

# **(Architecture) Decision Records**

What are ADRs and why should you use them?

# Context

- we live in a agile world
- documentation needs time, which we usually don't have
- teams are constantly in change

# What is an ADR?

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

- can be also used and modified for any kind of decision

# Problems they try to solve

- documenting decisions in a lightweight way
- make solutions trackable
- allow new members to easily understand why decisions were made
- revisit choices in case something turns out to be the wrong
- create a knowledge base
- keep it simple, no complex format

# Contents

ADRs can contain different fields depending on the usecase, these are the typical ones:

| Field                    | Description   |
|--------------------------|---|
| <b>title</b>             | short description about the problem or decision to make                     |
| <b>status</b>            | position in lifecycle (e.g. <code>proposed</code> , <code>accepted</code> ) |
| <b>context / problem</b> | conditions that led to the decision or a specific problem we try to solve   |
| <b>pros / cons</b>       | pro and cons for each possible option                                       |
| <b>consequences</b>      | list all positive and negative consequences                                 |

# How to use

- **synchronous**
- asynchronous

# synchronous

- meet
- discuss and use as protocol
- document final decision
- implement

# How to use

- synchronous
- **asynchronous**



# asynchronous

- someone or a subgroup creates a proposal
- discuss the proposal with a tool (using comments, chat etc.)
- finalize decision and get approval from everyone

# Typical lifecycle

| Status      | Meaning  |
|-------------|--|
| wip         | work in progress, not usable yet                                   |
| proposed    | ready, but not approved  |
| accepted    | everyone agreed, ready   |
| discarded   | thrown away, will not be used (e.g. problem doesn't exist anymore) |
| deprecated  | information no longer up to date                                   |
| superseeded | has been replaced by new decision (provide reference)              |

# Example (001-Written-language-in-the-team)

```
# Basic information
number: 001
title: Written language in the team
status: accepted
context: |
  In the team we need to agree which language to use for public communication and documentation inside TrustedShops.
  Previously there were a lot of uncertainties and unclear usages of different
  languages in different situations.
  To be more consistent as a team inside the company we need to decide for one.
# Choices we have
choices:
- choice: "English"
  description: We use english for the entire written communication
  pros:
    - company language
    - flexiblility for new team members
  cons:
    - language level in team is not high, which leads to miscommunication
- choice: "German"
  description: We use german for the entire written communication
  pros:
    - native language for all current members
    - no language barrier
  cons:
    - company language is english
    - new team members might be unable to understand it properly
- choice: "German & English"
  description: We write texts in german and provide a english translation with a tool (e.g. DeepL)
  pros:
    - includes company language
    - flexibility for new team members
  cons:
    - translation in english might be wrong at some places
# Final decision section
decision: "German & English"
consequences:
- we align with company language
- no room for misunderstandings due to german version
- proof reading for translated english version to avoid wrong keywords
```

# The End

Spread the word and use (A)DRs to make the world a better (documented) place.

