







ARCHITECTURAL DECISION RECORDS

by Oleksandr Ponomarov

WHY ADRS?

 Technology / 

  Edit  New doc 

How to write Design Docs

Marek updated 10 days ago · Viewed by 30 people

There are two kinds of designs:

- functional design
- technical design

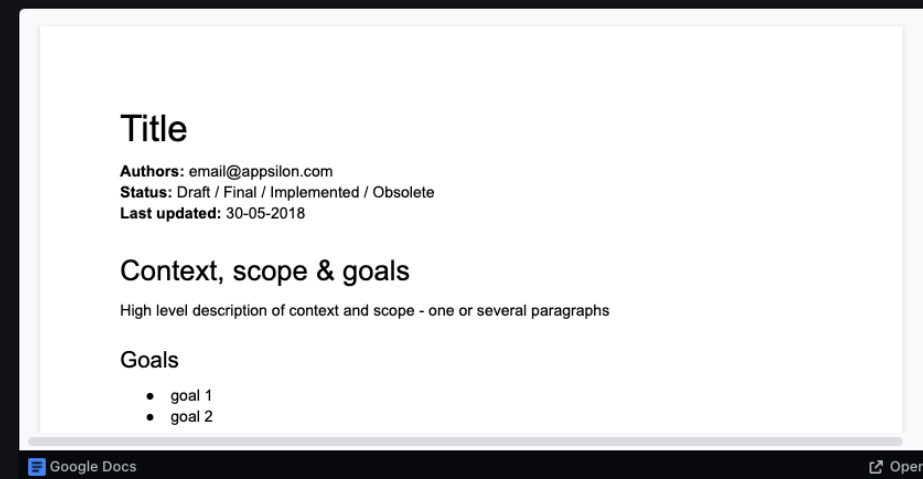
It's important to know which you are writing!

Functional design

Describes **what** should be implemented. This is what you discuss with the client (internal or external). It's about how it should look and work from user's perspective.

We don't currently have a template for that - just write it in a document.

There is a good article about technical design docs from Google. It is written in a form of a design doc itself, so it is simultaneously an explanation and an example of a design doc. Please read it:
<https://medium.com/@cramforce/design-docs-a-design-doc-a152f4484c6b>



GOOGLE TEMPLATE

Title

Authors: email@appsilon.com

Status: Draft / Final / Implemented / Obsolete

Last updated: 30-05-2018

Context, scope & goals

High level description of context and scope - one or several paragraphs

Goals|

- goal 1
- goal 2

Non-goals

- non-goal 1
- non-goal 2

Overview

Malte Ubl

6.4K Followers

About

Follow

Sign in

Get started



Design docs - A design doc



Malte Ubl

May 20, 2015 · 9 min
read



Originally published on my blog
Industrial Empathy

WHAT IS (SOFTWARE) ARCHITECTURE?

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

WHAT IS ARCHITECTURE DECISION?

An Architectural Decision (AD) is a software design choice that addresses a functional or non-functional requirement that is architecturally significant.

WHAT IS ARCHITECTURE DECISION RECORD?

An Architectural Decision Record (ADR) captures a single AD, such as often done when writing personal notes or meeting minutes; the collection of ADRs created and maintained in a project constitutes its decision log.

It should be as easy as possible to:

- write down the ADRs
- version them!

Markdown 🔥

ADR TEMPLATE

Decision record template by Michael Nygard

This is the template in [Documenting architecture decisions - Michael Nygard](#). You can use [adr-tools](#) for managing the ADR files.

In each ADR file, write these sections:

Title

Status

What is the status, such as proposed, accepted, rejected, deprecated, superseded, etc.?

Context

What is the issue that we're seeing that is motivating this decision or change?

ADR TEMPLATE IN MARKDOWN

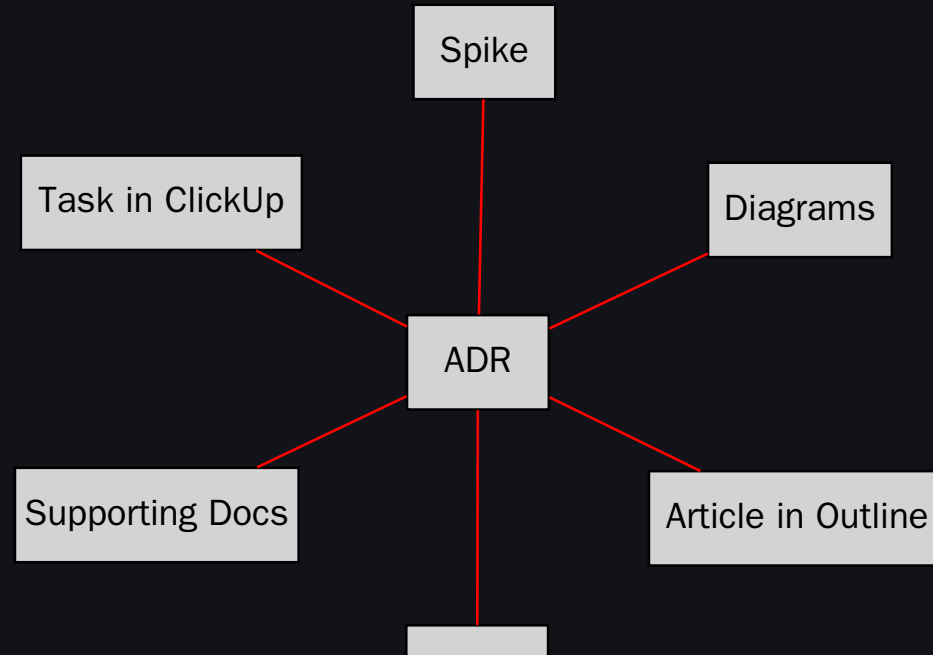
```
1  # Title
2
3  ## Status
4  What is the status, such as proposed, accepted, rejected, c
5
6  ## Context
7  What is the issue that we're seeing that is motivating this
8
9  ## Decision
10 ..
```


ADR =~ Google Template

ADR IN A NUTSHELL

ADRs should be considered as *lightweight* records of decision making, linking in other documents and content when required, keeping ADR itself compact and to-the-point. It can live *anywhere*, in *any* format.

ADR IN A NUTSHELL



WHO WRITES ADRS AND WHY?

- ADRs can and should be written by any engineer in the team
- Should be considered as living documents
- Common framework for communication
- Stronger team alignment, share best practices

TOOLING

- [adr-tools](#) - bash scripts to manage ADRs in the Nygard format
- [log4brains](#) - CLI and web UI to log and publish your ADRs as a static website
- .. & many more!

Example