		9			2			5
5	3	8		6	4			9
1	6	2					3	
		3		2	7			
	5	4	6			1		
		7		1	5	3	4	
3			8		1	9		6
7			3			8	5	
	9	1				4	7	

		9			2			5
5	3	8		6	4			9
1	6	2					3	
		3		2	7			
	5	4	6			1		
		7		1	5	3	4	
3			8		1	9		6
7			3			8	5	
	9	1				4	7	

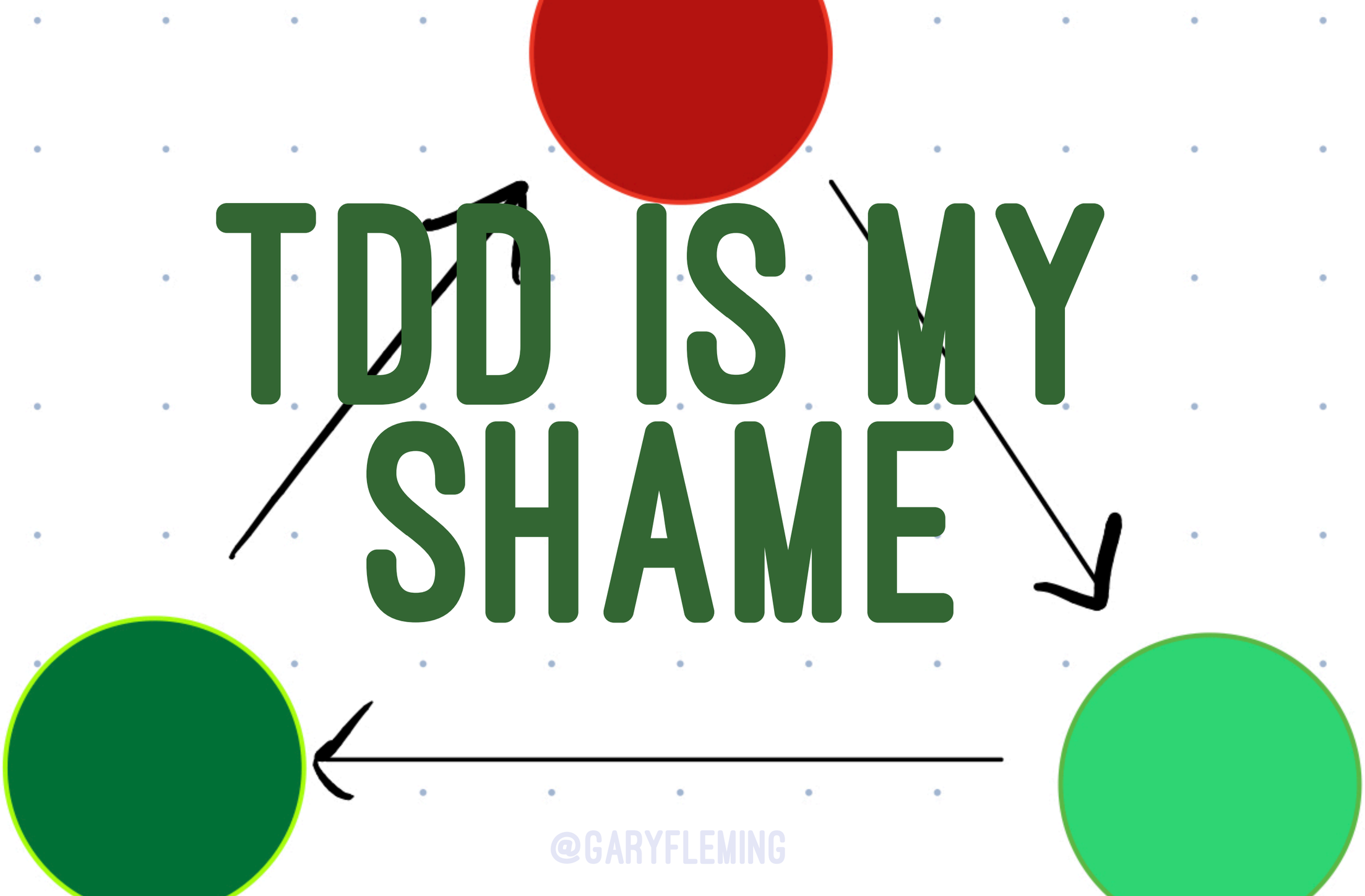
	3	5						6
				7		8		
		1			9			
9	2						7	8
	5						2	
3						5		
			5				1	
	9	4				2		
			6		7			4

I WASN'T GOOD AT IT RIGHT AWAY. SO I QUIT.

		9			2			5
5	3	8		6	4			9
1	6	2					3	
		3		2	7			
	5	4	6			1		
		7		1	5	3	4	
3			8		1	9		6
7			3			8	5	
	9	1				4	7	

	3	5						6
				7		8		
		1			9			
9	2						7	8
	5						2	
3						5		
			5				1	
	9	4				2		
			6		7			4

TDD IS MY SHAME



@GARYFLEMING

PART 0:
A JOURNEY INTO SHAME

@GARYFLEMING

CODECRAFT

@GARYFLEMING

CODECRAFT

@GARYFLEMING

PUB / PAUL

@GARYFLEMING

"I'M A LITTLE ASHAMED, BUT I DON'T TDD AS MUCH AS I THINK I SHOULD."

@GARYFLEMING

@GARYFLEMING

"DO YOU TDD?"

@GARYFLEMING

SHAME

@GARYFLEMING

WHAT IS SHAME?

@GARYFLEMING

WHAT IS SHAME?

- SELF-AWARENESS
  - SELF-BLAME
  - STANDARDS
- PERSONAL TRAIT
  - SELF-ESTEEM
  - ... AND MORE

@GARYFLEMING

"LISTENING TO SHAME" BY BRENÉ BROWN

@GARYFLEMING

"SHAME, FOR WOMEN, IS THIS WEB OF UNOBTAINABLE, CONFLICTING, COMPETING EXPECTATIONS ABOUT WHO WE'RE SUPPOSED TO BE. AND IT'S A STRAIGHT-JACKET."
– BRENÉ BROWN

@GARYFLEMING

DON'T BE WEAK

@GARYFLEMING

TDD AND SHAME

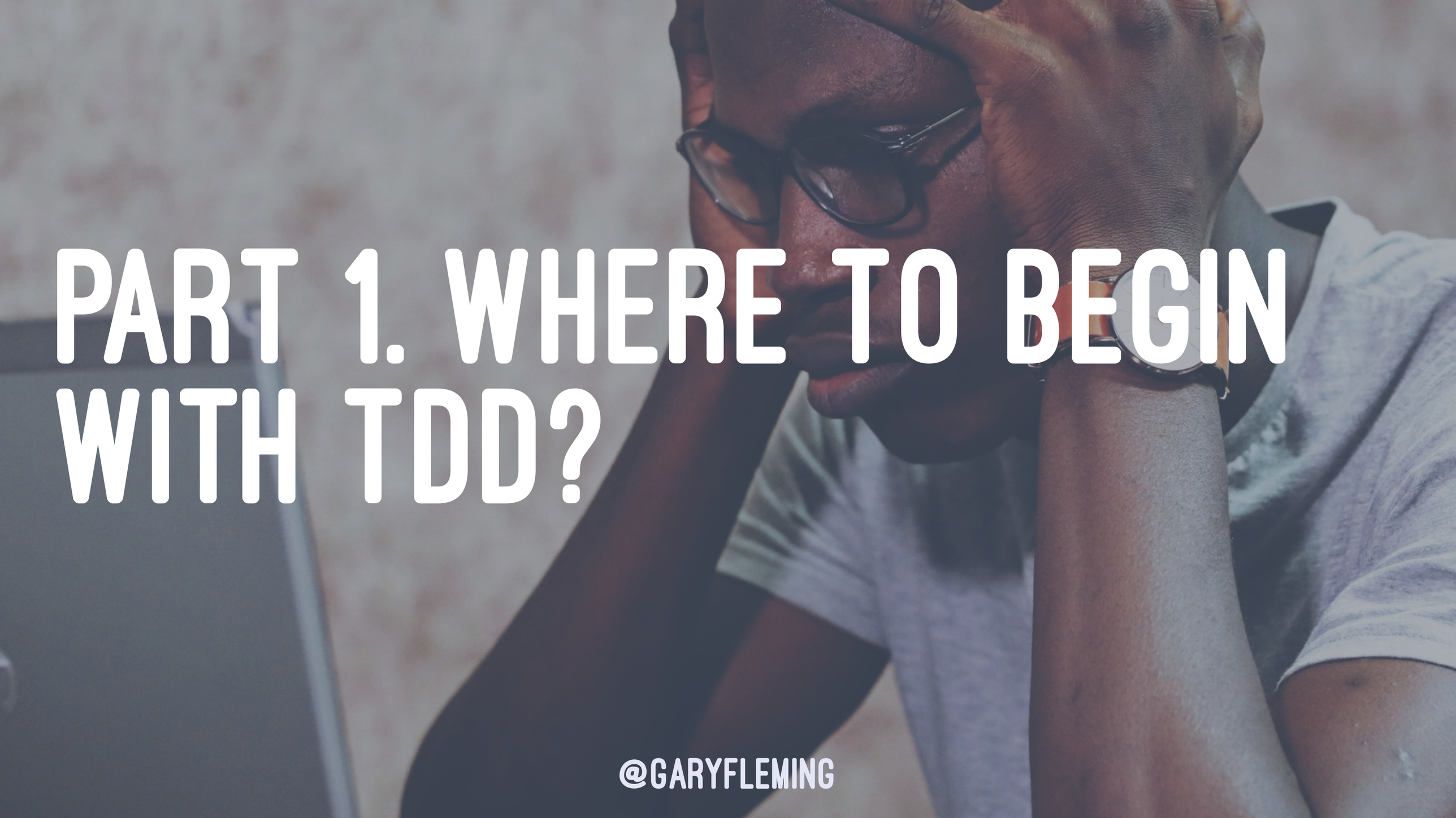
@GARYFLEMING

YOU'RE GOOD ENOUGH

@GARYFLEMING

EASY

		9			2			5
5	3	8		6	4			9
1	6	2					3	
		3		2	7			
	5	4	6			1		
		7		1	5	3	4	
3			8		1	9		6
7			3			8	5	
	9	1				4	7	



PART 1. WHERE TO BEGIN WITH TDD?

@GARYFLEMING

- TESTING PRACTICE... NOT REALLY
  - DESIGN STRATEGY... MAYBE
- DEVELOPMENT PRACTICE... PROBABLY

AntonyMarcano
@AntonyMarcano

Replying to @machielg and @allenholub

I would say TDD is a software development practice... design, coding, testing are all elements of software development.

(Too often, "development" is used synonymously with "writing code" which does the definition of the word a disservice)

@GARYFLEMING

NO ONE TRUE TDD

@GARYFLEMING

ANCHOR TERMS

@GARYFLEMING

UNIT TESTS

@GARYFLEMING

UNIT TESTS

- NO DATABASE
- NO NETWORK
- NO FILESYSTEM
- MUST BE PARALLELISABLE
- CAN'T HAVE ANY WEIRD ENVIRONMENT SET-UP

METZ SAYS

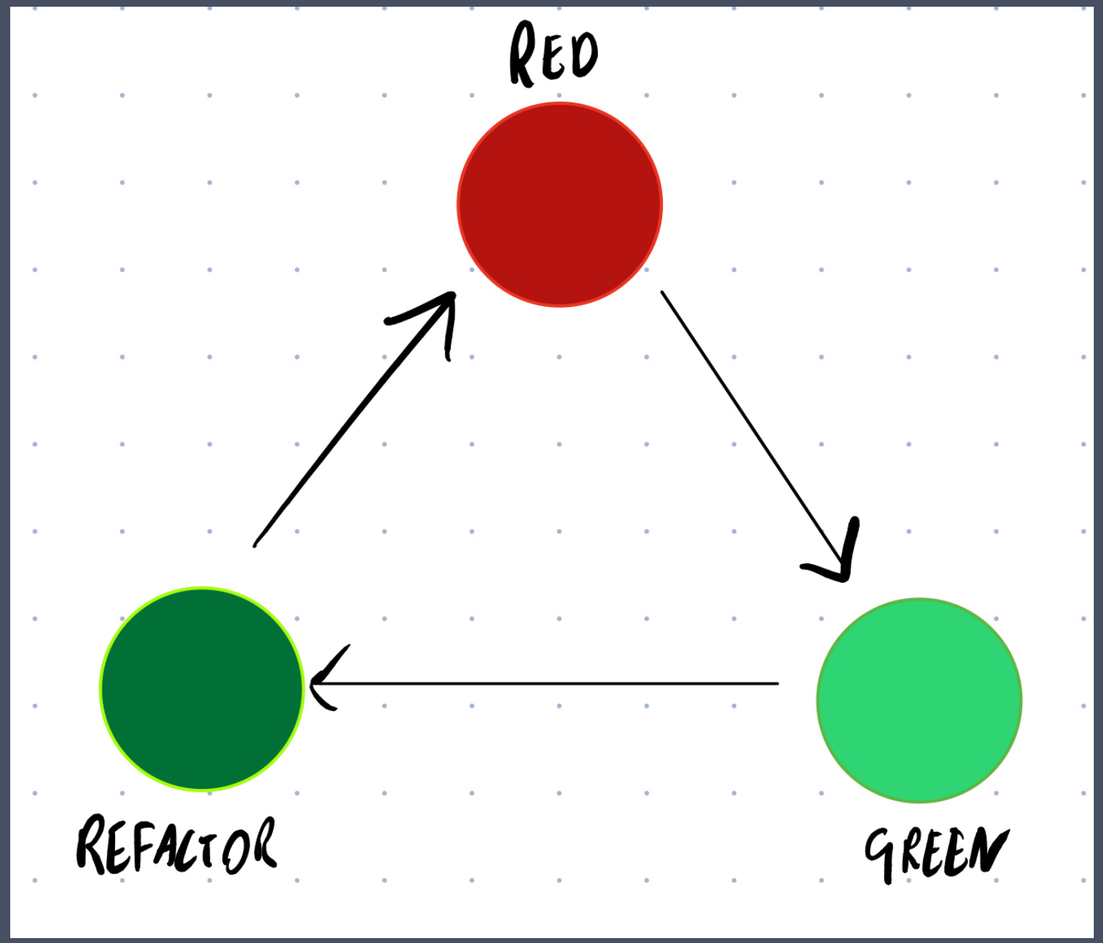
- THOROUGH
- STABLE
- FAST
- FEW

@GARYFLEMING

TEST FIRST / TEST-DRIVEN DEVELOPMENT

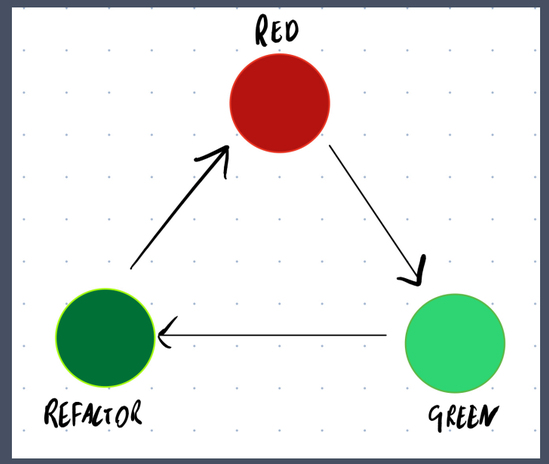
@GARYFLEMING

THE BASICS: RED - GREEN - REFACTOR



@GARYFLEMING

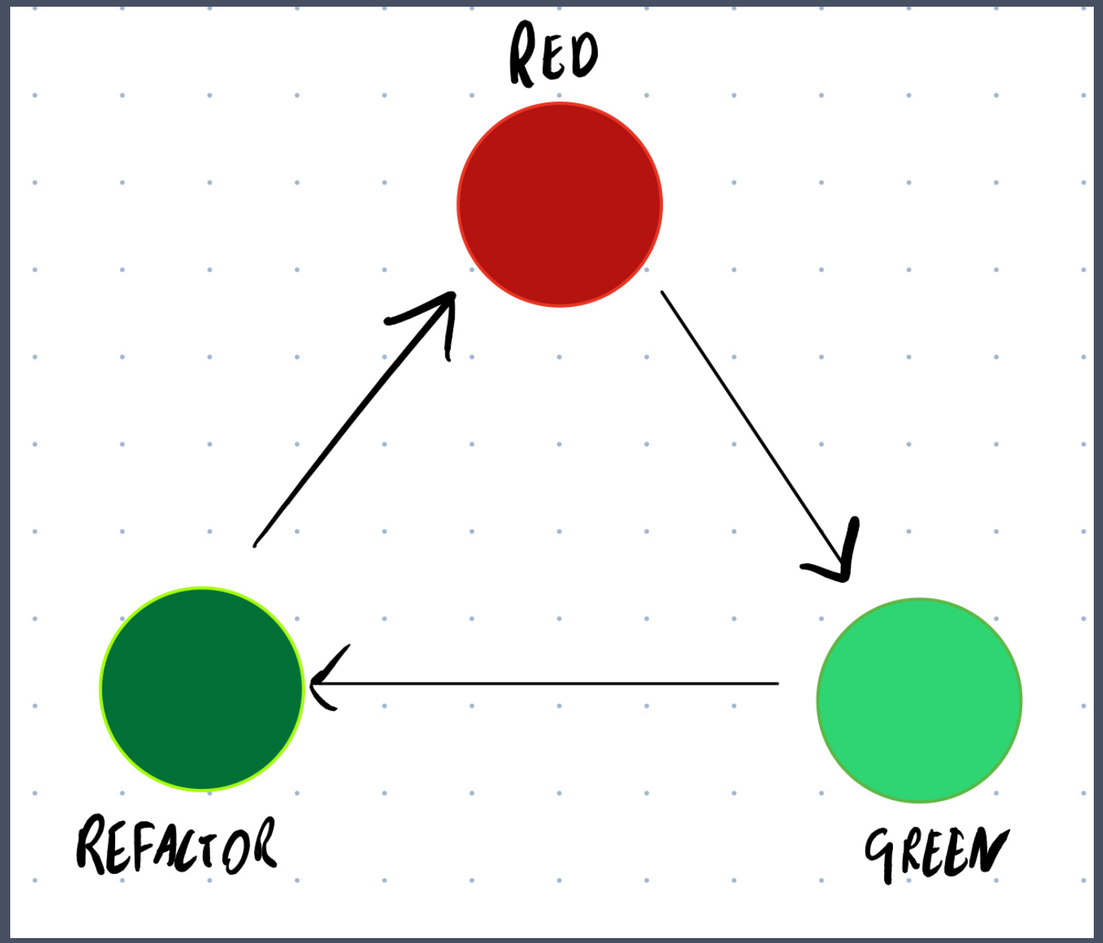
THE BASICS: RED



- ASSERTION FAILURE
- UNEXPECTED EXCEPTIONS
- MISSING CODE

@GARYFLEMING

THE BASICS: GREEN



@GARYFLEMING

THE BASICS: REFACTOR

$$5 \times 291 = 1455$$

$$5 + 3 + 97 = 1455$$

CHANGE CODE WITHOUT CHANGING FUNCTIONALITY.

@GARYFLEMING

COMMON GRIPES: MUST. SEE. TEST. FAIL

Marco Rogers
@polotek

Listen y'all. I'm not a person to push strict adherence to TDD or any particular testing methodology. But one thing you *must* do is *watch your test fail* before getting it to pass. However you do tests is fine. This is the only hard rule IMO.

COMMON GRIPES: THE LOOP SHOULD BE MUCH SMALLER.

@GARYFLEMING

COMMON GRIPES: PEOPLE THINK REFACTORING MEANS REWRITE

@GARYFLEMING

COMMON GRIPES: PEOPLE FORGET TO REFACTOR

@GARYFLEMING

COMMON GRIPES: RED – GREEN – REFACTOR ISN'T ENOUGH

@GARYFLEMING

TWIST: SELECTION AND NAMING

@GARYFLEMING

TWIST: SELECTION AND NAMING

@GARYFLEMING

MEDIUM

				4			7	
	8				3	4		
	7			9	8	5	6	
4	9	6				8		2
7					4			
5		1	7		2	9	8	
			3	8	1	2		
			4			6		3

MEDIUM

				4			7	
	8				3	4		
	7			9	8	5	6	
4	9	6				8		2
7					4			
5		1	7		2	9	8	
			3	8	1	2		
			4			6		3

PART 2: NOT NOW

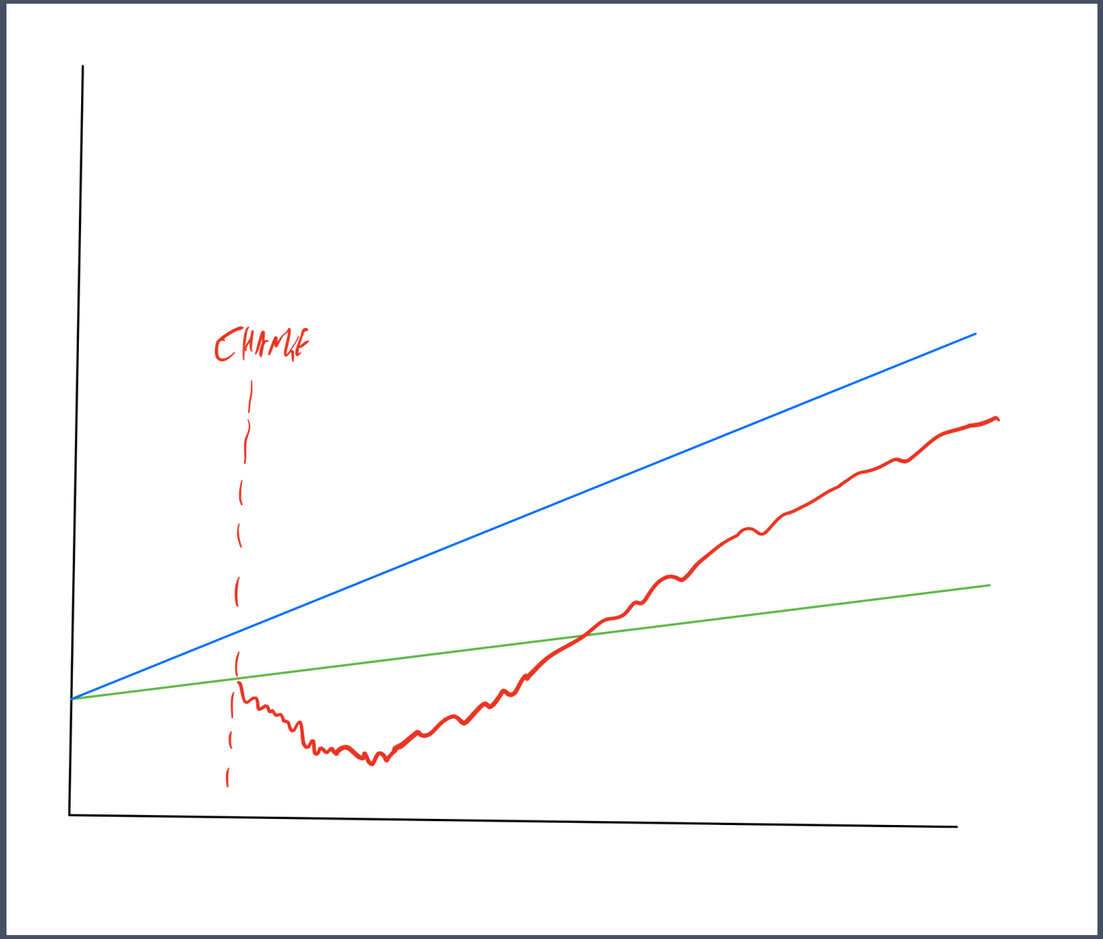
@GARYFLEMING

@GARYFLEMING

BUT, IT'S SLOOOOOW...

@GARYFLEMING

SATIR(ISH)



@GARYFLEMING

DON'T DO IT.

@GARYFLEMING

RON JEFFRIES DOESN'T TDD WHEN...

- ...IT'S SIMPLE/THROWAWAY
- ...THERE ISN'T A DECENT TOOL AT HAND
- ...THE OUTPUT IS VISUAL
- ...HE CAN'T THINK HOW TO TEST SOMETHING

RON JEFFRIES DOESN'T TDD WHEN...

- ...IT'S SIMPLE/THROWAWAY
- ...THERE ISN'T A DECENT TOOL AT HAND
- ...THE OUTPUT IS VISUAL
- ...HE CAN'T THINK HOW TO TEST SOMETHING

RON JEFFRIES DOESN'T TDD WHEN...

- ...IT'S SIMPLE/THROWAWAY
- ...THERE ISN'T A DECENT TOOL AT HAND
- ...THE OUTPUT IS VISUAL
- ...HE CAN'T THINK HOW TO TEST SOMETHING

RON JEFFRIES DOESN'T TDD WHEN...

- ...IT'S SIMPLE/THROWAWAY
- ...THERE ISN'T A DECENT TOOL AT HAND
- ...THE OUTPUT IS VISUAL
- ...HE CAN'T THINK HOW TO TEST SOMETHING

OTHER REASONS TO NOT...

- ...YOU PREFER LARGER METHODS, WITH LESS FOCUS
- ...YOU DON'T KNOW HOW TO KEEP THE TEST LOOP SMALL
- ...YOU DON'T KNOW HOW YOUR SYSTEM SHOULD BEHAVE

HARD

9				2				5
					3	9		1
	7	8						
4		7	1				5	
5			7		4			
		6		9			4	
3		4		7		2		9
		2		4				
	1		2					4

PART 3: WHAT NOW?

BECK'S FOUR ELEMENTS OF SIMPLE DESIGN

- PASSES THE TESTS
- REVEALS INTENTION
- NO DUPLICATION
- FEWEST ELEMENTS

WHAT DOES THIS ACTUALLY DO?

- PASSES THE TESTS
- REVEALS INTENTION
- NO DUPLICATION
- FEWEST ELEMENTS

REMOVE DUPLICATION AND IMPROVE NAMES IN SMALL CYCLES

@GARYFLEMING

ABSTRACT

OBJECT

INSTANCE

PROCESS CREATE

BUILD MANAGE

REQUEST

COMMAND

CONCRETE

GOOD NAMES

- ...AVOID UNNECESSARY AMBIGUITY
- ...ARE JUST CONCISE ENOUGH TO CONVEY THE RELAVENT INFO AND NO MORE
- ...USE THE SHARED DOMAIN LANGUAGE AS MUCH AS POSSIBLE.

TIME CHECK

@GARYFLEMING

A NEAT TRICK

Brian Marick @marick · Jun 15, 2019

One of those "trapped in a local maximum" situations is that programmer test frameworks *start* by assuming the fundamental assertion is "these two things are equal".

1/7

2 10 15

Brian Marick @marick · Jun 15, 2019

Instead, the fundamental assertion should be "apply this predicate to the actual value. If the result is false, the predicate knows how to report failures well." Equality testing is a special case.

@GARYFLEMING

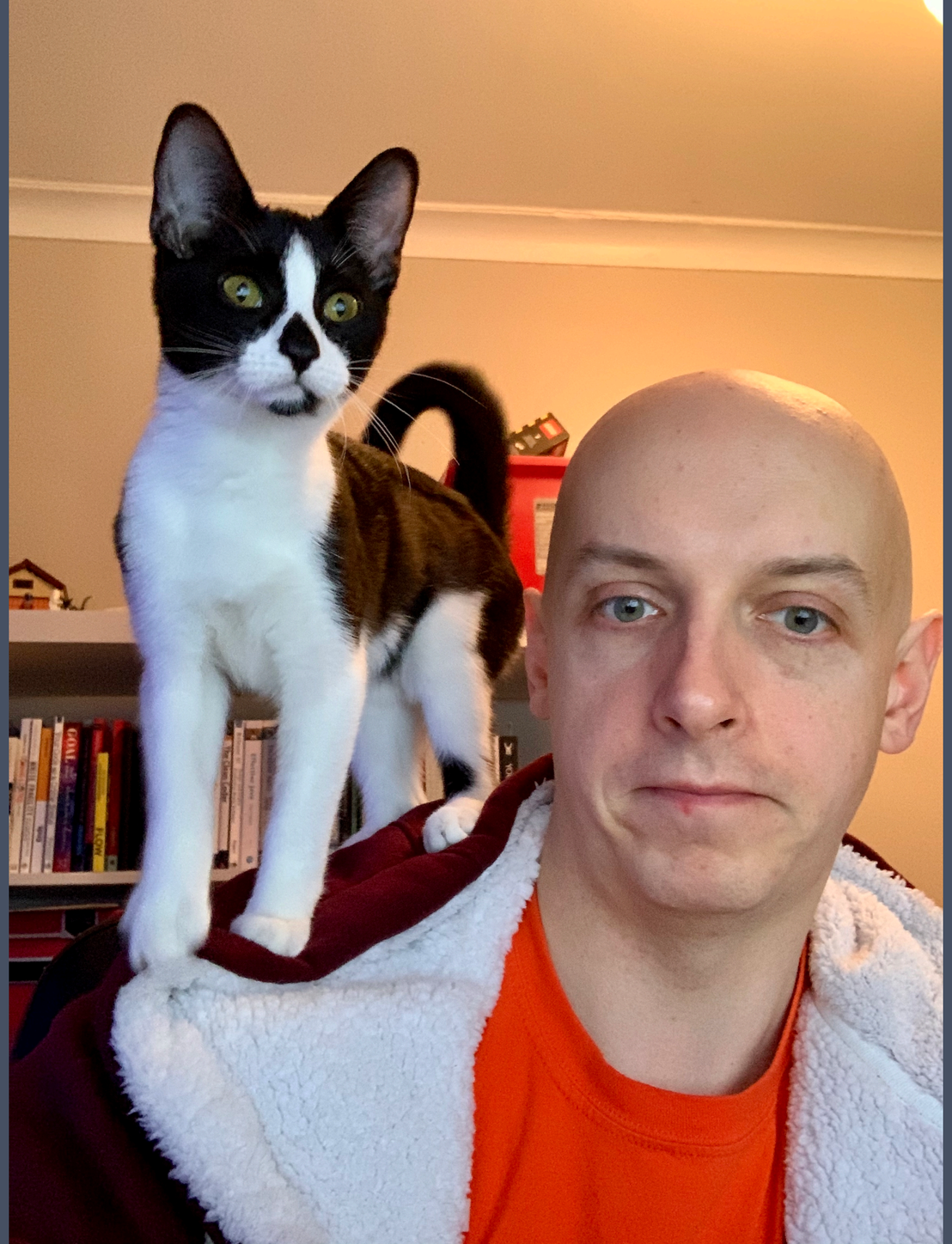
EXPERT

	3	5						6
				7		8		
		1			9			
9	2						7	8
	5						2	
3						5		
			5				1	
	9	4				2		
			6		7			4

EXPERT

- LEARNING
- STUDYING
- SEEKING NEW TOOLS
- TIME
- DELIBERATE PRACTICE

THANK YOU

@GARYFLEMING

BIT.LY/TDD-SHAME