



Architecture decision records

Alejandro Aguirre López

Alberto Díez Bajo

Juan Rodríguez de la Fuente

Index

[What is an ADR?](#)

[Benefits of using ADRs](#)

[When should we write ADRs](#)

[Taking decisions while writing an ADR](#)

[Components of an ADR](#)

[Where should ADRs be stored?](#)

[Tooling](#)



What is an Architecture Decision Record?

- It is a document that captures a decision.
- It also includes its context.
- Developers and stakeholders are able to see ADRs.
- It helps to discover the motivation behind certain decisions.
- It has a lot of benefits.

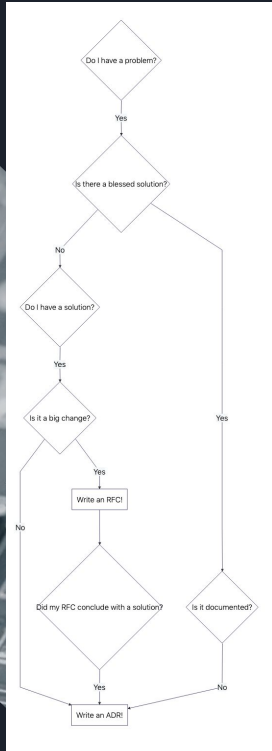




Benefits of using Architecture Decision Records

- 01 They are especially useful for capturing long-term intentions.
- 02 They are small and modular, which makes the maintenance of a system easier.
- 03 As they are records for significant questions, the quantity of valueless information will be minimum.

When should we write ADRs?



- Everytime we make an architecturally significant decision.
- When the solution of a problem is not documented yet.
- When we have a problem and a solution, but it involves changes (or not).



Taking decisions while writing an ADR

- Collection of records for “architecturally significant” decisions
- Formatting language
- Enumeration
- Keep all the ADRs
- Components (Michael Nygard)
- Size
- Conversation



Components of an ADR (Michael Nygard)

- Title
- Context
- Decision
- Status
- Consequences





Example

Pattern 1:

Context: Mud Hut

Problem: Allow natural light into the building

Solution: Make an opening in the wall

Pattern 2:

Context: Skyscraper

Problem: Allow natural light into the building

Solution: Make an opening in the wall

Pattern 3:

Context: Pattern 1

Problem: Prevent wind from entering

Solution: Put palm leaves in front of the opening

Pattern 4:

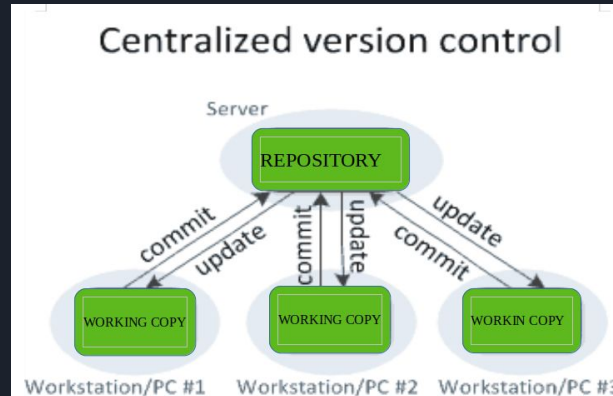
Context: Pattern 2

Problem: Prevent wind from entering

Solution: Add glass windows to the opening

Where should ADRs be stored?

- With the project documentation.
- In a VCS, along with the project source code.
- Synchronized with the rest of the project.





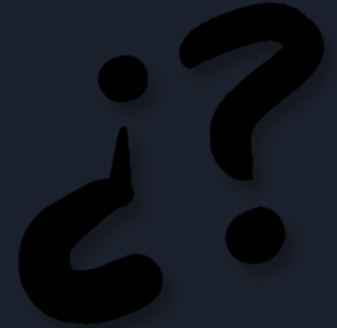
Tooling

- Your favorite text editor.
- Management tools.
 - adr-tools
- Templates.
 - Michael Nygard
 - MADR (Markdown ADR)
 - For business cases






Some questions of our partners...



- What are the major and practical benefits of ADR?
- How can we know that the decisions we made are correct before knowing the results in the near future?
- When should the architecture decisions be made? What are the consequences of failing in choosing the right time?
- Why is it important to use a standardized decision documentation? What are some of the other formats besides ADRs?
- What are the sections we need to include in an ADR?
- Is there any relationship between ADR and Lean software development?
- Is it useful to document decisions/solutions even if they were rejected?



What are the steps we need to follow to make a good ADR?

01	Decision identification	<ul style="list-style-type: none">• How urgent?• Experience• Decision todo list
02	Decision making	<ul style="list-style-type: none">• Technique to be used• Active research topic
03	Decision enactment and enforcement	<ul style="list-style-type: none">• Accepted by stakeholders• ADs have to be (re-)considered
04	Decision sharing (optional)	<ul style="list-style-type: none">• Repetition
05	Decision documentation	<ul style="list-style-type: none">• Templates and tools
06	Decision guidance	<ul style="list-style-type: none">• Decision making techniques

Now, questions time!

