

On the role of a Software architect

Autoría

- Jorge Puente García UO294228@uniovi.es
- Manuel García Baldó UO295497@uniovi.es
- Ignacio Llana Aller UO294914@uniovi.es

Introducción

El rol del arquitecto de software suele verse como algo estático: se diseña al inicio del proyecto y listo. Pero en realidad, el software está en constante evolución, y su arquitectura debe adaptarse con él. En el episodio 616 de Software Engineering Radio, Jeff Doolittle conversa con Ori Saporta sobre la importancia del diseño continuo y el papel clave del arquitecto en todo el ciclo de vida del software. Este documento resume los puntos más destacados de la entrevista.

El Rol del Arquitecto de Software: Más Allá del Diseño Inicial

En el desarrollo de software, el papel del arquitecto es crucial para garantizar que un sistema no solo funcione, sino que evolucione de manera sostenible. A menudo se piensa en la arquitectura como una fase inicial del proyecto, pero como destaca Ori Saporta en su entrevista en Software Engineering Radio, el diseño es un proceso continuo.

Diseño Continuo y Evolutivo

El diseño de software no es un evento aislado; debe ajustarse a medida que el sistema crece. Un arquitecto presente en todo el ciclo de vida del software evita la acumulación de deuda técnica y mantiene la estructura del sistema flexible ante los cambios en los requisitos.

Mediación y Toma de Decisiones

El arquitecto no solo es responsable del diseño técnico, sino que también actúa como un mediador entre los equipos de desarrollo y los stakeholders. Su rol es traducir necesidades de negocio en soluciones técnicas viables, tomando decisiones que equilibren escalabilidad, mantenibilidad y costos.

Habilidades Interpersonales: Clave del Éxito

Además del conocimiento técnico, un arquitecto efectivo debe poseer habilidades blandas. La escucha activa, la capacidad de comunicación y la gestión del cambio son esenciales para guiar a los equipos sin generar fricciones.

La Observabilidad Arquitectónica

En este contexto, herramientas como vFunction, donde trabaja Saporta, juegan un papel crucial. La observabilidad arquitectónica permite visualizar las dependencias dentro de un sistema complejo, identificar áreas problemáticas y priorizar acciones para mejorar la

calidad del software. Esto no solo ayuda a gestionar la deuda técnica, sino que también facilita la modernización continua de las aplicaciones.

Conclusión

La entrevista resalta el **papel clave del arquitecto de software** en todo el **ciclo de vida** de un **sistema**:

1. **Diseño continuo**: La arquitectura evoluciona con el software. El arquitecto debe ajustarla constantemente para evitar deuda técnica.
2. **Mediación**: Facilita la comunicación entre stakeholders y desarrolladores para alinear soluciones con el negocio.
3. **Decisiones estratégicas**: Transforma problemas de negocio en soluciones técnicas escalables y mantenibles.
4. **Visión global**: Asegura que los componentes trabajen en conjunto para un diseño flexible y eficiente.
5. **Habilidades interpersonales**: Liderazgo, negociación y escucha activa son clave para guiar equipos.

En conclusión, el **arquitecto no solo diseña soluciones**, sino que **garantiza la evolución, calidad y sostenibilidad** del sistema.

Referencias

- Podcast con Jeff Doolittle y Ori Saporta. [Enlace](#)
- vFunction y observabilidad arquitectónica. [Enlace](#)

Este enfoque continuo y dinámico del diseño es lo que permite construir software que no solo funciona hoy, sino que también está preparado para el mañana