

Why Do We Still Not We Give A @#\$! About Testing?

Trisha Gee @trisha_gee

Developer Advocate at JetBrains and OCD Geek

2010 IT Project Success Rates

Project Style	Successful	Challenged	Failure
Ad-hoc	49%	37%	14%
Iterative	61%	28%	11%
Agile	60%	28%	12%
Traditional	47%	36%	17%

[Dr Dobbs, 2010](#)

...when it comes to quality, 40% prefer to deliver on time and on budget and 57% prefer to deliver high-quality, easy-to-maintain systems.

Software Project Success Rates

Successful	32%
Challenged	44%
Failure	24%

[Gartner, 2012](#)

So, Why Don't We Test?

Focus is on
delivering features

Boring

Too much effort for
too little value

"I don't know what
it's supposed to do"

Code wasn't designed
for testing

Too Hard

Just a box
to tick

Framework
gets in the way

"It's only a prototype"

Takes too
much time

I'm the only
maintainer

The code is
self describing

Manual
Process

Why Should We Care?

Documentation Freedom to
Refactor

Knowledge
Sharing

Usability

Help us
with Design

So we know
what doesn't
work

Identify missing
features

So we know
it works!

What Can We Do?

Tap into Motivation

- OCD
- Frame it as a difficult problem
- Gamification
- Make it as easy as possible
- Lead by example
- Exploration (what if...?)
- Set as primary task
- Add to Objectives
- Break emotional attachment to code

We Don't Have To Care About
Testing

We Need To Encourage
Behaviour That Prioritises Quality