
Test-Driven Development

CS480 Software Engineering

Yu Sun, Ph.D.

<http://yusun.io>

yusun@cpp.edu



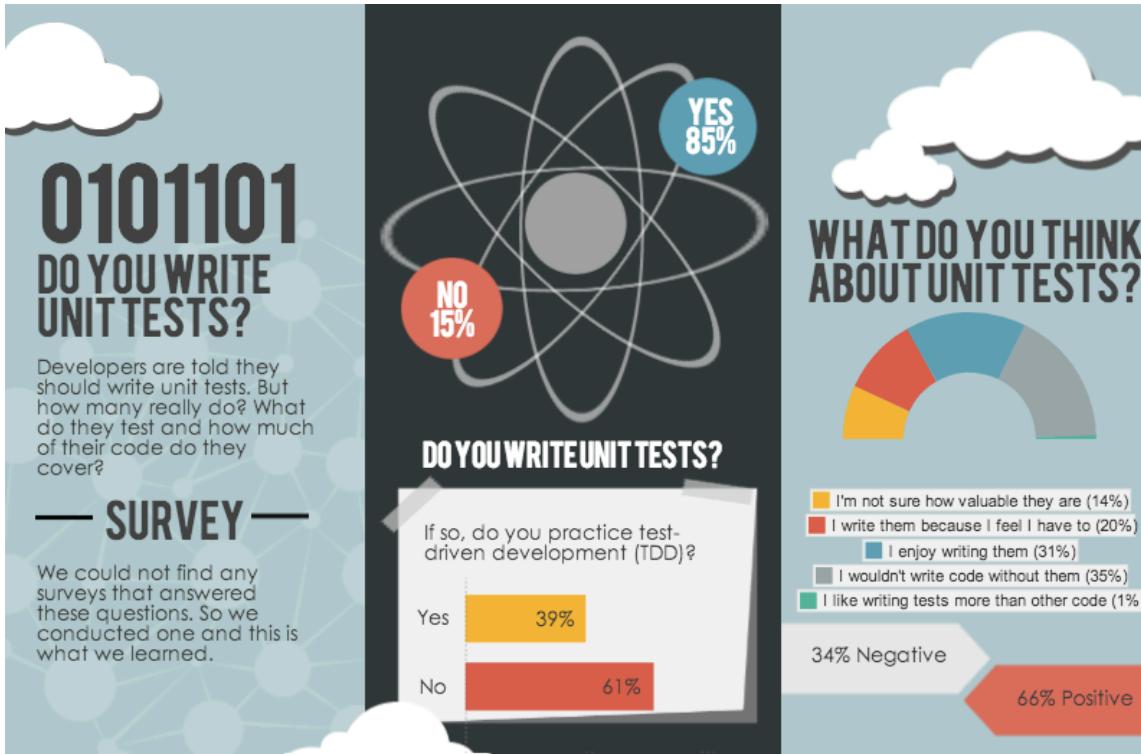
CAL POLY POMONA

Software Testing in Practice

- ◆ Unit Test (JUnit)
- ◆ Integration Test (JUnit)
- ◆ Test in Build Automation (Maven)
- ◆ Code Coverage (Cobertura)
- ◆ Static Analysis (Checkstyles, FindBugs)
- ◆ Continuous Integration (Jenkins)

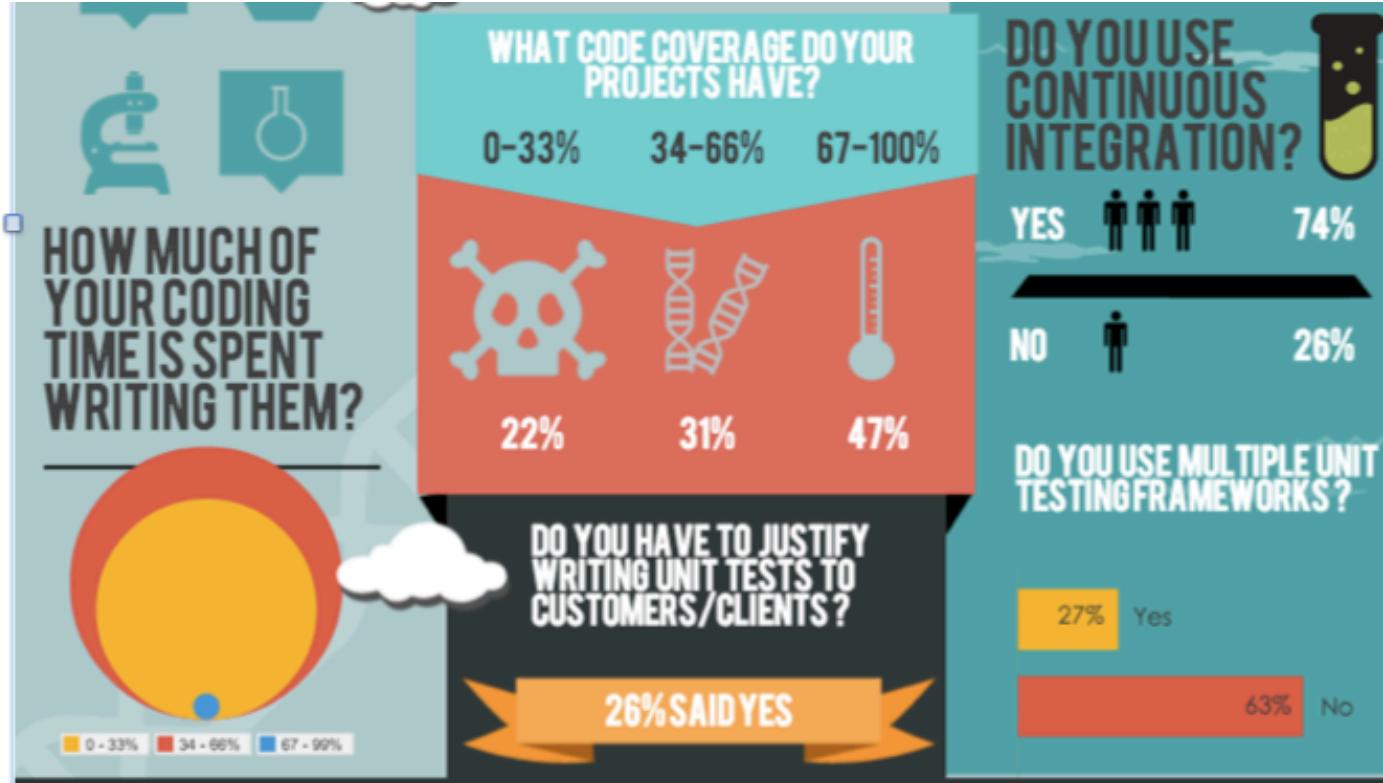


Survey on Unit Tests



•<http://blog.hubstaff.com/survey-many-developers-write-unit-tests/>

Survey on Unit Tests



• <http://blog.hubstaff.com/survey-many-developers-write-unit-tests/>

How do you like writing tests?



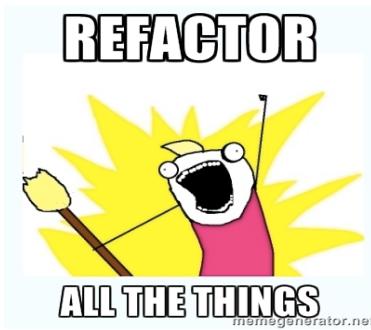
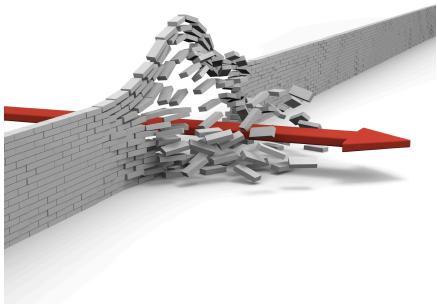
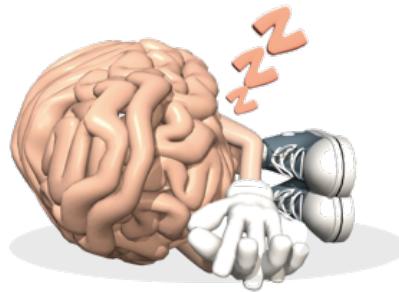
Cameron Purdy, SVP Engineering of Oracle Middleware

140 upvotes by Karthik Balaji, Satyanarayana Shanmugam, Simon Kinahan, (more)

It's simple: Writing unit tests sucks. I personally (and absolutely) hate writing unit tests for code that I have written. I would rather go to the dentist than write unit tests. Maybe even the proctologist.

- <http://www.quora.com/Why-do-so-many-developers-not-deliver-unit-tests-with-their-code>

Why?



There must be a better way!



DECISIONS

"There has to be a better way to decide who makes the bed!"

Test-Driven Development (TDD)



**ALL CODE IS GUILTY
UNTIL PROVEN INNOCENT**

TDD in Detail

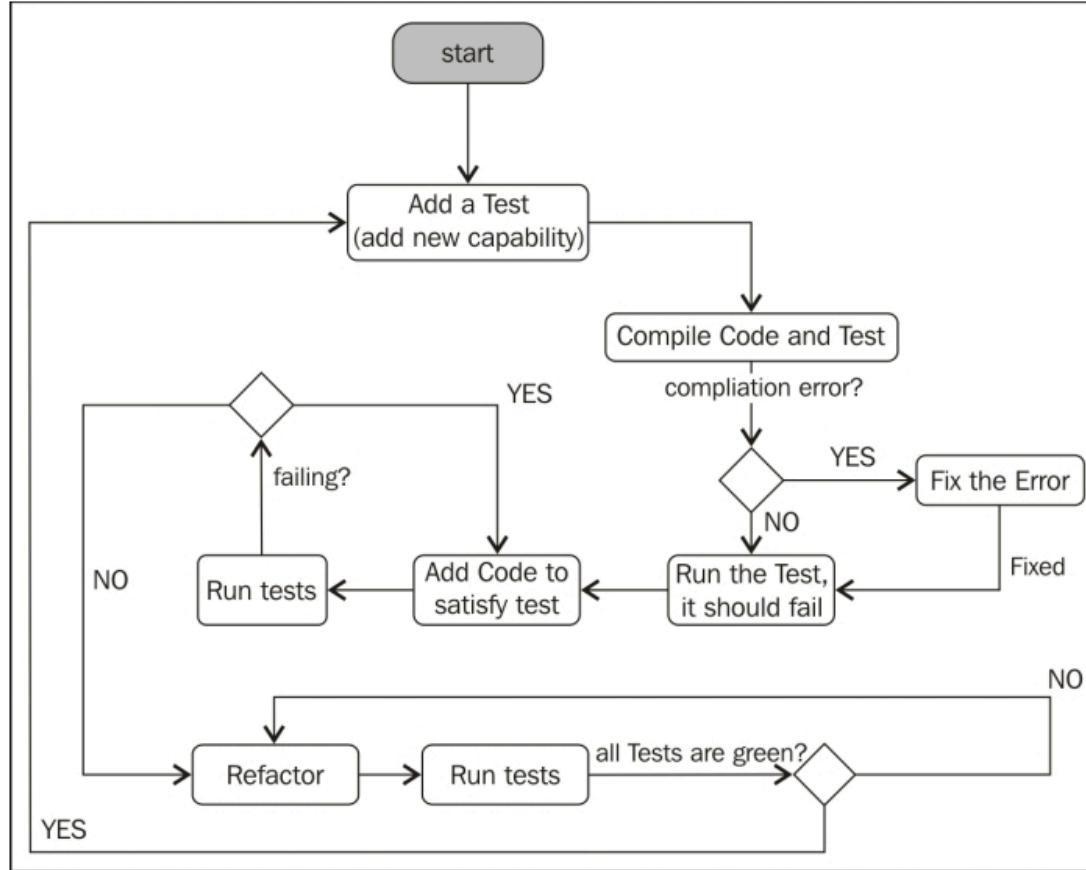
1. Write a test to add a new capability (automate tests)
2. Write code only to satisfy tests
3. Re-run the tests—if any test is broken, revert the change
4. Refactor and make sure all tests are green
5. Continue with step 1

TTD by Example

- ◆ Let us write a program to calculate tax and test it.
 - ◆ If the taxable income is less than USD 500,000, then deduct 10 percent as tax
 - ◆ The tax is 20 percent for taxable income between USD 500,000 and USD 1,000,000
 - ◆ The tax is 30 percent for taxable income above USD 1,000,000



TDD in Detail



Benefits of TTD

- ◆ Clean, testable, and maintainable code



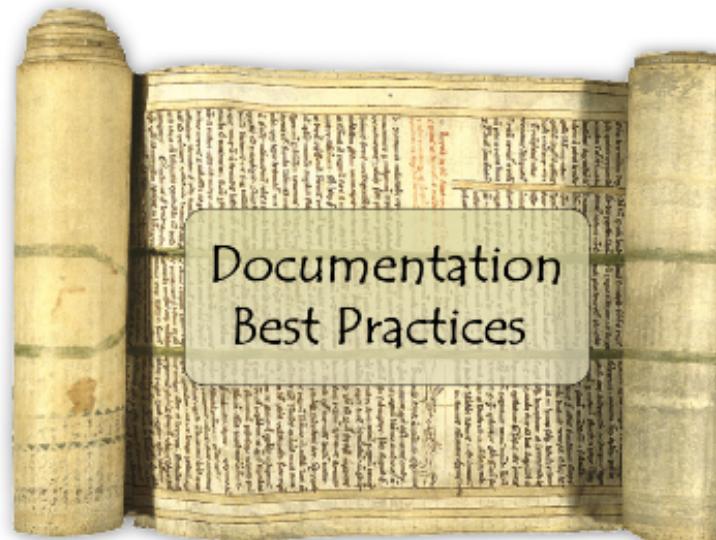
Benefits of TTD

- ◆ Incrementally building your code
- ◆ Your API is easier to work with, because the code is written and used at the same time



Benefits of TTD

- ◆ Tests become the best documentation



Benefits of TTD

- ◆ Immediate test triggers efficient development cycles



Benefits of TTD

- ◆ Avoid over engineering the class that you are writing
- ◆ Just write what's needed to make all tests green

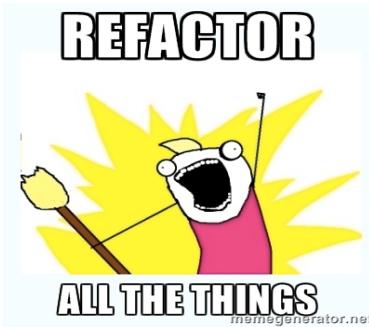
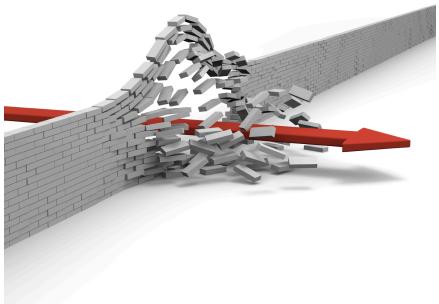
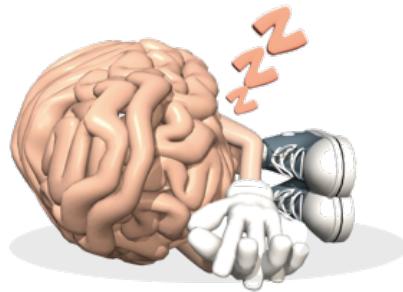


Benefits of TTD

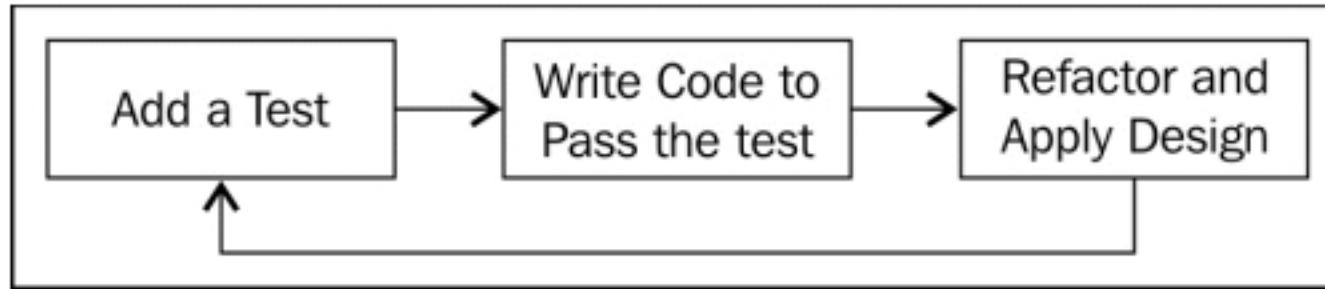
- ◆ You get more confidence about your code



Solving the Problems



TDD – The Big Picture



*A test is written for a requirement,
a code is added to pass the test,
the code is cleaned,
and the design is identified.*

Functionality is Broken after Enhancement



Functionality is Broken after Enhancement

 Victor Sui ⌚ 10:59 PM (32 minutes ago) ⭐ ↺ ▾

to me, Jeff, Carol ▾

I have been patching things up lately and it seems like every time I fix something something else breaks. Right now it seems like importing the computation code and download file code is broken somehow, even though I had it working earlier.
When I just had everything written in one file everything worked though so I can always revert back to that. You are supposed to run everything by running index.js

Thanks,

Victor

On Wed, Jan 28, 2015 at 12:46 AM, Yu Sun <yu.sun.cs@gmail.com> wrote:
Hi Victor,
...
[Message clipped] [View entire message](#)

Functionality is Broken after Enhancement



This can happen if this rule of thumb is not followed—Do not write any code unless you have a failing unit test and do not write more code than is sufficient to make the test pass. Big upfront coding creates this issue.