# DaViL: Developer Guide

## Introduction

This document shows how to install the Data Visualization Tool and how to collaborate on it using git and BitBucket. It has been tested on Windows 7, 10 and Ubuntu 16.04LTS (64 bit).

### Requirements

* [git](https://git-scm.com/)
* [Python 2.7 (Anaconda)](https://www.continuum.io/downloads)
* [pip](https://pypi.python.org/pypi/pip)
* [BitBucket Account](https://bitbucket.org/product)
* Access to the repository. You will need to ask for an invite to one of the admins of the project:
* Gabriel Chica Fernandez (gabrielchicafernandez@gmail.com)
* Alberto Sanchez Campos ([alberto.sanchez@urjc.es](mailto:alberto.sanchez@urjc.es))
* Manuel Rubio Sanchez ([manuel.rubio@urjc.es](mailto:manuel.rubio@urjc.es))

Make sure that you have both *python* and *conda* commands available. Open a new terminal and type:

python -v

You should see something like: “...*Python* ***2.7.13*** *|Anaconda 4.4.0 (****64-bit****)...”*

conda

You should see conda’s options

On Windows, if any of these commands was not available, make sure it’s been added to the system’s path variable. By default:

* python: *D:\Users\<YourUserName>\Anaconda2*
* conda: *D:\Users\<YourUserName>\Anaconda2\Scripts*

### Set up

Clone the repository

git clone https://**YourUserName**@bitbucket.org/urjc\_data\_visualization/davil.git

In order to install the libraries, on the root folder, execute:

pip install –r requirements.txt

conda install -c bokeh nodejs

Test from the root folder:

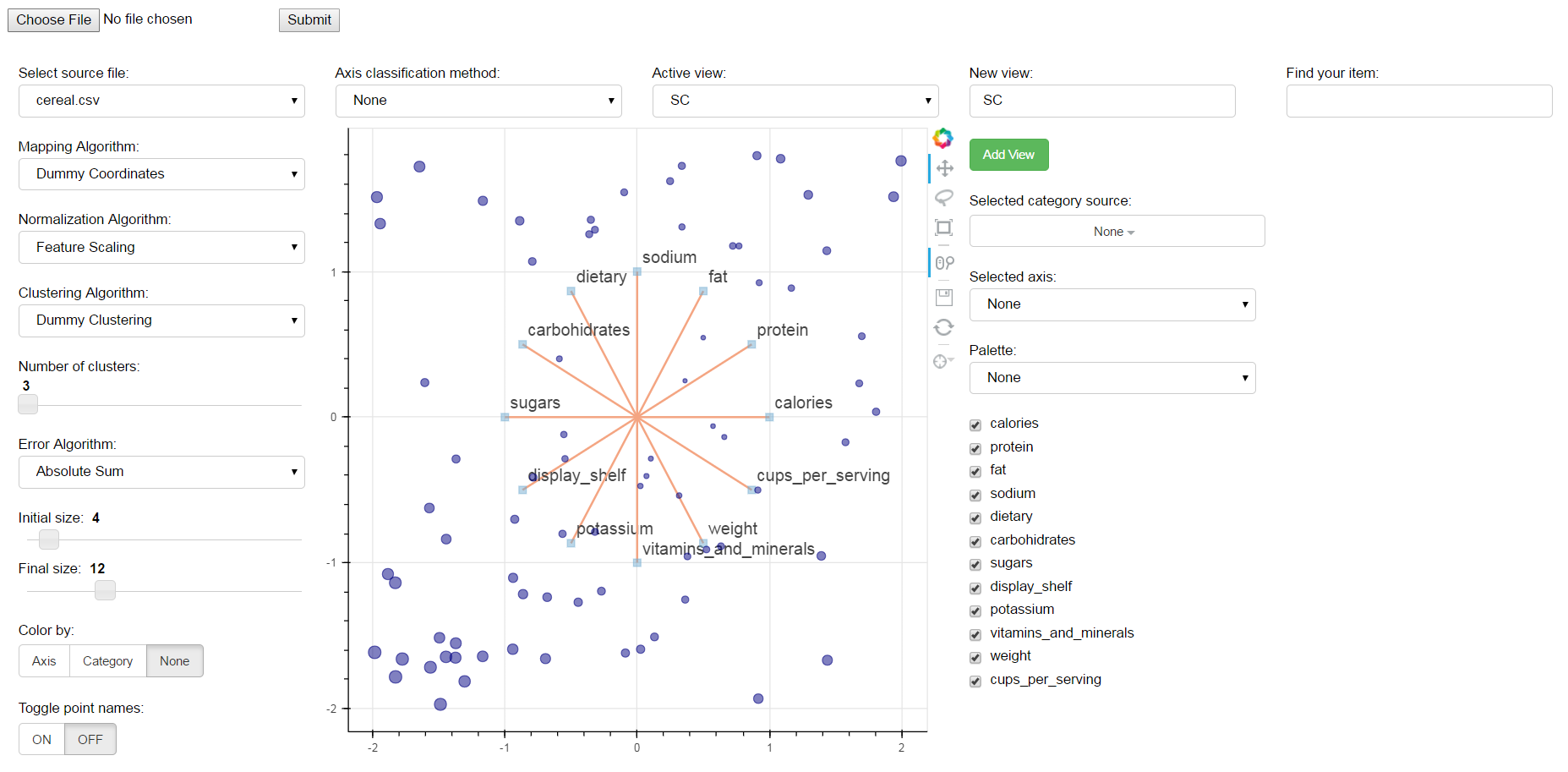
UNIX:

./runserver.sh

Windows:

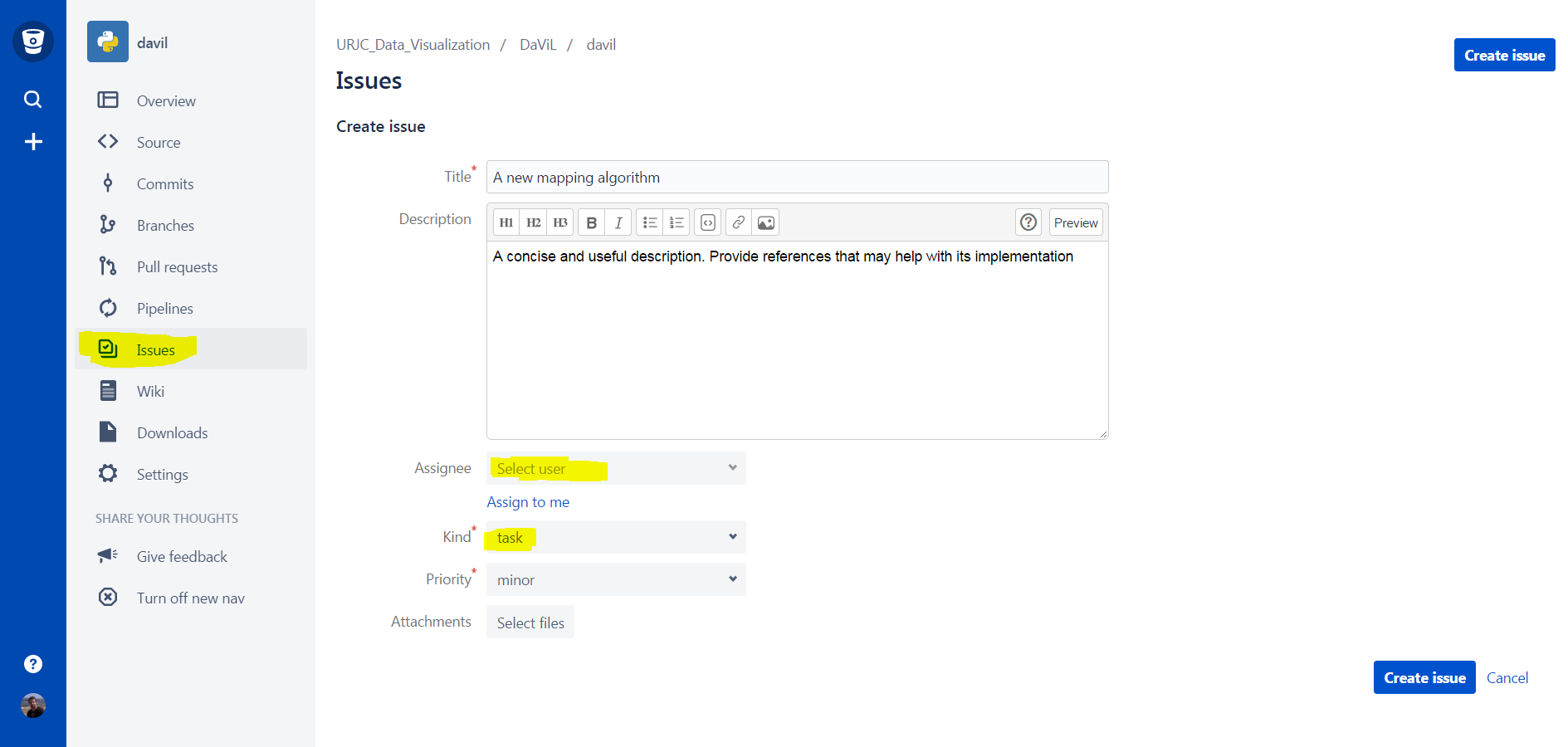
runserver.bat

Should open the browser and display the app:



### Develop

First, create a new task



It will generate a task number.

Then, branch off from the develop branch as follows:

git checkout -b **YourUserName**/<ISSUE\_TYPE><ISSUE\_NUMBER> origin/develop

ISSUE\_TYPE:

* Defect (bug): **DE**
* Task/Feature: **US**

Make your changes, create unit testing under mymodule/tests to validate your changes and then:

git fetch

git commit –m “<ISSUE\_TYPE><ISSUE\_NUMBER> - Meaningful description”

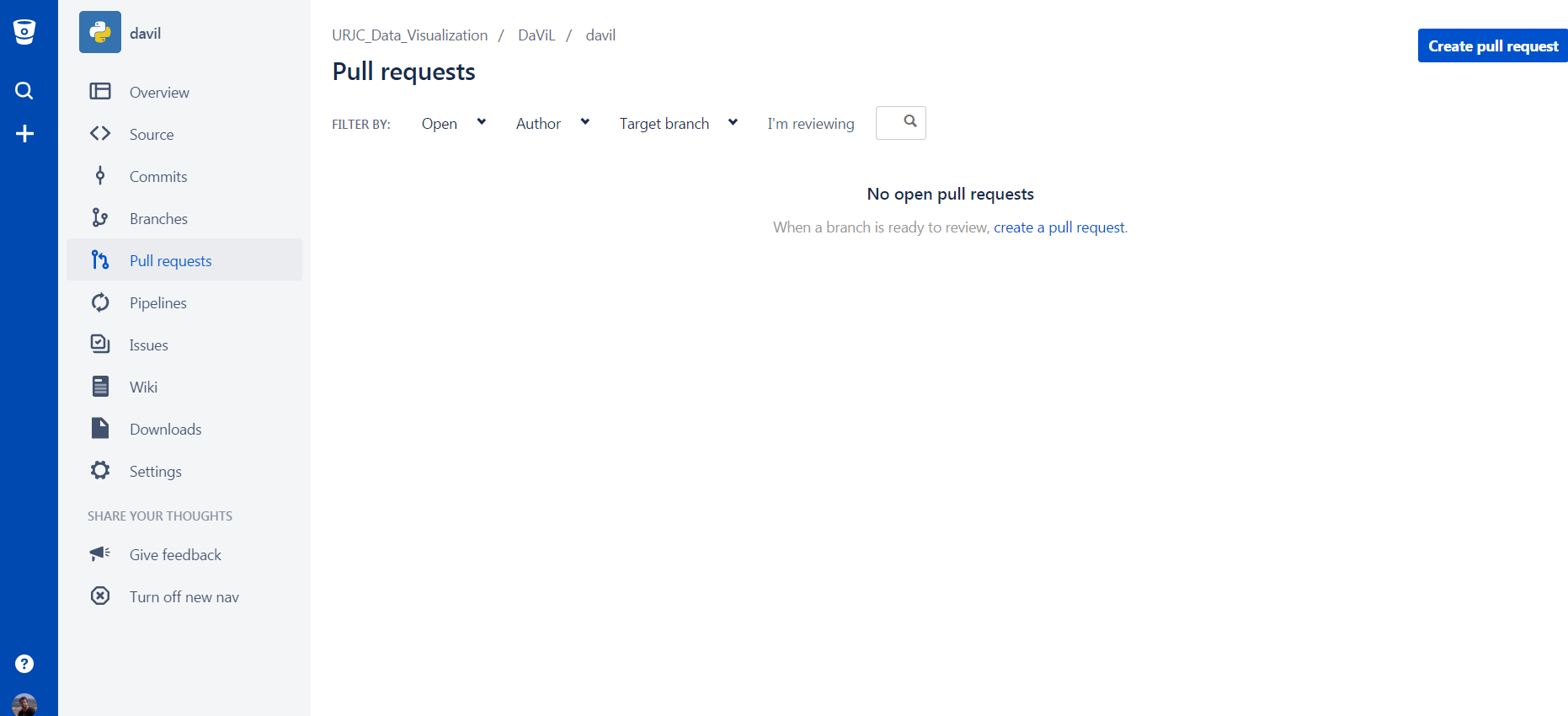
git rebase --onto origin/develop

git push

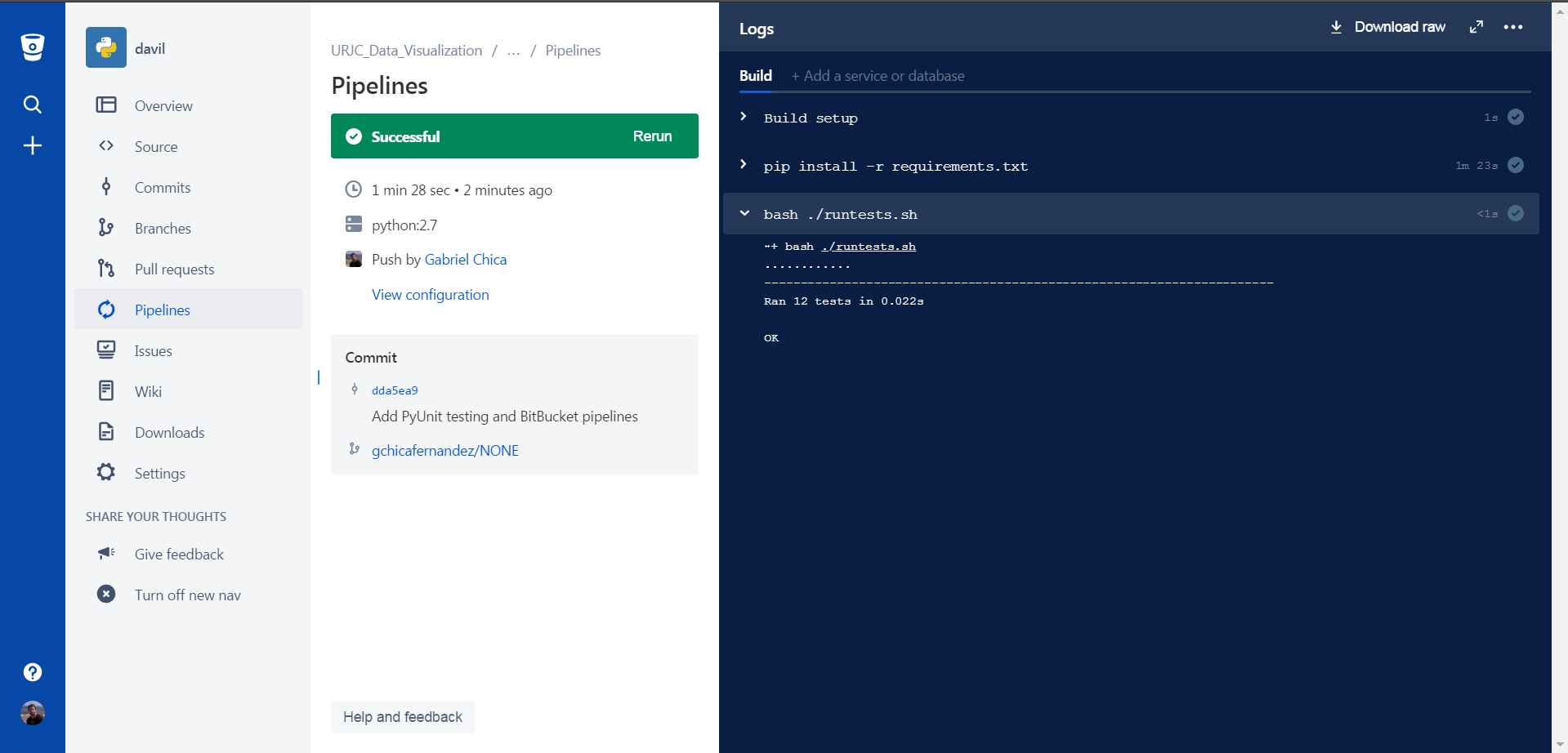
Keep your changes in **one commit**. If you need to modify something, use

git commit –amend

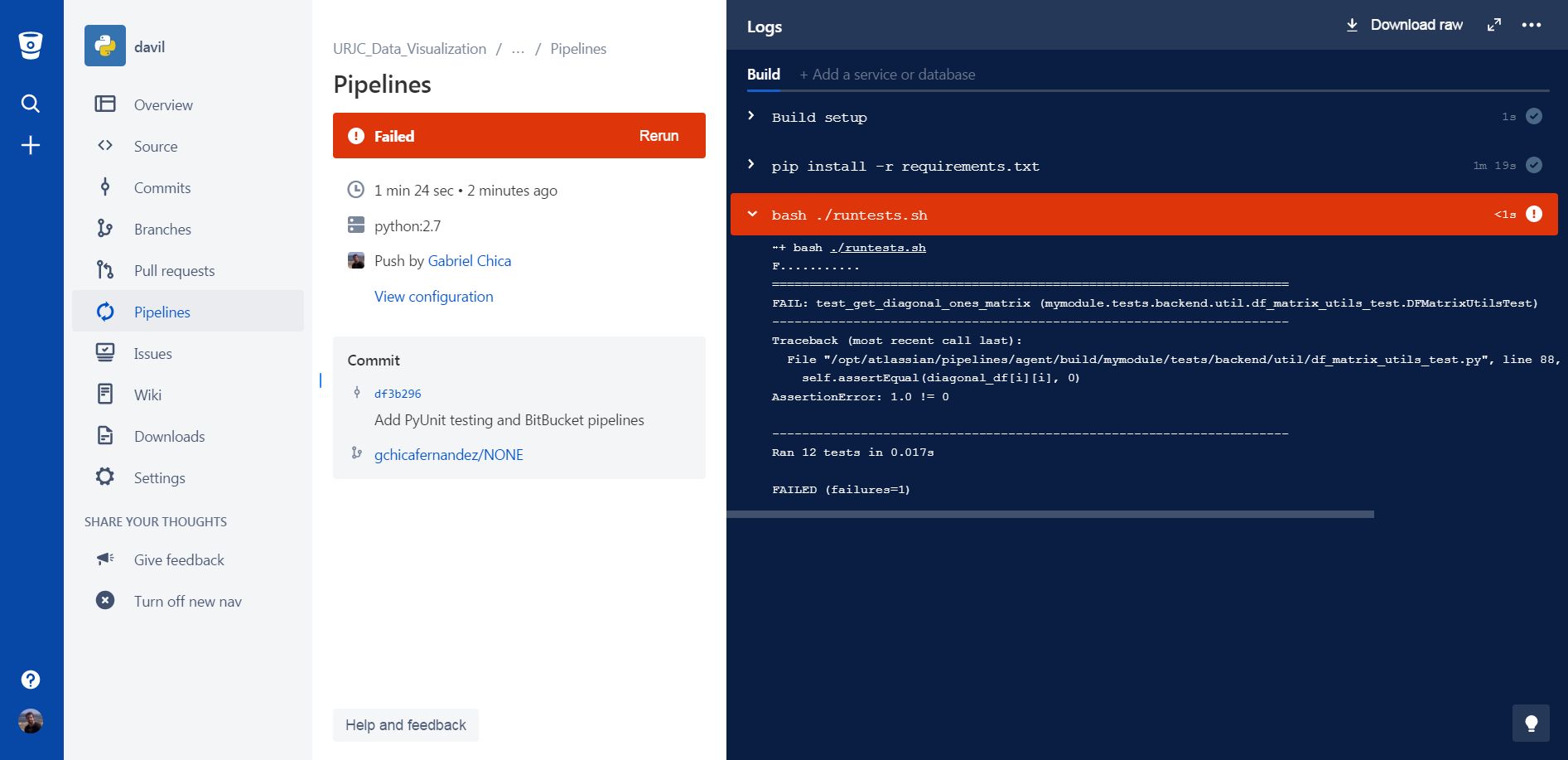
Then, open a PR **to develop** and assign at least two people to review it.



For the PR to be approved, it must have **at least one approval** and **a green build status**.



The build will show up green if all dependencies can be successfully installed and all tests are passed. If any dependency or test is not passed, it will explain why:



**Never** merge code with a failing build since it will cause all subsequent builds to fail.

If some tests are failing, you can debug it by making modifications and running locally the *runtests* script. **Do not push constantly for the build to run again, since the pipeline time is limited monthly to 50 minutes.**

