Today I'd like to talk with you a little about writing technical documentation, ADRs in particular.

Architectural Decision Records

by Oleksandr Ponomarov

So, why is this a hot topic, why somebody should be concerned about writing proper technical documentation? When I joined Appsilon, the first thing I went though was a bunch of pages on our Outline and this one in particular.

Why ADRs?

It briefly describes Appsilon's approach to writing documentation. There's a reference at the end to an article from Google, which explains how to write design docs and provides a template. What's more appealing is that entire article is using the template itself.

In a nutshell, the template from Google consists of a title, some meta keys like author, status, last time updated. It sets up a context for your design, goals you'd like to achieve, overview, detailed design, cross-cutting concerns whatever that means, and also alternatives considered.

Google Template

The article is from 2015, therefore I would have suspected that it has already been widely adopted. When I did some research - it turned out that there's actually a dedicated term for this particular kind of document describing your design or architecture, and it has done even more influence comparing to the Google's one. The term was introduced by Michael Nygard in 2011. And this is essentially how we're going to get familiar with ADRs.

What is (Software) Architecture?

So let's move a bit closer to the ADRs and answer a simple question: what is architecture? A common definition would be like this: it represents the design decisions that shape a system.

“Architecture represents the significant design *decisions* that shape a system, where significant is measured by cost of change.”— Grady Booch