

The 3 biases of software development

...and how to overcome them

How many CSMs do we have in Sweden?

2113 CSM's

How many CSDs do we have in Sweden?

Less than 2113

Less than 113

it's even less than 13...

5 CSD's

The same thing happen in the whole world

	CSM's	CSD's
Brazil	4.333	201
Germany	7.626	471
South Africa	1.442	5
Japan	8.038	400
Mexico	1.541	121

Developers and Scrum, how's it going?

It's failing, because agile is being used to micro-manage people



861



Reply

How did we get to this point?

Mario, the junior developer



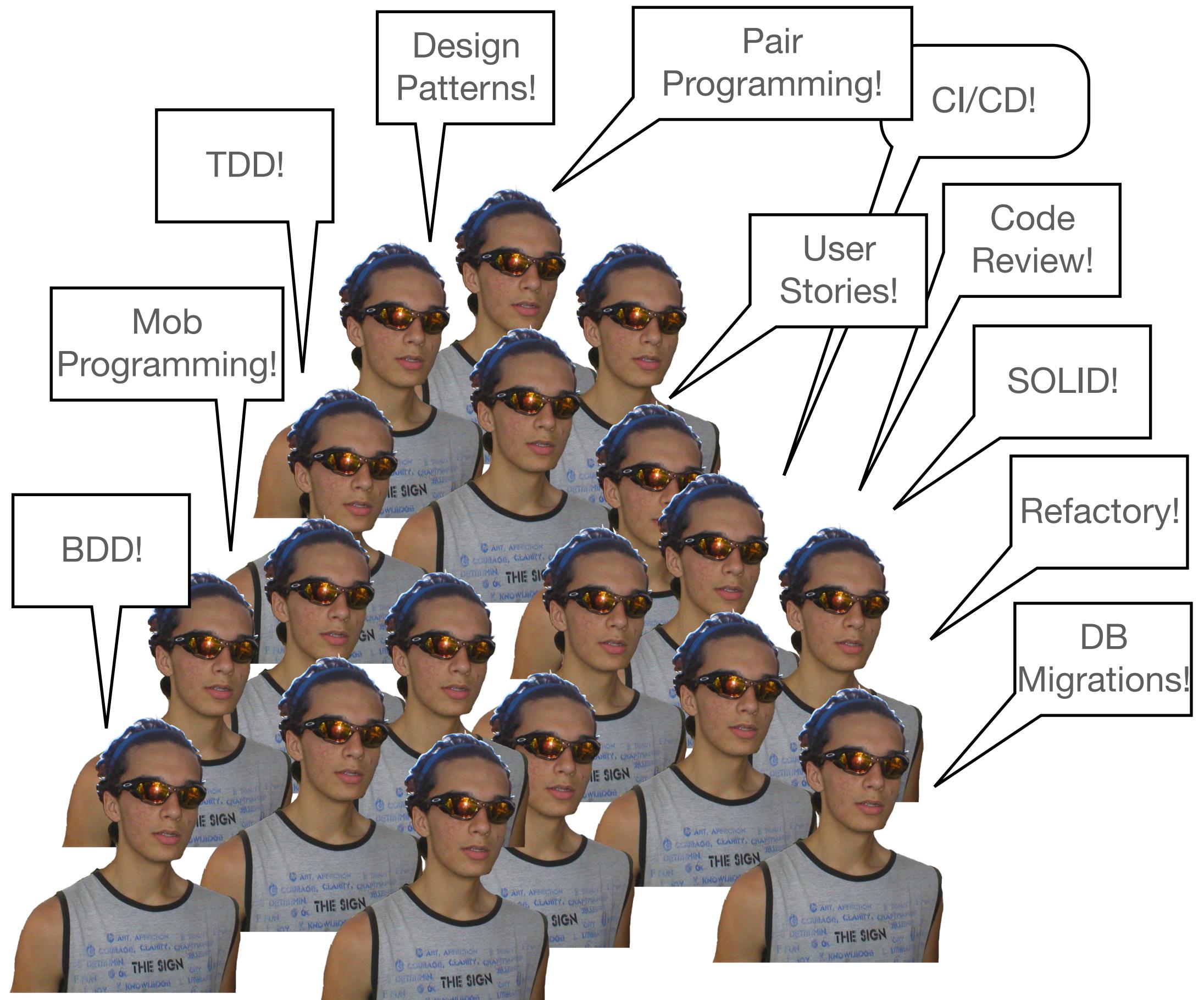
We need something that could help us with the constant changes the customer demands...

I mean... I would do ANYTHING to write less documentation and more code. And specially: waste less time.

I'll introduce the idea to my manager.

Agile Begins

Developers were annoyed by the traditional management models.



Whenever the customer asked for changes, they had to redo a lot of stuff. So they came up with technical solutions to minimize this waste.

They also wanted to waste less time in useless meetings and writing pointless documents.

So, they rebelled. A little.

The years passed...

Mario, the Tech Lead/Manager

Mario, the junior developer

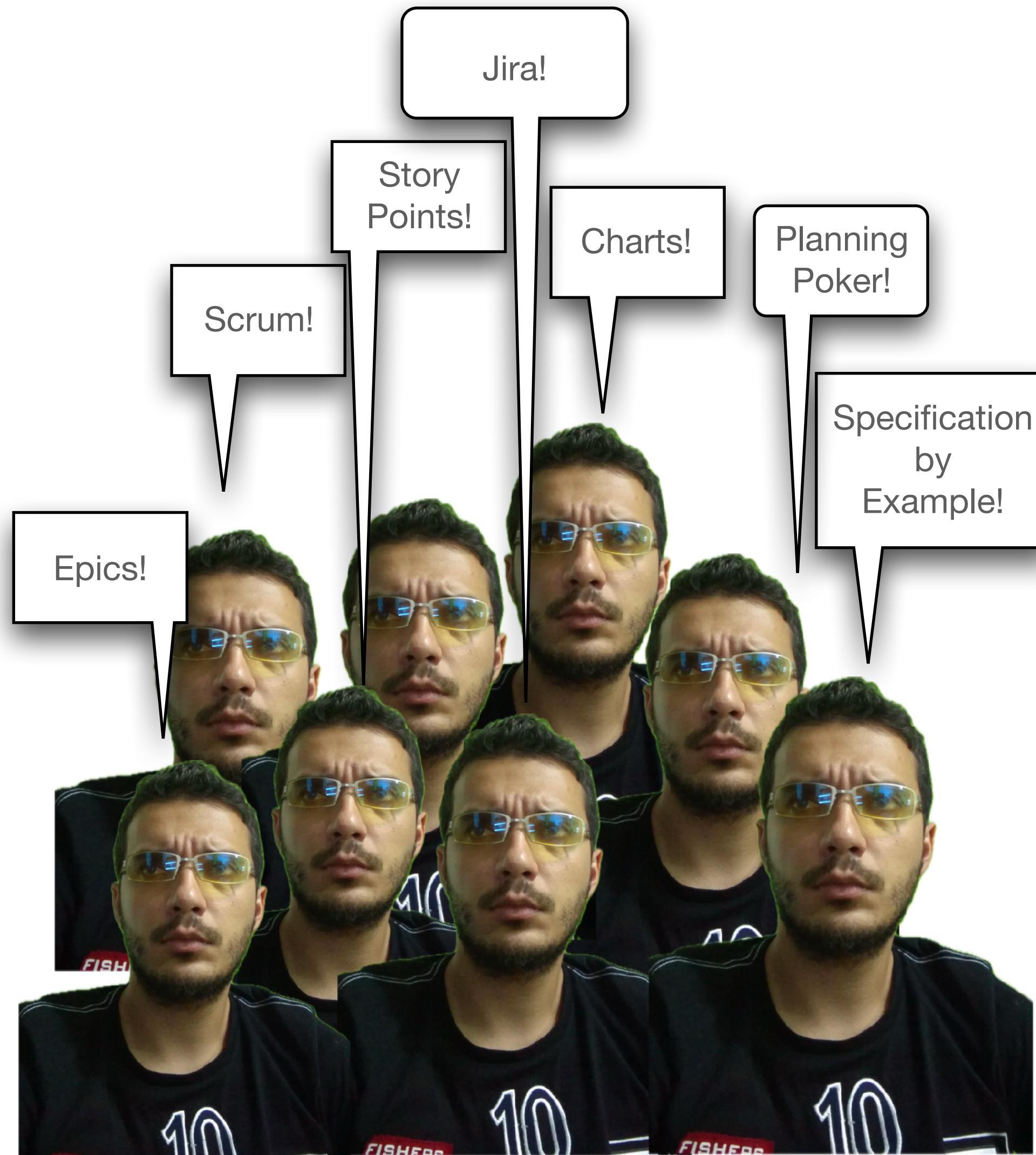


We still have problems with rigid contract models, but I still have to find out how to sell a flexible model to the customers.

I also don't know how could I demonstrate progress to both my stakeholders and my boss.

We need to figure that out.

Different challenges, different solutions



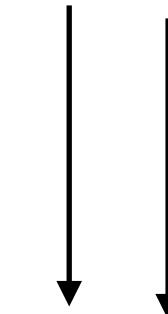
We improved the technical aspects that were hindering our ability to react to changes, and moved forward to the next challenge.

We started talking more about management, and less about the technical aspect of our work, because our current problems resided there.

We started to use Agile as substantive, and not as an adjective.

More years passed...

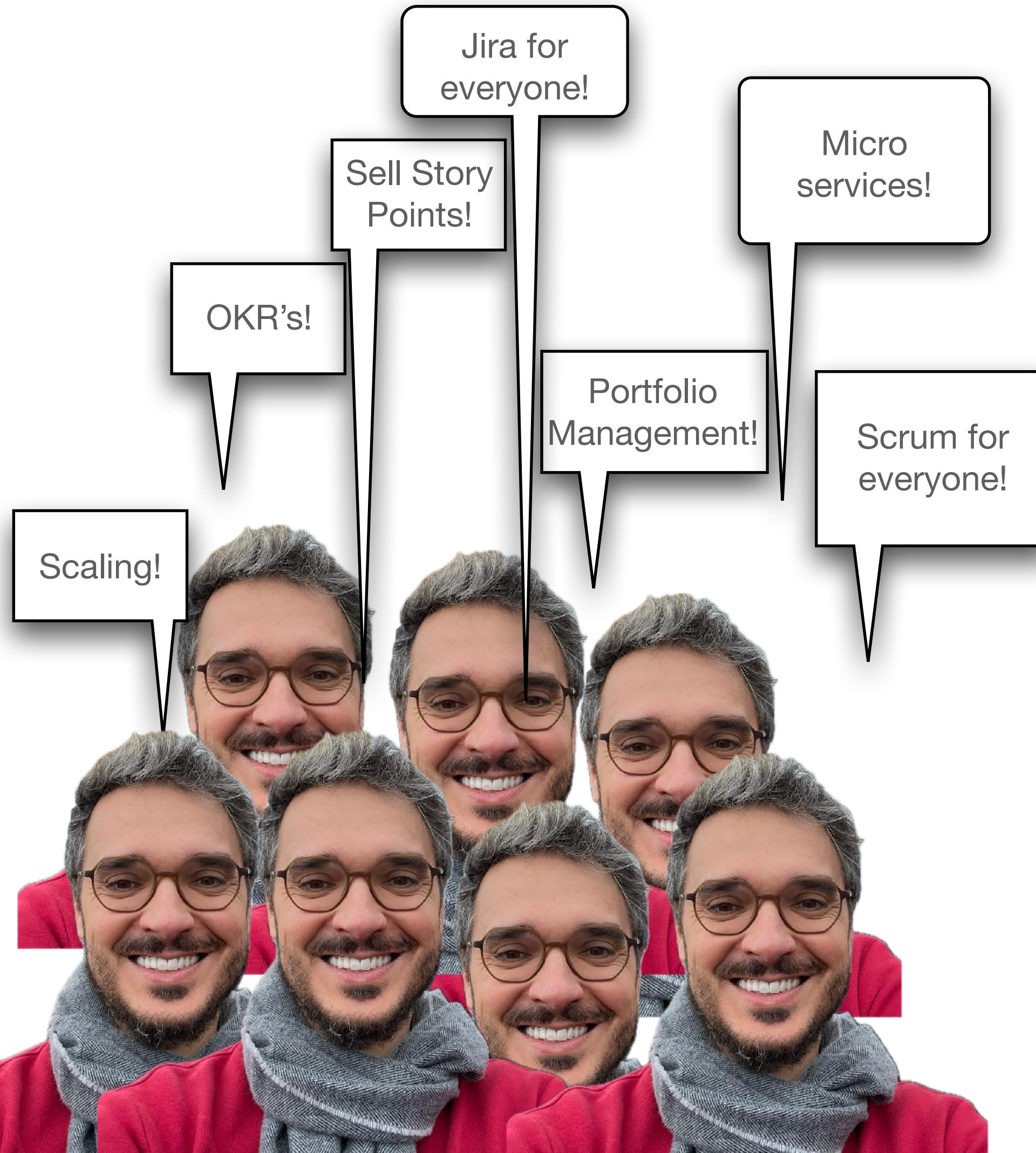
Mario, the Consultant / The C-Level



How could I help multiple teams that work on the same product to coordinate themselves?

How should I manage a portfolio of products while our business keep growing?

Agile: Far From Home



We were bringing the concepts we learned to the upper management.

We became upper management.

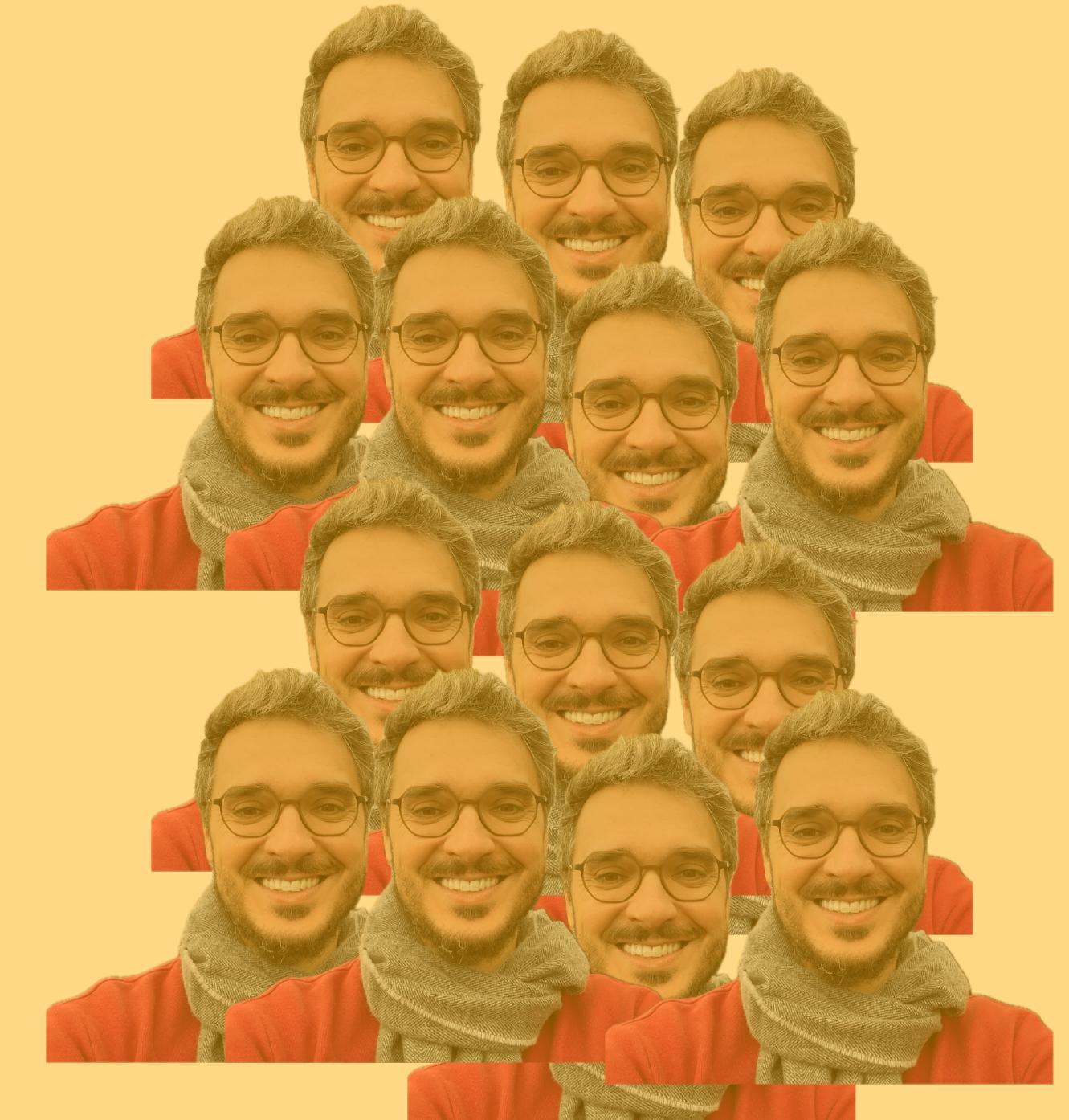
That part was ok, I guess.

The problem was thinking that the technical problems we first faced were gone forever.

The agile "blanket" became too short



Focus of agile conferences, conversations, and work

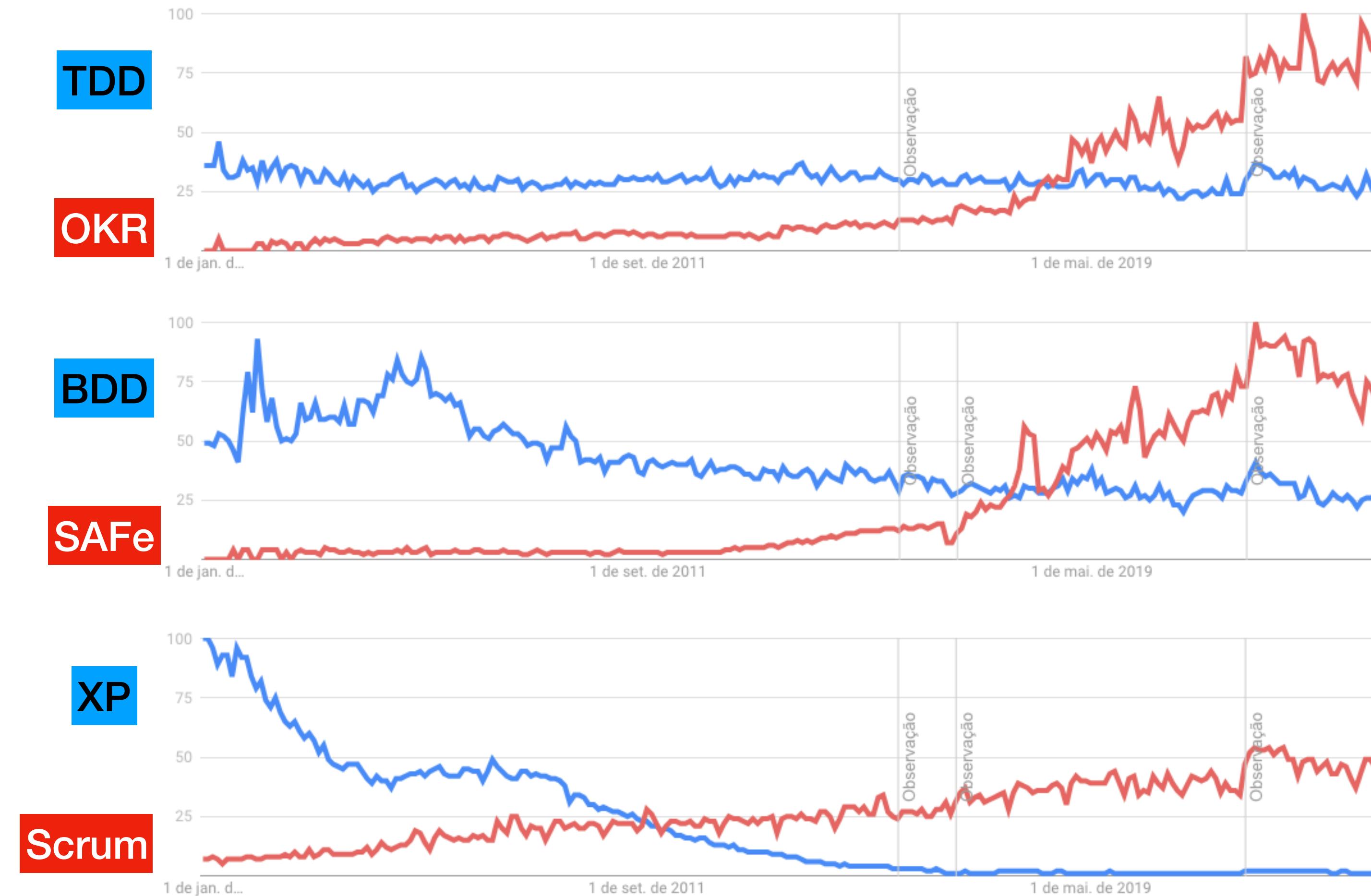


Technical People

Middle Management

Leadership Roles

We moved away from technical people...



Source: <https://trends.google.com> - Data comprised between 2004 and 2024

... and we need to fix that!

**Continuous attention to technical
excellence
and good design enhances agility.**

Source: some kind of manifesto I found on the internet

We'll do that by fixing 3 biases:

Stratification

Chronology

Modularity

I came up with these
complicated names so
I could look smart
during the presentation



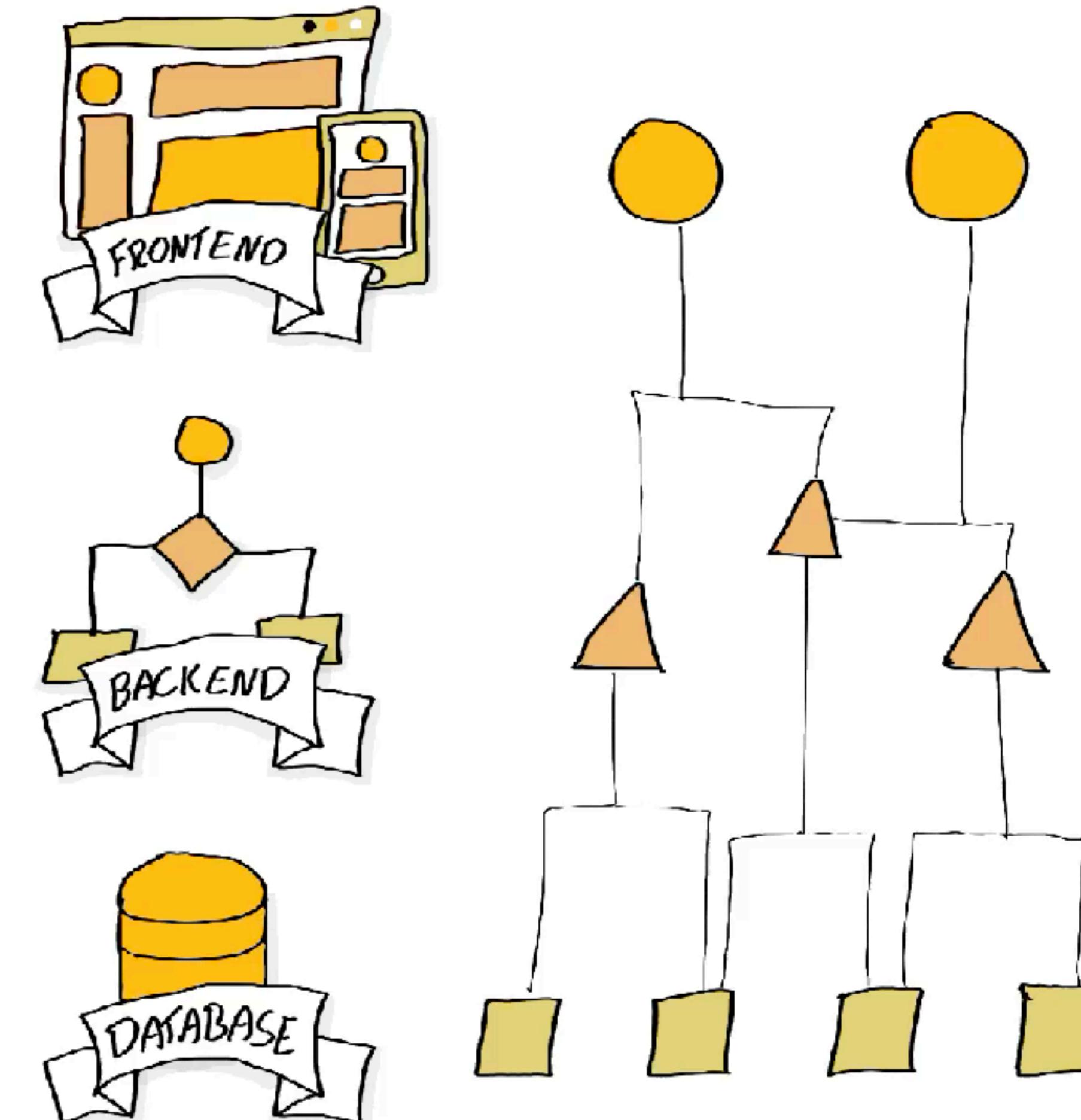
So, what is a Stratification Bias?

Software is often divided
in technical layers



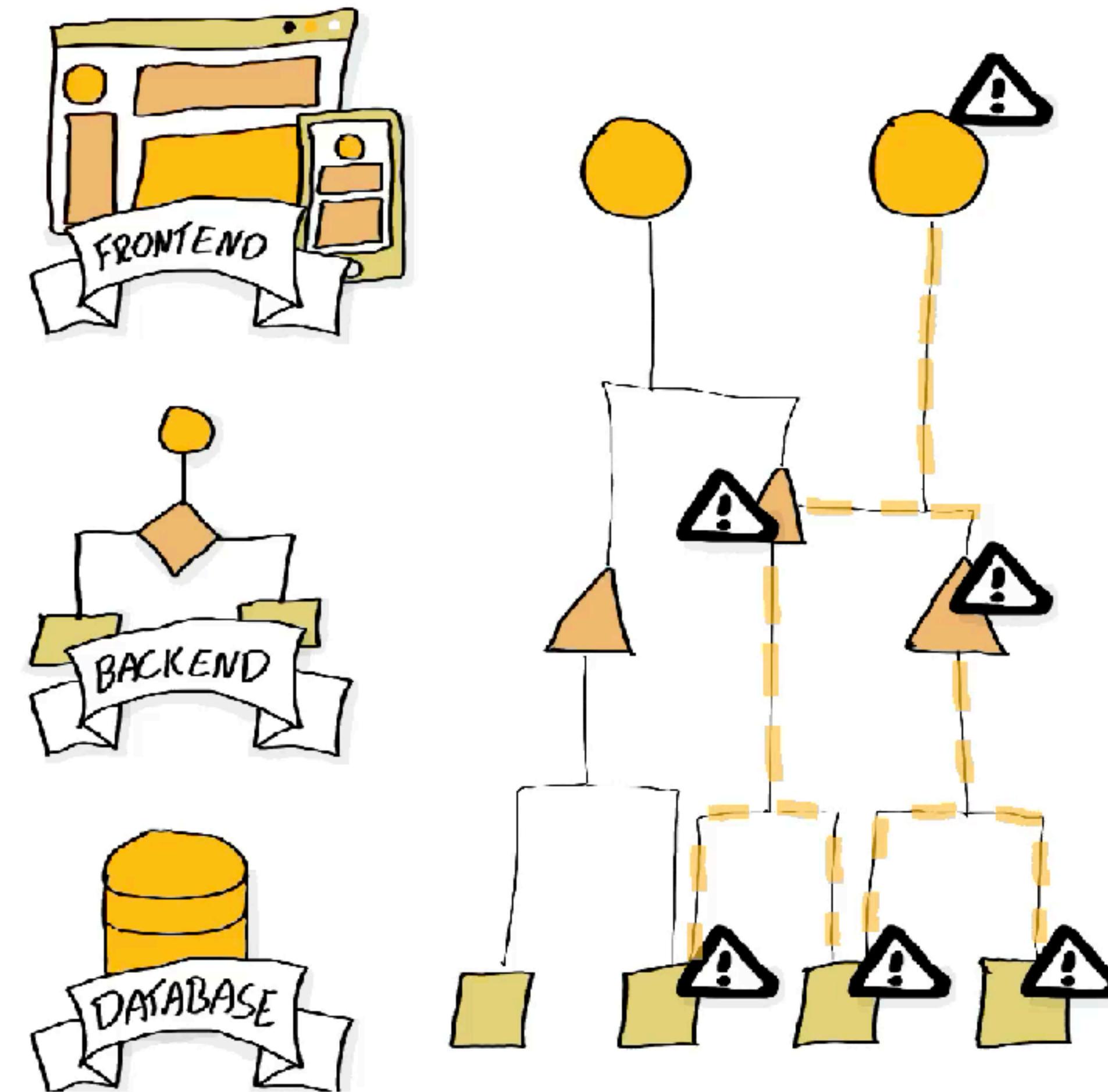
So, what is a Stratification Bias?

That might lead a team to pick a certain layer as a starting point, and move to next one only when they finish the first one



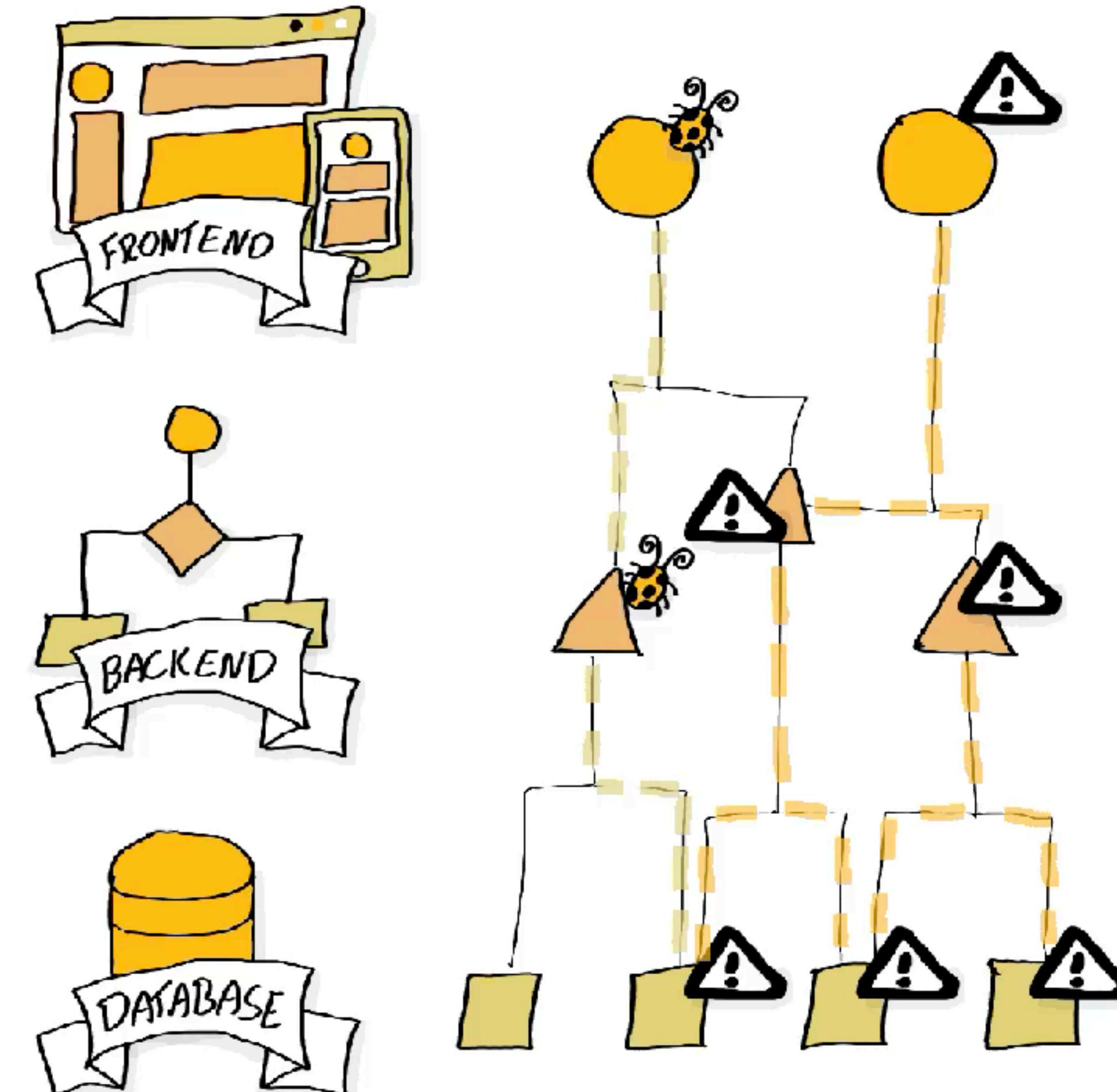
So, what is a Stratification Bias?

In this way, when a change is necessary, it often demands a lot of work



So, what is a Stratification Bias?

These chances also tend to generate many undesired side effects



How do we fix that?

Mario, the not so junior developer



This bias is a consequence of a technical limitation: if only the databases could grow along with the rest of the product...

Which means... I can solve it with a technical solution:

Database Migrations!

leaderboard

id	user_id	score	date
1	3	25	01-01-24
2	4	3100090	12-12-23
3	2	100	02-01-24
4	3	95	06-04-24

users

id	username	email	password
1	mario@.....	*****
2	zlatan@.....	*****
3	i_bergman@.....	*****
4	b_borg@.....	*****

words

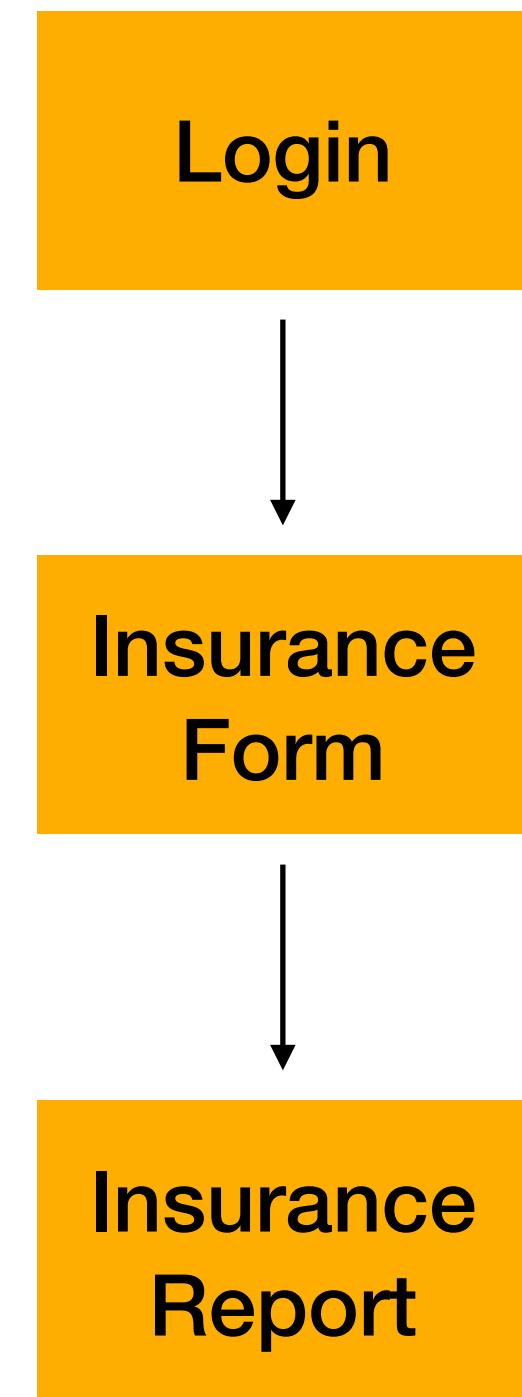
id	language	word	difficulty
1	eng	agile	easy
2	eng	jazz	hard
3	eng	crown	hard
4	swe	herrgård	medium

Feedback Received:

Well.. I don't have
much too say until I
can see the actual
product...

The Chronology Bias

It happens when the order of the user flow is used as a constraint for the development process



Chronology Bias: The Hangman Game, pt 1

Mario, the Tech Lead



Since there's no way to create a Hangman Game without establishing a word, we **MUST START THERE**.

We need your feedback on what we've done so far:



Find a random word

Accept user guesses

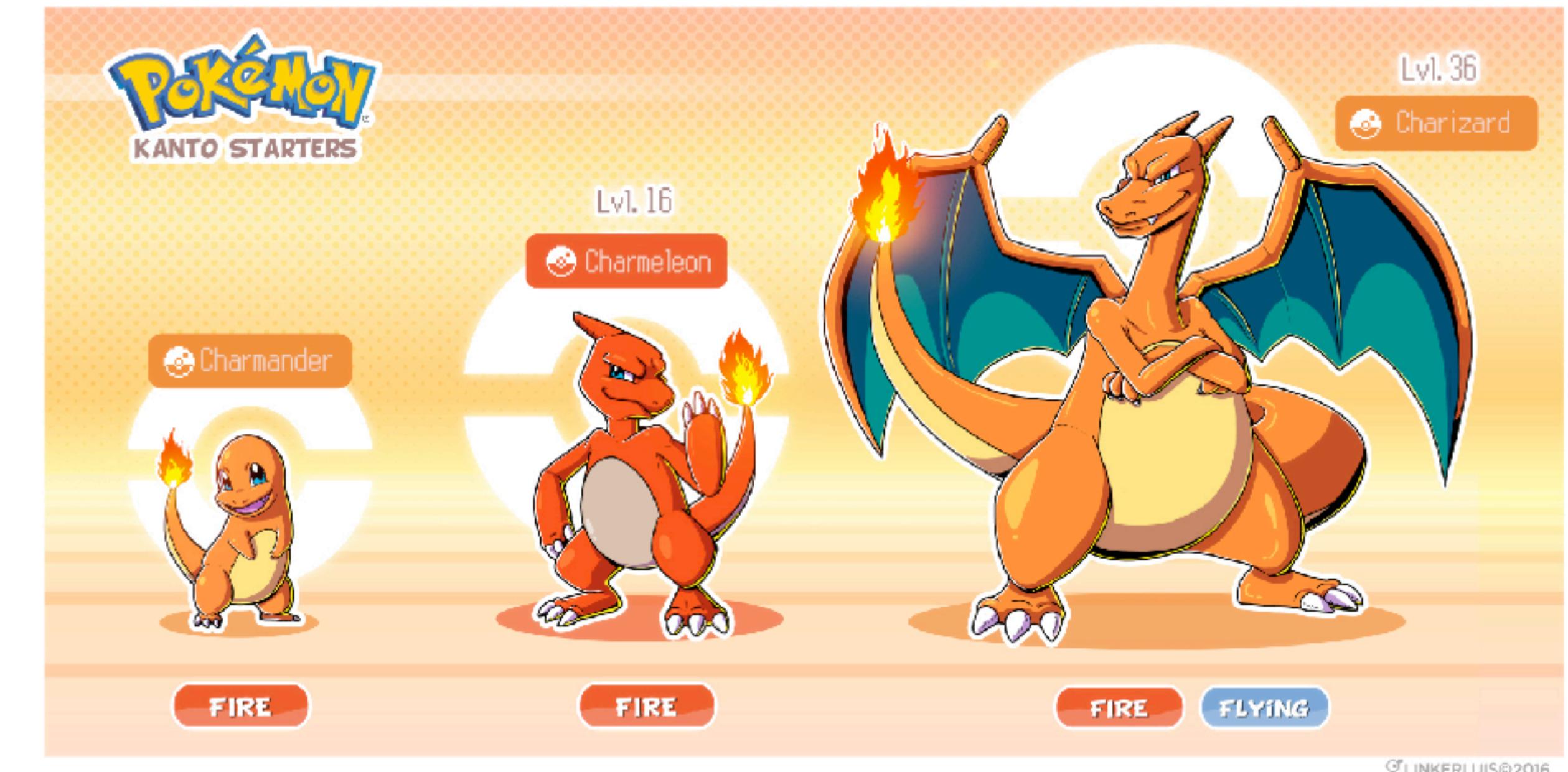
Show Results

Feedback Received:

We should not accept
words with less than
5 letters

The Chronology Bias

If we work this way, our software will evolve just a like a Pokemon: in a predictable and linear way.



The younger me would love this reference!

“Younger me...”
Come on, you still play it.



Please Nintendo, don't sue me.



Breaking the Chronology Bias

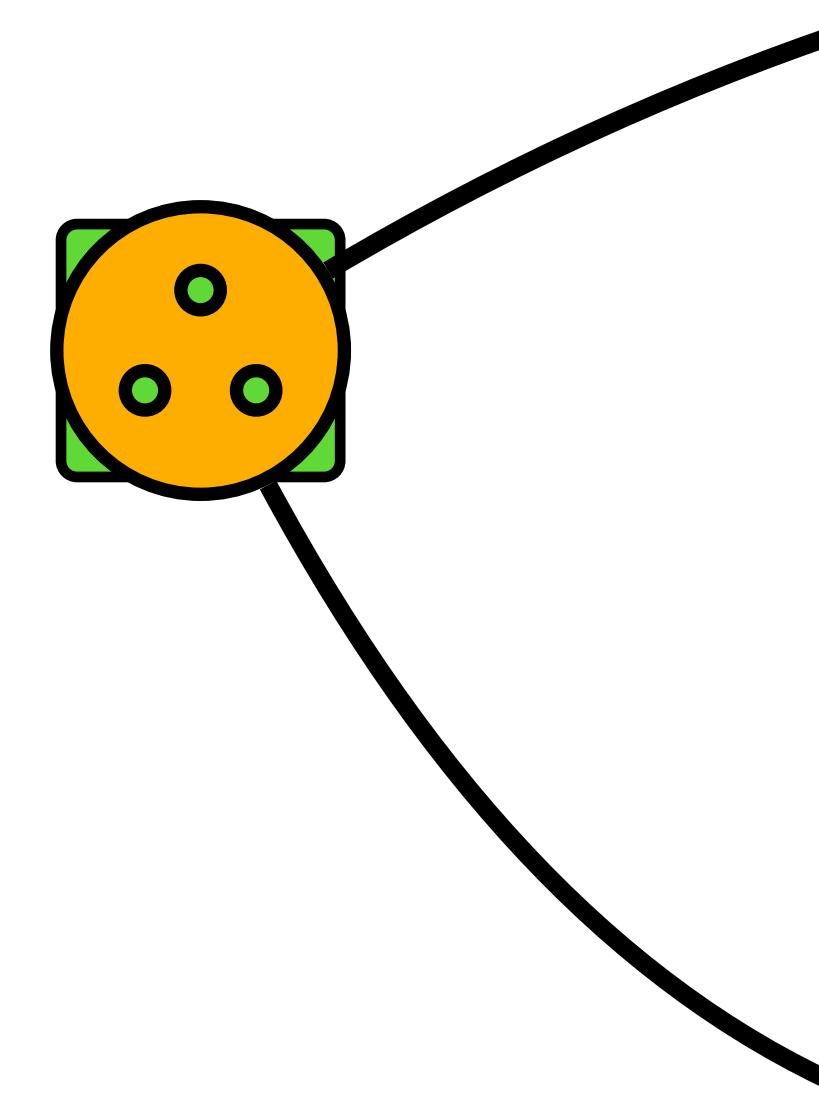
```
def new_game(player \\ "Unknown") do
  word = Dictionary.get_word()
```

So, to start a game, I call
this function and it returns
me a word

```
defmodule Dictionary.ImprovedDictionary do
  @api_key_name "NINJA_API_KEY"
  @api_endpoint "https://api.api-ninjas.com/v1/randomword"

  def get_word do
    api_key = System.get_env(@api_key_name)
    Req.get!(@api_endpoint, headers: %{"X-Api-Key" => api_key}).body["word"]
    |> Enum.at(0)
    |> String.upcase
  end
end
```

Breaking the Chronology Bias



Breaking the Chronology Bias

```
def new_game(player \\ "Unknown") do
  word = Dictionary.get_word()
```

So, as long as we don't
change the interface, we can
postpone this development

```
defmodule Dictionary.ImprovedDictionary do
  @api_key_name "NINJA_API_KEY"
  @api_endpoint "https://api.api-ninjas.com/v1/randomword"

  def get_word do
    api_key = System.get_env(@api_key_name)
    Re- defmodule Dictionary.PoorDictionary do
      ▷
      ▷
    end
    end
  end

  def get_word do
    "ELEPHANT"
  end

end
```

Chronology Bias: The Hangman Game, pt 2

Mario, the Tech Lead



Ok, so for now the word will always be “ELEPHANT”, because we haven’t implemented the Dictionary yet.

What can you tell us about the product?



Find a random word

Accept user guesses

Show Results

Problem fixed, right?

Because we are now able to break dependencies, we have more options to choose from when we think about what to do next



Feedback Received:

Show the result when
the game ends

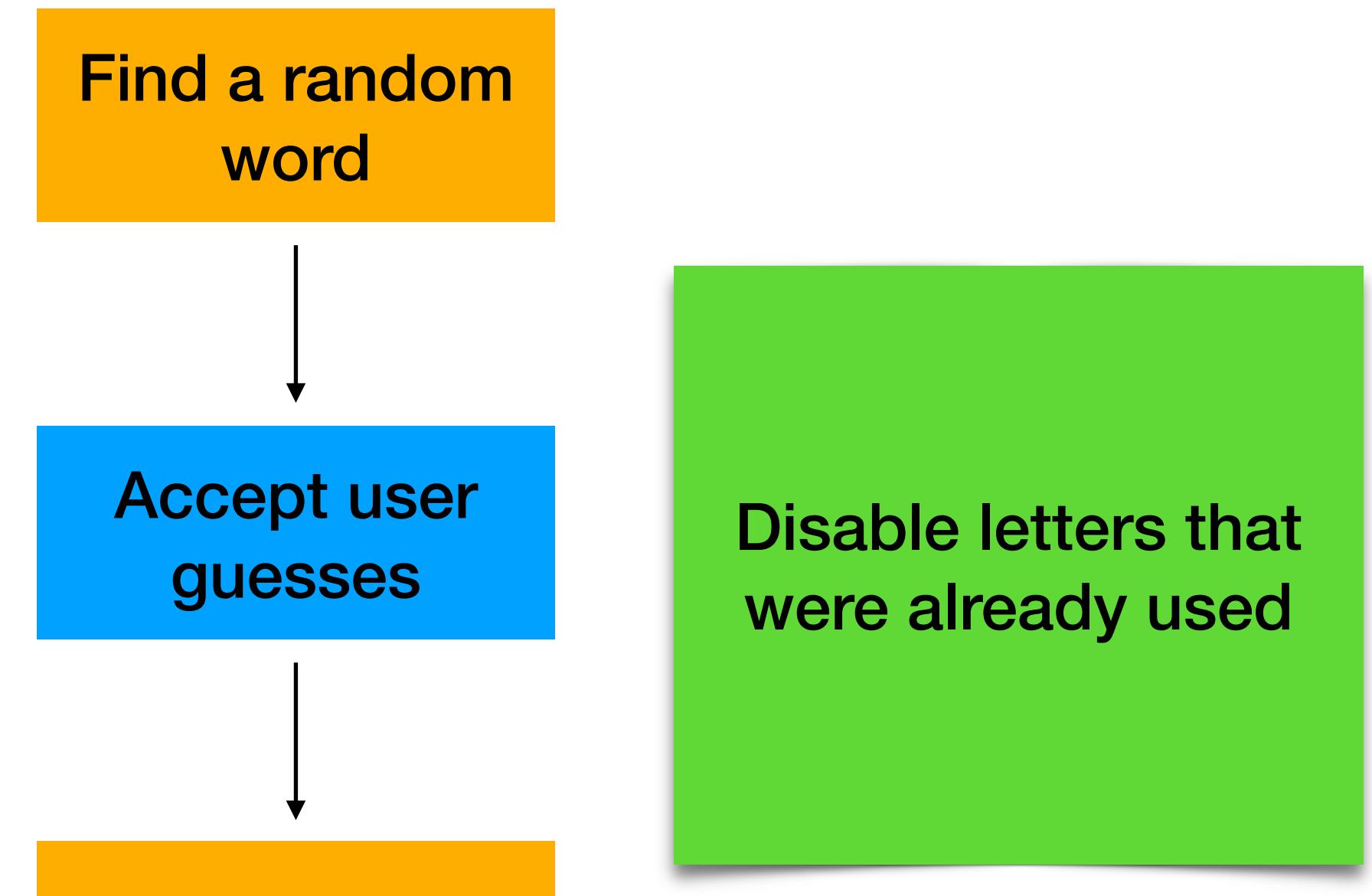
Add a time limit for
the game

Calculate a score
based on the time/
lives spent

Disable letters that
were already used

The Modularity Bias

It makes us think that we must finish a part of the software before moving to the next one



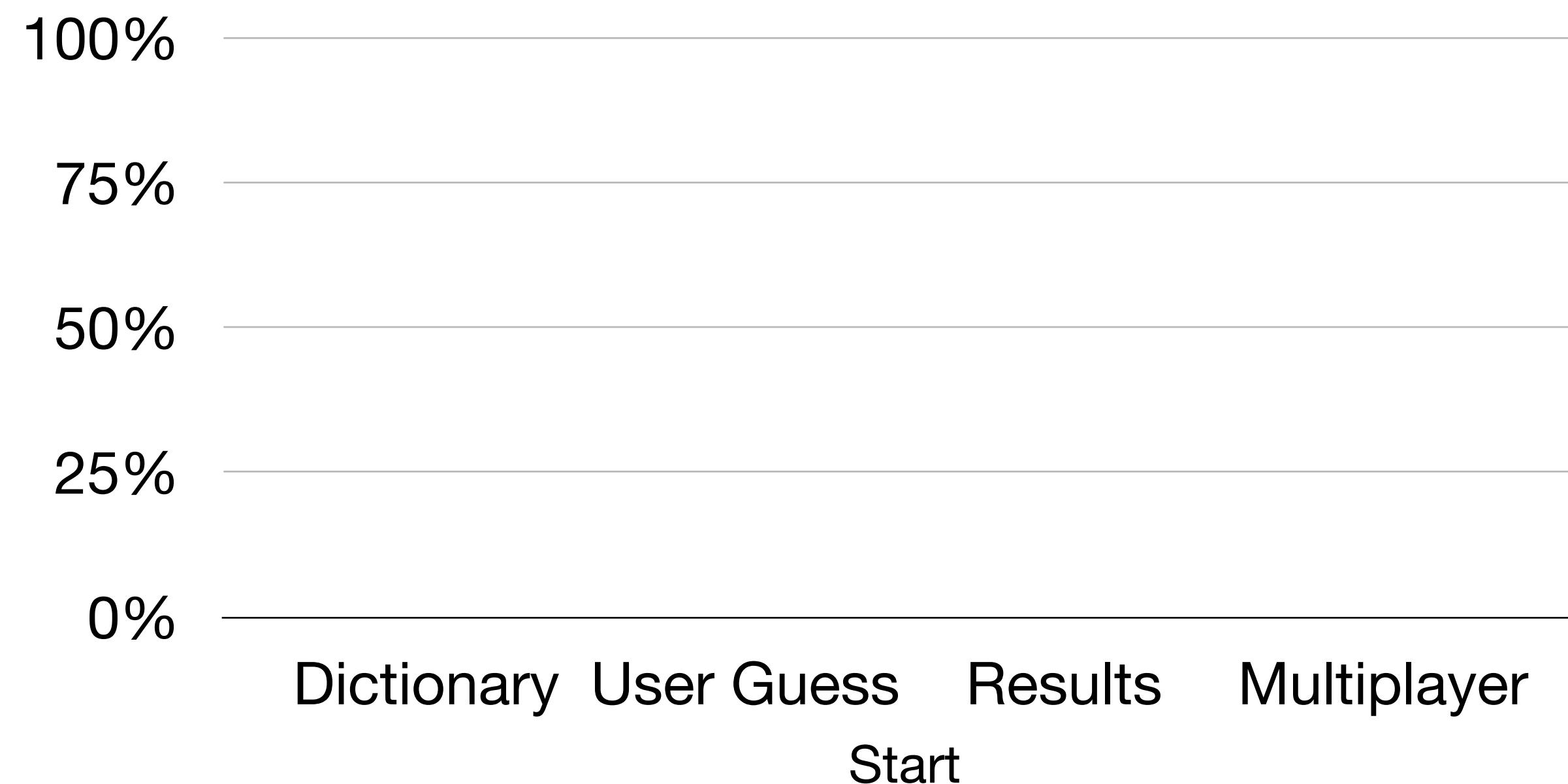
What??
How come it's not finished?



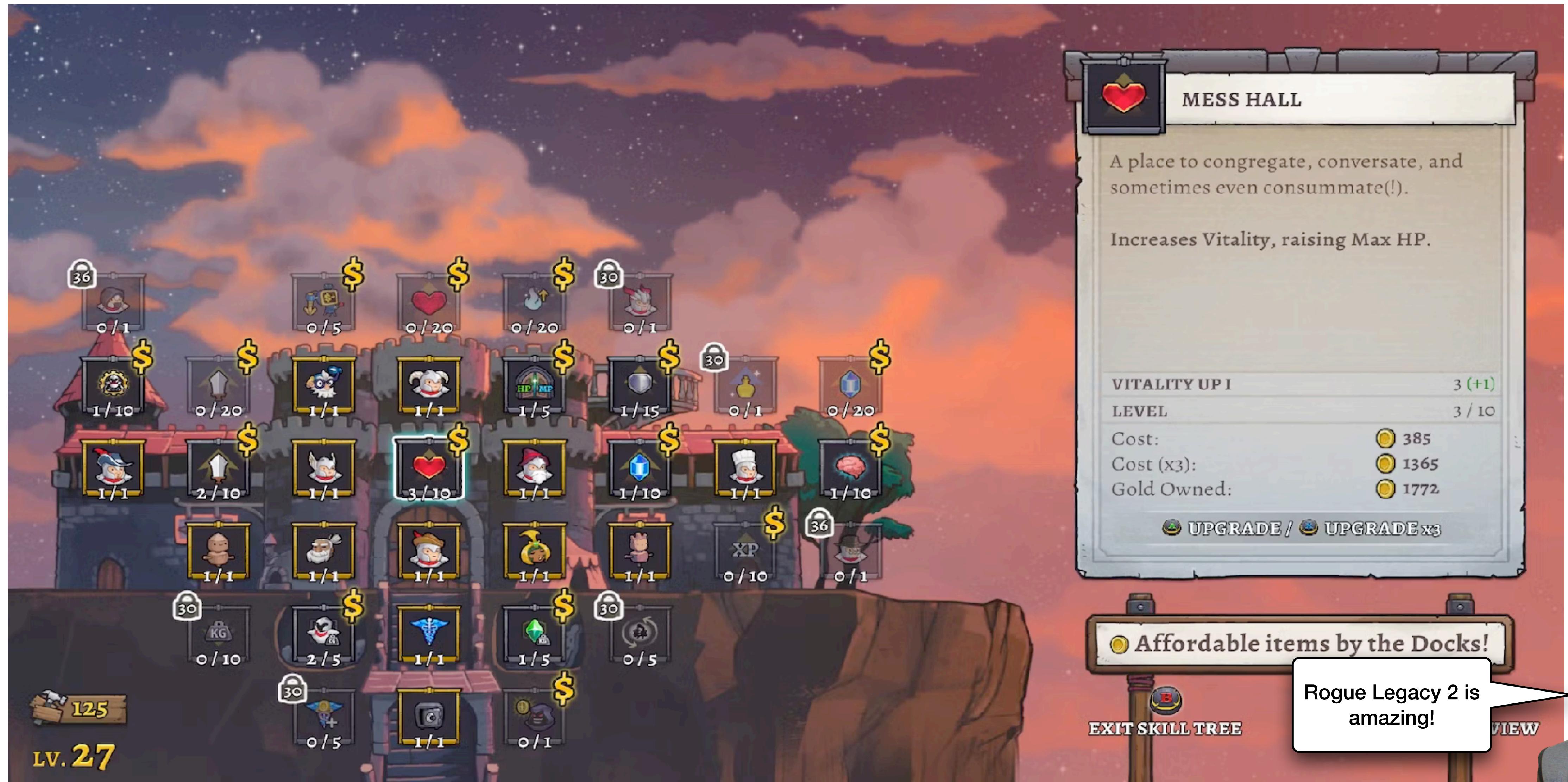
The Modularity Bias

In a modular approach, we have less options as we keep moving forward.

It also means that maintenance and change costs are rising quickly.



Breaking The Modularity Bias



Breaking the Modularity Bias

If the developers have the **freedom** and the **ability** to see beyond the “modules”, they’ll always provide more options to the Product Owner

Dictionary User Guess Results Multiplayer
Start

I don't even know what a 100% is!



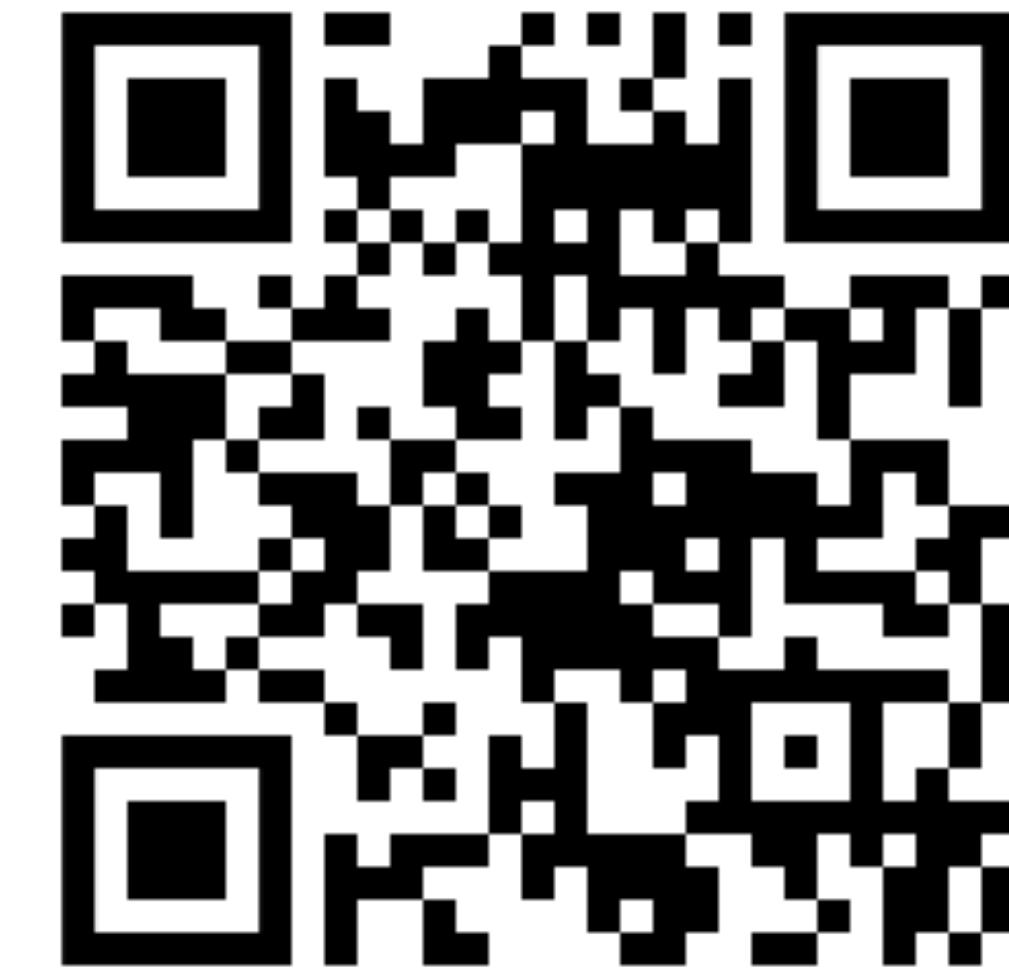
Chronology Bias: The Hangman Game, pt 3

Mario, the Tech Lead



Ok, so nothing is completely done, but I think we can maximize feedback value with what we have in hands.

What can you tell us about the product?



Feedback Received:

Dictionary

Game is too hard, it would be nice if the words belonged to a certain theme

Hint Mechanism (?)

We need a way to provide hints so the game doesn't bore me when the word is a hard one

Game Mode

I wanna see the top scorers and have my name there

Multiplayer

We could create a new way to play: 2 words at once, like they did with that game "wordle"

We could have an infinite game mode, where I keep getting words until I lose all my lives

Breaking the Modularity Bias

How could we implement a Hint Mechanism?

Make a correct letter flash on the screen

Disable a wrong letter every 5 seconds

Provide a contextual hint

Or... we could have a thief running across the screen carrying a bag with the hint!

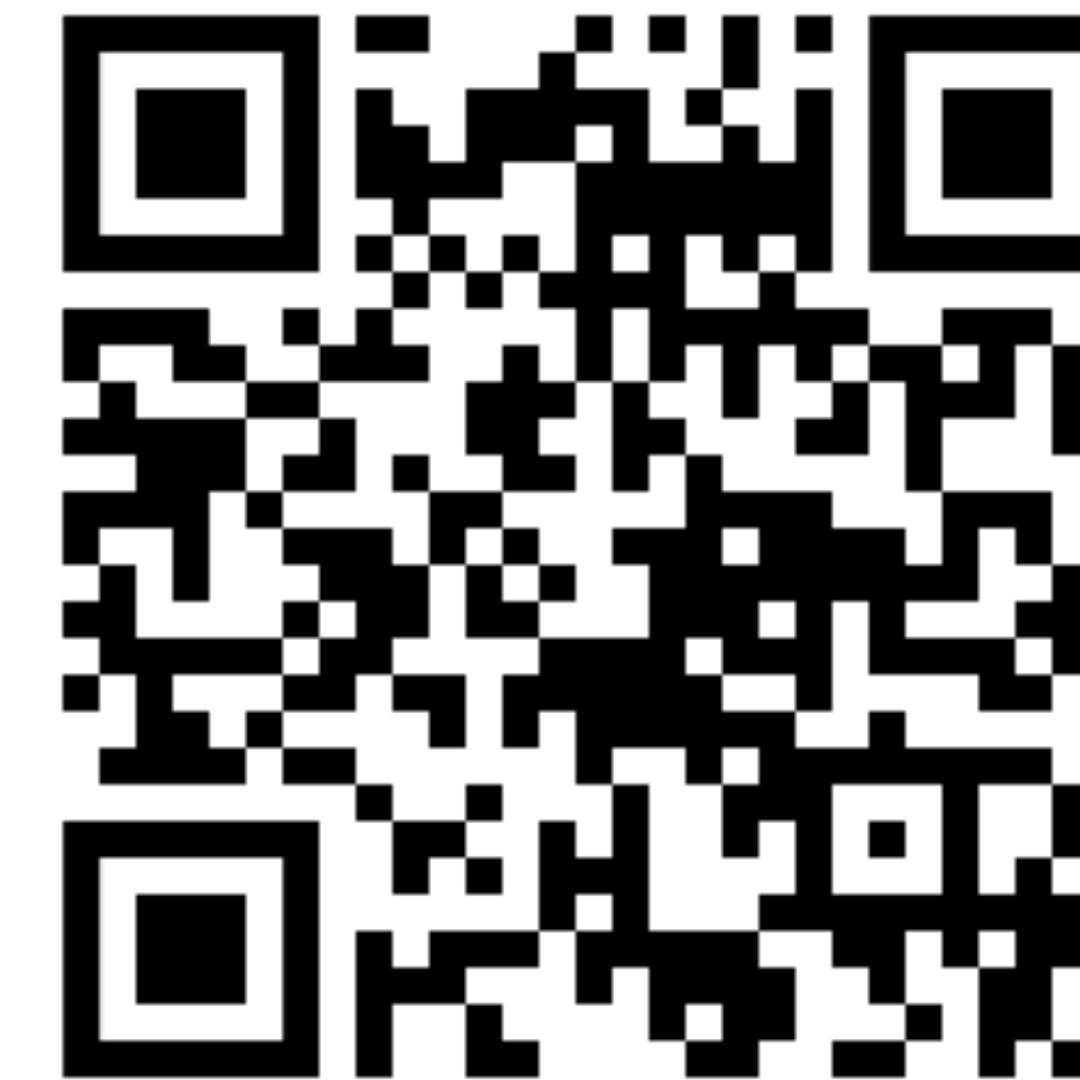
If you catch him, one letter is revealed!



Chronology Bias: The Hangman Game, pt 4

So.. We created a Hangman Game without the Hangman.

We can still do it, but... should we?



The fundamental truth about software development is that we might be wrong

Kent Beck

We are exposed to a lot of other software development biases.

And I'm certain there's a technical solution just around the corner.

Your developers will find it.

If you let them.

And help them.

How to break the biases

Stratification Bias

- * Database Migrations
- * Behavior Driven Development
- * Impact Mapping

Chronology Bias

- * Interfaces / Protocols
- * Design Patterns: Factory, Strategy
- * Mocks

Modularity Bias

- * Specification by Example
- * Test Driven Development
- * Hamburger Method

Mário Melo

- Software Developer since 2004
- Certified Scrum Trainer ®
- Brazilian, living in Bologna
- Soccer Player (amateur)
- Vegan Cook (novice)
- Musician (journeyman)
- Vim and Elixir user
- <https://mariomelo.com>

