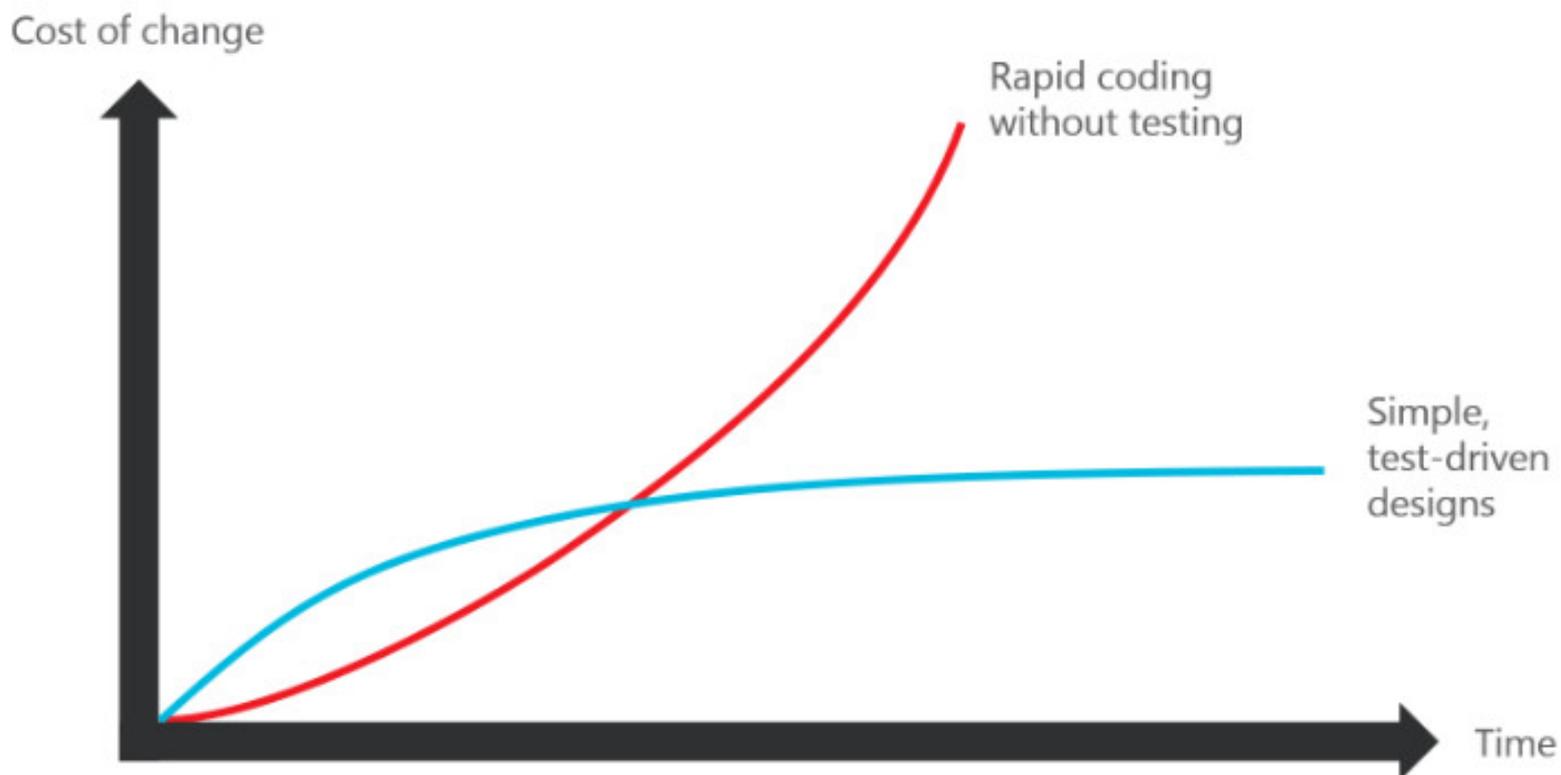


Technical debt from a business perspective

Sometimes, in software development, you may not be doing things the right way. Not doing the thing right can land you in future trouble maintaining and further developing your software. When you are in this way burdened with the pain of past sins, we call it technical debt.



The analogy goes as follows: when you are cutting corners and not spending your time writing unit tests and documenting your code, it is similar to be incurring debt. Debt that has to be paid back one day. Debt that comes with interest. Interest that may keep piling on top of your debt until the day when it becomes un-serviceable - equal to the cost of maintaining your software becoming prohibitively high and you may decide to start allover.

However, technical debt can be more than a curse - it can also be a blessing. When short on resources, budget, time, you may proactively decide to put proper development culture on hold and send your developers “hacking away”. This can give you a burst of fast results towards an important milestone. And often with some surprising added benefits like strengthening the team spirit from the late night pizzas and moving the understanding of the team what it is *really* capable of achieving.

You need to remember though that incurring technical debt is going to the loan shark. And you really want to repay him quickly before the interest piles up. And that requires cleaning up after each "hacking" period - and that will often take as much as half of the time that you were developing away. While doing nothing may get you stuck in the mud with no way out.

Jesper Lindholt, CEO at Livatek