RESUM Fundamentals of Software Architecture (Ch1)

June 16, 2025

Prefaci *Això és un document orientatiu que resumeix undamentals of Software Architecture de Mark Richards*

1 Preface

Axioms don't hold up to the rapid evolution of software, today's technologies might become obsolete in a decade, the ecosystem is always changing.

As developers it's easy to be biased towards a technology or an approach, but as architects we must soberly asses the good, bad and the ugly of EVERY choice (and they all have trade-offs).

2 Chapter 1: Introduction

The scope of what a software architect encompasses is just massive, to the point of being indefinible. Part of the job is taking decisions within a changing ecosystem.

Many ways to define what is architecture based on decisions, characteristics and structure, none well defined or catch-all. Rather than waste time defining the role, focus on the expectations:

• Make architecture decisions: Able to define the architecture ecisions and design principles used to guide technology within the team/department/outside *Guide*. There's a difference between helping guide the team to make a decision VS making the decision for them, although sometimes you *need* to make a decision to satisfiy an architectural need, is a fine line to walk.

• Continually analyze the architecture: Able to analyze the architecture and current technology envionment and then recommen solutions for improvement. Be able to check the architecture vitality, comparing it to the past and seeing the changes in business and tech.

• Keep current with latest trends: Able to keep with the latest tech and industry trends. Since the decisions are long lasting and difficult to change, understanding and following key trends helps the architect prepare for the future, be informed to make sound decisions.

• Ensure compliance with decisions: Able to ensure compliance with the design principles. The architect is able to continually verify that the development teams are following the architecture decisions and principles. Basically policing your app and enforcing guidelines.

• Diverse exposure and experience: Having experienced diverse techonologies, frameworks, platforms, environments. Doesn't mean to be an expert of all, just have context. Stretch your comfort zone, aggressively seek oportunities to experience multiple languages, platforms and technologies. , focus on technical breadth rather than technical depth. Master of all trades and so.

• Have business domain knowledge: Must have a certain business domain expertise, being confortable in the domain of a problem space. *Imagine being an architect at a large financial institution and not understanding common financial terms such as an average directional index, etc*. You have to understand the context in which you work.

• Possess interpersonal skills: Excepcional interpersonal skills, including teamwork, facilitation and leadership. *No matter what they tell you, it's always a people problem* (Speaking of axioms....), it's half the job.

• Understand and navigate politics: Able to understand the climate of the company and navigate it. textitAlmost every decision an architect makes will be challanged. Will be challenged for the effort involved, will be challenged by ego, must justify and fight for almost every one of those decisions.

There are certain intersections between the role of architect and other parts of an organization:

- Engineering Practices; It referes on the mechanics of how people organize and interact, the software ones on the other hand are such like continous integration. One of our achiles heels of software development is estimation of time. Unknown unknowns are the bane of our existence

- Operations/DevOps: Now we gotta handle this, the small side effect is that design gives a more complex architecture

- Process: The Process DOES have impact on the architecture (like Agile)

- Data: Code and Data go hand to hand.

We don't have Axioms but we do have laws of Software Architecture: - Everything in software architecture is a trade-off - Second law is read the first law again - Why is more important than how