

開発者は仕事で リーダブルなコードを 書けるのか？

須藤功平

株式会社クリアコード

Developer Migration 2013
2013/03/02

対象

開発者

リーダーダブルな
コード？

例1: リーダブル?

```
class Person
  def initialize
    @mutex = Mutex.new
  end
  def name=(name)
    @mutex.synchronize do
      @name = name
    end
  end
end
```

例2: リーダブル?

```
class Person
  def initialize
    @mutex = Mutex.new
  end
  def name=(name)
    synchronize do
      @name = name
    end
  end
  def synchronize
    @mutex.synchronize do
      yield
    end
  end
end
```

参考

Ruby コードの感想戦

- ✓ <http://jp.rubyist.net/magazine/?0040-CodePostMortem>
- ✓ <http://jp.rubyist.net/magazine/?0041-CodePostMortem>

ヒント

- ✓ コードはコンピューターが実行するだけではない
- ✓ 読む人がいる
- ✓ コードは書いた人の意図を語る

リーダブルなコード

- ✓ わかりやすい (書いた人の意図が)
- ✓ 直しやすい
- ✓ 調べやすい
- ✓ 試しやすい

て

?

リーダブルな
コードって
うれしいの？

開発者として

ストレスが減る (1)

ストレス

なにしてるか
全然わからない
んだけど！

ストレス

なにしたいか
全然わからない
んだけど！

開発者として

楽しい
(2)

楽しい

すぐに
対応できる

すぐに対応

- ✓ こういうのが欲しいんだけど
- ✓ あ、ちょっと待ってください。
- ✓ こんな感じですか？
- ✓ →そう！こういうの！
- ✓ →ああ、こうなるのかあ。
じゃあ、なしで！

すぐに対応

オレってば スゲー感

<http://kitaj.no-ip.com/tdiary/20040404.html#p05>

オレってばスゲー感を

- ✓ 他の人のコードで、じゃなく
- ✓ 自分たちのコードで！

作らないのがいい？

- ✓ 本当に必要？検討しましょう！
 - ✓ 必要？本当に？本当に？
- ✓ 考えている間に試したら？
 - ✓ コードを書かない事が目指す事？
 - ✓ 必要ないものに時間をかけない事じゃない？

コストが
下がれば
試せる

コストが下がる

- ✓ 実サーバー → 仮想マシン
 - ✓ すぐに追加できる
 - ✓ すぐに削除できる

試せる

- ✓ 実サーバー → 仮想マシン
- ✓ どのくらいあらかじめ用意する？ →
- ✓ いつどうやって増減させる？

コード
だって
同じ

リーダブルなコードなら

素早く 対応できる

素早い対応

“

うわぁ！
こんなに素早く対応して
もらえるなんて！
ありがとう！

”

素早くない対応

- ✓ このくらいで終わりそう
 - ✓ でも、なんかあったらイヤだなあ
 - ✓ バッファで2倍かかることにしよう
- ✓ 困ったらすぐ相談じゃダメ？
 - ✓ 全体でよくなるように進めないの？

信用

“いつも素早く対応してくれる人が時間がかかるっていうんだから大変なんだろう。”

チームとして

誰でも直せる

誰でも直せる

- ✓ あるべき設計を維持できる
 - ✓ × 担当じゃないからわからない…
 - ✓ × 自分のところでなんとかするか…
- ✓ 助け合える

誰でも直せる？

- ✓ 直せても直しちゃダメ？
 - ✓ エンバグしたら責任が…
- ✓ できる人にだけ負荷がかかる？
 - ✓ できる人ほどツライ…
 - ✓ できない人の方が得？

開発者は
仕事で
リーダブルな
コードを
書けるのか？

ムリ？

コードレビュー

“リーダブルなコードにするためにコードレビューっていうのを導入したいけど他の人が読んでくれないからできない。”

答え

“まず、
あなたが読んでみませんか？”

時間がない

“自分は他の人のコードを読んでいる時間ないし…”

答え

“時間を決めて読んでみませんか？

ペースがわかったらどうやって
取り組んでいけばよいかアイ
ディアが浮かぶと思いますよ。

”

でも

“でも、でも、でも…”

”

コミットへの コメントサービス

“ クリアコードは、よいコードを書くことを当たり前にするためには、まず「みんながみんなのコードを読む」文化にすることからはじめるのがよいと考えます。

”

<http://www.clear-code.com/services/commit-comment.html>

B2D

Business to
Developer

なんで読むの？

- ✓ 読まないと
読みやすいコードは書けない
- ✓ 読む人を想像しろなんてムリ
- ✓ 自分が読む人になる

あなたが読む事を支援

- ✓ どうやって読むの？
 - ✓ よいところを学ぶ
 - ✓ 悪いところ探しではない！
- ✓ どういうタイミングで？
 - ✓ pushとかcommitした後
 - ✓ 一区切り付いているでしょ？

みんなが読む事を支援

- ✓ こうすると読みやすいですよ
 - ✓ 読みにくいコードばかりだと続けられない
- ✓ よいコードを見た！ → 真似してコミット
 - ✓ いいね！しているだけじゃない
 - ✓ 行動しているからより本物

コミットコミュニケーション

- ✓ 他の方が何をしているかわかる
 - ✓ あの方はあそこらへん詳しくそう
 - ✓ 相談してみよう
- ✓ 他の方が困っていないかわかる
 - ✓ この1時間ひとつもコミットしていないけど大丈夫？
 - ✓ このコミットすごいごちゃごちゃしてるけど悩んでる？

毎日↓をコミット

- ✓ 今日どれくらい読めたか
 - ✓ 感覚で。70%とか。
- ✓ 今日印象に残ったコミット
 - ✓ メモリリーク直った！
- ✓ 今日の一言感想
 - ✓ 自分のコミットが真似されていた！

さいごに

- ✓ リーダブルなコードが大事そうな気がしましたか？
- ✓ 仕事でリーダブルなコードを書けそうですか？
- ✓ 支援があれば書けそうですか？

いや、でも…

“自分の会社は横のつながりを
推奨していないので…”

答え

“ オープンソースソフトウェアの開発に参加してみませんか？
そこで体験してよかったら自分の言葉で説明できるはず。 ”