



Git and GitHub

General Applications

Garrett Speed & Aristotelis Kandylas

Data Stewards – Geo Data Team

The background is split horizontally. The top half is orange and features a faint, stylized network diagram of interconnected circles and lines. The bottom half is solid black. A white, torn-paper-like border separates the two sections.

Git

What is Git?

- A versioning program intended for tracking changes in code
- Allows many people to make and track changes independently, and then merge those changes together.
- Created by the creator of Linux, Linus Torvalds, in 2005 for managing development of the Linux kernel

Tracking Changes in Code

- A versioning program intended for tracking changes in code

Tracking Changes in Code

- A versioning program intended for tracking changes in code
- Actually tracks changes in text-based files
 - Text-based like .txt
 - Not text like Word .docx files... unless you save your .docx in a certain way

Tracking Changes in Code

- You can use Git to track changes in
 - Program code
 - Javascript (JS), Python, R, C, C++, C#, Java, Julia, MatLab, Rust, CSS, HTML, etc...
 - Jupyter Lab Notebooks
 - Documents
 - XML-based Word Documents (.docx)
 - Open Document Foundation documents (.odt)
 - Markdown
 - LaTeX

How does it track changes?

- When you initially start tracking a file with Git, it saves the state of the file at that time.
- When you update changes with Git, like changing text on a line, it will record what line number was changed, and whether it was altered, added, or deleted.
- Also able to ignore a situation where unchanged lines are now on different line numbers

How does it track changes?

protis-api/__init__.py

+6 -2

@@ -1,13 +1,17 @@

```
1 - from web import project, department, person,
   publication
```

```
2 # from web import workpackage,
   workpackage_department_template,
   workpackage_funder_template
```

```
3 from fastapi import FastAPI
```

```
4 from fastapi.middleware.cors import CORSMiddleware
```

```
5 - from os import getenv
```

```
6
7 app = FastAPI()
```

```
1 + # LOCAL LIBRARIES
```

```
2 + from web import project, department, person,
   publication, funder
```

```
3 # from web import workpackage,
   workpackage_department_template,
   workpackage_funder_template
```

```
4 + # 3RD PARTY LIBRARIES
```

```
5 from fastapi import FastAPI
```

```
6 from fastapi.middleware.cors import CORSMiddleware
```

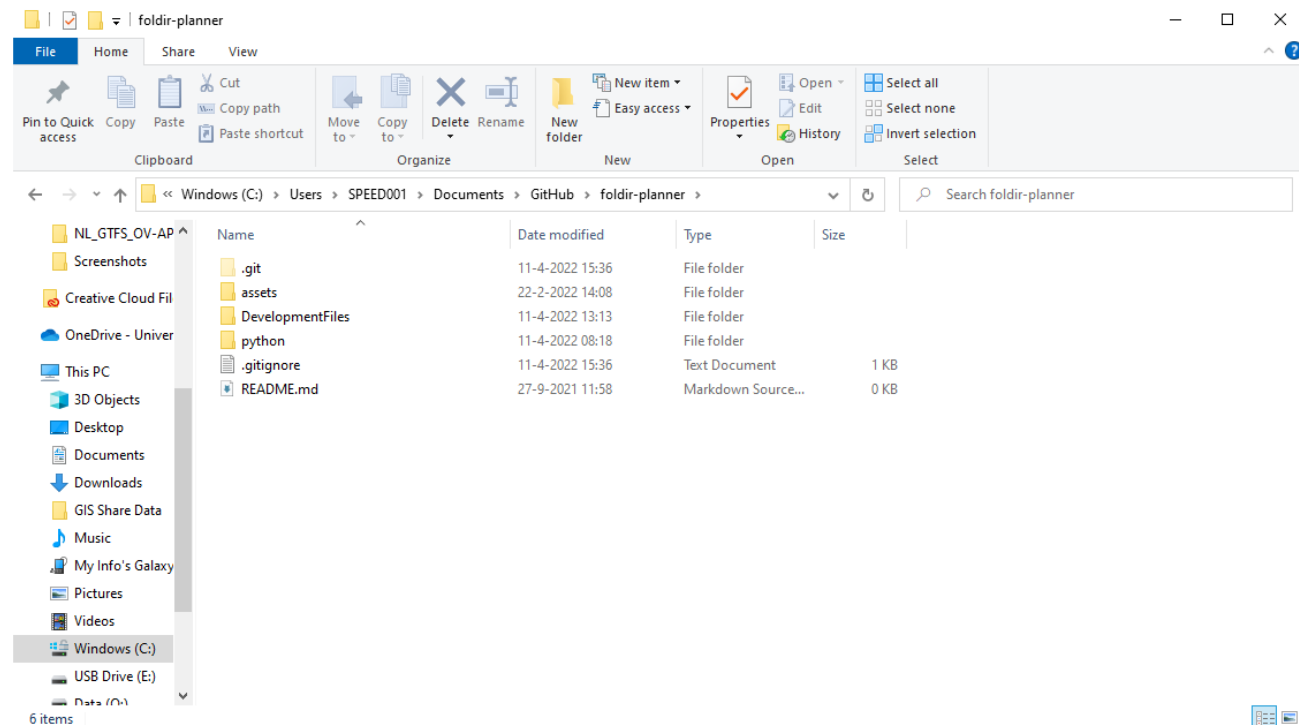
```
7 + # BATTERIES INCLUDED LIBRARIES
```

```
8 + #from os import getenv
```

```
9
10 app = FastAPI()
```

Repositories

- A repository is a project folder that will have the changes in it tracked.
- On your computer, it is a folder with all of the project files in it
- On a Git server is a clone of your local folder



Git Commands

- `git init`
- `git add`
- `git commit`
- `git clone`
- `git checkout`
- `git pull`
- `git merge`
- `git push`

<https://edu.nl/mfnxu>

<https://www.gitkraken.com/learn/git/commands>



edu.nl/mfnxu

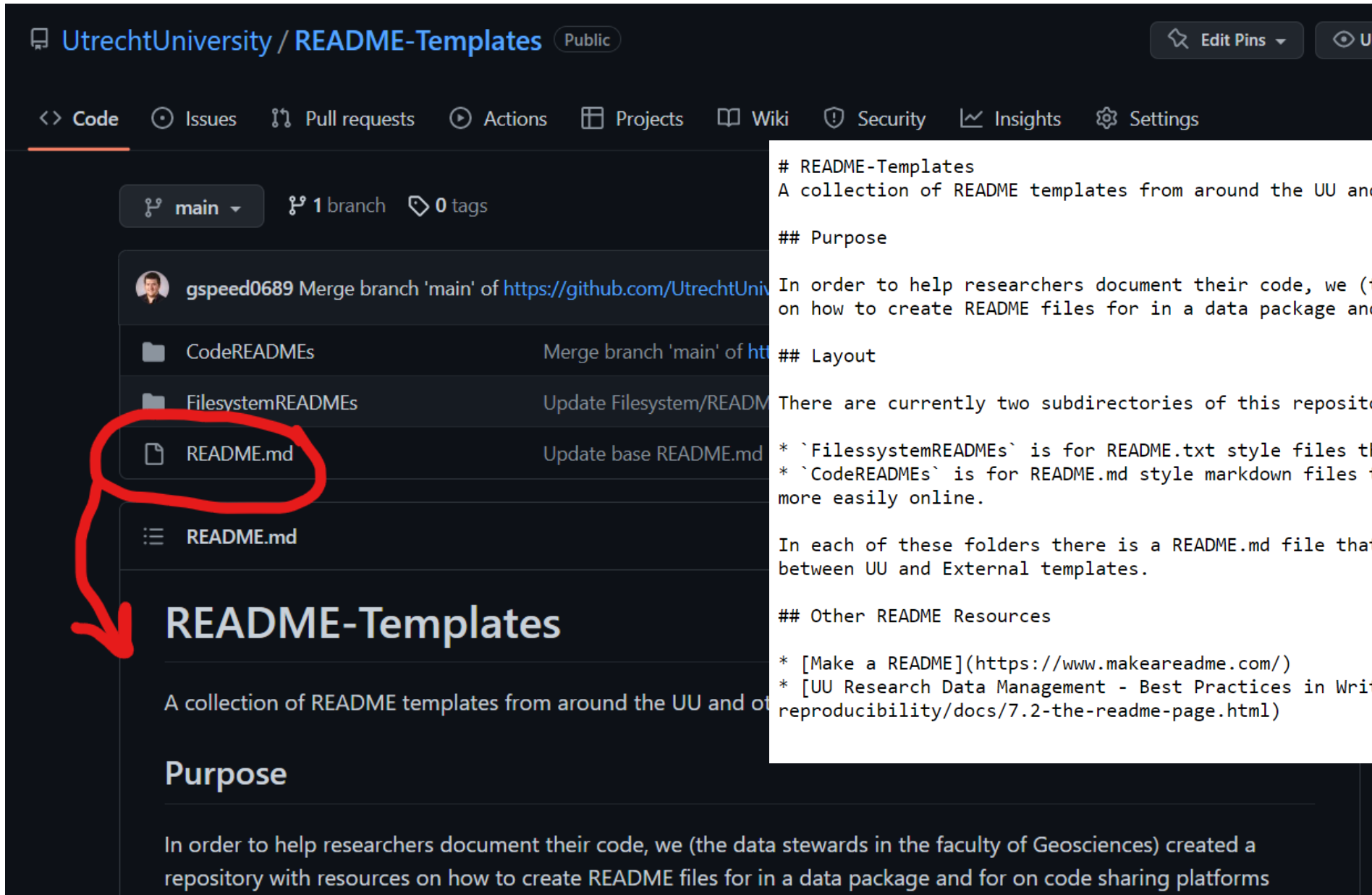
Creating a Repository

- Navigate to your parent folder
- Use the git commands
 - `git init`
 - `git add`
 - `git commit -m {short text about what changes you made}`

README.md

- README markdown documentation file
- GitHub, GitLab, Gitea all recognize README.md filename as the documentation for a folder in a Git repository, and will display the stylized Markdown
- Place a README.md file on most folder levels.
- Find README templates on out GitHub:
[UtrechtUniversity/README-Templates](https://github.com/UtrechtUniversity/README-Templates)

README.md



UtrechtUniversity / README-Templates Public

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags

gspeed0689 Merge branch 'main' of https://github.com/UtrechtUniversity/README-Templates

- CodeREADMEs Merge branch 'main' of https://github.com/UtrechtUniversity/README-Templates
- FilesystemREADMEs Update Filesystem/READMEs
- README.md** Update base README.md

README.md

README-Templates

A collection of README templates from around the UU and other institutions.

Purpose

In order to help researchers document their code, we (the data stewards in the faculty of Geosciences) created a repository with resources on how to create README files for in a data package and for on code sharing platforms such as

README-Templates

A collection of README templates from around the UU and other institutions.

Purpose

In order to help researchers document their code, we (the data stewards in the faculty of Geosciences) created a repository with resources on how to create README files for in a data package and for on code sharing platforms such as

Layout

There are currently two subdirectories of this repository, FilesystemREADMEs and CodeREADMEs.

- * `FilesystemREADMEs` is for README.txt style files that should accompany data packages on the command line
- * `CodeREADMEs` is for README.md style markdown files that should be included with code in GitHub repositories

In each of these folders there is a README.md file that will explain more about each folder, between UU and External templates.

Other README Resources

- * [Make a README](https://www.makeareadme.com/)
- * [UU Research Data Management - Best Practices in Writing Reproducible Code](https://utrecht-research-data-management.github.io/docs/7.2-the-readme-page.html)

Markdown

- Text based document with basic formatting
- Popular in programming communities
- How to write documentation for GitHub and GitLab
- Reddit comments use Markdown
- <https://edu.nl/cer6v>
- <https://www.markdownguide.org/cheat-sheet>



.gitignore

- Sometimes in your folder you have things you don't want tracked:
 - SSH Keys, Passwords, Config Files, AWS Cloud Keys
 - Data, especially personal data
- .gitignore is basic text document of files you don't want to be tracked, or templates of files you don't want tracked.
- [github/gitignore: A collection of useful .gitignore templates](https://github.com/gitignore-io)

.gitignore

```
1  local_config.conf
2  *.ai
3  *checkpoint.ipynb
4  *.pyc
5  */fconfig.py
6  PythonServer/fconfig.py
7  structure_versions/*
8  current_files/*
9  test_results/GeoweteschappenSurvey.foldir
10 PythonServer/IPYNBs/Untitled.ipynb
11 *.pyc
12 PythonServer/__pycache__/*
13 PythonServer/IPYNBs/.ipynb_checkpoints/*
14 DevelopmentFiles/*
15 python/generic_function_testing.py
16 rest-web
```

Code Licenses

- A file named LICENSE should be placed in the repository, GitHub will recognize standard licenses.
- Find out more on our website:
<https://geo-data-support.sites.uu.nl/open-science-open-data/code-licenses/>
- Our recommendations:
 - MIT
 - GNU

Code Licenses

The screenshot shows a GitHub repository interface. At the top, there are buttons for 'Go to file', 'Add file', and '<> Code'. Below these, the repository name 'gspeed0689' is displayed, along with 'Changes made by RE to files' and 'cc80fa8 20 hours ago 9 commits'. A list of files is shown, including 'ChatGPT', 'apache2', 'drupal-tracker', 'focalboard', 'odoo', 'portainer-1', 'portainer-2', 'traefik', 'LICENSE', and 'README.md'. The 'LICENSE' file is circled in red. A red line starts from the 'LICENSE' file and points to the 'MIT license' option in the sidebar. The sidebar also contains sections for 'About', 'Releases', and 'Packages'.

| File | Commit Message | Commit Time |
|----------------|---|--------------|
| ChatGPT | Adding ChatGPT created docker-compose files | yesterday |
| apache2 | Changes made by RE to files | 20 hours ago |
| drupal-tracker | initial commit | last week |
| focalboard | Adding version to docker-compose files | 5 days ago |
| odoo | Adding version to docker-compose files | 5 days ago |
| portainer-1 | initial commit | last week |
| portainer-2 | Fixing \$PWD with ./ | yesterday |
| traefik | Changes made by RE to file | 20 hours ago |
| LICENSE | initial commit | last week |
| README.md | Updating README with short desc | 5 days ago |

About

Docker Compose (and other docker stuff) files for the Geo Data Team

- Readme
- MIT license
- Code of conduct
- 0 stars
- 1 watching
- 0 forks

Releases

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

Packages

No packages published

The background of the image is a vibrant yellow-orange color. It features a network of interconnected circles and lines, resembling a molecular structure or a social network, in a slightly darker shade of the background color. A white, torn-paper-like edge separates the yellow-orange top section from a solid black bottom section.

GitHub

What is GitHub

- An online Git server with a friendly user interface for seeing, downloading, and managing Git projects.
- Now owned by Microsoft
- Most popular code repository

Creating a Repository

- You can create a new empty repository on the GitHub website
- You will need a name for your repository, and it will be addressable as
`{your_username}/{repository_name}`
- Options to add a default license
- Options to add a ReadMe
- Options to add a .gitignore file

GitHub Desktop

- A GUI for managing Git projects.
 - Integrated with GitHub
 - Not required to use a GitHub hosted Git repository
- Today, we're going to use GitHub Desktop, for ease
 - The command line is easier to use on Mac and Linux as those operating systems are Unix based
- No need for SSH keys if working with GitHub

GitHub Desktop

The screenshot shows the GitHub Desktop application window. The top bar includes a menu (File, Edit, View, Repository, Branch, Help) and window controls. Below the menu, the current repository is 'GEE_ClassDemo_2022-05-04_v01' and the current branch is 'main'. A 'Fetch origin' button is available, showing it was last fetched 8 minutes ago. A notification bar states: 'An updated version of GitHub Desktop is available and will be installed at the next launch. See [what's new](#) or [restart GitHub Desktop](#).' The main area is divided into three panes. The left pane shows 'Changes' with 5 files: .gitignore, GEE_Scripts\Landsat_89_NDVI.js, GEE_Scripts\Landsat_89_NDWI.js, README.md, and Samples\NDVI_062003.png. The middle pane shows the 'History' tab. The right pane shows the 'README.md' file content, which includes a header, a description of the class demo, and a script for NDVI analysis. The bottom pane shows the commit message 'Initial commit of first content' and a description: 'Adding a .gitignore, file placeholders for the Landsat NDWI and NDVI analysis, and example of NDVI analysis for documentation.' A 'Commit to main' button is at the bottom.

File Edit View Repository Branch Help

Current repository: GEE_ClassDemo_2022-05-04_v01

Current branch: main

Fetch origin
Last fetched 8 minutes ago

An updated version of GitHub Desktop is available and will be installed at the next launch. See [what's new](#) or [restart GitHub Desktop](#).

Changes (5) History README.md

5 changed files

- .gitignore
- GEE_Scripts\Landsat_89_NDVI.js
- GEE_Scripts\Landsat_89_NDWI.js
- README.md
- Samples\NDVI_062003.png

Initial commit of first content

Adding a [.gitignore](#), file placeholders for the Landsat NDWI and NDVI analysis, and example of NDVI analysis for documentation.

Commit to main

```
@@ -1,2 +1,11 @@
1  # GEE_ClassDemo_2022-05-04_v01
2  Class Demo for a Google Earth Engine Repository layout, course taking place 2022-05-04. v01 Pre-class
3  +
4  +# Landsat_89_NDVI.js
5  +
6  +This script will perform an NDVI analysis on Landsat 8 and Landsat 9 datasets.
7  +![NDVI_062003.png from Wikipedia used as an example. Shared CC-BY-SA by Gennaro Cappelluti](Samples/NDV
8  +
9  +# Landsat_89_NDWI.js
10 +
11 +This script will perform an NDVI analysis on Landsat 8 and Landsat 9 datasets. 📄
```



Gist


- Quick sharing site for single file code projects
- gist.github.com

GitHub Gist

Search...

All gists Back to GitHub

 + 

 **gspeed0689** / **Kadaster_koopsominformatie.py**
Created 3 months ago


Unsubscribe Edit Delete Star 0

<> Code

Revisions 1

Embed <script src="https://" Copy Download ZIP

Tool to extract the table from Kadaster koopsominformatie reports.

 **Kadaster_koopsominformatie.py**

Raw

```
1  # pip libraries
2  from pypdf import PdfReader
3  import pandas as pd
4  # builtins
5  from datetime import datetime
6  import argparse
7
8  def args():
9      parser = argparse.ArgumentParser()
10     parser.add_argument("-i", "--input", help="Input Kadaster Koopsominformatie Report")
```

UU GitHub Organization

- Utrecht University has its own GitHub organization, allowing for collaboration officially as the university
- Unlocks more tools and storage for projects
- Requires Multi-Factor Authentication activated on your GitHub account
- Sign-in at <https://github.com/UtrechtUniversity>

Other Git Platforms



GitHub



GitLab



Gitea

Software as a Service



Self-hostable



CI/CD



Social Media Like +
Code Discoverability
Tools



Let's git practical

g.t.speed@uu.nl

datateam.geo@uu.nl

Tasks

- Log into GitHub
- Create a new repository
- Create a Markdown Readme
- Create folders
 - Data
 - Code
- Create a .gitignore file
- Commit changes to code and push to GitHub