

How to take control of code quality

Joel Chippindale — CTO at Unmade

LRUG — April 2020

“

I am not **allowed** to spend sufficient
time on code quality

Many, many software engineers

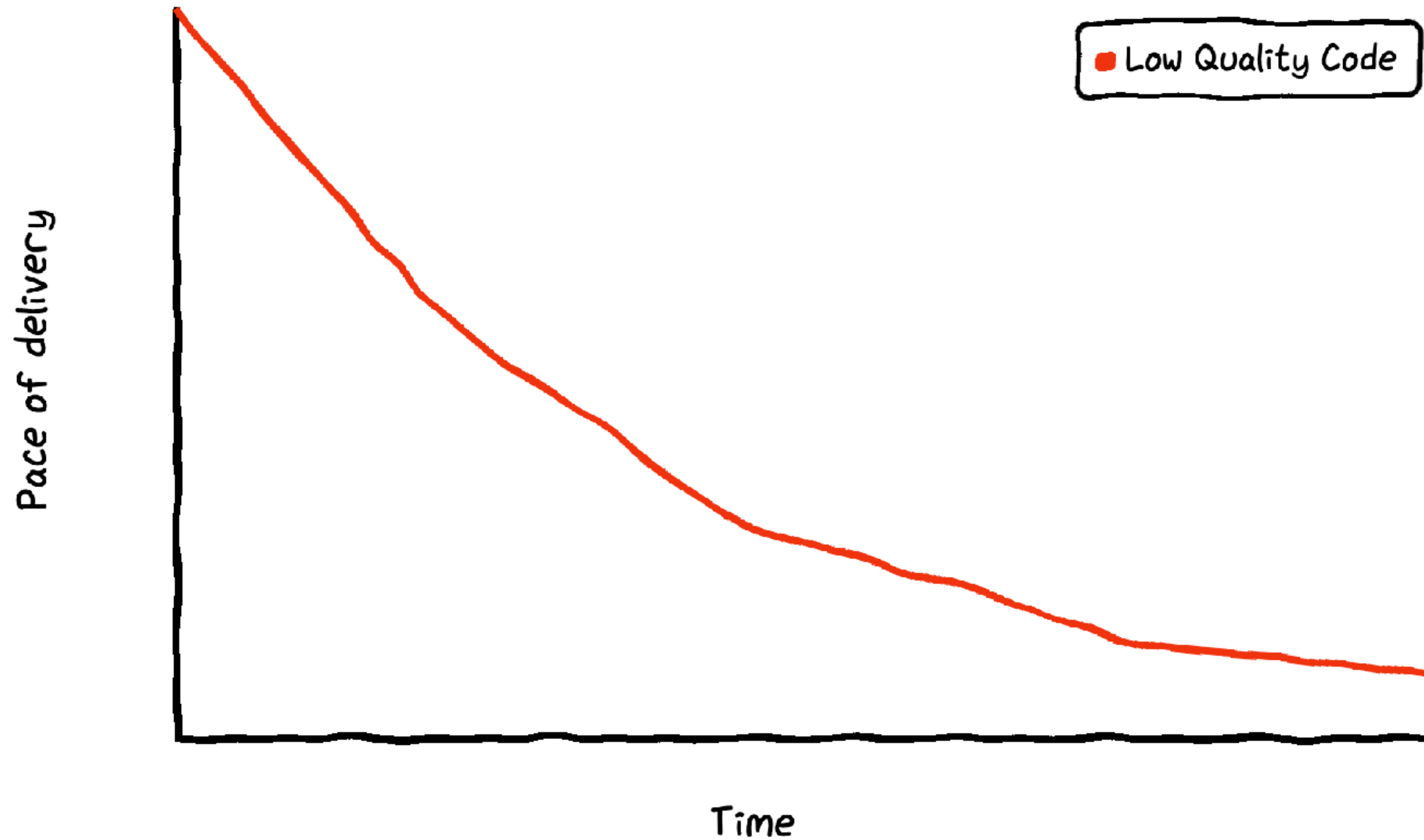
@joelchippindale

Do you feel that
you are spending **sufficient time**
improving the quality of your code?

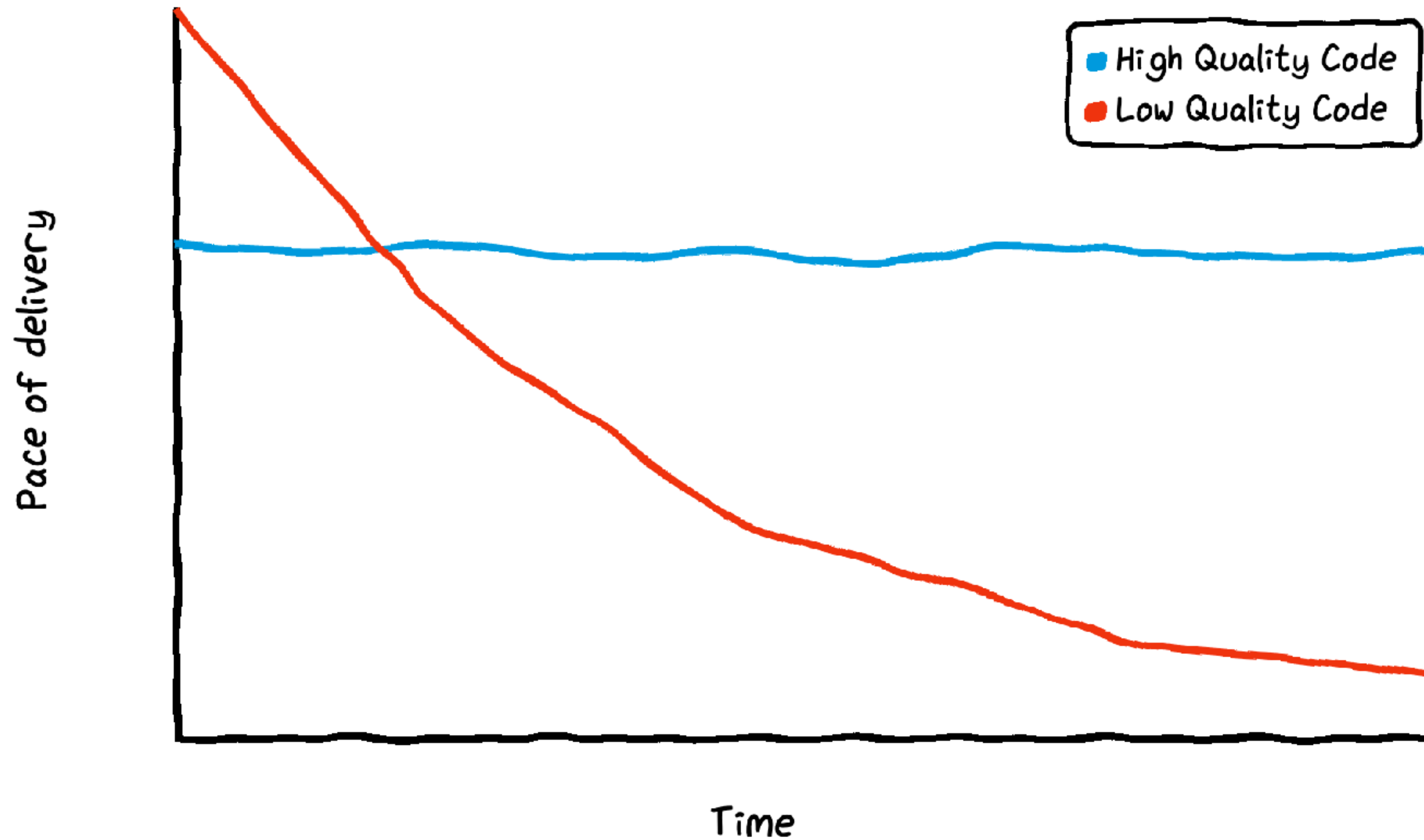
Do you feel that
you are spending **too little time**
improving the quality of your code?

WHAT DO WE MEAN BY HIGH QUALITY CODE?

Made up graph of pace of delivery over time



Made up graph of pace of delivery over time



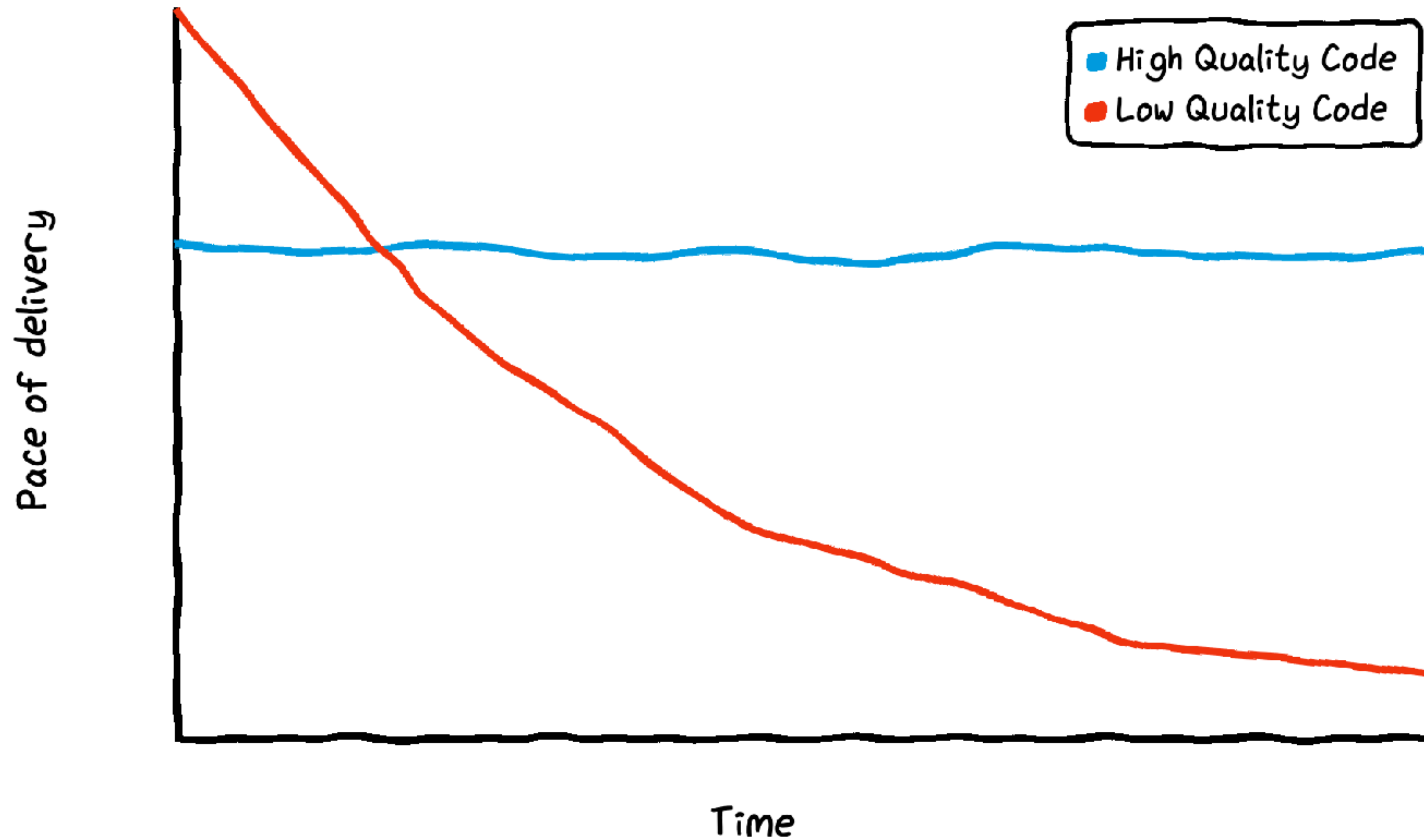
High quality code is easy to keep changing

“

Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Agile Principle No 8. - <https://agilemanifesto.org/>

Made up graph of pace of delivery over time



**WE ARE RESPONSIBLE FOR
CODE QUALITY**

UNMADE

BUILD TRUST TO MAINTAIN CODE QUALITY

@joelchippindale



1. **Speak in terms that make sense to others**

“

**We need to refactor the model classes
to reduce technical debt before we can
work on that**

What we say

@joelchippindale

UNMADE

“

**Blah blah...have to whiffle the slithy
toves and gimble the frumious
Bandersnatch...blah blah blah**

[it's going to be slow and delayed]

What they hear

@pelchior

2. Show that you value the same things

“

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software

Agile Principle No 1. - <https://agilemanifesto.org/>

3. Don't bullshit

PRACTICES TO SUPPORT CODE QUALITY

1. Shape the solution

2. Scout rule



“

Continuous attention to technical excellence and good design enhances agility.

Agile Principle No 9. - <https://agilemanifesto.org/>

3. Make a quantified case for larger improvements

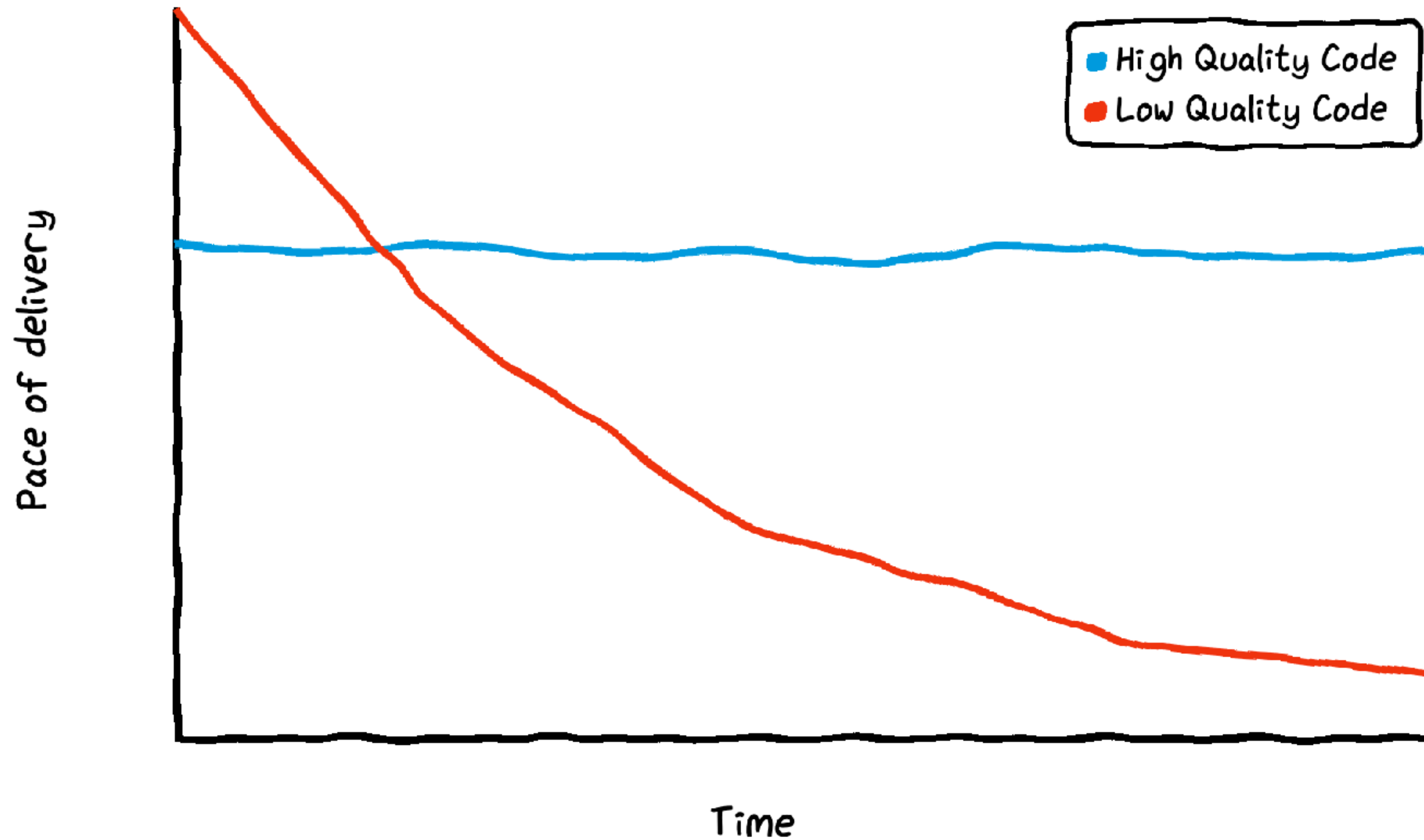
UNMADE

IN SUMMARY

@joelchippindale

We are responsible for code quality

Made up graph of pace of delivery over time



- 1. Shape the solution**
- 2. Scout rule**
- 3. Make a quantified case for larger improvements**

THANK YOU

Joel Chippindale - CTO at Unmade
@joelchippindale

Further reading and watching

- This talk and the slides - <https://blog.mocoso.co.uk/talks/2020/02/27/take-control-of-code-quality/>
- Principles behind the Agile Manifesto - <https://agilemanifesto.org/principles.html>
- Running an Agile Principles study group - <https://medium.com/making-unmade/running-an-agile-principles-study-group-7744774da053>
- Towards an understanding of technical debt - <http://laughingmeme.org/2016/01/10/towards-an-understanding-of-technical-debt/>
- Technical debt - Bad metaphor or the worst metaphor? - <https://ronjeffries.com/articles/015-11/tech-debt/>
- Why not fix your technical debt? - <https://www.youtube.com/watch?v=TNcJqyNWTNY>
- Intercom's engineering principles - <https://www.intercom.com/blog/intercom-product-principles/>