Continuous Integration

Jérôme Dockès

Nilearn Dev Days 2020

Continuous Integration (CI)

- One shared master branch
- Small changes merged in frequently
- master kept bug-free and ready to deploy

Relies on:

- Version control
- Automated testing and quality control

Software should be like a firefighter

- Ready to jump into action
- Reliable once on site

This entails

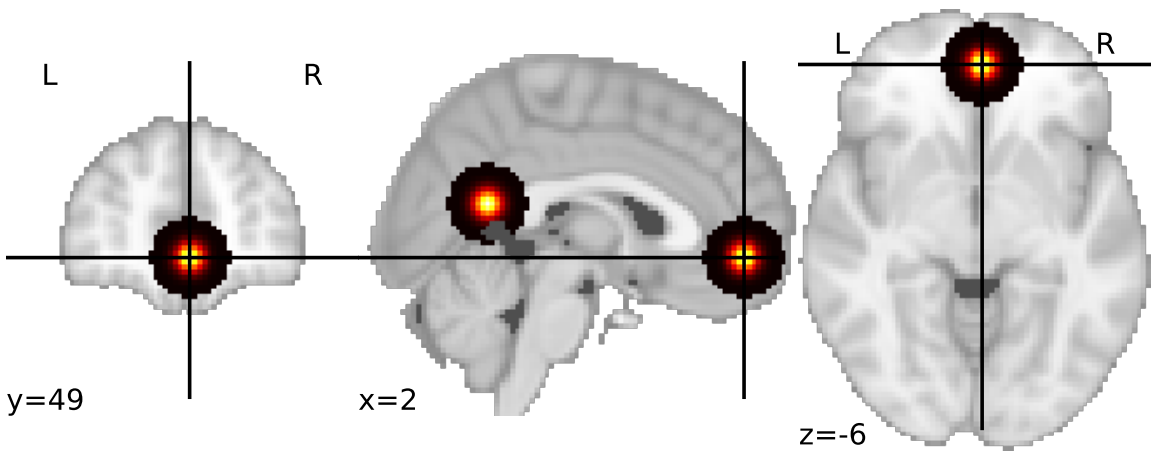
- Keeping master up-to-date: incremental changes
- Constantly testing on diverse environments

We can get it easily and for free!

Illustration on a toy package: brain_coords

https://github.com/jeromedockes/brain_coords

```
from nilearn import plotting
from brain_coords.coordinates import (
    coords_to_img, load_coords)

img = coords_to_img(load_coords())
plotting.plot_stat_map(img, colorbar=False)
```



brain_coords

```
brain_coords/
├── brain_coords/
│   ├── data/
│   │   ├── __init__.py
│   │   └── dmn_coords.txt
│   ├── tests/
│   │   ├── __init__.py
│   │   └── test_coordinates.py
│   ├── __init__.py
│   └── coordinates.py
├── examples/
│   └── demo.py
├── README.md
└── setup.py
```

See this and this for arguments for a different layout.

brain_coords.coordinates.load_coords

```
def load_coords(coords_file=None):
    if coords_file is None:
        coords_file = (
            pathlib.Path(__file__).parent
            / "data"
            / "dmn_coords.txt"
        )
    coords_file = pathlib.Path(coords_file)
    if (ext := coords_file.suffix) != ".txt":
        print("We expect .txt format")
    return np.loadtxt(str(coords_file))
```

setup.py

```
setup(
    name="brain_coords",
    packages=find_packages(),
    install_requires=["nilearn"],
    python_requires=">=3",
)
```

pytest --cov=brain_coords

```
============================= test session starts ==============================
platform linux -- Python 3.8.2, pytest-5.4.1, py-1.8.1, pluggy-0.13.1
plugins: cov-2.8.1
collected 2 items

brain_coords/tests/test_coordinates.py ..                                [100%]

----------- coverage: platform linux, python 3.8.2-final-0 -----------
Name                                             Stmts   Miss  Cover
--------------------------------------------------------------------
brain_coords/__init__.py                             0      0   100%
brain_coords/coordinates.py                         18      0   100%
brain_coords/data/__init__.py                        0      0   100%
brain_coords/tests/__init__.py                       0      0   100%
brain_coords/tests/test_coordinates.py              12      0   100%
--------------------------------------------------------------------
TOTAL                                               30      0   100%


============================== 2 passed in 1.38s ===============================
```

"Works on my machine / my container"

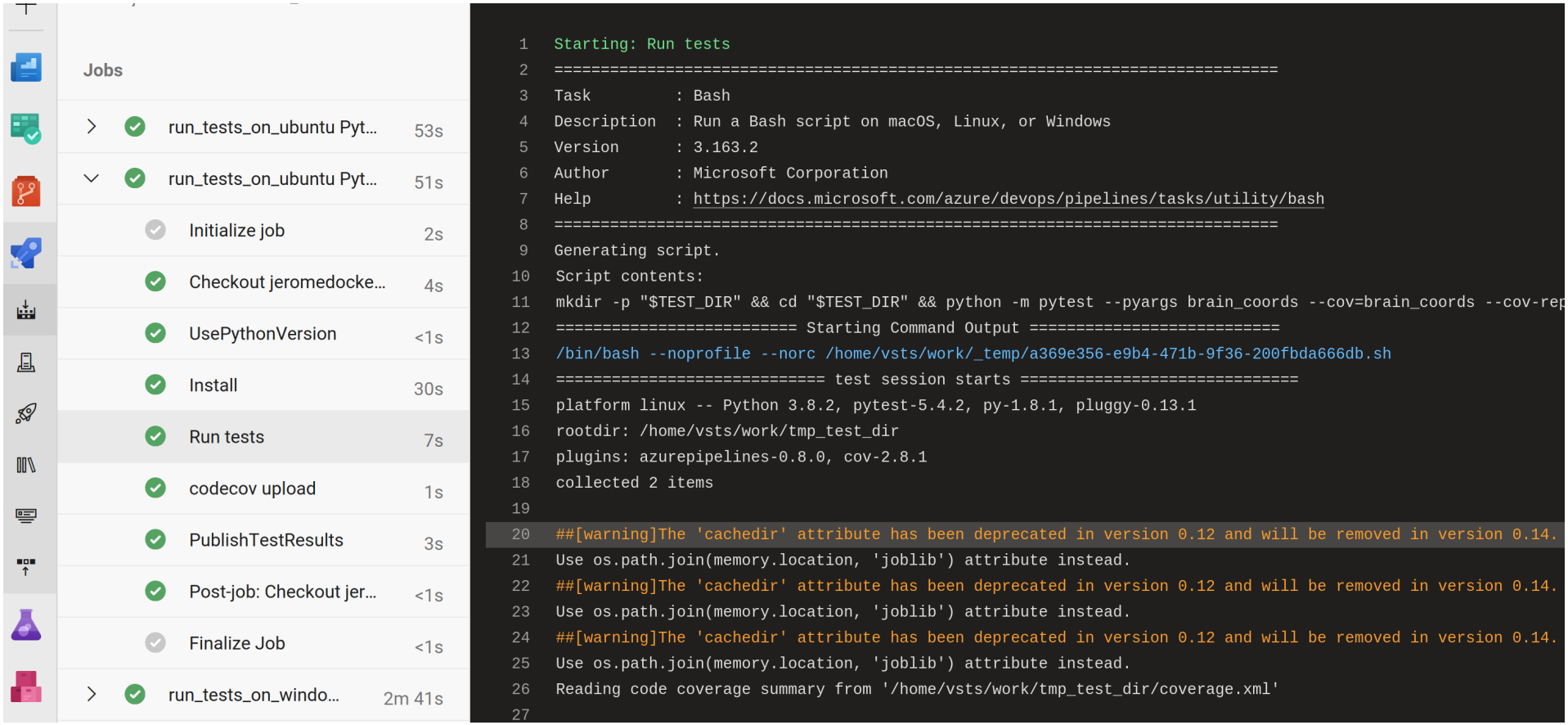
(for better local automated testing use tox – can be combined with what follows)

Continuous integration platforms

- Azure pipelines, Travis, CircleCI, AppVeyor . . .
- set up an account; have it follow our repository
- add a configuration file to the repository
- CI tools listens to some events:
  - pushing to a branch
  - pull requests
- events trigger a "build":
  - obtain resources on the provider’s infrastructure
  - clone the repository
  - run our tests or whatever we ask for in the configuration
  - store results ("artifacts")
- for PRs, checkout is a merge commit into the target branch

Setting up Azure pipelines

- Add configuration file `azure-pipelines.yml`
- Set up account
- Add uploading codecov reports and saving artifacts



documentation

adding the configuration file

```
brain_coords/
├── README.md
├── azure-pipelines.yml
├── ci/
│   └── azure_test_steps.yml
├── brain_coords/
│   └── __init__.py
└── setup.py
```

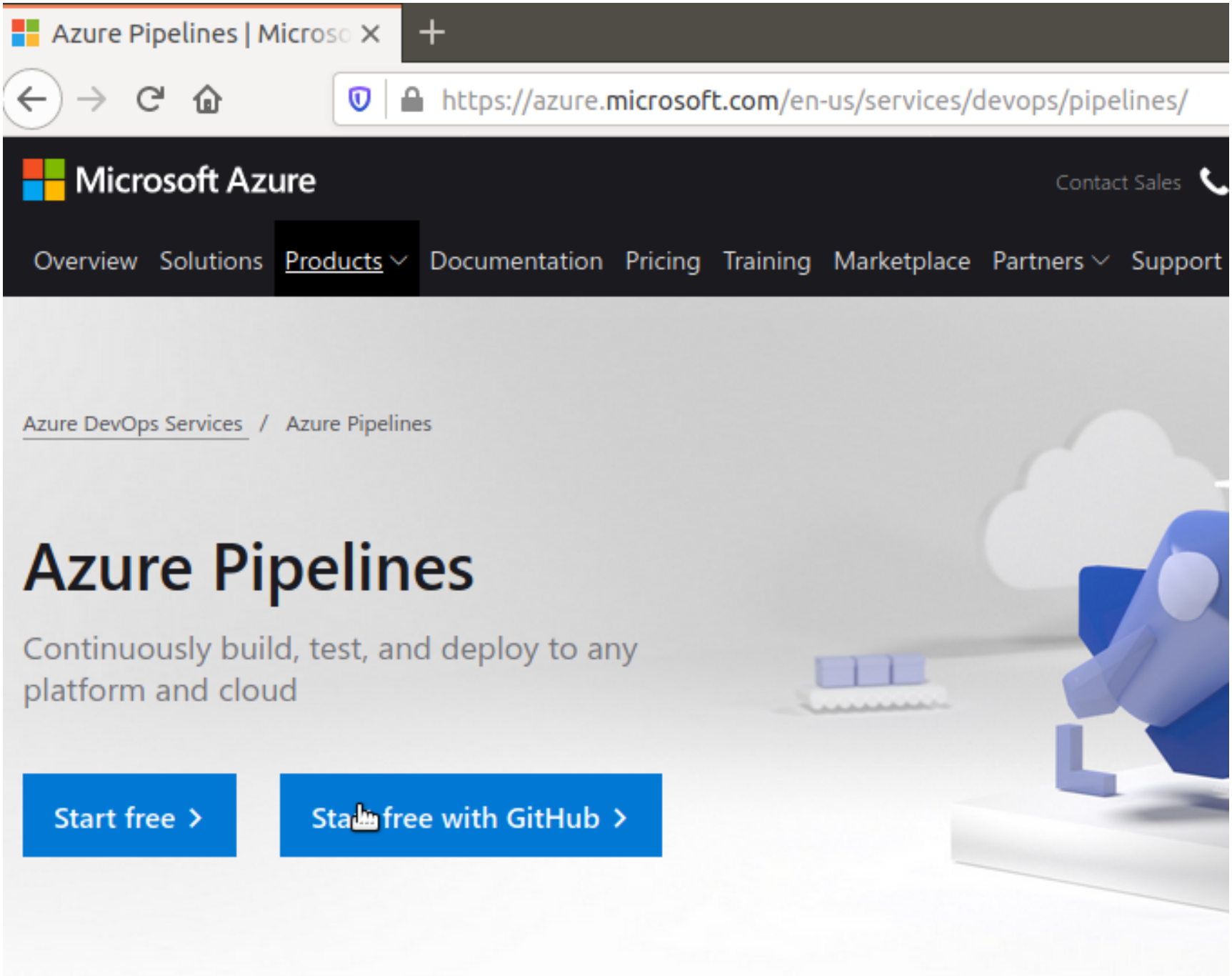
azure-pipelines.yml

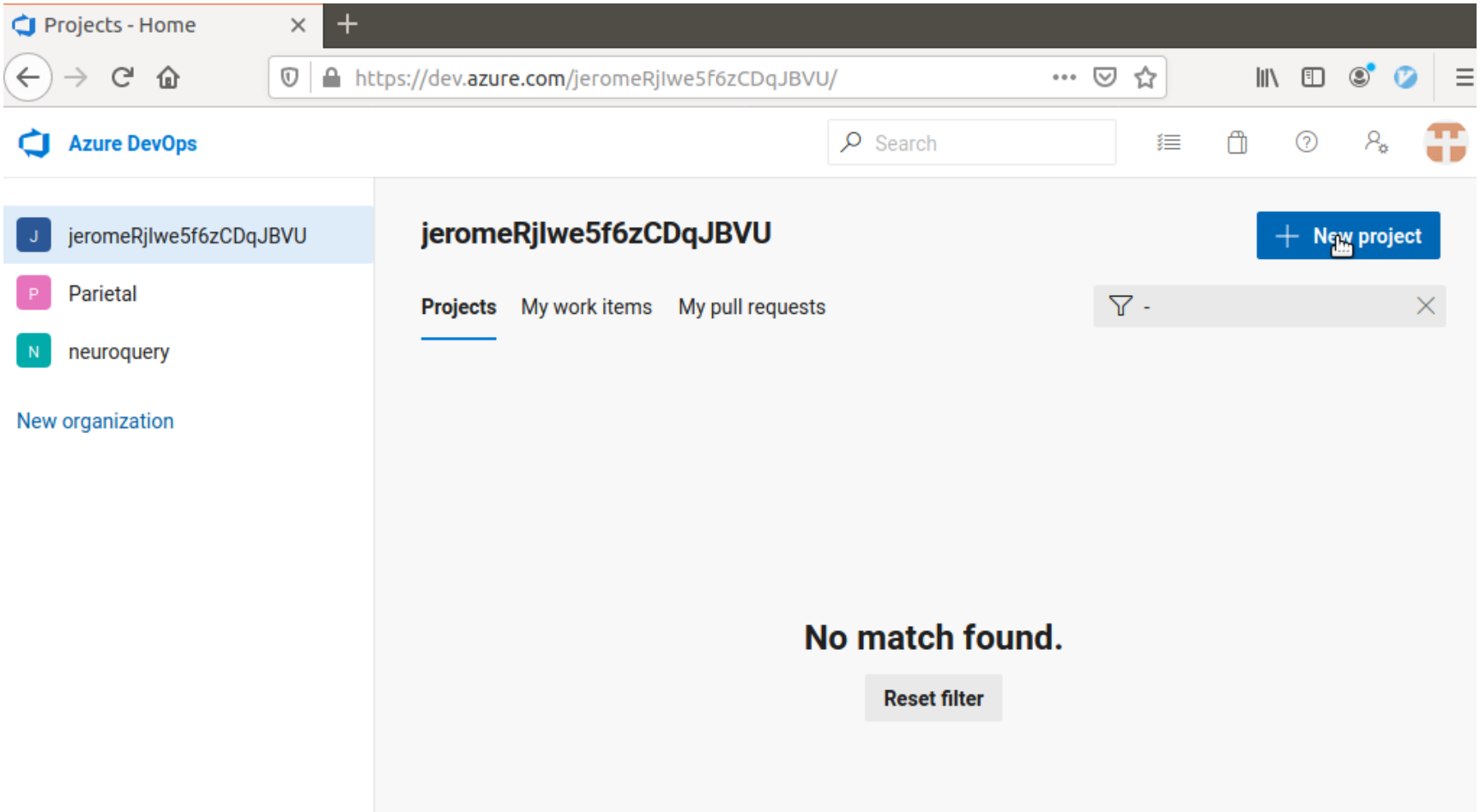
```yaml
trigger:
  - master

jobs:
  - job: 'run_tests_on_ubuntu'
    pool:
      vmImage: 'ubuntu-latest'
    variables:
      TEST_DIR: '$(Agent.WorkFolder)/tmp_test_dir'
    strategy:
      matrix:
        Python36:
          python.version: '3.6'
          numpy.spec: 'numpy==1.16'
        Python38:
          python.version: '3.8'
          numpy.spec: 'numpy'
    steps:
      - template: ci/azure_test_steps.yml
```

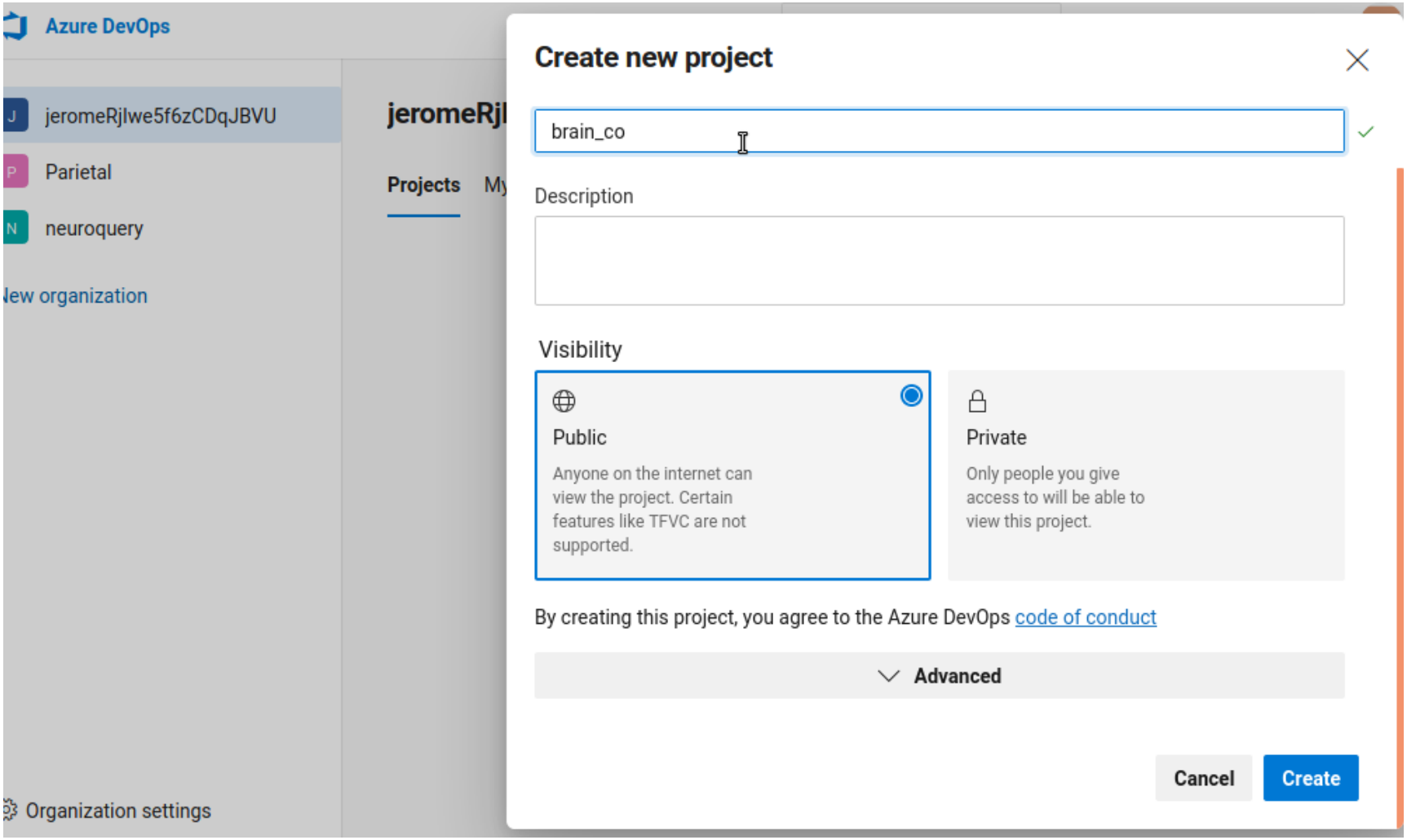
ci/azure_test_steps.yml

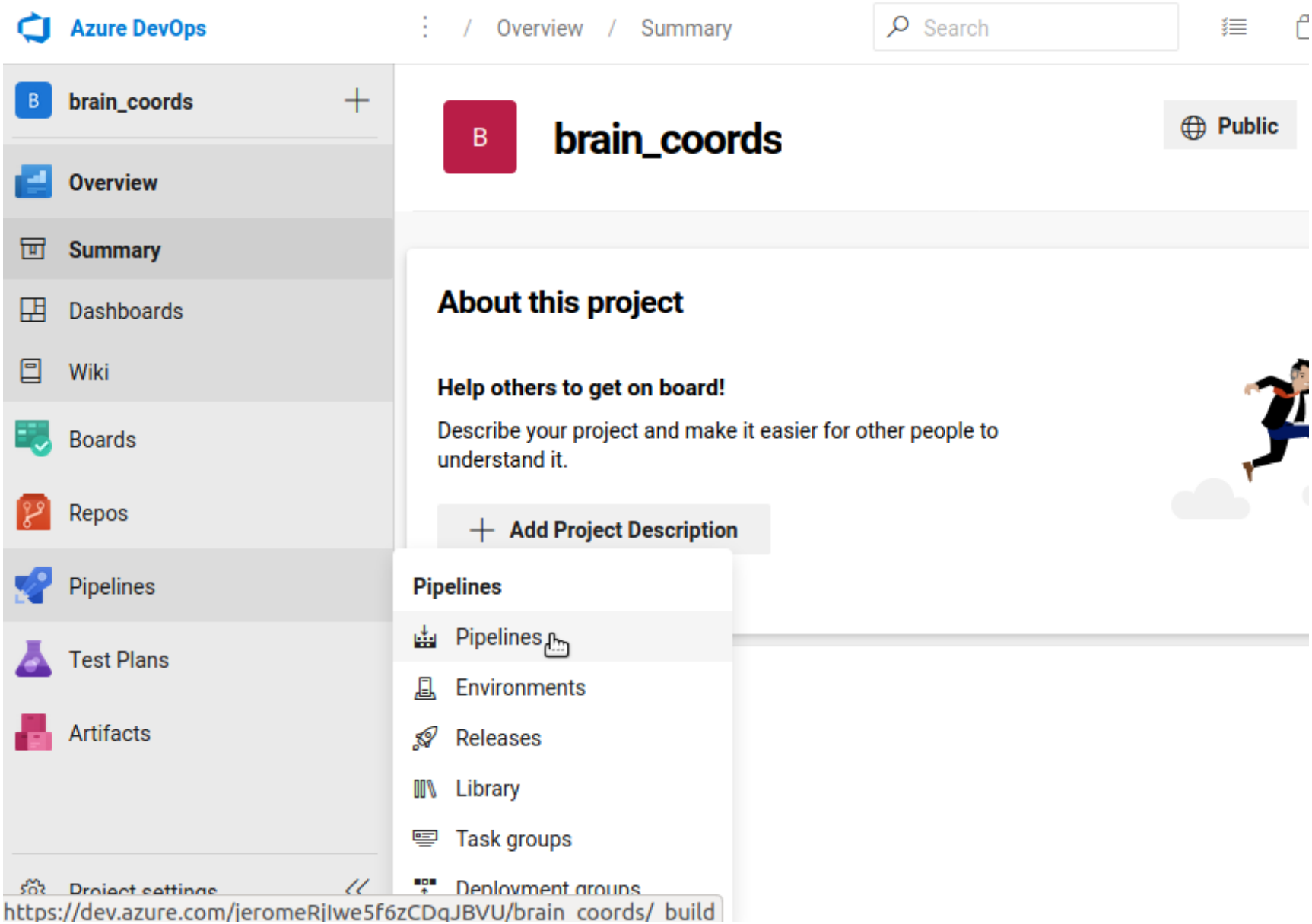
```yaml
steps:
  - task: UsePythonVersion@0
    inputs:
      versionSpec: '$(python.version)'
  - script: |
      python -m pip install --upgrade pip
      pip install pytest pytest-azurepipelines $(numpy.spec)
      pip install .
    displayName: 'Install'
  - bash: >
      mkdir -p "$TEST_DIR" && cd "$TEST_DIR" &&
      python -m pytest --pyargs brain_coords
      -o junit_family=xunit2 --junitxml=test-results.xml
    displayName: 'Run tests'
  - task: PublishTestResults@2
    inputs:
      testResultsFiles: '$(TEST_DIR)/test-results.xml'
      testRunTitle: 'Python $(python.version)'
    condition: succeededOrFailed()
```

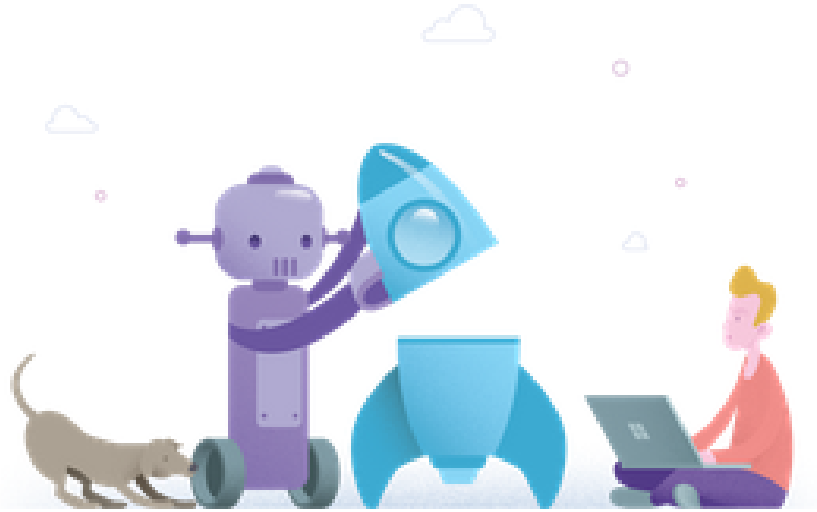
pipeline tasks documentation


Azure Pipelines | Microsc
https://azure.microsoft.com/en-us/services/devops/pipelines/
Microsoft Azure
Contact Sales
Overview
Solutions
Products
Documentation
Pricing
Training
Marketplace
Partners
Support
Azure DevOps Services / Azure Pipelines
Azure Pipelines
Continuously build, test, and deploy to any platform and cloud
Start free >
Start free with GitHub >


Projects - Home
https://dev.azure.com/jeromeRjlwe5f6zCDqJBVU/
Azure DevOps
Search
jeromeRjlwe5f6zCDqJBVU
Parietal
neuroquery
New organization
jeromeRjlwe5f6zCDqJBVU
New project
Projects
My work items
My pull requests
No match found.
Reset filter


Azure DevOps
jeromeRjlwe5f6zCDqJBVU
Parietal
neuroquery
New organization
Organization settings
jeromeRj
Projects
Create new project
brain_co
Description
Visibility
Public
Anyone on the internet can view the project. Certain features like TFVC are not supported.
Private
Only people you give access to will be able to view this project.
By creating this project, you agree to the Azure DevOps code of conduct
Advanced
Cancel
Create

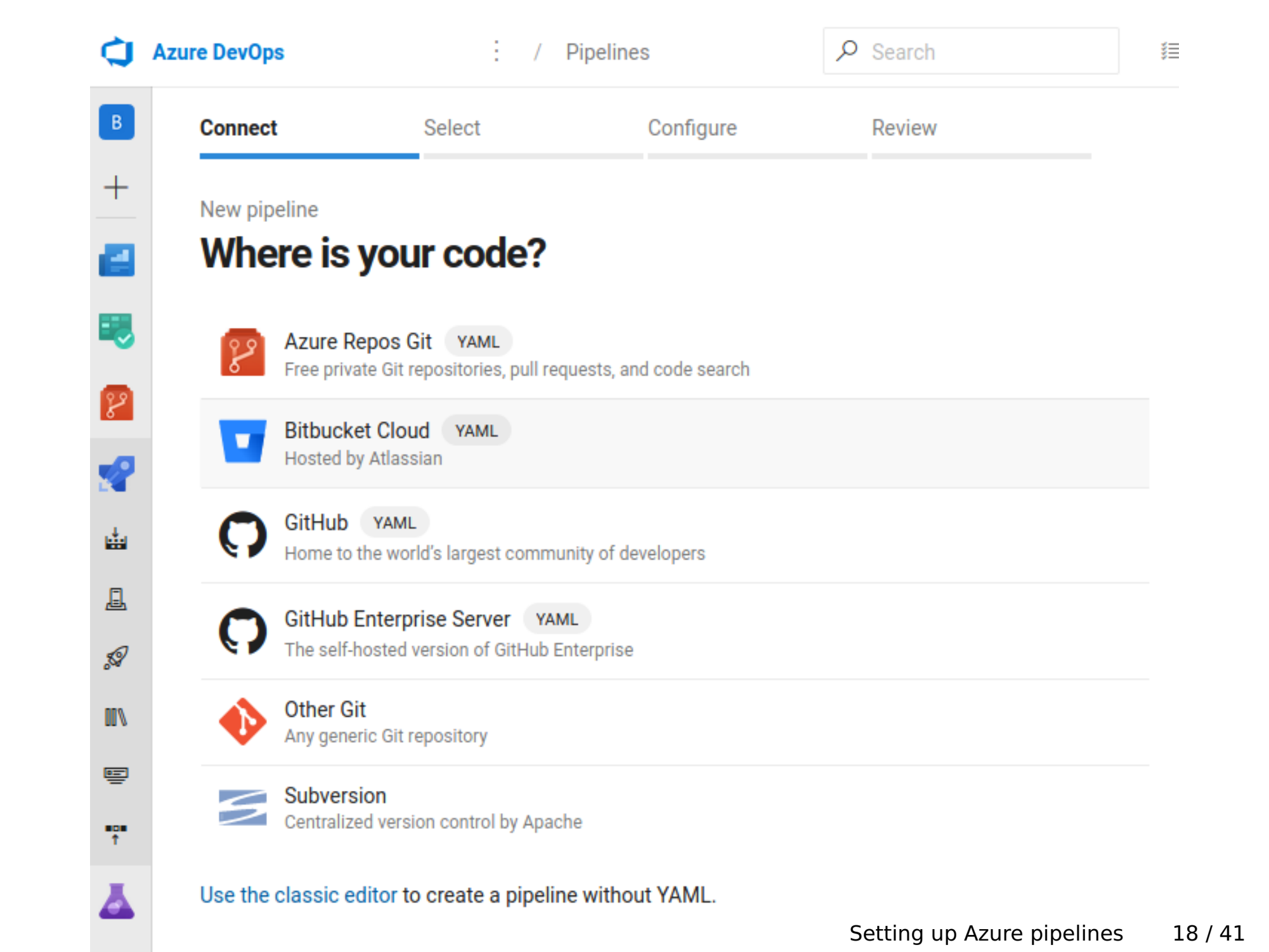

Azure DevOps
Overview
Summary
Search
brain_coords
Overview
Summary
Dashboards
Wiki
Boards
Repos
Pipelines
Test Plans
Artifacts
Project settings
B
brain_coords
Public
About this project
Help others to get on board!
Describe your project and make it easier for other people to understand it.
Add Project Description
Pipelines
Pipelines
Environments
Releases
Library
Task groups
Deployment groups
https://dev.azure.com/jeromeRilwe5f6zCDqJBVU/brain_coords/_build

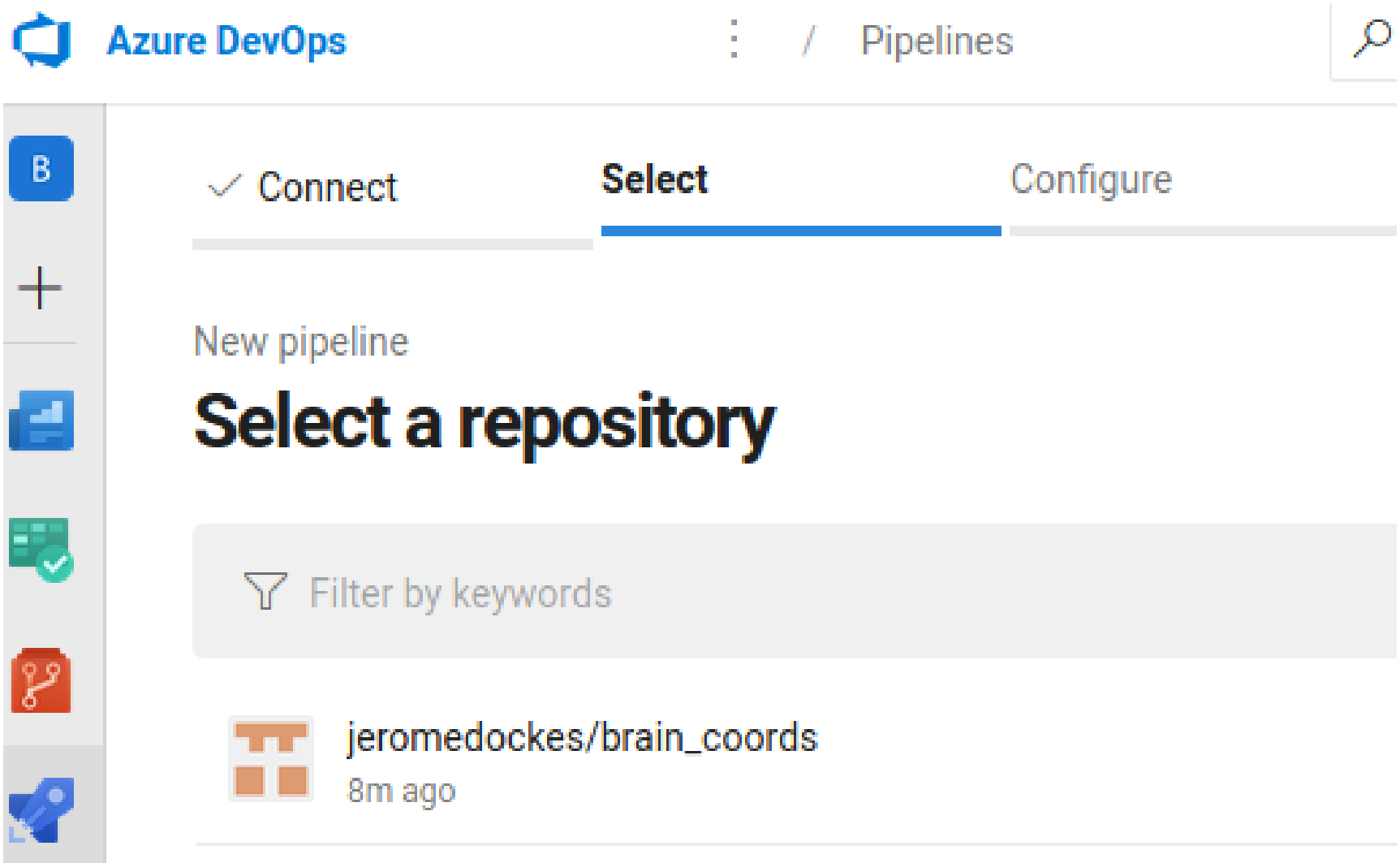


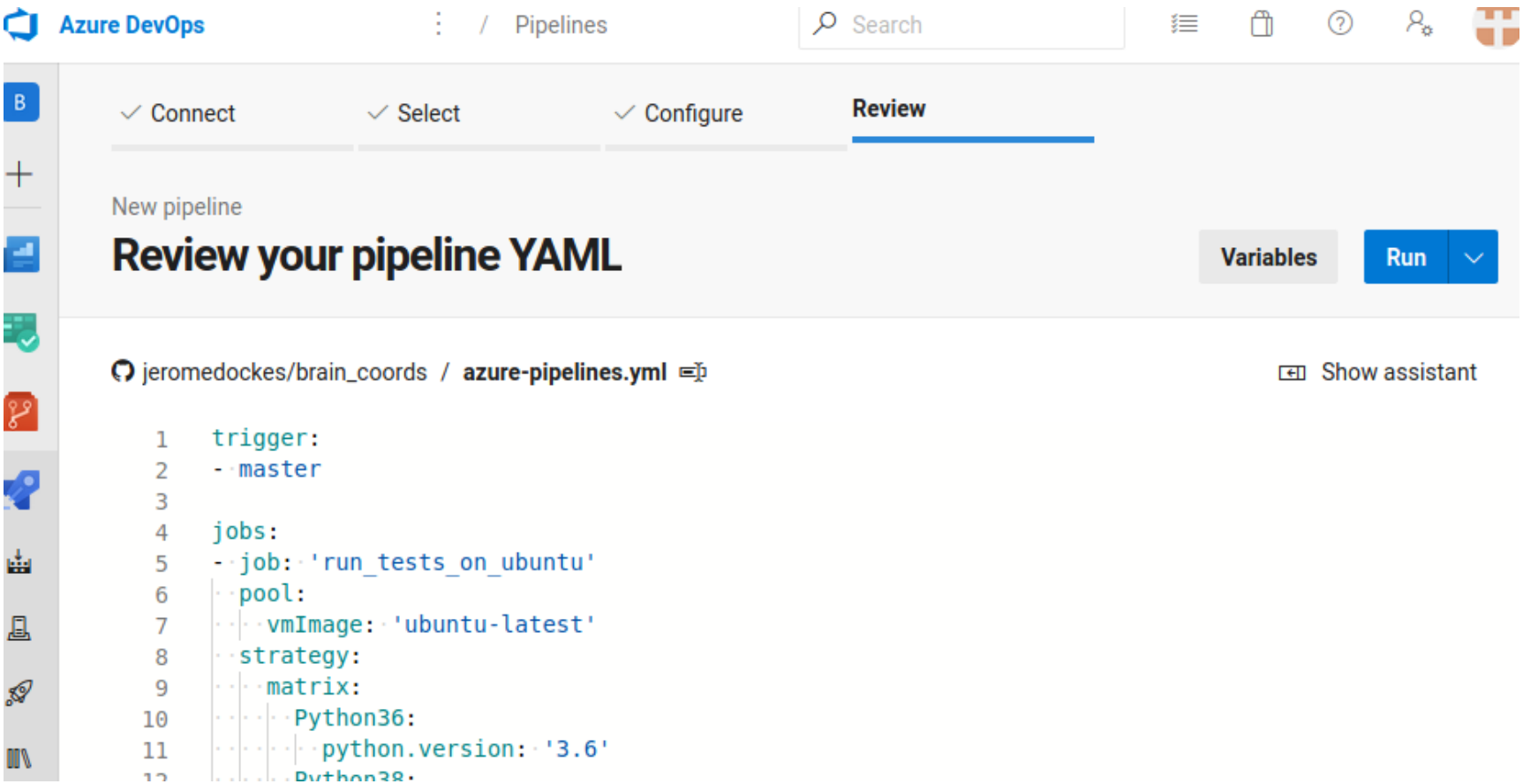
Create your first Pipeline

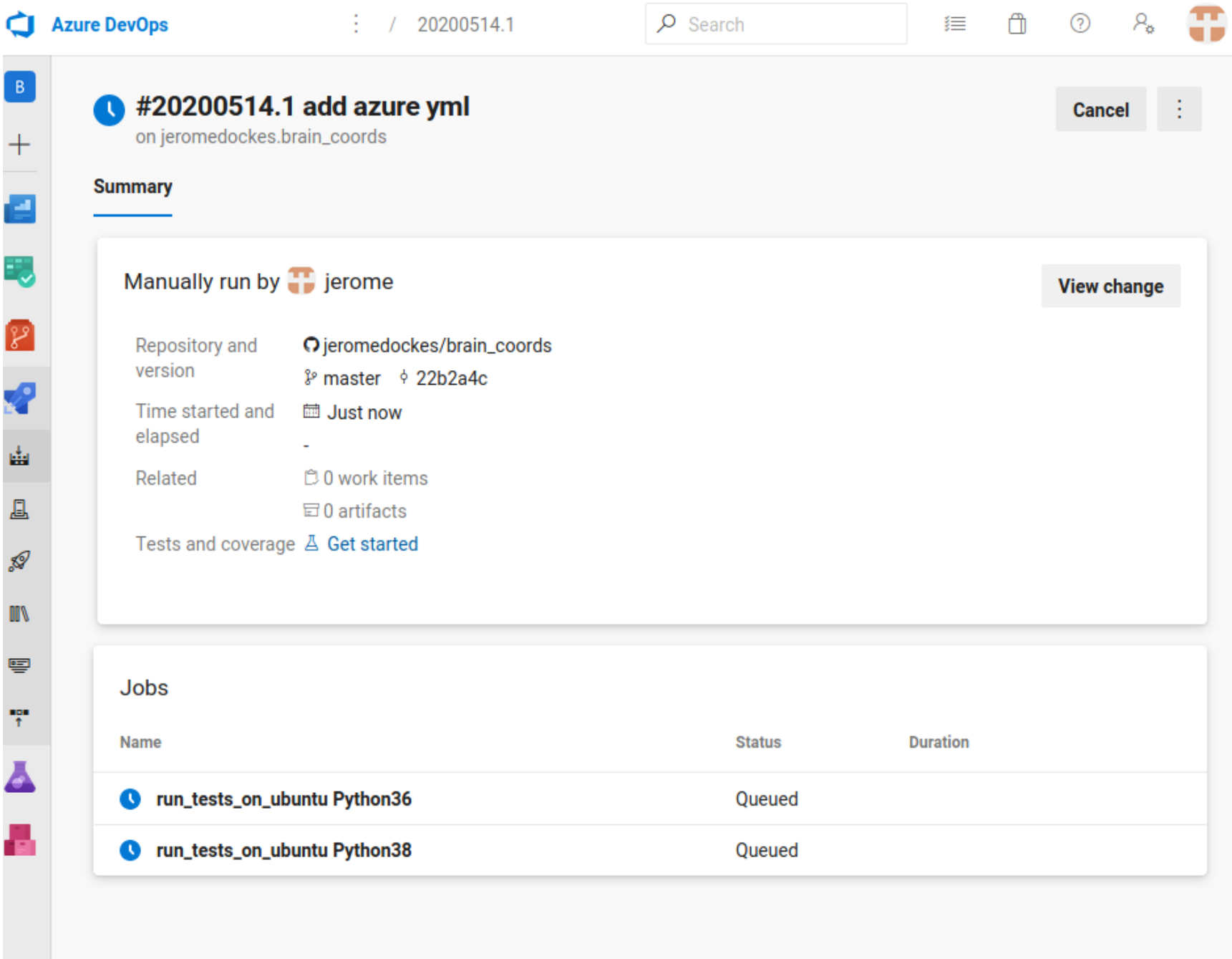
Automate your build and release processes using our wizard, and go from code to cloud-hosted within minutes.

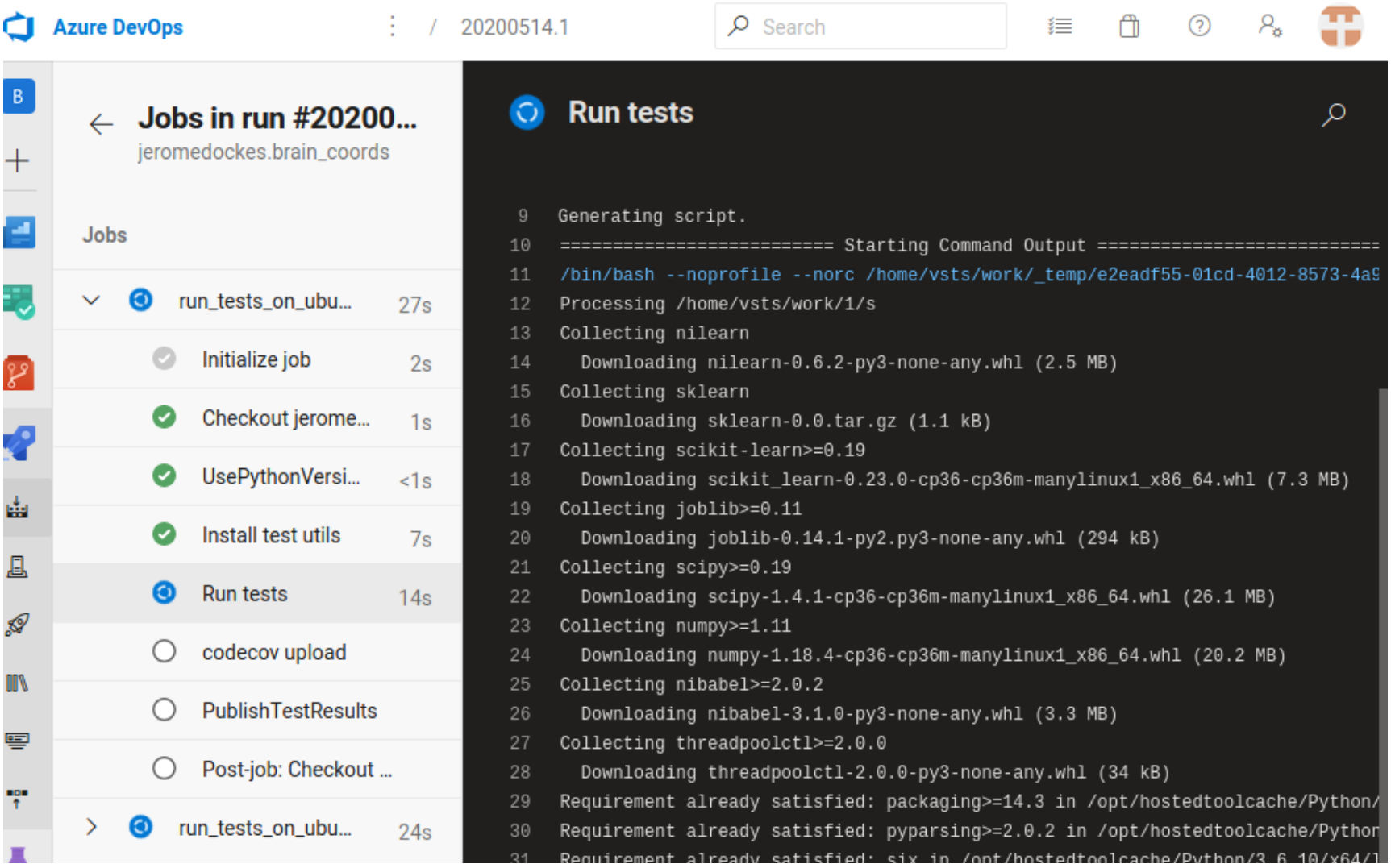
Create Pipeline ⋮


Azure DevOps
Pipelines
Search
Connect
Select
Configure
Review
New pipeline
Where is your code?
Azure Repos Git
YAML
Free private Git repositories, pull requests, and code search
Bitbucket Cloud
YAML
Hosted by Atlassian
GitHub
YAML
Home to the world's largest community of developers
GitHub Enterprise Server
YAML
The self-hosted version of GitHub Enterprise
Other Git
Any generic Git repository
Subversion
Centralized version control by Apache
Use the classic editor to create a pipeline without YAML.

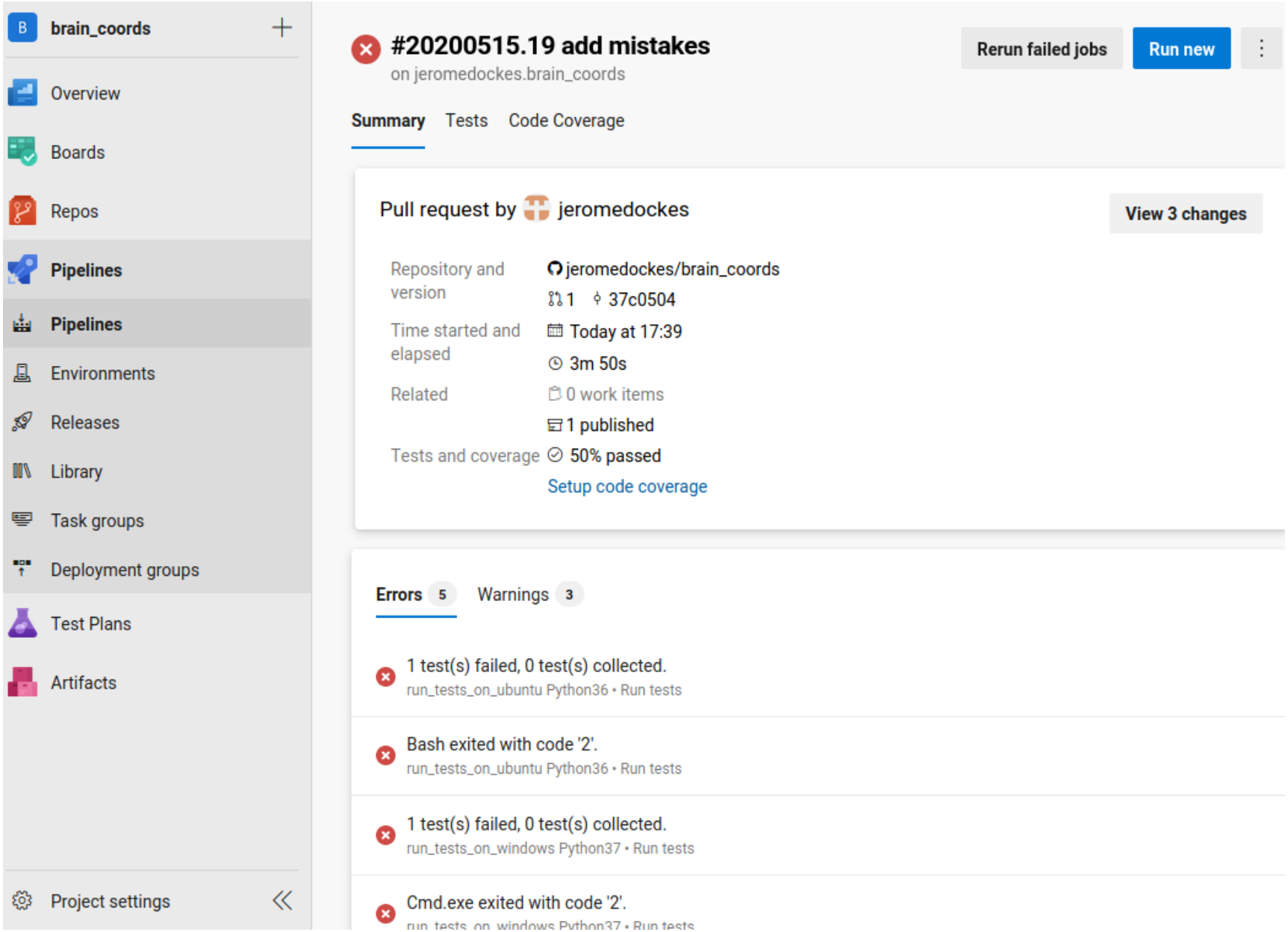

Azure DevOps
/ Pipelines
Connect
Select
Configure
New pipeline
Select a repository
Filter by keywords
jeromedockes/brain_coords
8m ago


Azure DevOps
Pipelines
Search
Connect
Select
Configure
Review
New pipeline
Review your pipeline YAML
Variables
Run
jeromedockes/brain_coords / azure-pipelines.yml
Show assistant
trigger:
- master
jobs:
- job: 'run_tests_on_ubuntu'
pool:
vmImage: 'ubuntu-latest'
strategy:
matrix:
Python36:
python.version: '3.6'

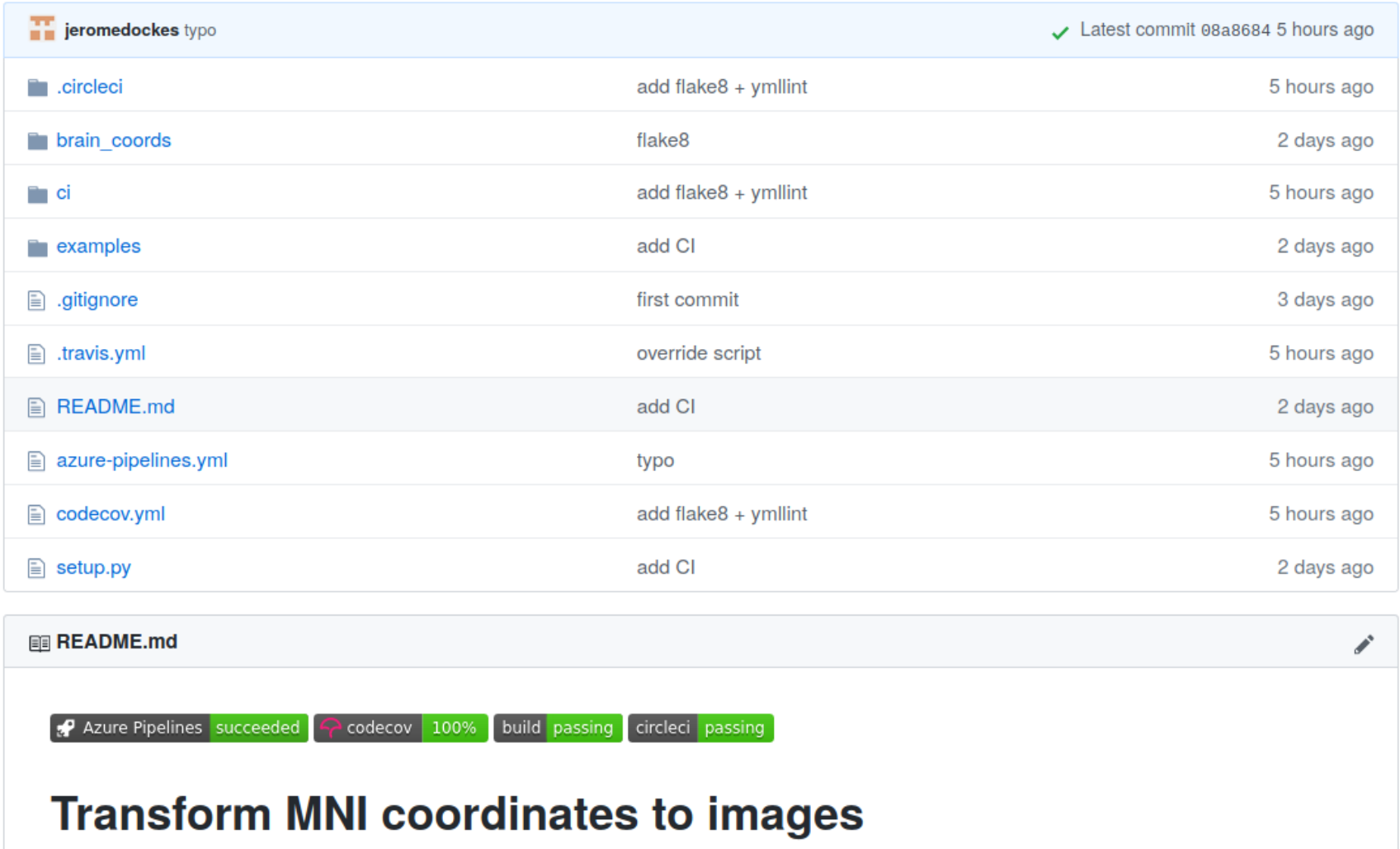

Azure DevOps
20200514.1
Search
#20200514.1 add azure yml
on jeromedockes.brain_coords
Cancel
Summary
Manually run by jerome
View change
Repository and version
jeromedockes/brain_coords
master
22b2a4c
Time started and elapsed
Just now
-
Related
0 work items
0 artifacts
Tests and coverage
Get started
Jobs
Name
Status
Duration
run_tests_on_ubuntu Python36
Queued
run_tests_on_ubuntu Python38
Queued


Azure DevOps
20200514.1
Search
Jobs in run #20200...
jeromedockes.brain_coords
Jobs
run_tests_on_ubu... 27s
Initialize job 2s
Checkout jerome... 1s
UsePythonVersi... <1s
Install test utils 7s
Run tests 14s
codecov upload
PublishTestResults
Post-job: Checkout ...
run_tests_on_ubu... 24s
Run tests
Generating script.
========== Starting Command Output ==========
/bin/bash --noprofile --norc /home/vsts/work/_temp/e2eadf55-01cd-4012-8573-4a9
Processing /home/vsts/work/1/s
Collecting nilearn
Downloading nilearn-0.6.2-py3-none-any.whl (2.5 MB)
Collecting sklearn
Downloading sklearn-0.0.tar.gz (1.1 kB)
Collecting scikit-learn>=0.19
Downloading scikit_learn-0.23.0-cp36-cp36m-manylinux1_x86_64.whl (7.3 MB)
Collecting joblib>=0.11
Downloading joblib-0.14.1-py2.py3-none-any.whl (294 kB)
Collecting scipy>=0.19
Downloading scipy-1.4.1-cp36-cp36m-manylinux1_x86_64.whl (26.1 MB)
Collecting numpy>=1.11
Downloading numpy-1.18.4-cp36-cp36m-manylinux1_x86_64.whl (20.2 MB)
Collecting nibabel>=2.0.2
Downloading nibabel-3.1.0-py3-none-any.whl (3.3 MB)
Collecting threadpoolctl>=2.0.0
Downloading threadpoolctl-2.0.0-py3-none-any.whl (34 kB)
Requirement already satisfied: packaging>=14.3 in /opt/hostedtoolcache/Python/
Requirement already satisfied: pyparsing>=2.0.2 in /opt/hostedtoolcache/Python

CI uncovers hidden mistakes



Seen from GitHub



jeromedockes typo — Latest commit 08a8684 5 hours ago

.circleci	add flake8 + ymllint	5 hours ago
brain_coords	flake8	2 days ago
ci	add flake8 + ymllint	5 hours ago
examples	add CI	2 days ago
.gitignore	first commit	3 days ago
.travis.yml	override script	5 hours ago
README.md	add CI	2 days ago
azure-pipelines.yml	typo	5 hours ago
codecov.yml	add flake8 + ymllint	5 hours ago
setup.py	add CI	2 days ago

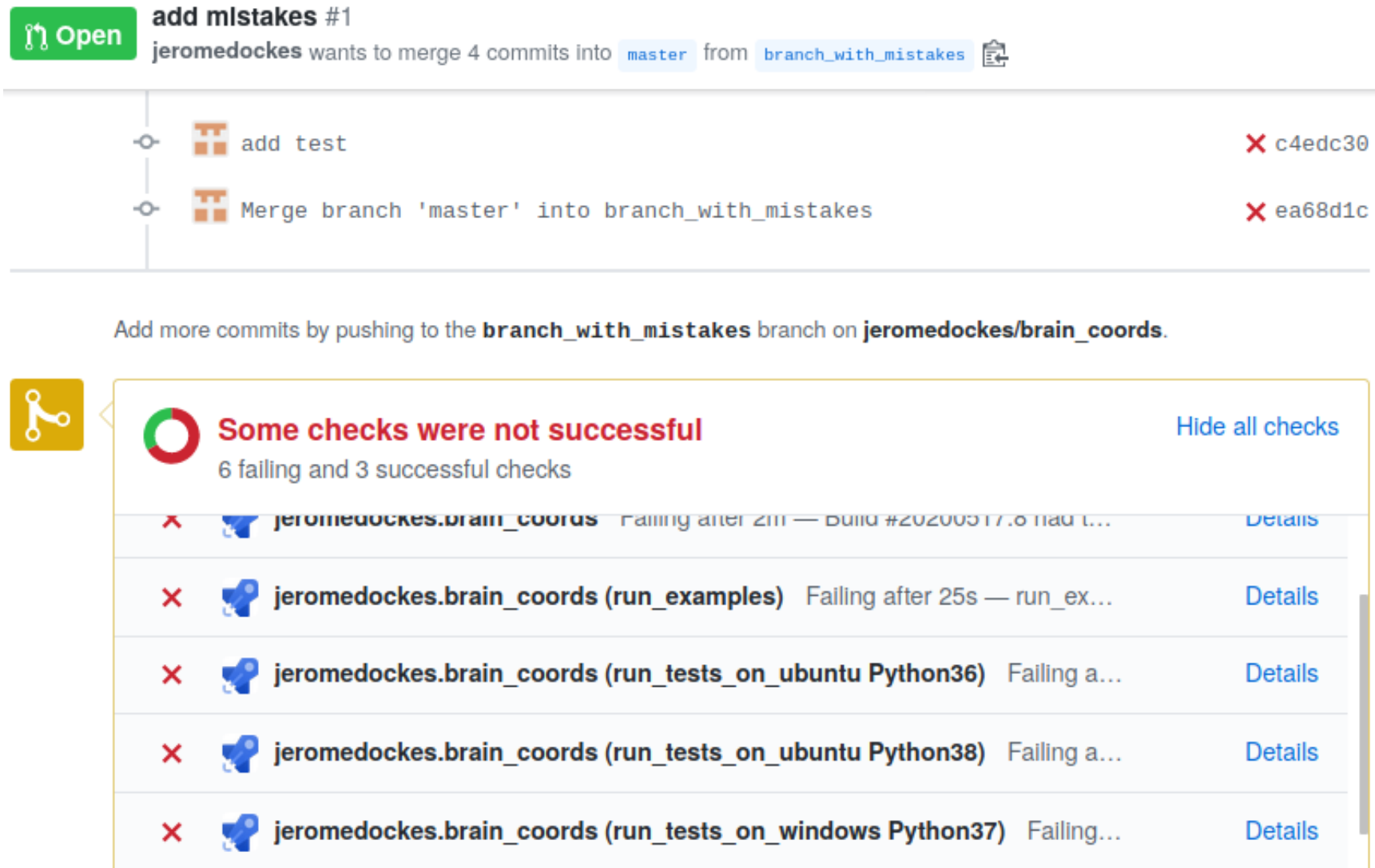
README.md

Azure Pipelines succeeded | codecov 100% | build passing | circleci passing

Transform MNI coordinates to images

`https://dev.azure.com/USERNAME/PROJECT/_apis/build/status/GH_USERNAME.REPONAME?branchName=master`

Seen from GitHub



Test also on windows

azure-pipelines.yml

```
- job: 'run_tests_on_windows'
  pool:
    vmImage: 'windows-latest'
  variables:
    TEST_DIR: '$(Agent.WorkFolder)\\tmp_test_dir'
  strategy:
    matrix:
      Python37:
        python.version: '3.7'
        numpy.spec: 'numpy'
  steps:
    - template: ci/azure_test_steps.yml
```

Storing build artifacts

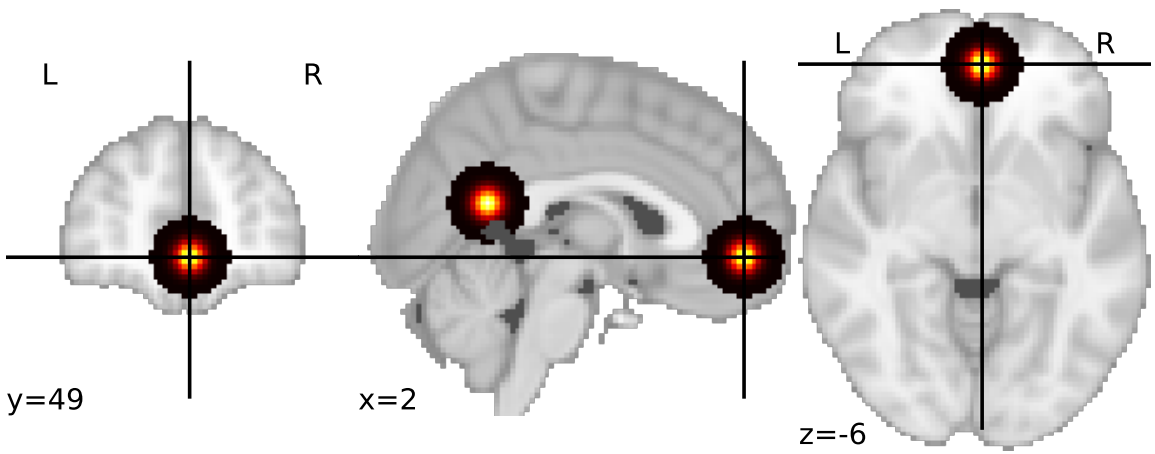
azure-pipelines.yml

```
- job: 'run_examples'
  pool:
    vmImage: 'ubuntu-latest'
  steps:
    - task: UsePythonVersion@0
      inputs:
        versionSpec: '3.8'
    - script: |
        python -m pip install --upgrade pip
        pip install .
      displayName: 'Install package'
    - script: |
        python examples/demo.py
      displayName: 'Run example'
    - publish: dmn_img.png
      artifact: dmn_img
      displayName: 'Publish artifacts'
```

Triggered by jeromedockes

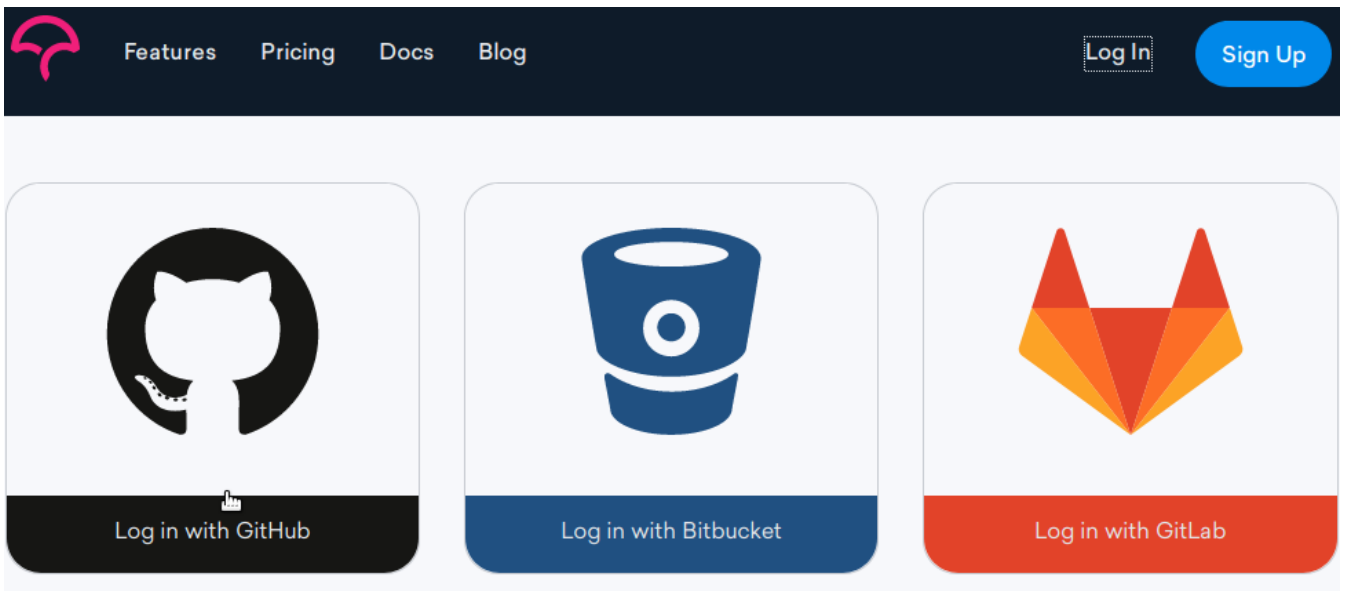
View 6 changes

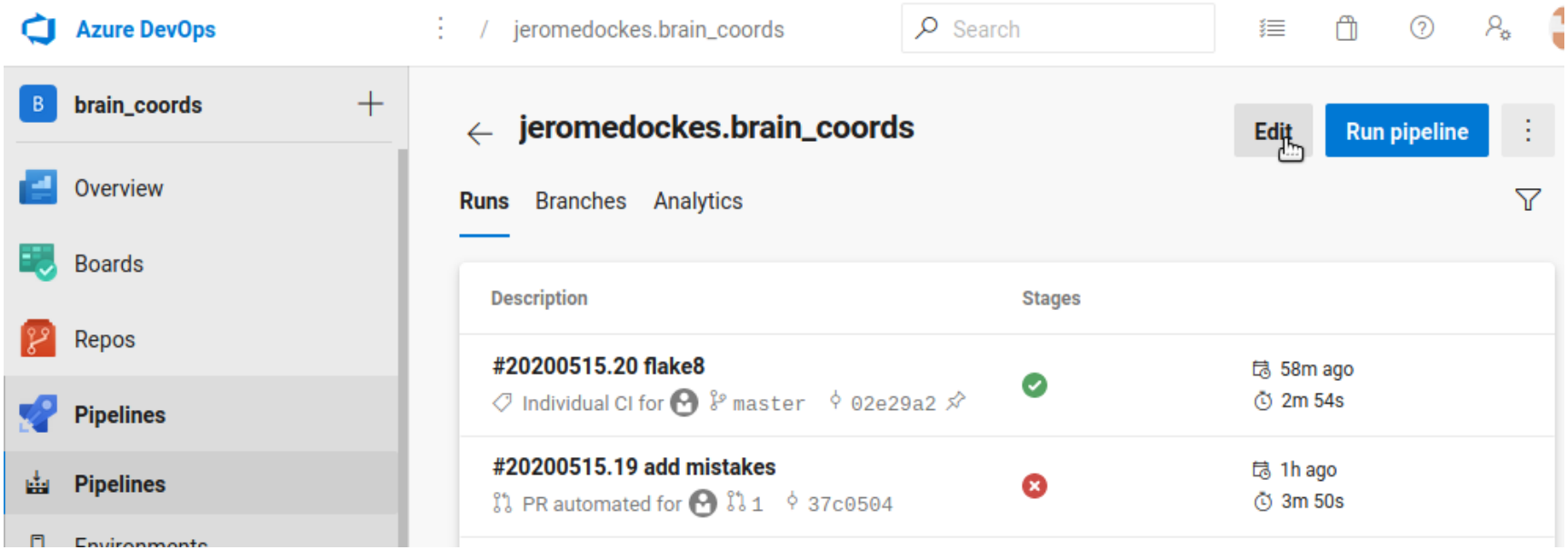
Repository and version	jeromedockes/brain_coords master 08a8684
Time started and elapsed	Today at 11:42 3m 4s
Related	0 work items 2 published
Tests and coverage	100% passed

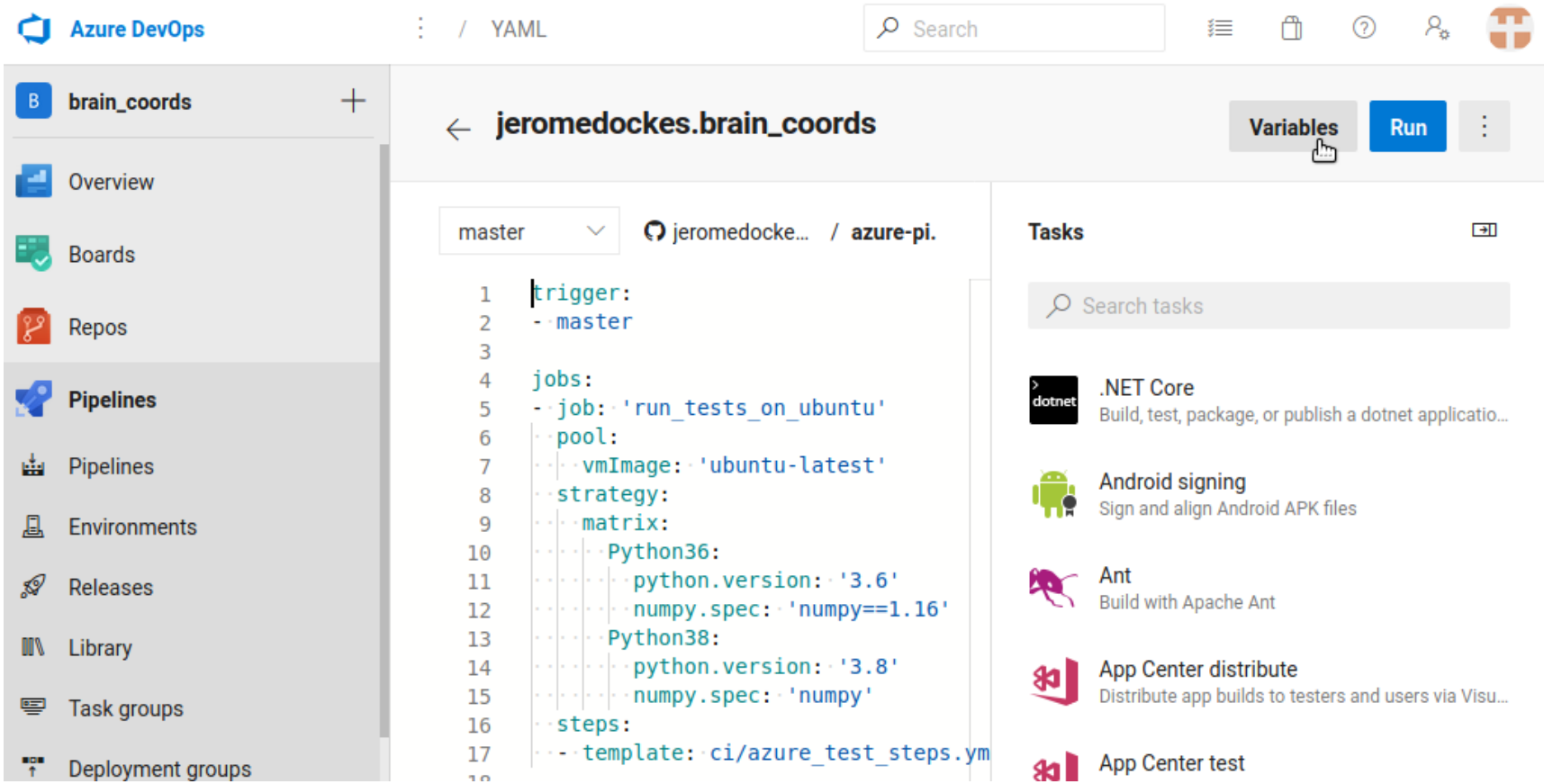


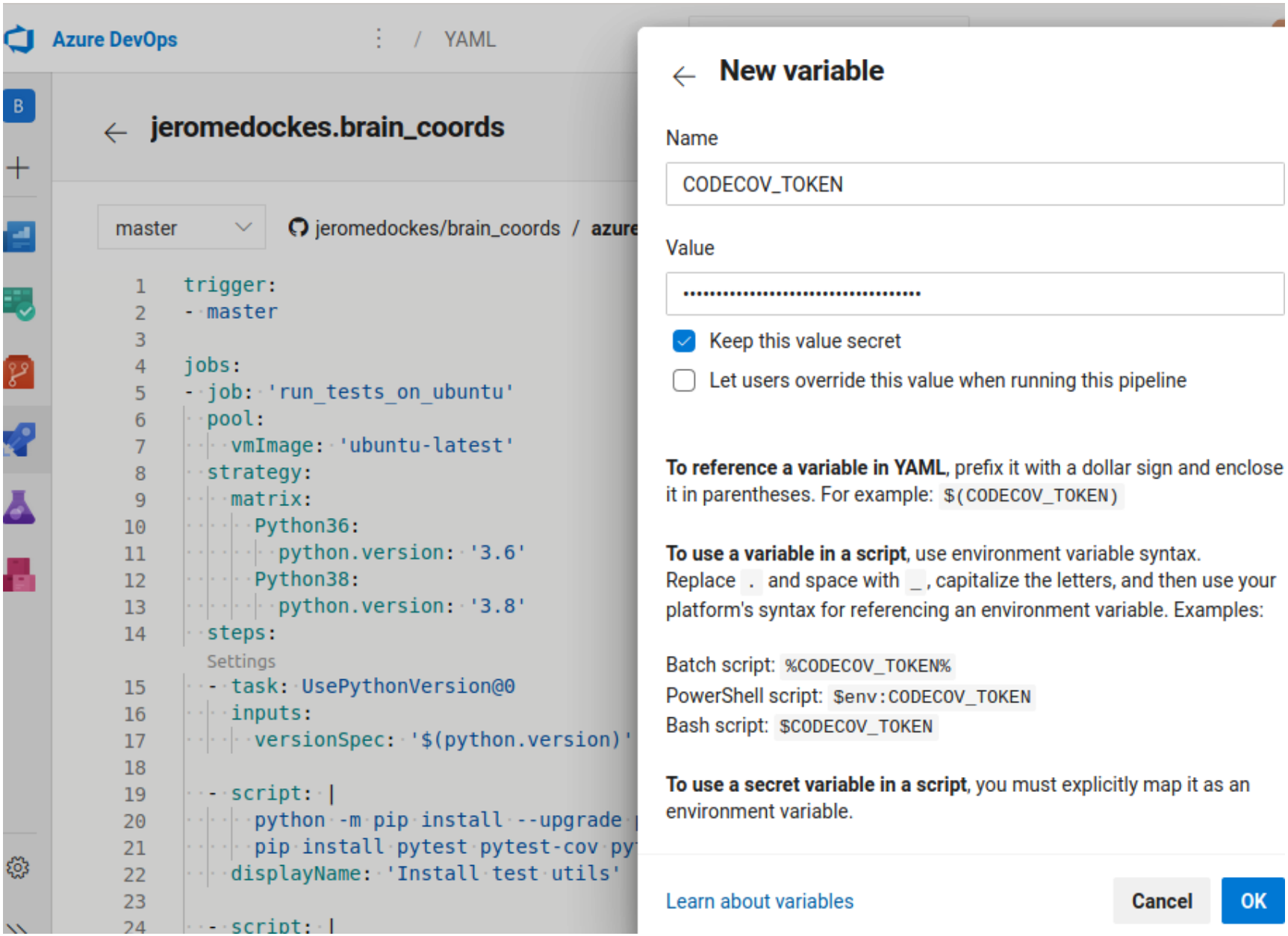
Adding codecov reports

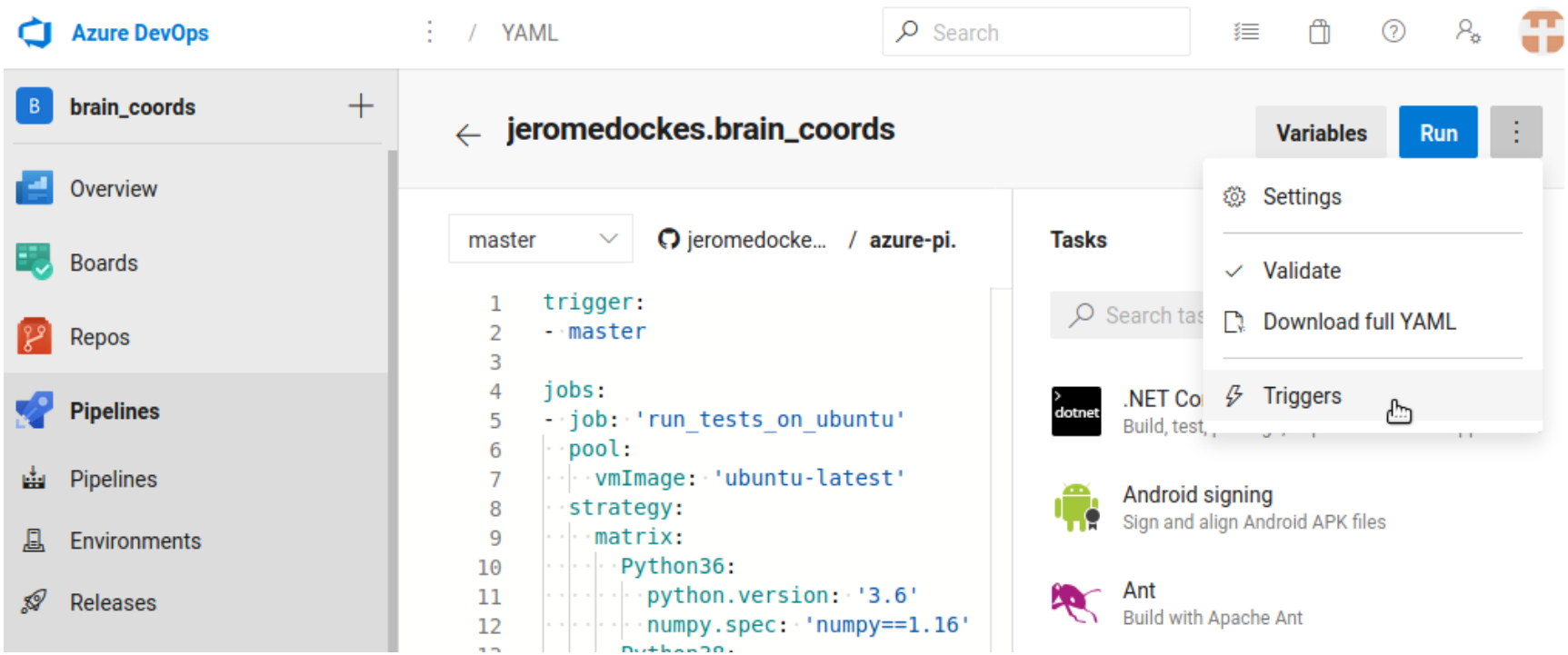
- Get codecov token for this project
  - visit https://codecov.io/gh/USERNAME/REPONAME
  - (signing in with github)
  - copy token
  - add it to pipeline variables
- Upload reports to codecov.io as a pipeline step

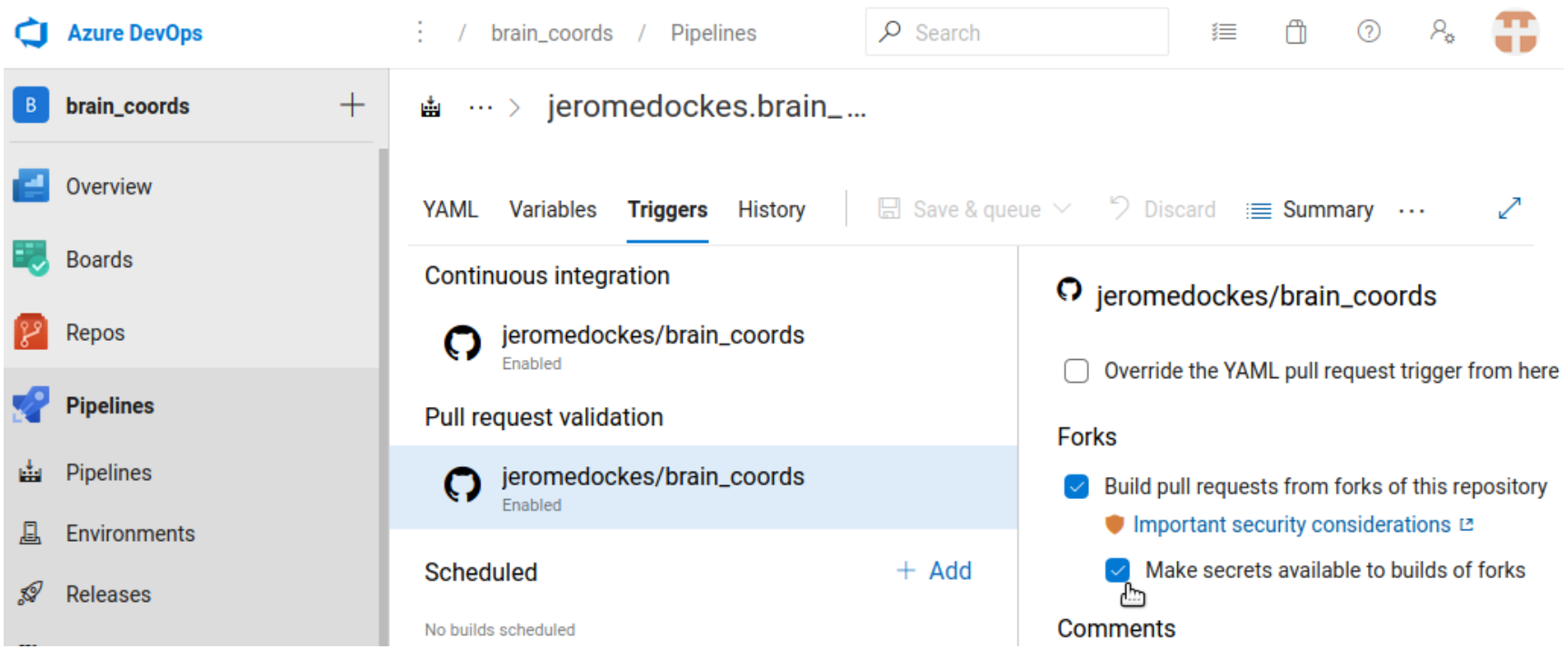



Azure DevOps
jeromedockes.brain_coords
Search
brain_coords
Overview
Boards
Repos
Pipelines
Pipelines
jeromedockes.brain_coords
Edit
Run pipeline
Runs
Branches
Analytics
Description
Stages
#20200515.20 flake8
Individual CI for
master
02e29a2
58m ago
2m 54s
#20200515.19 add mistakes
PR automated for
1
37c0504
1h ago
3m 50s


Azure DevOps
YAML
Search
brain_coords
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Releases
Library
Task groups
Deployment groups
jeromedockes.brain_coords
Variables
Run
master
jeromedocke...
/ azure-pi.
1 trigger:
2 - master
3
4 jobs:
5 - job: 'run_tests_on_ubuntu'
6 pool:
7 vmImage: 'ubuntu-latest'
8 strategy:
9 matrix:
10 Python36:
11 python.version: '3.6'
12 numpy.spec: 'numpy==1.16'
13 Python38:
14 python.version: '3.8'
15 numpy.spec: 'numpy'
16 steps:
17 - template: ci/azure_test_steps.ym
Tasks
Search tasks
.NET Core
Build, test, package, or publish a dotnet applicatio...
Android signing
Sign and align Android APK files
Ant
Build with Apache Ant
App Center distribute
Distribute app builds to testers and users via Visu...
App Center test


Azure DevOps
YAML
jeromedockes.brain_coords
master
jeromedockes/brain_coords / azure
1 trigger:
2 - master
3
4 jobs:
5 - job: 'run_tests_on_ubuntu'
6 pool:
7 vmImage: 'ubuntu-latest'
8 strategy:
9 matrix:
10 Python36:
11 python.version: '3.6'
12 Python38:
13 python.version: '3.8'
14 steps:
Settings
15 - task: UsePythonVersion@0
16 inputs:
17 versionSpec: '$(python.version)'
18
19 - script: |
20 python -m pip install --upgrade
21 pip install pytest pytest-cov
22 displayName: 'Install test utils'
23
24 - script: |
New variable
Name
CODECOV_TOKEN
Value
Keep this value secret
Let users override this value when running this pipeline
To reference a variable in YAML, prefix it with a dollar sign and enclose it in parentheses. For example: $(CODECOV_TOKEN)
To use a variable in a script, use environment variable syntax. Replace . and space with _, capitalize the letters, and then use your platform's syntax for referencing an environment variable. Examples:
Batch script: %CODECOV_TOKEN%
PowerShell script: $env:CODECOV_TOKEN
Bash script: $CODECOV_TOKEN
To use a secret variable in a script, you must explicitly map it as an environment variable.
Learn about variables
Cancel
OK


Azure DevOps
YAML
Search
brain_coords
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Releases
jeromedockes.brain_coords
Variables
Run
Settings
Validate
Download full YAML
Triggers
master
jeromedocke...
/ azure-pi.
Tasks
Search tas
.NET Co
Build, test,
Android signing
Sign and align Android APK files
Ant
Build with Apache Ant
1 trigger:
2 - master
3
4 jobs:
5 - job: 'run_tests_on_ubuntu'
6 pool:
7 vmImage: 'ubuntu-latest'
8 strategy:
9 matrix:
10 Python36:
11 python.version: '3.6'
12 numpy.spec: 'numpy==1.16'


Azure DevOps
brain_coords / Pipelines
Search
brain_coords
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Releases
jeromedockes.brain_...
YAML
Variables
Triggers
History
Save & queue
Discard
Summary
Continuous integration
jeromedockes/brain_coords
Enabled
Pull request validation
jeromedockes/brain_coords
Enabled
Scheduled
+ Add
No builds scheduled
jeromedockes/brain_coords
Override the YAML pull request trigger from here
Forks
Build pull requests from forks of this repository
Important security considerations
Make secrets available to builds of forks
Comments

Upload to codecov

ci/azure_test_steps.yml

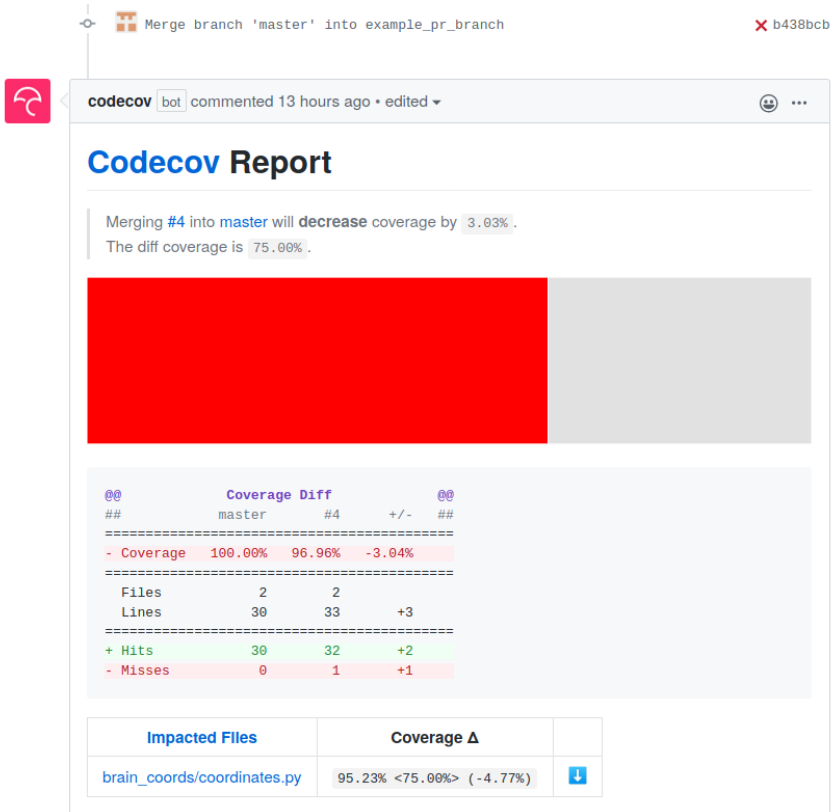
```
- script: |
    python -m pip install --upgrade pip
    pip install pytest pytest-cov pytest-azurepipelines codecov
    pip install $(numpy.spec)
    pip install .
  displayName: 'Install'
- bash: >
    mkdir -p "$TEST_DIR" && cd "$TEST_DIR" &&
    python -m pytest --pyargs brain_coords
    --cov=brain_coords --cov-report=xml --cov-append
    -o junit_family=xunit2 --junitxml=test-results.xml
  displayName: 'Run tests'
- bash: |
    cp "$TEST_DIR"/{.coverage,coverage.xml} .
    bash <(curl -s https://codecov.io/bash) -t "$CODECOV_TOKEN"
  displayName: 'codecov upload'
  env:
    CODECOV_TOKEN: $(CODECOV_TOKEN)
```

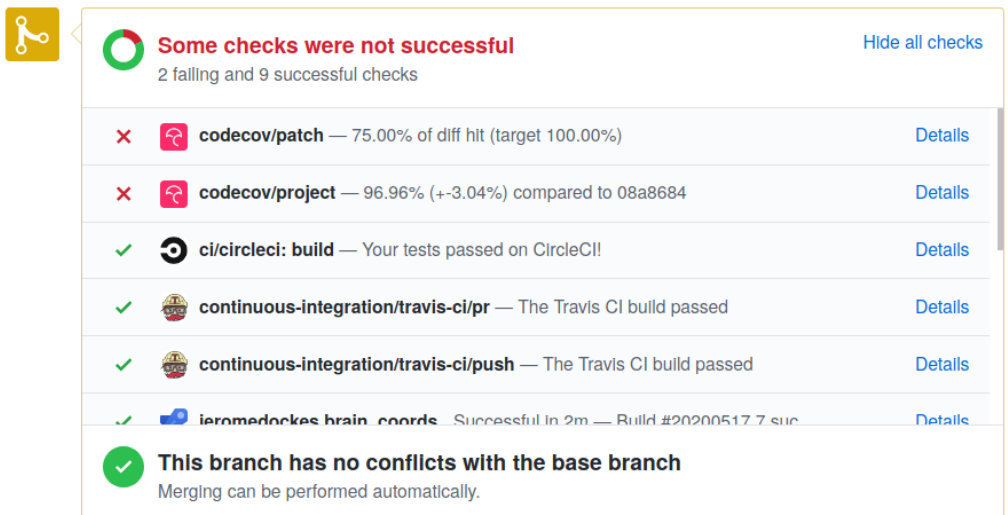
Optional: add codecov.yml

```
coverage:
  status:
    project:
      default:
        target: auto
        threshold: 1%
    patch:
      default:
        target: auto
        threshold: 1%
```

```
brain_coords/
├── brain_coords/
├── ci/
│   └── azure_test_steps.yml
├── examples/
├── azure-pipelines.yml
├── codecov.yml
└── setup.py
```

- install codecov GitHub APP: https://github.com/marketplace/codecov





Summary

- repo on GitHub
- create Azure pipelines account and project
- add `azure-pipelines.yml`
- install codecov GitHub app
- that's it!

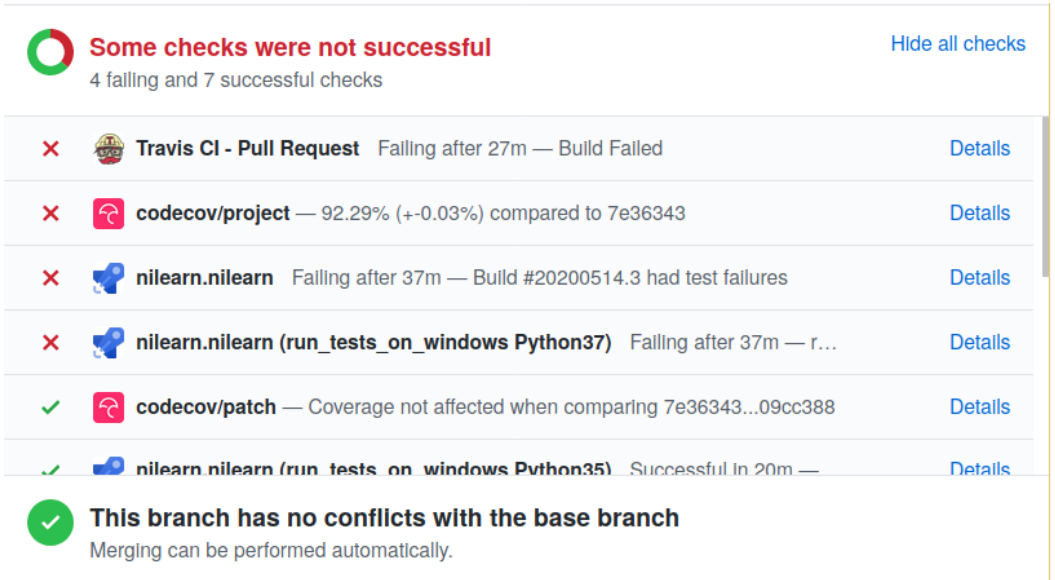
- very similar with other CI platforms
- See `brain_coord` repo for travis, circleci(, tox)
- Read the CI configuration of your favorite projects!
- Use tox

- Complementary to code reviews

Nilearn’s CI infrastructure

- Travis: unit tests, doctests and flake8 on Linux
- circleci: building documentation and example gallery
- Azure pipelines: unit tests on Windows
- codecov: assembles and displays coverage reports

```
nilearn
├── .travis.yml
├── setup.py*
├── azure-pipelines.yml
├── codecov.yml
├── .circleci/
│   └── config.yml
└── nilearn/
    └── __init__.py
```



Building doc: dataset download caching

- after build on circleci, `nilearn_data`, `miniconda3` and `doc` are saved
- `full-build` ignores the cache. runs:
  - when pushing to master
  - once per day
- `quick-build` runs for PRs.
  - `nilearn_data` and `miniconda3` always restored
  - `doc` restored if:
    - cache was written today
    - `.circleci/docs-cache-timestamp` has not changed

TL;DR if you want to force running sphinx:

```
date -Is > .circleci/docs-cache-timestamp
```

A few tips

- Before pushing:
  - test new code:
    `pytest --pyargs nilearn.some.test.module`
  - test code and doctests: `make test`
  - run new examples
  - `make doc` to build doc without running examples
- CI failures are usually genuine
- But not always: failing downloads, dependency releases
- Answer is usually in the logs
- Download log and search for `"FAIL"`
- Ask for help!