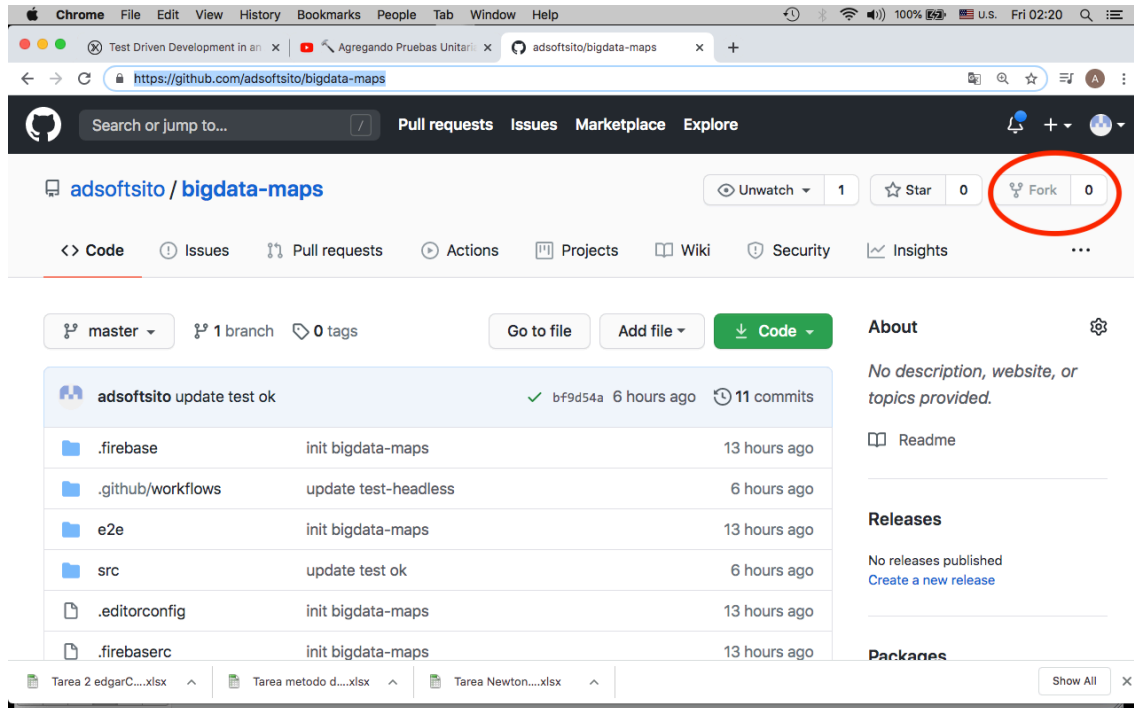


Test Driven Development & Continuous Integration

1.- Go to <https://github.com>

2.- Fork the repository: <https://github.com/adsoftsito/bigdata-maps>



3.- Clone you own repository in wokstation and Change to repo directory

```
$ git clone https://github.com/<your-user>/project-demo.git
```

```
$ cd bigdata-maps
```

4.- Check url repository, install, serve, test and build.

```
$ pwd
```

```
$ git remote -v
```

```
origin  https://github.com/<your-user>/bigdata-maps.git (fetch)
```

```
origin  https://github.com/<your-user>/bigdata-maps.git (push)
```

```
$ npm install
```

```
$ ng serve
```

```
$ ng test code--coverage
```

```
$ npm test-headless
```

```
$ ng build
```

4.- Edit README.md, add # v0.1 tag

```
$ vi README.md
```

BigdataMaps

```
# v0.1 tag
```

```
...  
...  
:wq
```

5.- Commit, push changes

```
$ git add .  
$ git commit -m "update README maps with v0.1 tag"  
$ git push -u origin master
```

6.- Tag base code with v0.1

```
$ git tag -a v0.1 -m "my maps version 0.1"  
$ git tag  
$ git show v0.1
```

7.- Publish tag

```
$ git push origin v0.1
```

8.- create branch hotfixes, release and develop

```
$ git checkout master  
$ git branch hotfixes  
$ git checkout hotfixes
```

```
$ vi HOTFIXES.md  
# hotfixes maps here  
:wq
```

```
$ git add .  
$ git commit -m "init hotfixes branch"  
$ git push -u origin hotfixes
```

```
#####
```

```
$ git checkout master  
$ git branch release  
$ git checkout release
```

```
$ vi RELEASE.md  
# release maps here  
:wq
```

```
$ git add .  
$ git commit -a -m "init release branch"  
$ git push -u origin release
```

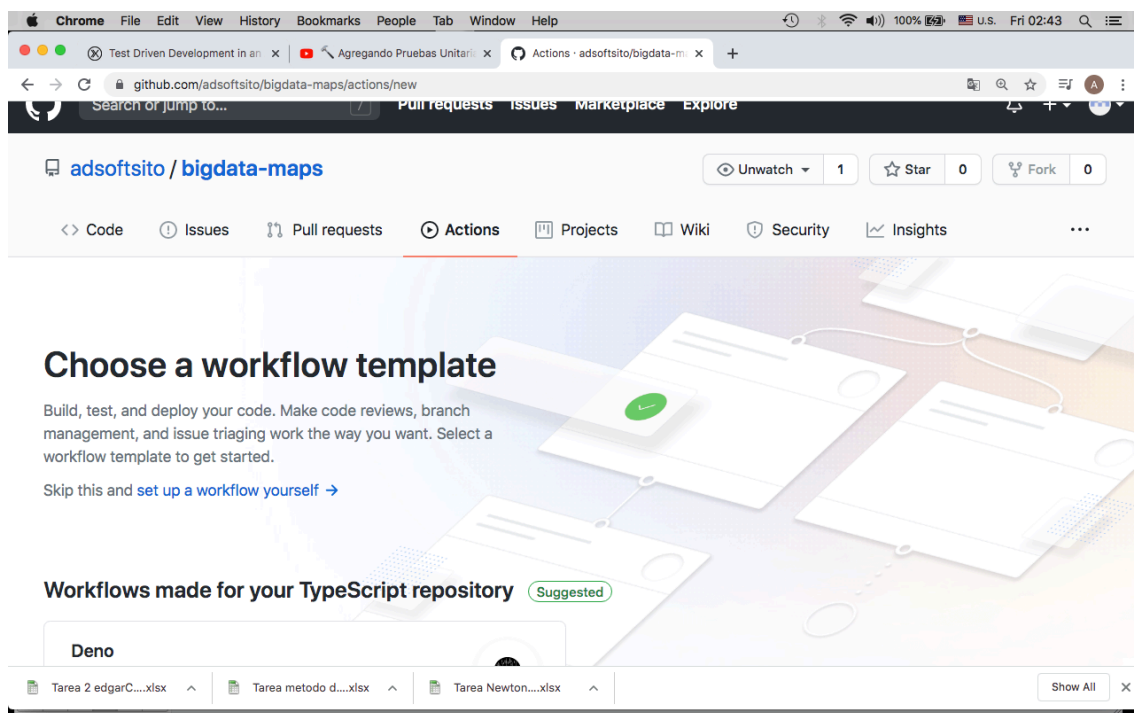
```
#####
```

```
$ git checkout master
$ git branch develop
$ git checkout develop
$ vi DEVELOP.md
# develop maps here
:wq

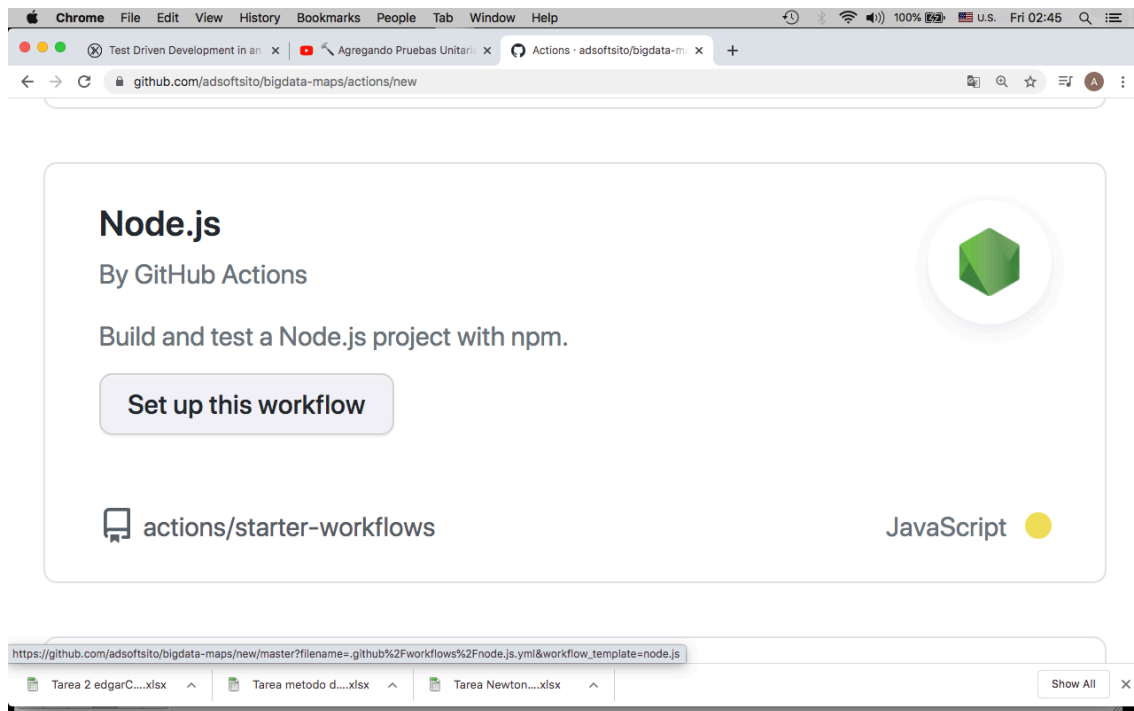
$ git add .

$ git commit -a -m "init develop branch"
$ git push -u origin develop
```

9.- Configure CI in Github Actions



Select nodejs template



Edit nodejs template :

name:

Node.js

CI

on:

push:

branches: [master, hotfixes, release, develop]

pull_request:

branches: [master, hotfixes, release, develop]

jobs:

build:

runs-on: ubuntu-latest

strategy:

matrix:

node-version: [10.x]

steps:

- uses: actions/checkout@v2

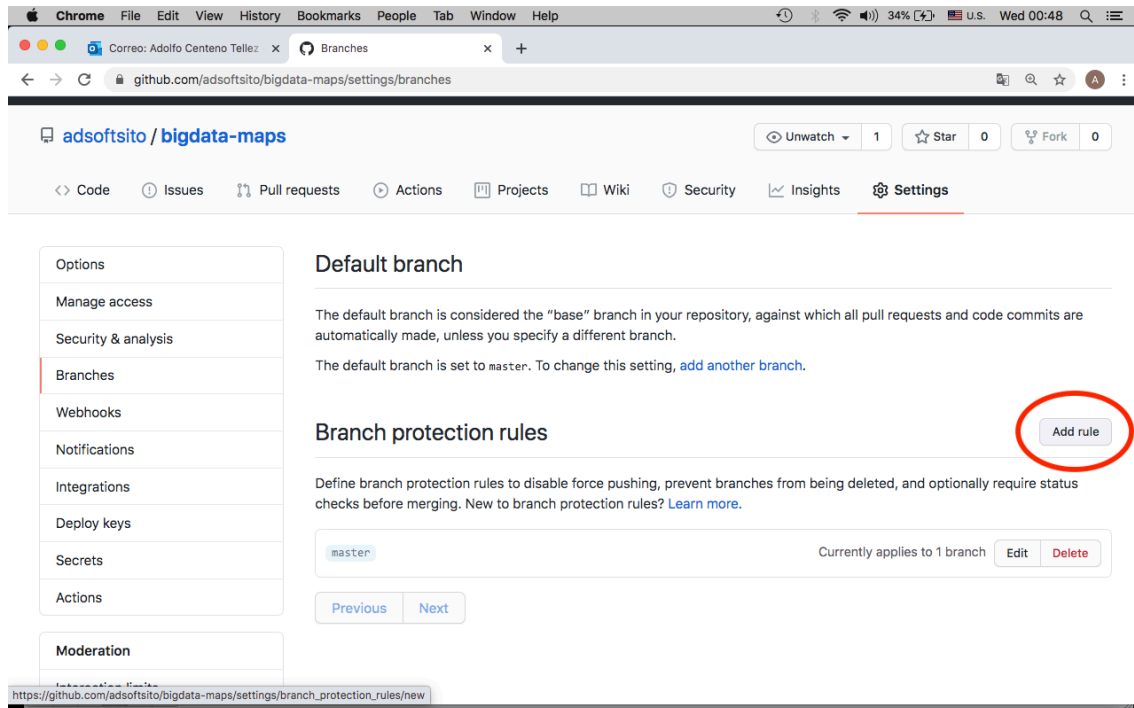
- name: Use Node.js \${ matrix.node-version }

uses: actions/setup-node@v1

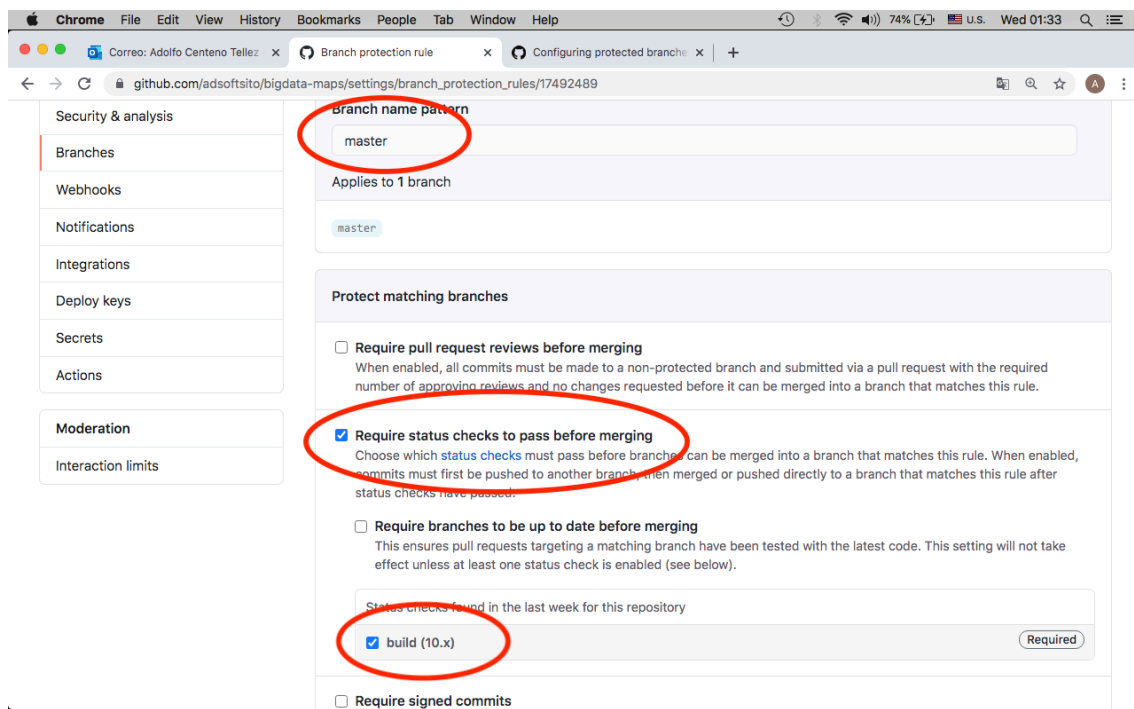
with:

```
node-version: ${matrix.node-version}}
- run: npm ci
- run: npm run test-headless
- run: npm run build --if-present
```

10.- Configure branches protection



- Create master rule



- Repeat for hotfixes, release and develop branch

