CS499 - Open Source software development

Tests and CI

Dr. Igor Steinmacher

e-mail: Igor.Steinmacher@nau.edu

Twitter: @igorsteinmacher

NAU NORTHERN ARIZONA UNIVERSITY

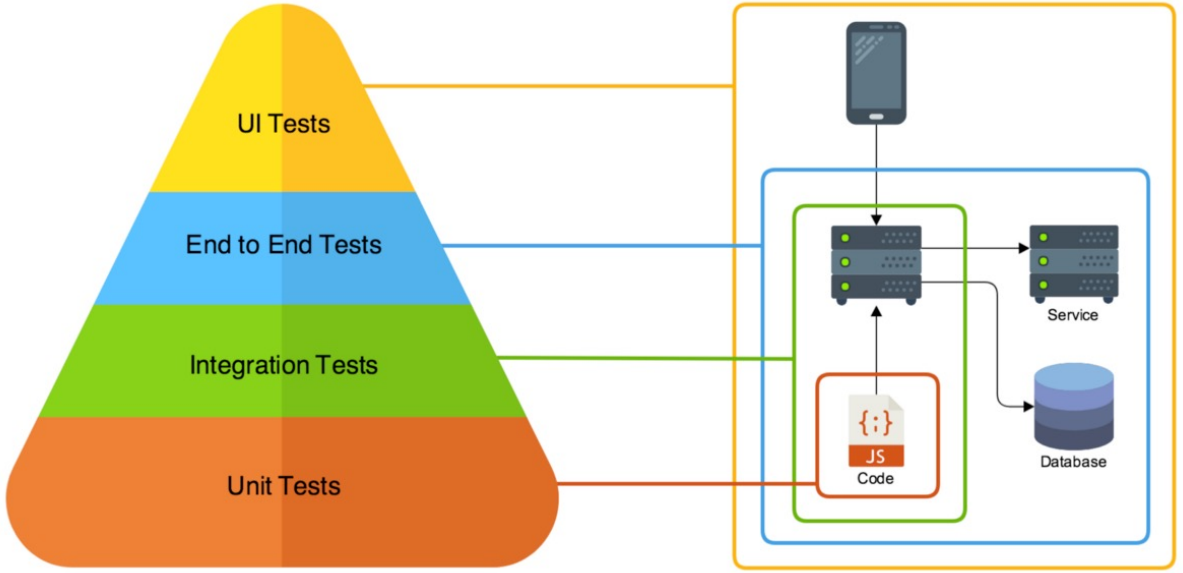
Based on the material of @gustavopinto and @filipesaraiva

What are unit tests?

- Functions/methods to test the behavior of small(est) parts of a software in isolation
- May or may not be executed in production
- The goal is to validate that each small unit of software will have the expected behavior
- Basically:
  - Input the data in a method and check if it is returning the correct result

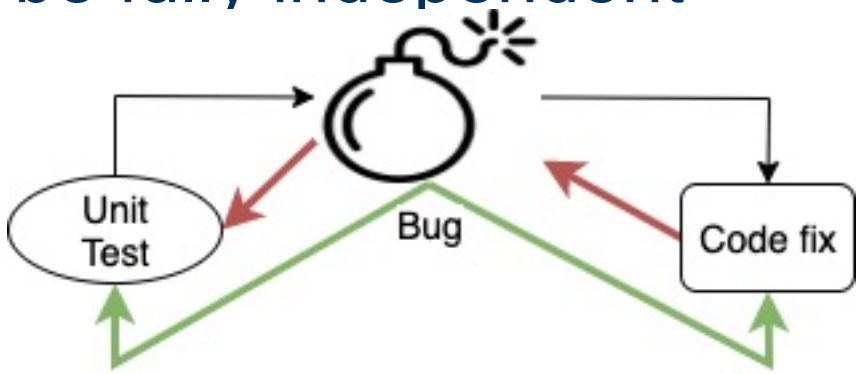
Why?

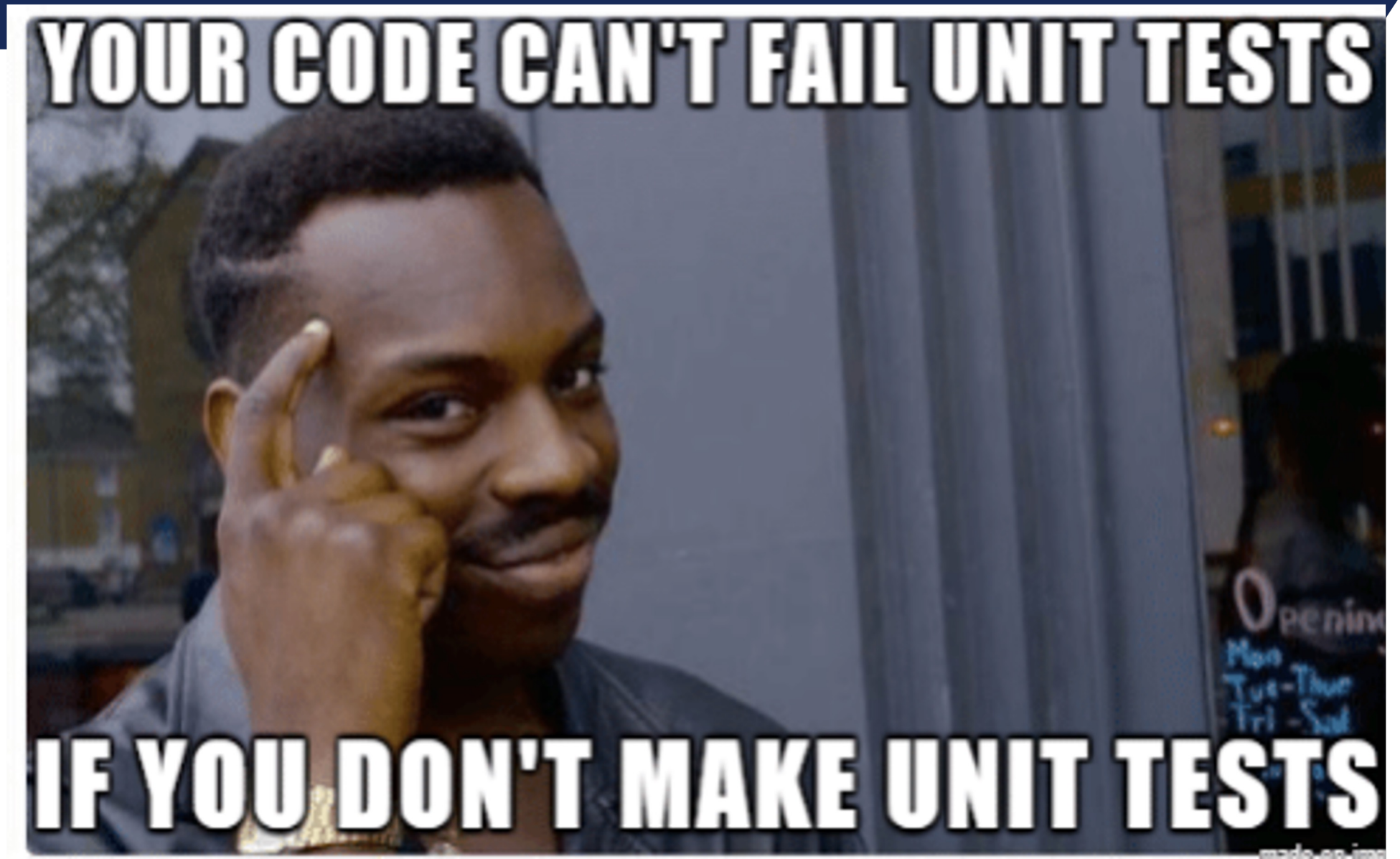
- Check if pieces are working after you change the system
  - Unit testing increases confidence in changing/ maintaining code.
- Point out defects while developing
- Enforces more reusable code → You need to go modular
- Debugging is easier (s
- Helps you finding wher



What else?

- First level of software testing
- The smallest testable parts of a software are tested
- Validate each unit of the software
- Each test unit must be fully independent




YOUR CODE CAN'T FAIL UNIT TESTS
IF YOU DON'T MAKE UNIT TESTS

Unit Tests: Basic!

```
import unittest class TestSum(unittest.TestCase):

def test_sum(self):
        self.assertEqual(sum([1, 2, 3]), 6, "Should be 6")

def test_sum_tuple(self):
        self.assertEqual(sum((1, 2, 2)), 6, "Should be 6")

if __name__ == '__main__':
        unittest.main()
```

```
.F
======================================================================
FAIL: test_sum_tuple (__main__.TestSum)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "basicTest.py", line 8, in test_sum_tuple
    self.assertEqual(sum((1, 2, 2)), 6, "Should be 6")
AssertionError: 5 != 6 : Should be 6

----------------------------------------------------------------------
Ran 2 tests in 0.000s

FAILED (failures=1)
```

Unit Tests: Basic!

```
def is_prime(number):
            for element in range(number):
                        if (number % element == 0):
                                    return False
            return True


def print_next_prime(number):
            index = number
            while True:
                        index += 1
                        if is_prime(index):
                                    print(index)
```

The smallest unit here is the **is_prime** function (**print_next_time** uses it)

Let's test it using something that we know
Is 5 prime?

Thanks
https://jeffknupp.com/blog/2013/12/09/improve-your-python-understanding-unit-testing/

Unit Tests: Basic!

```
def is_prime(number):
            for element in range(number):
                        if (number % element == 0):
                                    return False
            return True


def print_next_prime(number):
            index = number
            while True:
                        index += 1
                        if is_prime(index):
                                    print(index)
```

```
import unittest
from prime import is_prime

class PrimesTestCase(unittest.TestCase):
   def test_is_five_prime(self):
      self.assertTrue(is_prime(5))
      #self.assertEquals(is_prime(5), True)

if __name__ == '__main__':
            unittest.main()
```

```
E
======================================================================
ERROR: test_is_five_prime (__main__.PrimesTestCase)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "PrimesTestCase.py", line 6, in test_is_five_prime
    self.assertTrue(is_prime(5))
  File "/Users/igorsteinmacher/Downloads/prime.py", line 3, in is_prime
    if (number % element == 0):
ZeroDivisionError: integer division or modulo by zero

----------------------------------------------------------------------
Ran 1 test in 0.001s

FAILED (errors=1)
```

NORTHERN ARIZONA UNIVERSITY

School of Informatics, Computing, and Cyber Systems

CS499 - Open Source software development

Tests and CI

Dr. Igor Steinmacher

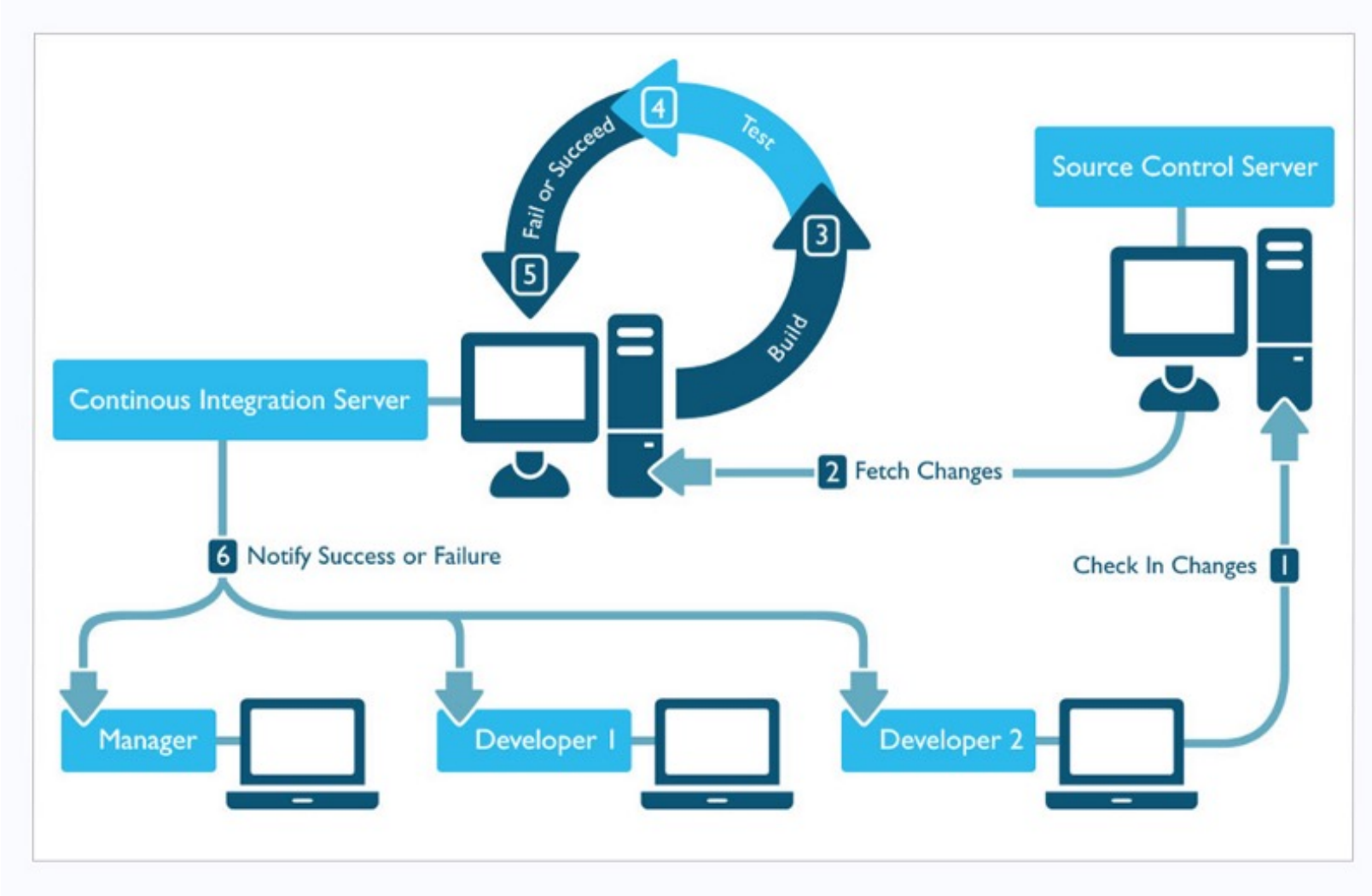
e-mail: Igor.Steinmacher@nau.edu

Twitter: @igorsteinmacher

Based on the material of @gustavopinto and @filipesaraiva

What is Continuous Integration?

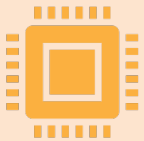
- Continuous Integration is a set of related practices that guide development teams to implement minor changes and check in code to version control repositories frequently.


Source Control Server
Continous Integration Server
Fail or Succeed
4
Test
3
Build
5
2 Fetch Changes
6 Notify Success or Failure
Check In Changes 1
Manager
Developer 1
Developer 2

What is it for?

Find and investigate bugs faster

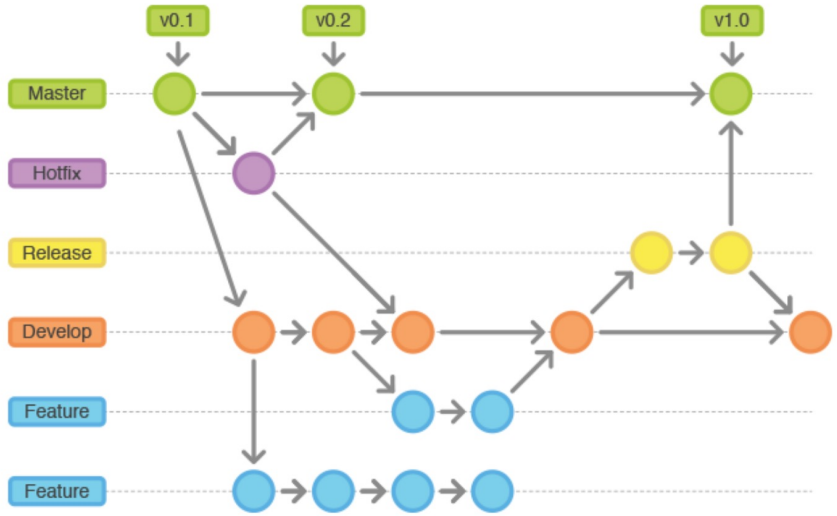
Improve Software Quality



Reduce the time to validate and release the software

When to use it?

- When there is a long time until branches are merged with the master
- When there is a problem to integ
- When the integration gets more ange



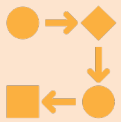
What do you need?

Version Control

single repositor

Automated build process

Command line compilation (IDE build)



Team acceptance

Integration is not exactly a tool (but a practice)

Team used to develop in the master

All need to share the same vision

“Requirements” for CI

Frequent check-ins (commits)

- Throughout the day
- Smaller changes
- Less chances of failing

A comprehensive automated test set

Keep the compilation and test process short

- Increase the chances of running the CI locally with the time is small

Travis CI

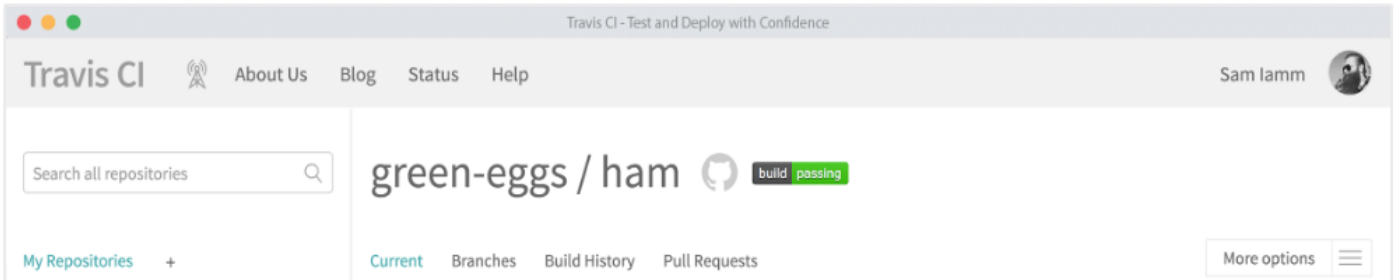
- Integration with users/projects on GitHub
- Free for public repos
- Easy to use
- Support for Mac, Windows, iOS

Travis CI About Us Blog Status Documentation Help Sign in with GitHub

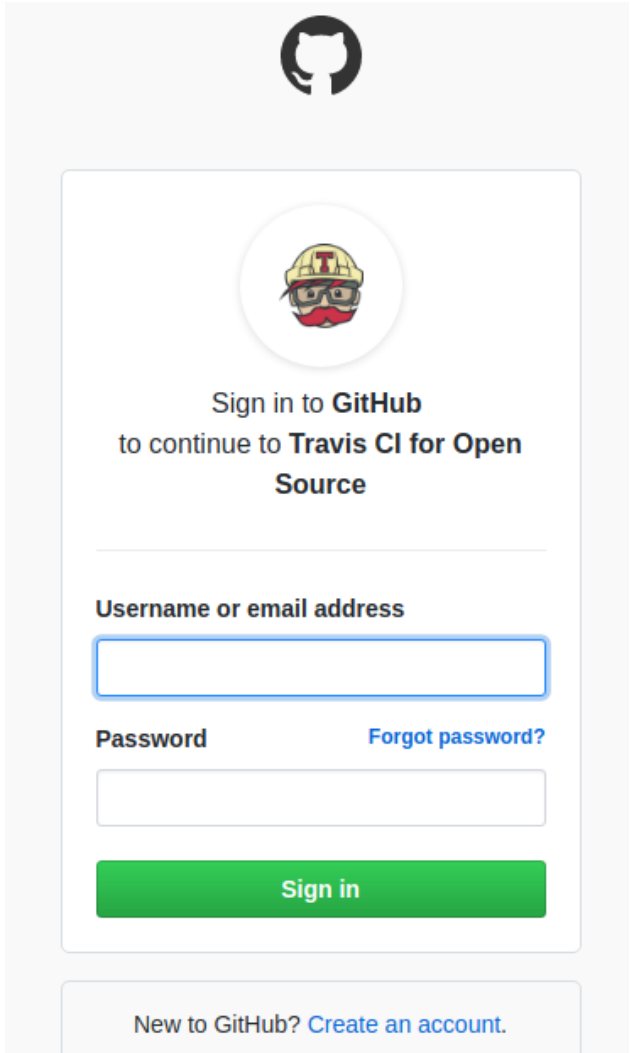
Test and Deploy with Confidence

Easily sync your GitHub projects with Travis CI and you'll be testing your code in minutes!

Sign Up



Sign up to Travis



Igor Steinmacher

@igorsteinmacher

Repositories | Settings | Subscription | Migrate

GitHub Apps Integration

Activate the GitHub Apps integration to start testing and deploying on Travis CI.

The GitHub Apps integration supports both private and open source repositories, while providing enhanced security when interacting with GitHub.

Activate

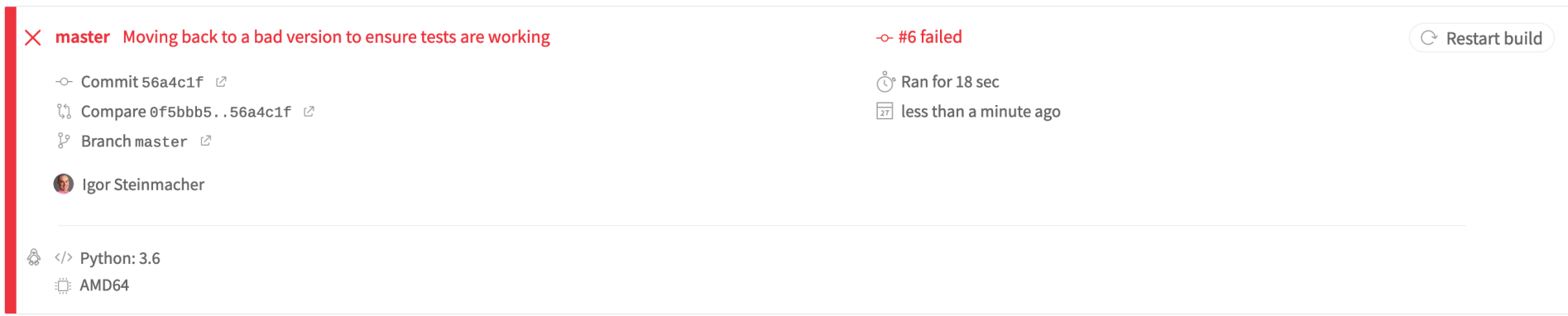
We are only able to migrate accounts that have 50 or fewer repositories using the Legacy Services Integration. Please refer to our documentation on how to migrate your account.

On GitHub: your .travis.yml

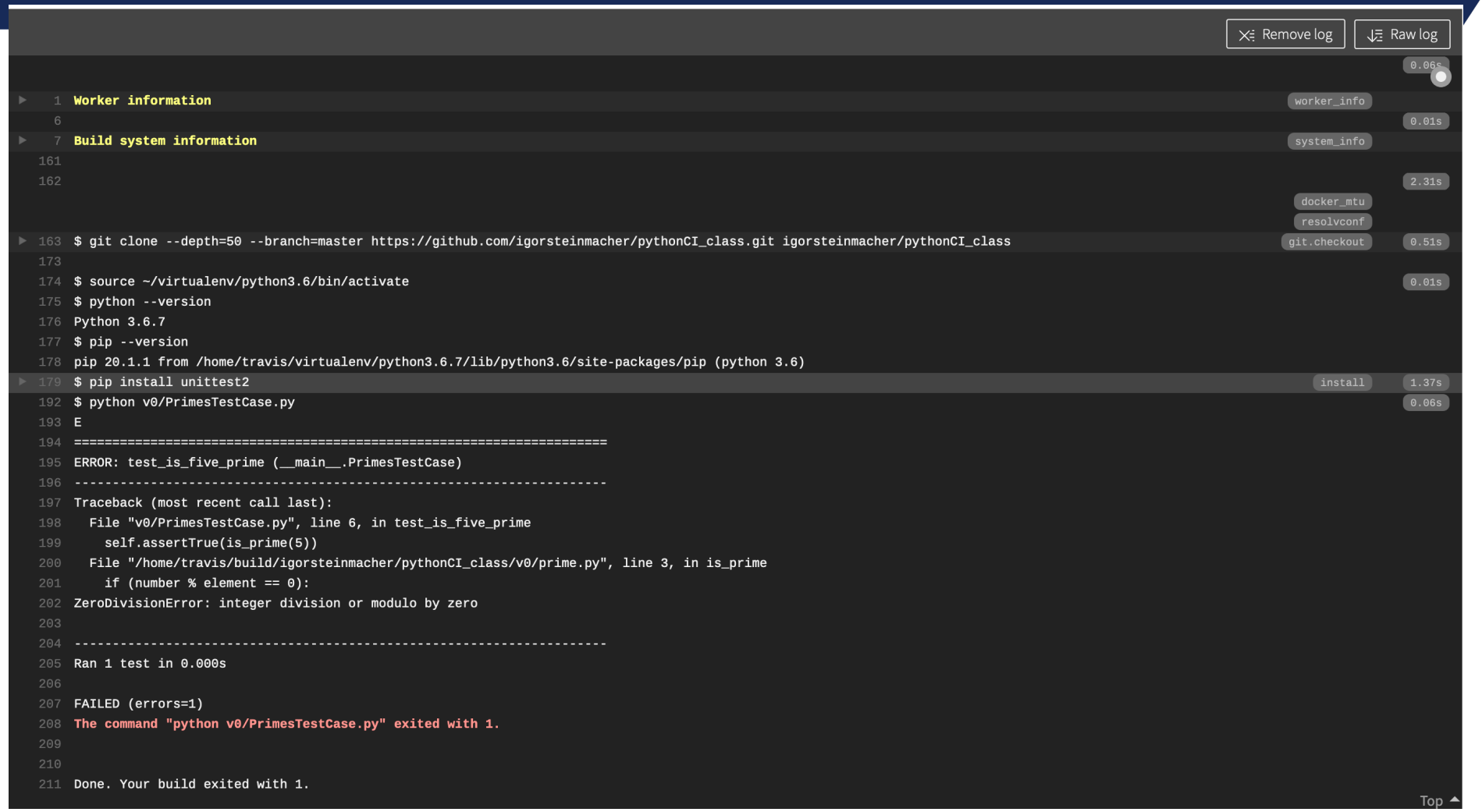
12 lines (8 sloc) | 129 Bytes

```
1  language: python
2
3  python:
4    - "3.6"
5
6  sudo: required
7
8  install:
9    - pip install unittest2
10
11 script:
12   - python v0/PrimesTestCase.py
```

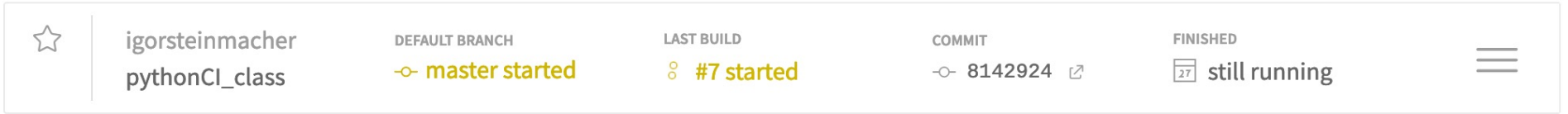

igorsteinmacher
pythonCI_class
DEFAULT BRANCH
master started
LAST BUILD
#6 started
COMMIT
56a4c1f
FINISHED
still running

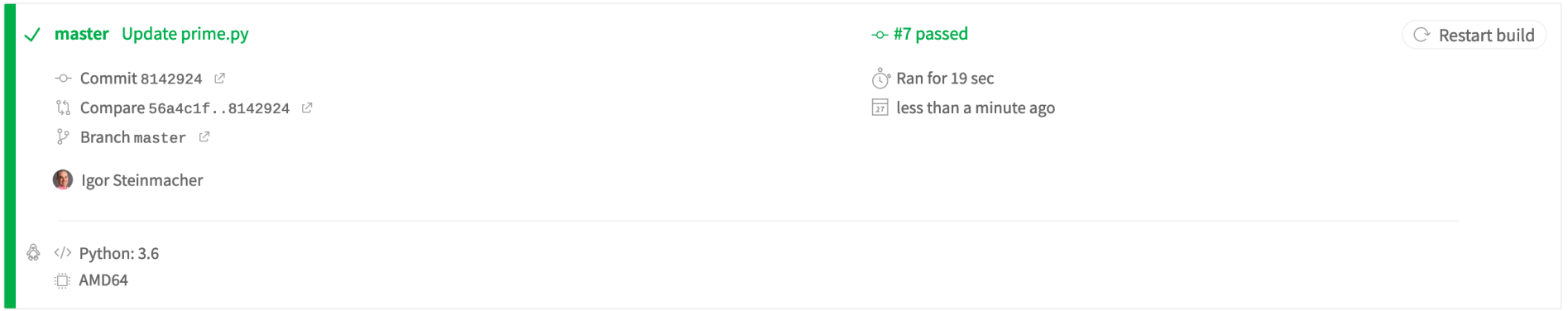

master Moving back to a bad version to ensure tests are working
#6 failed
Restart build
Commit 56a4c1f
Compare 0f5bbb5..56a4c1f
Branch master
Igor Steinmacher
Ran for 18 sec
less than a minute ago
Python: 3.6
AMD64

Oh No!


Remove log
Raw log
0.06s
1 Worker information
worker_info
6
0.01s
7 Build system information
system_info
161
162
2.31s
docker_mtu
resolvconf
163 $ git clone --depth=50 --branch=master https://github.com/igorsteinmacher/pythonCI_class.git igorsteinmacher/pythonCI_class
git.checkout
0.51s
173
174 $ source ~/virtualenv/python3.6/bin/activate
0.01s
175 $ python --version
176 Python 3.6.7
177 $ pip --version
178 pip 20.1.1 from /home/travis/virtualenv/python3.6.7/lib/python3.6/site-packages/pip (python 3.6)
179 $ pip install unittest2
install
1.37s
192 $ python v0/PrimesTestCase.py
0.06s
193 E
194 ==
195 ERROR: test_is_five_prime (__main__.PrimesTestCase)
196 --
197 Traceback (most recent call last):
198 File "v0/PrimesTestCase.py", line 6, in test_is_five_prime
199 self.assertTrue(is_prime(5))
200 File "/home/travis/build/igorsteinmacher/pythonCI_class/v0/prime.py", line 3, in is_prime
201 if (number % element == 0):
202 ZeroDivisionError: integer division or modulo by zero
203
204 --
205 Ran 1 test in 0.000s
206
207 FAILED (errors=1)
208 The command "python v0/PrimesTestCase.py" exited with 1.
209
210
211 Done. Your build exited with 1.
Top

Changed the code and commiTted back





YES!

```
163 $ git clone --depth=50 --branch=master https://github.com/igorsteinmacher/pythonCI_class.git igorsteinmacher/pythonCI_class
173
174 $ source ~/virtualenv/python3.6/bin/activate
175 $ python --version
176 Python 3.6.7
177 $ pip --version
178 pip 20.1.1 from /home/travis/virtualenv/python3.6.7/lib/python3.6/site-packages/pip (python 3.6)
179 $ pip install unittest2
192 $ python v0/PrimesTestCase.py
193 .
194 ----------------------------------------------------------------------
195 Ran 1 test in 0.000s
196
197 OK
198 The command "python v0/PrimesTestCase.py" exited with 0.
199
```