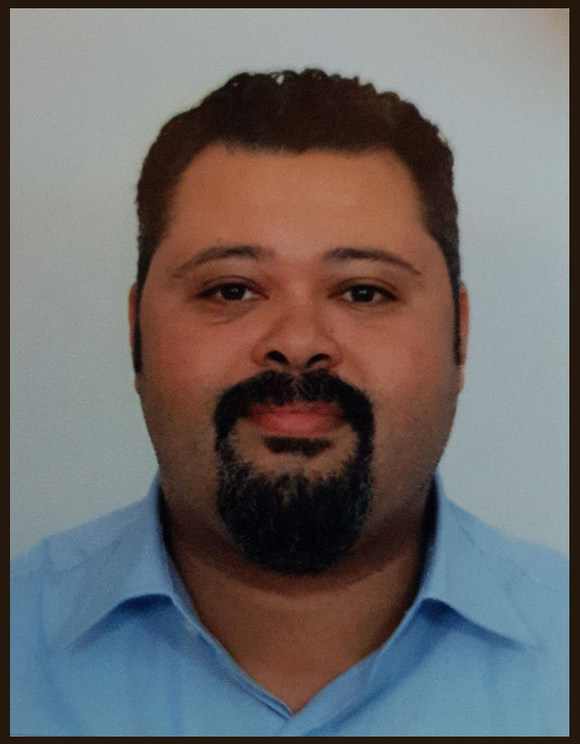
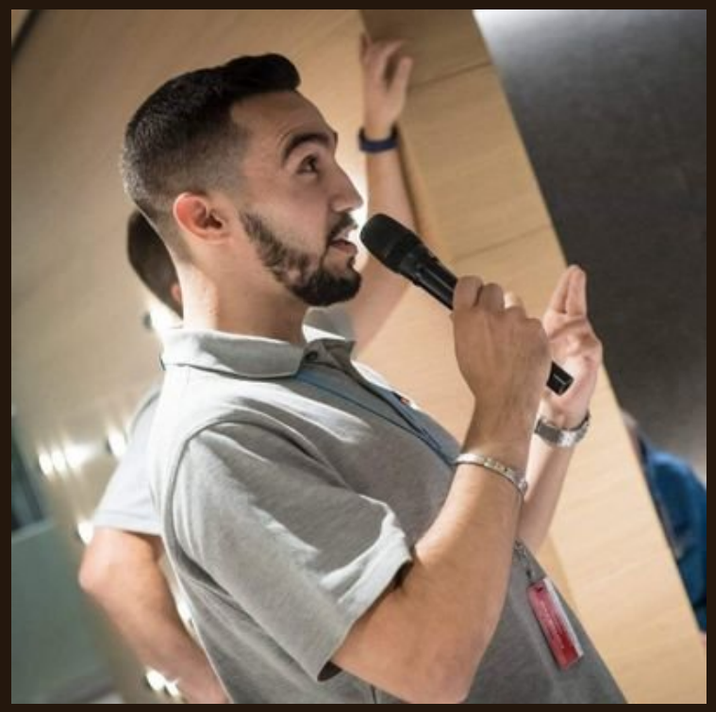
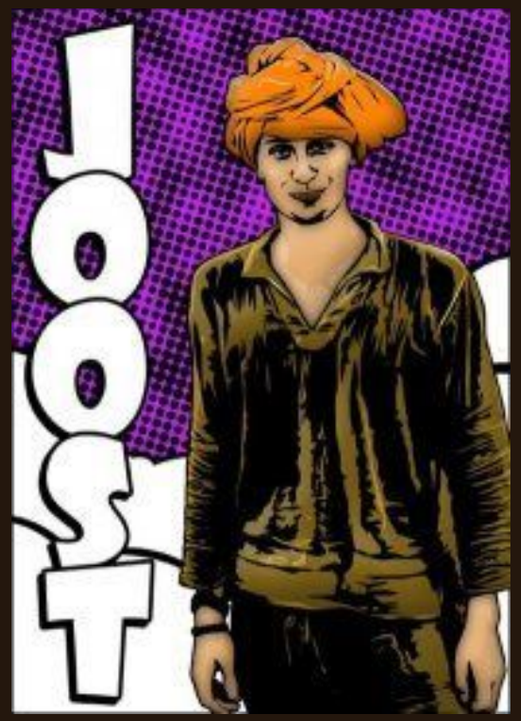
Coderetreat
17-11-2018

Utrecht

Who







What do we learn?

- Design skills
  - minute-by-minute decisions we do daily
- Delete code

What is (a typical) coderetreat

- 1 day of coding
- Conway's game of life
  - (simple rules, learning to program - not the domain)
- X pairing partners
- ~5 constraints
- Language agnostic

What to expect today?

- Practice the fundamentals of programming
  - OO, functional, clean code, TDD, refactoring
- An intense day of coding
- And a lot of fun

Structure of the day

- 15-30 min intro
- 1 session of 45' + 15’ retro
- Coffee break
- 1 session of 45min + 15min retro
- Lunch + something cool!

Structure of the day

- 1 session of 45' + 15’ retro
- Snack break
- 1 session of 45’ + 15' retro
- Coffee 15’
- 1 session of 45’ + 15' retro
- 30’ RETRO

principles

- We are all valued - everyone counts
- We are all here to learn.
- Focus on practice & experiment
- Try not to finish the problem
- DELETE YOUR CODE after each session
- Have Fun

prerequisites

- Computer
- Coding environment
- Test environment
  - unless you want to start by creating one of your own
- Source control (git, SVN, mercurial, etc.)

The 4 elements of simple design

1. Passes its tests
2. Maximizes clarity
3. Minimizes duplication
4. Has fewer elements

1. Passes its tests

Not “automated tests”

About correctness and verification.

Faster feedback cycle is better.

“If you are to ask how fast your test suite should be, it should be faster”

2. Maximizes clarity

How quickly can you find the part that should be changed?

A source of code smells - if it is hard to give an expressive name, maybe the (unit) is doing too much

3. No Duplication

Not about code, but knowledge.

“Every piece of knowledge should have one and only one representation”

4. Small

Code that is no longer used?

Duplicate abstractions - could I combine

Duplication in behaviour -> sign of a missing Abstraction?

Conway's game of life

Having an **infinite 2D orthogonal** universe. Being given an initial generation called a **seed**. The following rules are applied **simultaneously**

The rules

A live cell having less than 2 live neighbors dies

A live cell having 2 or 3 live neighbors lives

A live cell having more than 3 neighbors dies

A dead cell having 3 neighbors becomes alive

First session

- Find a pair
- Choose programming language
- Setup the environment
- Make a decision on how to start
- We'll start in 5 minutes.
- https://github.com/thodorisbais/coderetreat

Why coderetreats?

- Learn through pairing
  - … out of comfort zone
- deliberate practice
- experiment
  - ...safely in safe environment

2nd session – strict ping pong

Ahmed?

2nd session – retro

And then lunch!

3nd session - no talking

Joost?

3nd session - retro

- … (does this imply “no talking”?)

Snack

- For 15 min

4th session –
4 statements per method

- Ahmed?

4th session - retro

5th session - Legacy

Joost?

5th session - Retro

Closing circle

- What, if any, did you learn today?
- What, if any, did surprise you?
- What, if any, will you take to work with you ‘tomorrow’?

Thank you

- Thanks CodeSquad!