
ENGINEERING PRACTICE · TEMPLATE

Architecture Decision Records: The *1-Page* Version

KAANSYSTEMS.COM/LIBRARY/ONE-PAGE-ADRS · MAY 30, 2026

ABOUT THIS TEMPLATE

The canonical ADR format is 4-6 pages. Teams don't write them. The 1-page version engineers actually adopt: same value, an order of magnitude less friction.

THE TEMPLATE

The single-page format, copy-pasteable into a new `docs/decisions/NNNN-short-title.md` :

```

# ADR-NNNN: <Decision in present tense>

## Status

Proposed | Accepted | Superseded by ADR-NNNN | Deprecated

## Context

<3-4 sentences. What's the situation? What forces require a decision now?>

## Decision

<One sentence. The decision itself.>

## Consequences

- Positive: <one sentence>
- Negative: <one sentence>
- Neutral: <one sentence>

## References

- <PR link>
- <Related ADR>
- <External documentation>

```

That's the entire format. Six headings, four to six fields filled in, total length under a screen on a normal monitor.

— HOW TO USE IT

Make ADRs part of the PR template for significant changes. The PR template includes a line like "Does this require an ADR? If yes, link the file." That single question catches the changes that should have one.

In code review, ask for ADRs the way you ask for tests: not for every change, but for changes where the absence is conspicuous. "Why is this a separate service?" usually means an ADR is missing.

When an ADR is superseded, *don't delete it*. Mark the status. The history is part of what makes ADRs valuable. Six months from now, someone reads ADR-007 (superseded), realizes the original reasoning, and avoids recreating the same mistake.

The first month after introducing the practice, expect the team to write three or four ADRs. The second month, fewer. They've learned what counts. By month six, you have a small library of decisions that captures the architectural skeleton of the system in a form that's still readable three years and three engineering generations later.