter un répertoire « docs »


```
cd gallettes/docs
```
4. Ouvrez un terminal et via pip installez les outils nécessaires


```
pip install sphinx
pip install sphinx-quickstart
pip install recommonmark
pip install sphinx-rtd-theme
pip install sphinx-panels
```
5. Placez vous dans le dossier de travail


```
cd gallettes/docs
sphinx-quickstart
```

```
gaetan@DESKTOP-1HU3HJF MINGW64 /c:/xampp/htdocs/gallettes (master)
```

```
$ sphinx-quickstart
```

```
Welcome to the Sphinx 3.0.3 quickstart utility.
```

```
Please enter values for the following settings (just press Enter to
accept a default value, if one is given in brackets).
```

```
Selected root path: .
```

```
You have two options for placing the build directory for Sphinx output.
Either, you use a directory "_build" within the root path, or you separate
"source" and "build" directories within the root path.
```

```
> Separate source and build directories (y/n) [n]: n
```

```
The project name will occur in several places in the built documentation.
```

```
> Project name: gallettes
```

```
> Author name(s): gaetanbrl
```

```
> Project release []:
```

```
If the documents are to be written in a language other than English,
you can select a language here by its language code. Sphinx will then
translate text that it generates into that language.
```

```
For a list of supported codes, see
```

```
https://www.sphinx-doc.org/en/master/usage/configuration.html#confval-language.
```

```
> Project language [en]: de
```

```
Creating file .\conf.py.
```

```
Creating file .\index.rst.
```

```
Creating file .\Makefile.
```

```
Creating file .\make.bat.
```

```
Finished: An initial directory structure has been created.
```

```
You should now populate your master file .\index.rst and create other documentation
source files. Use the Makefile to build the docs, like so:
```

```
make builder
```

```
where "builder" is one of the supported builders, e.g. html, latex or linkcheck.
```

Exercice : Créer une documentation avec readthedocs

5. Ajouter un répertoire « docs »
6. Ouvrez le fichier index.rst et observer le contenu
7. Si besoin, adaptez les variables : project, copyright, author, release
8. Maintenant, poussez tout ces fichier sur votre dépôt Git

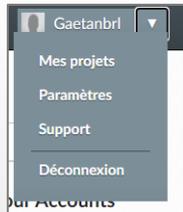
```
gaetan@DESKTOP-1HU3HJF MINGW64 /c/xampp/htdocs/galettes (master)
$ git add *
warning: LF will be replaced by CRLF in Makefile.
The file will have its original line endings in your working directory

gaetan@DESKTOP-1HU3HJF MINGW64 /c/xampp/htdocs/galettes (master)
$ git commit -m "Create doc"
[master b139d82] Create doc
4 files changed, 134 insertions(+)
 create mode 100644 Makefile
 create mode 100644 conf.py
 create mode 100644 index.rst
 create mode 100644 make.bat

gaetan@DESKTOP-1HU3HJF MINGW64 /c/xampp/htdocs/galettes (master)
$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 2.28 KiB | 2.28 MiB/s, done.
Total 6 (delta 0), reused 0 (delta 0)
To https://github.com/Gaetanbr1/galettes.git
059e442..b139d82  master -> master
```

Exercice : Créer une documentation avec readthedocs

10. Dans l'interface de <https://readthedocs.org/>, connectez-vous et sélectionnez « Mes projets »



11. Sélectionnez ensuite « Importer un projet »

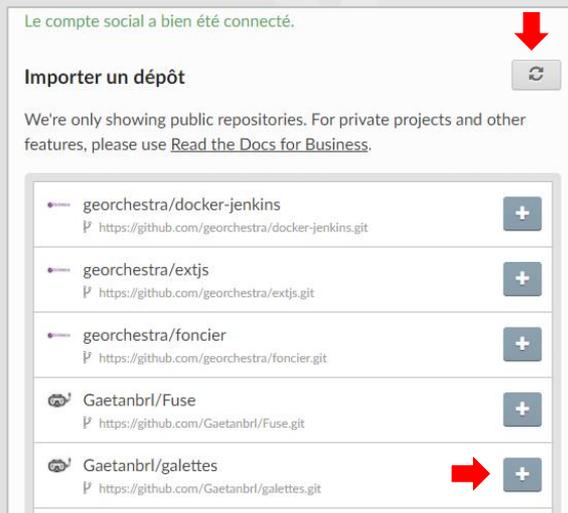
12. Choisissez « Connect to GitHub »

13. Cliquez sur le bouton « Rafraîchir »

14. Dans la liste, sélectionnez votre dépôt « galettes »

15. Faites suivant

10. A la fin cliquez sur « Compiler une version »



Détails du projet

To import a project, start by entering a few details and choosing options for your documentation in a [readthedocs.org](#) project.

Nom :

URL du dépôt :

URL du dépôt de la documentation hébergée :

Type de dépôt :

Modifier les options avancées du projet :

Exercice : Créer une documentation avec readthedocs

10. Vous devriez avoir une erreur du type

Sphinx error:

```
master file /home/docs/checkouts/readthedocs.org/user_builds/galettes/checkouts/latest/contents.rst not found
```

11. Cette erreur indique qu'il cherche un fichier content.rst, hors nous avons un index.rst à la place comme fichier principal

12. Dans le fichier config.py, ajouter ces deux lignes :

```
master_doc = 'index'  
source_suffix = '.rst'
```

13. Poussez ces modifications via git

14. Cliquez sur l'onglet « Compilation » et observez qu'une nouvelle compilation est en cours sur readthedocs.org

15. Cliquez sur « Afficher la doc » à la fin de la compilation si c'est une réussite

Exercice : Créer une documentation avec readthedocs

16. Cette doc n'est pas très séduisante... nous allons changer le thème par défaut :

- Dans /gallettes créez un répertoire « _themes »
- Récupérez tout le thème https://github.com/readthedocs/sphinx_rtd_theme/tree/master/sphinx_rtd_theme via un clone
- Copier le répertoire sphinx_rtd_theme vers /gallettes/_themes
- Dans le fichier config.py, remplacez la variable `html_theme` par ces lignes à la place:

```
# theme HTML
html_theme = "sphinx_rtd_theme"
html_theme_path = ["_themes",]
```

- Ajoutez en dessous ces lignes si elles n'existent pas dans le fichier:

```
# The name of the Pygments (syntax highlighting) style to use.
pygments_style = 'sphinx'
```

- Poussez ces nouveautés sur votre dépôt GitHub et attendez la fin de la compilation pour retourner sur la doc (**bien vider le cache !**)

17. On va maintenant ajouter un peu de contenu et un menu

Exercice : Créer une documentation avec readthedocs

17. On va maintenant ajouter un peu de contenu et un menu

- Dans le fichier `index.rst` remplacez le contenu par celui du fichier `index.rest` fourni avec le powerpoint dans le répertoire « files »

Cette syntaxe permet de modifier la page d'accueil et d'ajouter des menus de navigation à gauche à partir de fichiers qui doivent être trouvés dans l'arborescence du projet

```
.. gallettes documentation master file, created by sphinx-quickstart on Thu Dec 3 12:15:27 2020. You can adapt this file completely to your liking, but it should at least contain the root `toctree` directive.
```

En-tête non
affichée

```
La galette bretonne
```

Titre

```
=====
```

```
Découvrez l'histoire et la recette de la galette bretonne !
```

Du texte libre

```
.. image:: _img/galette.png
   :width: 600
```

Image

```
.. toctree::
```

```
   :hidden: Un niveau de navigation
```

```
   :maxdepth: 1
```

```
   :caption: Documentation utilisateur
```

```
doc_main/histoire
```

Les sous-niveau de navigation

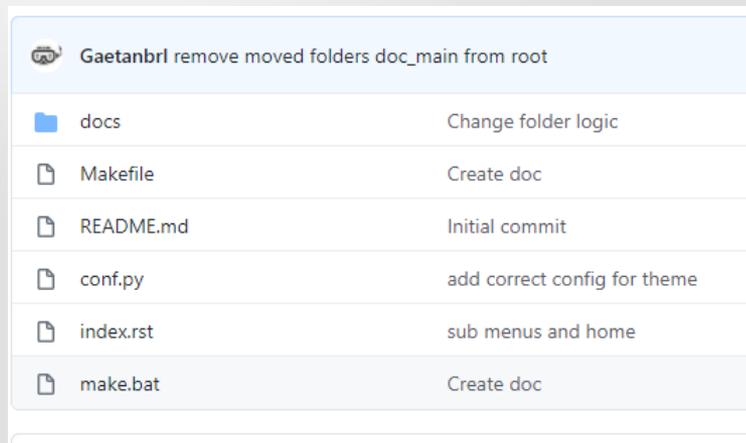
```
doc_main/recette
```

Exercice : Créer une documentation avec readthedocs

- Depuis le répertoire « files » qui est fourni, copiez les répertoires `_img` et `doc_main` puis les placer dans le dépôt `/galettes`
 - Dans le dossier `/doc_main` se situent les pages et leur contenu
 - Vous pouvez observer chaque page
- Poussez sur GitHub ces modifications
- Observez la doc après le build

Exercice : Créer une documentation avec readthedocs

18. On va modifier le répertoire et le rendre plus facile à maintenir
- Créez un répertoire gallettes/docs
 - Placez tous les fichiers dedans **sauf** les fichiers Make, make.bat, .git, README
 - Vous devriez avoir quelque chose comme ceci :



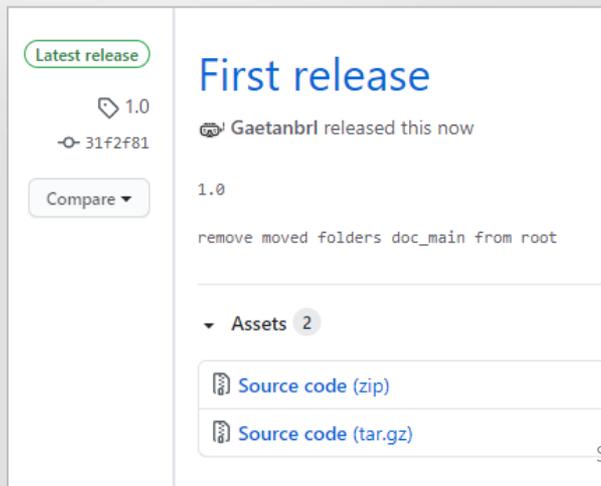
The screenshot shows a terminal window with the following content:

```
Gaetanbri remove moved folders doc_main from root
```

docs	Change folder logic
Makefile	Create doc
README.md	Initial commit
conf.py	add correct config for theme
index.rst	sub menus and home
make.bat	Create doc

Exercice : Créer une documentation avec Readthedocs

19. A présent, faite une release de votre dépôt GitHub avec le tag 1.0
20. Allez dans l'onglet « Version » readthedocs.org du projet « galettes »
21. Observez que votre release 1.0 est disponible
22. Cliquez sur « Activate »
 17. Cochez « Actif »
 18. Cliquez sur « Enregistrer »



Latest release

1.0

31f2f81

Compare ▾

First release

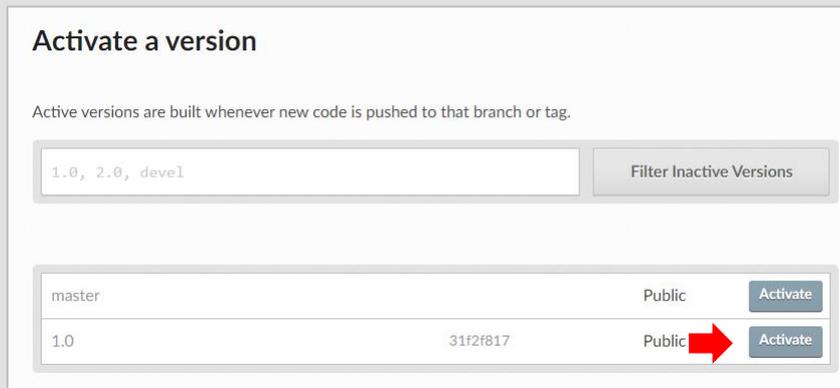
Gaetanbri released this now

1.0

remove moved folders doc_main from root

Assets 2

- Source code (zip)
- Source code (tar.gz)



Activate a version

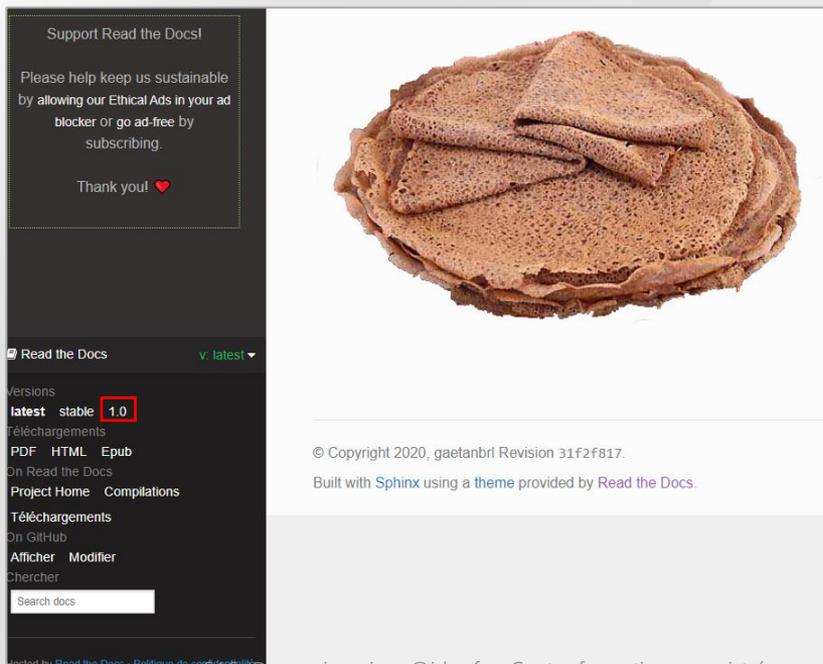
Active versions are built whenever new code is pushed to that branch or tag.

1.0, 2.0, devel Filter Inactive Versions

master	Public	Activate
1.0	Public	Activate

Exercice : Créer une documentation avec Readthedocs

23. Retournez sur la doc et en bas à gauche vous devriez voir une nouvelle version de disponible !



The screenshot shows a dark-themed sidebar on the left and a main content area on the right. The sidebar contains a 'Support Read the Docs' message, a 'Read the Docs' header with a 'v. latest' dropdown, and a 'Versions' section where '1.0' is highlighted in a red box. Below this are links for 'Téléchargements', 'PDF', 'HTML', and 'Epub'. The main content area features a large image of a stack of brown, porous crackers. Below the image, there is a copyright notice: '© Copyright 2020, gaetanbri Revision 31f2f817.' and a note: 'Built with Sphinx using a theme provided by Read the Docs.'

Exercice : Créer une documentation avec Readthedocs

Les sources : → <https://github.com/Gaetanbri/galettes>

Besoin d'aller plus loin ?

<https://sublime-and-sphinx-guide.readthedocs.io/en/latest/>



QUESTIONS ?

Pierre Jego – pierre.jego@jdev.fr
Gaëtan Bruel – gaetan.bruel@jdev.fr