Functional Reactive Programming

With Frappuccino

@steveklabnik

Thank you!

Thank you!

Thank you so much, Athens.

Dad

Frappuccino

Irresponsible Ruby

Kill Your Idols

Irresponsible Ruby

What is #irresponsibleruby?

Irresponsible Ruby

Ruby code that you wouldn't deploy at your day job because it's just too interesting or creative.

Irresponsible Ruby

- Neat hacks

- Weird metaprogramming

- Unusual techniques

- No Unit Tests

Creativity

Creativity

"A letter from _why"

Creativity

I do not write tests for my code. I do not write very many comments. I change styles very frequently.

Creativity

And, most of all, I shun the predominant styles of coding because that would go against the very essence of experimentation.

Creativity

In short: all I do is muck around.

Creativity

I admire programmers who take risks. They aren't afraid to write dangerous or "crappy" code.

Creativity

If you worry too much about being clean and tidy, you can't push the boundaries.

Creativity

Until an asteroid,
_why

Creativity

Steve Klabnik · 2 years ago

As someone who maintains _why's code these days, please, write some tests.

6 △ ▽   Reply   Share ›

Creativity

In the end, _why was right.

But so was I.

Two Modes

Programmers produce things.
We're builders.

Two Modes

"Linguistic Relativity"

Two Modes

- Responsible Mode

- Irresponsible Mode

Two Modes

- TDD Mode

- "Exploratory Mode"

Two Modes

We need both!

Shoes Introduction

Shoes Introduction

- App

Shoes Introduction

- Button

Shoes Introduction

- Alert

Shoes Introduction

- Para

Shoes Introduction

- edit_line

Shoes Introduction

- background

Shoes Introduction

- hide

Shoes Introduction

- animate

FRP Introduction

FRP has two parts:

- Functional

- Reactive

FRP Introduction

Functional programming: functions as a first-class concept.

FRP Introduction

Reactive Programming: data flows as a first-class concept.

FRP Introduction

FRP: using first-class functions to manipulate data flows

FRP Introduction

Data flows?

FRP Introduction

```
a = 5
b = a + 6
b # => 11
a = 6
b # => 11
```

FRP Introduction

```
a := 5
b := a + 6
b # => 11
a := 6
b # => 12
```

FRP Introduction

- "time varying value": "behavior" or "signal"

- "events": occurs many times over time

- "switching": react to events

FRP Introduction

FRP Introduction

Frappuccino Introduction

- Streams

Frappuccino Introduction

- Events

Frappuccino Introduction

- map

Frappuccino Introduction

- inject

Frappuccino + Shoes

- Button sample

Frappuccino + Shoes

- Circle Example

Frappuccino

The most irresponsible code
I've written lately.

Go Forth And Be Irresponsible

- Be Creative.

- Do 'bad' things.

- Let's have fun with Ruby!