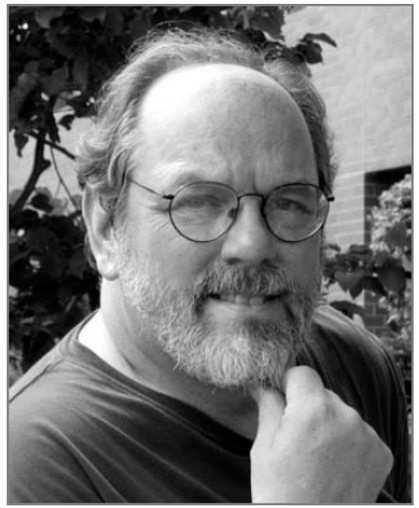
Tech Debt

Yegor Bugayenko

Lecture #14 out of 24
80 minutes

The slidedeck was presented by the author in this YouTube Video



“Shipping first time code is like going into debt. A little debt speeds development so long as it is paid back promptly with a rewrite. The danger occurs when the debt is not repaid. Every minute spent on not-quite-right code counts as interest on that debt.”

— Ward Cunningham, *Experience Report — The WyCash portfolio management system*, OOPSLA, 1992

Puzzle Driven Development: Motivating Example

Commit #1:

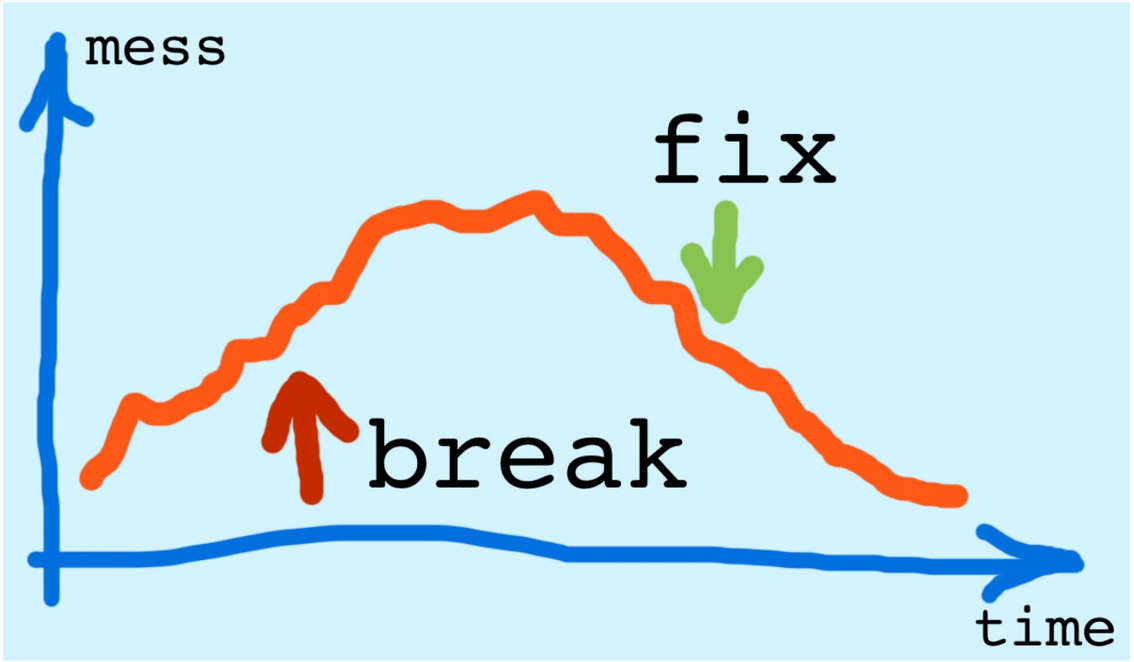
```
int fibonacci(int n) {
  if (n <= 2) {
    return 1;
  }
  // @todo I don't know
  // what to do when "n"
  // is larger than "2".
  // Implement it and uncomment
  // the assertion below.
  return 0;
}
assert fibonacci(0) == 1;
assert fibonacci(2) == 1;
// assert fibonacci(9) == 34;
```

Commit #2:

```
int fibonacci(int n) {
  if (n <= 2) {
    return 1;
  }
  if (n == 9) {
    return 34;
  }
  // @todo Implement others
  // too, but I don't know
  // how to do it right.
  return 0;
}
assert fibonacci(2) == 1;
assert fibonacci(9) == 34;
// assert fibonacci(10) == 55;
```

```
\end{ffcode*}
}
\par\columnbreak\par
Commit \#3:\par
{\scriptsize\begin{ffcode}
int fibonacci(int n) {
  if (n <= 2) {
    return 1;
  }
  return fibonacci(n-1)
    + fibonacci(n-2);
}
assert fibonacci(0) == 1;
assert fibonacci(2) == 1;
assert fibonacci(9) == 34;
assert fibonacci(10) == 55;
```

Break-and-Fix Cycle

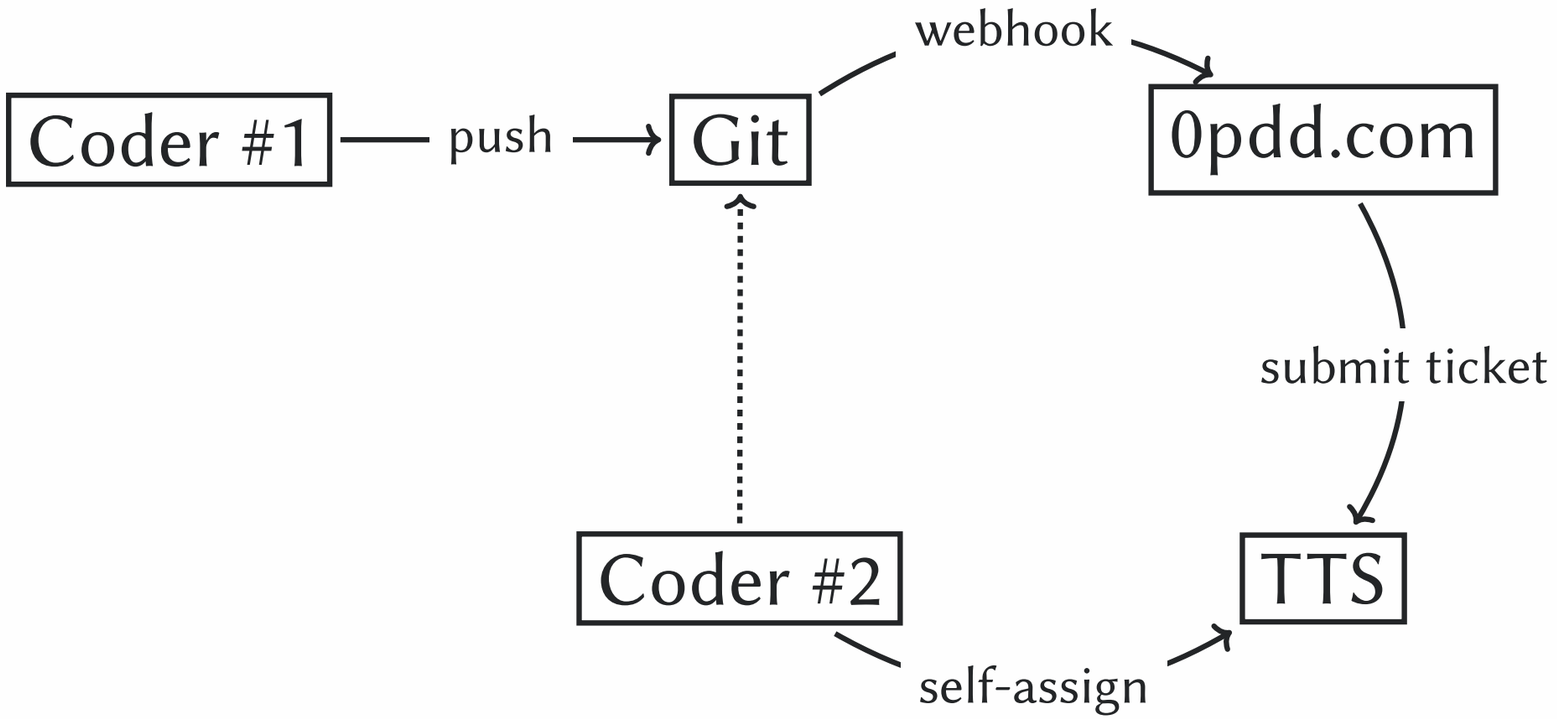


Source: https://www.yegor256.com/2014/04/12/puzzle-driven-development-by-roles.html

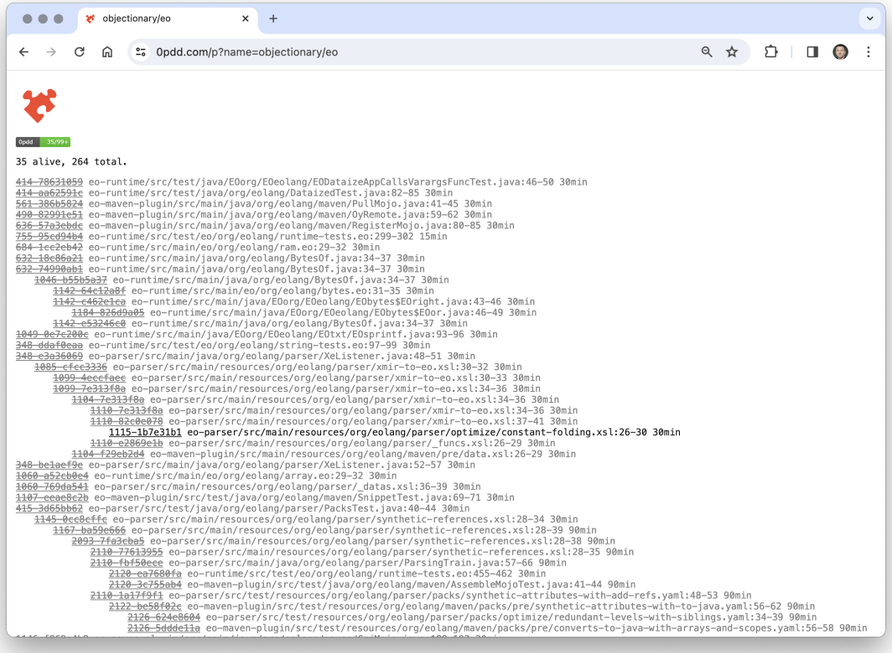
www.0pdd.com


0pdd
0pdd.com
PDD puzzles collector.
How does it work?
Recent commits:
zold-io/zold-score zold-io/zold-ruby-sdk zold-io/zold.github.io zold-io/zold zold-io/wts.zold.io zold-io/quick zold-io/papers zold-io/java-api zold-io/blog.zold.io zerocracy/farm
Made by @yegor256 for Zerocracy.
96.0006%
0/81
stars 120
BUILD 2.34.1 3.1.4
4561

PDD Pipeline

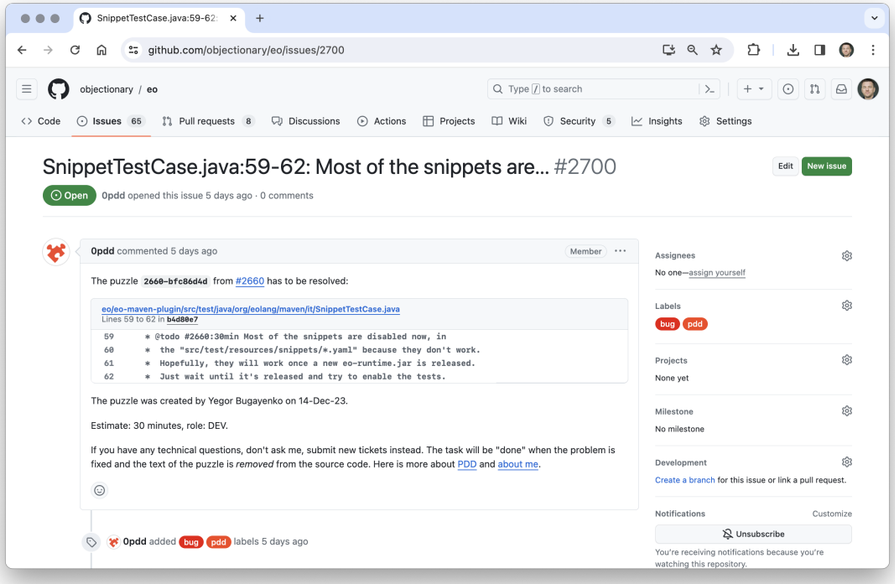


250+ Puzzles in objectionary/eo



Source: https://www.0pdd.com/p?name=objectionary/eo

Sample Ticket Submitted by 0pdd.com to objectionary/eo



Source: https://github.com/objectionary/eo/issues/2700

“This paper presents the benefits of considering the entire backlog when prioritizing tasks. We employ an iterative approach using Particle Swarm Optimization to optimize a linear model with various preprocessing methods to determine the optimal model for task prioritization within a backlog.”

— Yegor Bugayenko, Mirko Farina, Artem Kruglov, Witold Pedrycz, Yaroslav Plaksin, Giancarlo Succi, *Automatically Prioritizing Tasks in Software Development*, IEEE Access, Vol. 11, 2023

Read this:

Automatically Prioritizing and Assigning Tasks from Code Repositories in Puzzle Driven Development, Yegor Bugayenko, Ayomide Bakare, Arina Cheverda, Mirko Farina, Artem Kruglov, Yaroslav Plaksin, Giancarlo Succi, and Witold Pedrycz, Proceedings of the International Mining Software Repositories (MSR), 2022

Automatically Prioritizing Tasks in Software Development, Yegor Bugayenko, Mirko Farina, Artem Kruglov, Witold Pedrycz, Yaroslav Plaksin, and Giancarlo Succi, IEEE Access, Vol. 11, 2023

Puzzle Driven Development (2010)

PDD by Roles (2014)

PDD in Action (2017)

References