Nombre	Código
Esteban Durán	12103025

Jenkins

Instalacion de dependencias requeridas para usar jenkins

```
# yum install java-1.7.0-openjdk
# yum install wget -y
# yum install git -y
# sudo wget -O /etc/yum.repos.d/jenkins.repo http://pkg.jenkins-ci.org/redhat/jenkins.repo
# sudo rpm --import https://jenkins-ci.org/redhat/jenkins-ci.org.key
# sudo yum install jenkins
```

Iniciar el servicio de Jenkins y habilitar su respectivo puerto

```
# chkconfig jenkins on
# service jenkins start
# iptables -I INPUT 5 -p tcp -m state --state NEW -m tcp --dport 8080 -j ACCEPT
# service iptables save
```

Se crea el usuario y ambiente virtual para Jenkins

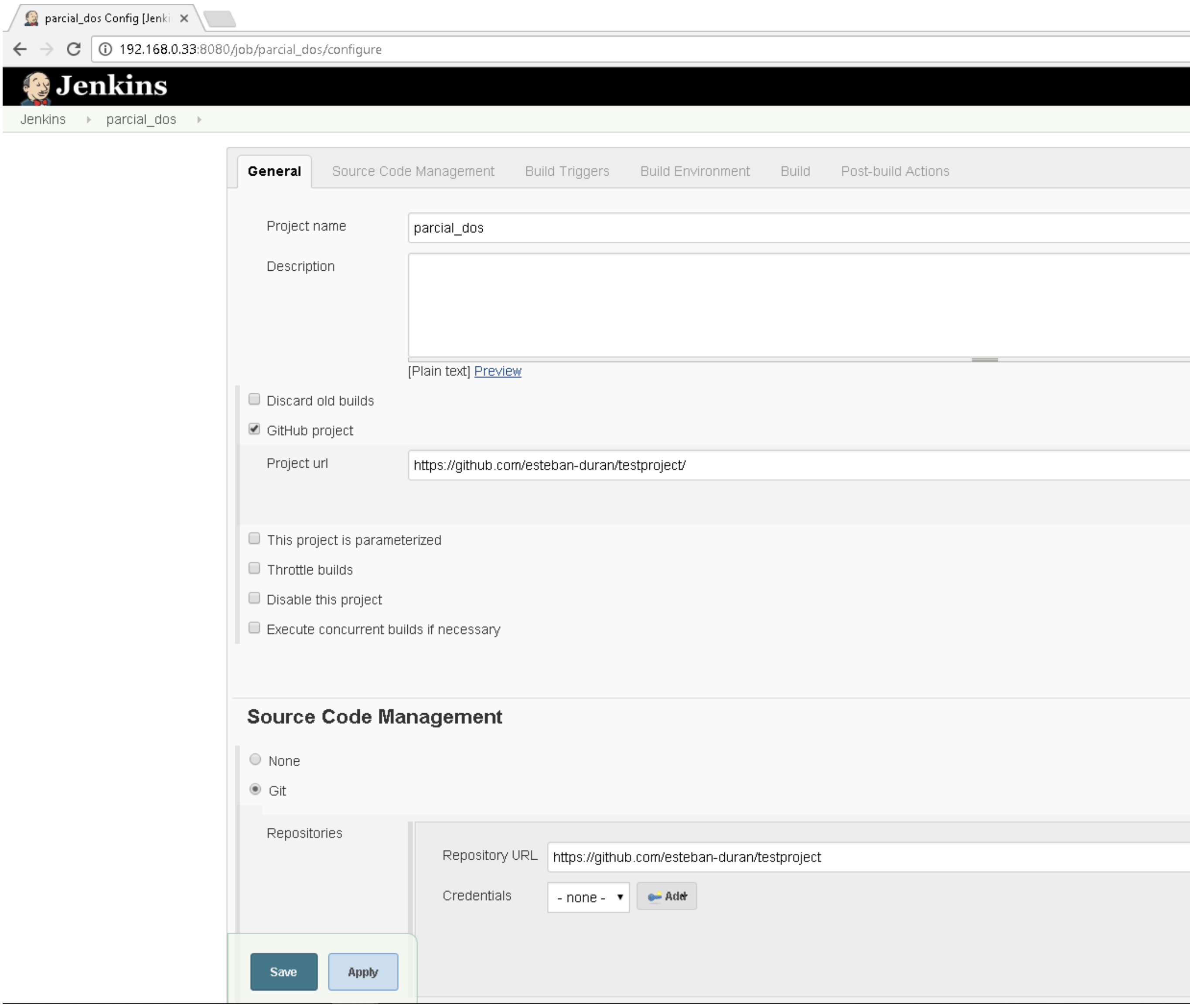
```
# useradd jenkins
# passwd jenkins
# su jenkins
$ mkdir /var/lib/jenkins/.virtualenvs
$ cd /var/lib/jenkins/.virtualenvs
$ virtualenv testproject
$ . testproject/bin/activate
```

Se instalan las herramientas requeridas para trabajar con pruebas en jenkins

```
$ pip install xmlrunner
$ pip install unittest2
$ pip install pytest
$ pip install pytest-cov
$ pip install flask

$ pip freeze > requirements.txt
```

Configuracion de jenkins con github



Project name: parcial_dos

Description:

[Plain text] Preview

☐ Discard old builds

☑ GitHub project

Project url: https://github.com/esteban-duran/testproject/

☐ This project is parameterized

☐ Throttle builds

☐ Disable this project

☐ Execute concurrent builds if necessary

Source Code Management

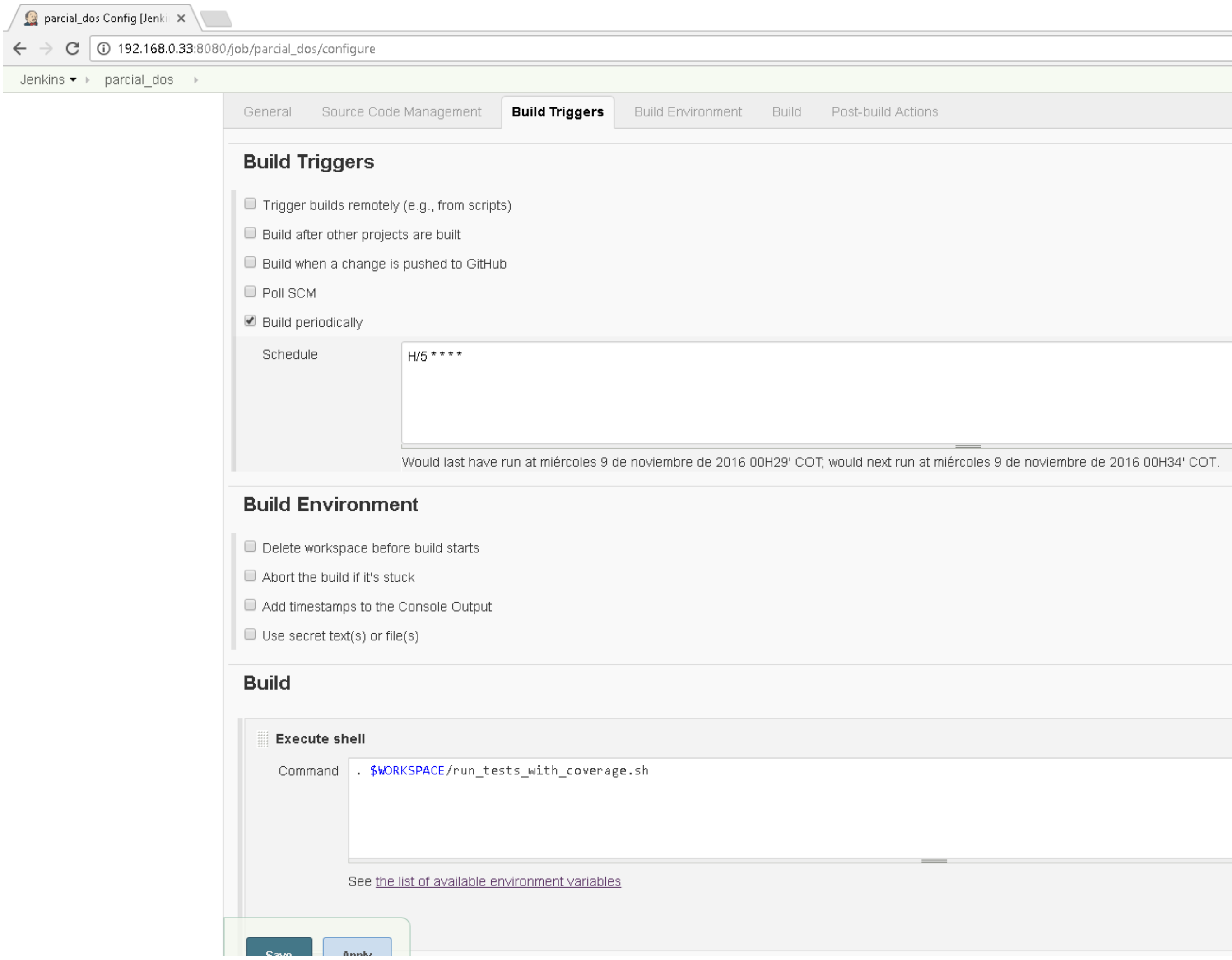
○ None

● Git

Repositories

Repository URL: https://github.com/esteban-duran/testproject

Credentials: - none - ▾ Add



Build Triggers

- □ Trigger builds remotely (e.g., from scripts)
- □ Build after other projects are built
- □ Build when a change is pushed to GitHub
- □ Poll SCM
- ☑ Build periodically

Schedule

```
H/5 * * * *
```

Would last have run at miércoles 9 de noviembre de 2016 00H29' COT; would next run at miércoles 9 de noviembre de 2016 00H34' COT.

Build Environment

- □ Delete workspace before build starts
- □ Abort the build if it's stuck
- □ Add timestamps to the Console Output
- □ Use secret text(s) or file(s)

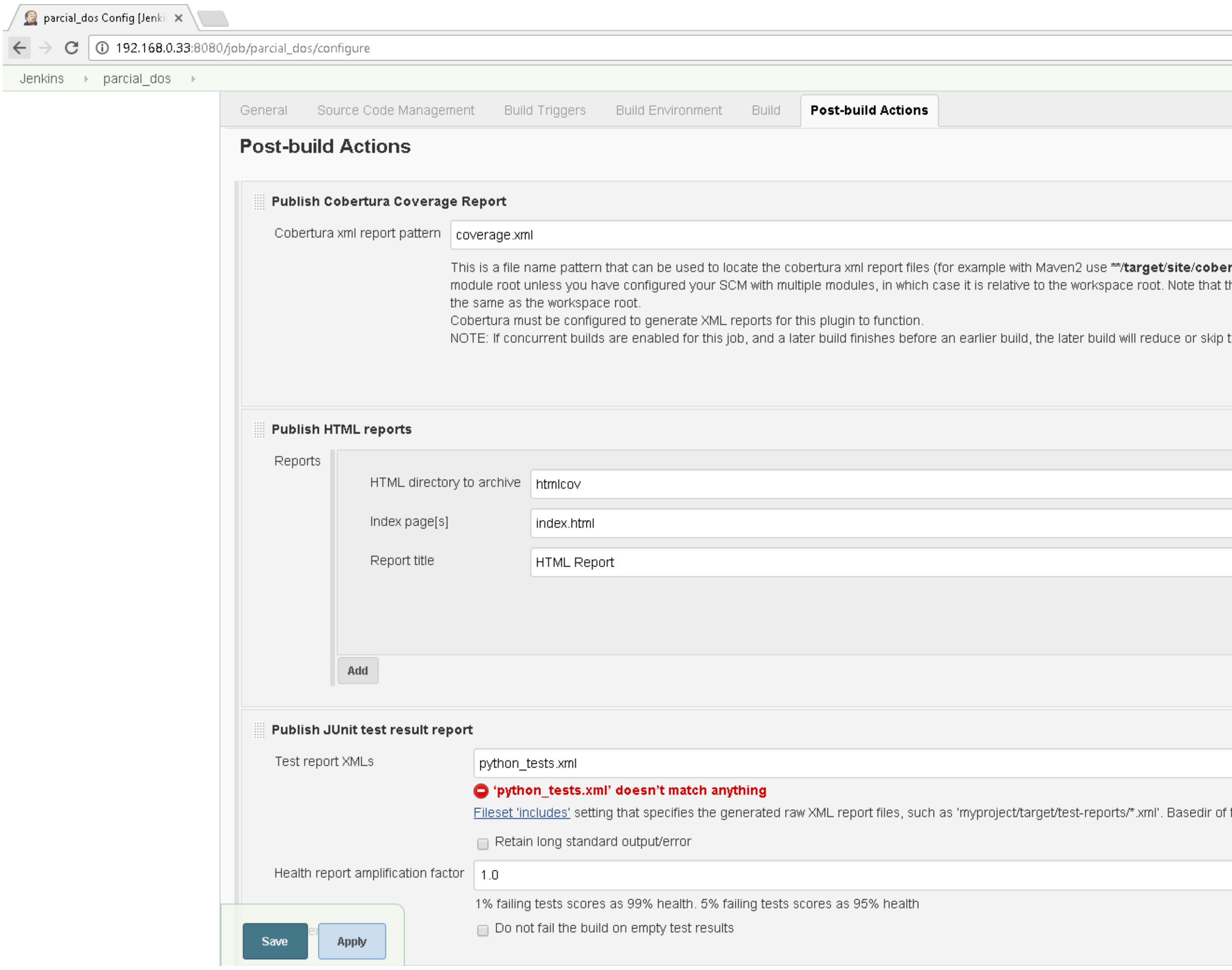
Build

Execute shell

Command

```
. $WORKSPACE/run_tests_with_coverage.sh
```

See the list of available environment variables



Post-build Actions

Publish Cobertura Coverage Report

Cobertura xml report pattern: coverage.xml

This is a file name pattern that can be used to locate the cobertura xml report files (for example with Maven2 use **/**target/site/cober** module root unless you have configured your SCM with multiple modules, in which case it is relative to the workspace root. Note that th the same as the workspace root.
Cobertura must be configured to generate XML reports for this plugin to function.
NOTE: If concurrent builds are enabled for this job, and a later build finishes before an earlier build, the later build will reduce or skip t

Publish HTML reports

Reports

HTML directory to archive: htmlcov

Index page[s]: index.html

Report title: HTML Report

Publish JUnit test result report

Test report XMLs: python_tests.xml

'python_tests.xml' doesn't match anything

Fileset 'includes' setting that specifies the generated raw XML report files, such as 'myproject/target/test-reports/*.xml'. Basedir of t

□ Retain long standard output/error

Health report amplification factor: 1.0

1% failing tests scores as 99% health. 5% failing tests scores as 95% health

□ Do not fail the build on empty test results

Creacion de las pruebas

Cree un archivo de nombre **test_files.py**

```
import pytest
import json, files

@pytest.fixture
def client(request):
    client = files.app.test_client()
    return client


@pytest.mark.order1
def test_create(client):

        testFile3 = {'filename': 'testFile3', 'content': 'This is a test
file'}
        testFile4 = {'filename': 'testFile4', 'content': 'This is a test
file'}
        testFile5 = {'filename': 'testFile5', 'content': 'This is a test
file'}
```

```
        execute3 = client.post('/files', data = json.dumps(testFile3),
content_type='application/json')
        execute4 = client.post('/files', data = json.dumps(testFile4),
content_type='application/json')
        execute5 = client.post('/files', data = json.dumps(testFile5),
content_type='application/json')

        assert execute3.status == '201 CREATED'
        assert execute4.status == '201 CREATED'
        assert execute5.status == '201 CREATED'

        #assert get_content("testFile3") == "This is a test file"
        #assert get_content("testFile4") == "This is a test file"
        #assert get_content("testFile5") == "This is a test file"


@pytest.mark.order2
def test_recent_file(client):

        execute1 =
client.get('/files/recently_created',follow_redirects=True)

        assert "testFile3" in execute1.data
        assert "testFile4" in execute1.data
        assert "testFile5" in execute1.data

@pytest.mark.order3
def test_get_all_files(client):

        execute1 = client.get('/files',follow_redirects=True)

        assert "testFile3" in execute1.data
        assert "testFile4" in execute1.data
        assert "testFile5" in execute1.data

@pytest.mark.order4
def test_remove_files(client):

        client.delete('/files', follow_redirects=True)

        execute1 = client.get('/files',follow_redirects=True)

        assert "testFile3" not in execute1.data
        assert "testFile4" not in execute1.data
        assert "testFile5" not in execute1.data
```

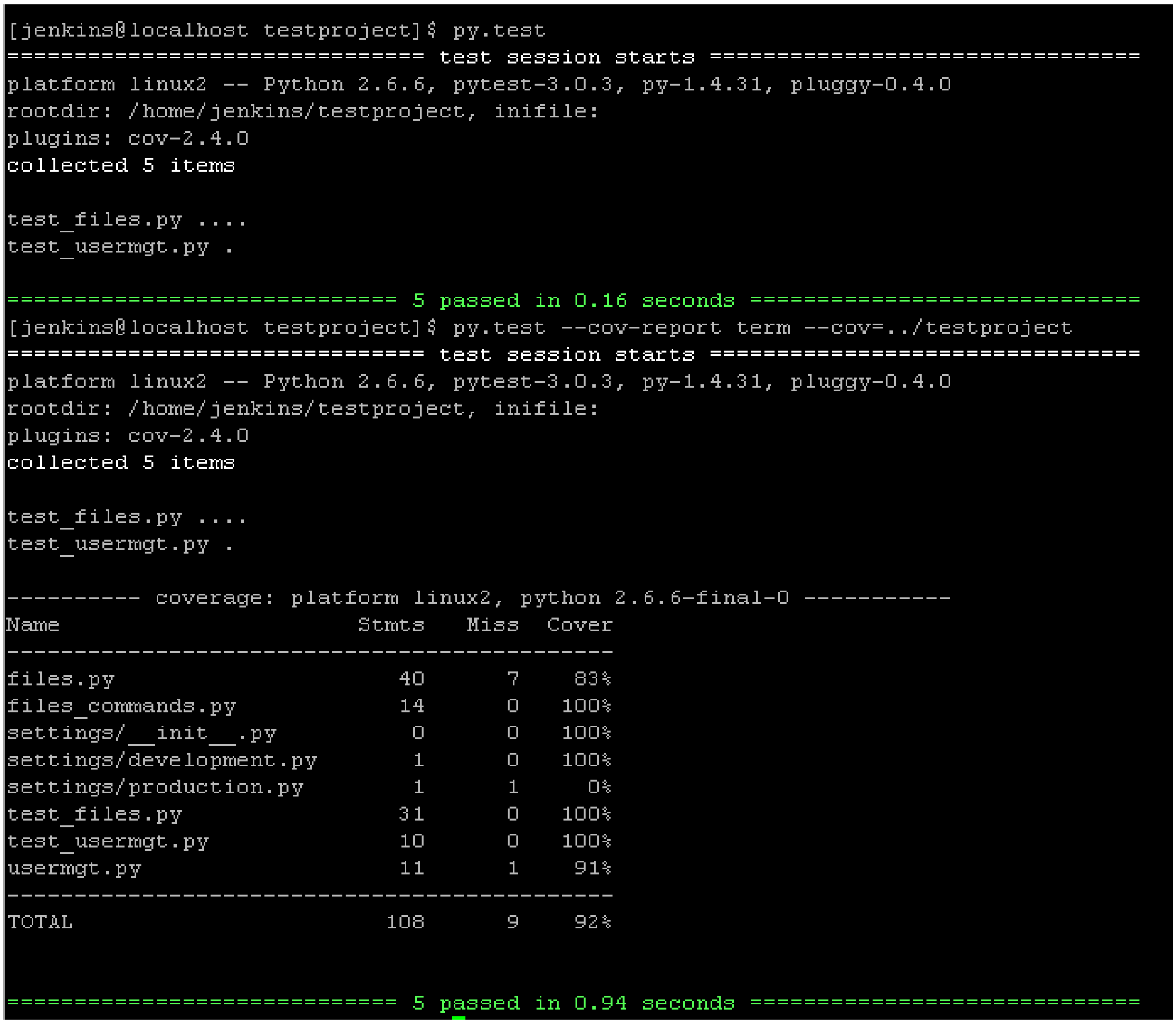
Pruebas locales

Ejecutar las pruebas con pytest

```
py.test
```

Ejecutar pruebas de cobertura desde la terminal

```
py.test --cov-report term --cov=../testproject
```



```
[jenkins@localhost testproject]$ py.test
=============================== test session starts ================================
platform linux2 -- Python 2.6.6, pytest-3.0.3, py-1.4.31, pluggy-0.4.0
rootdir: /home/jenkins/testproject, inifile:
plugins: cov-2.4.0
collected 5 items

test_files.py ....
test_usermgt.py .

============================= 5 passed in 0.16 seconds =============================
[jenkins@localhost testproject]$ py.test --cov-report term --cov=../testproject
=============================== test session starts ================================
platform linux2 -- Python 2.6.6, pytest-3.0.3, py-1.4.31, pluggy-0.4.0
rootdir: /home/jenkins/testproject, inifile:
plugins: cov-2.4.0
collected 5 items

test_files.py ....
test_usermgt.py .

---------- coverage: platform linux2, python 2.6.6-final-0 -----------
Name                          Stmts   Miss  Cover
-------------------------------------------------
files.py                         40      7    83%
files_commands.py                14      0   100%
settings/__init__.py              0      0   100%
settings/development.py           1      0   100%
settings/production.py            1      1     0%
test_files.py                    31      0   100%
test_usermgt.py                  10      0   100%
usermgt.py                       11      1    91%
-------------------------------------------------
TOTAL                           108      9    92%


============================= 5 passed in 0.94 seconds =============================
```

Dirección al github de la soluciónn

https://github.com/esteban-duran/testproject