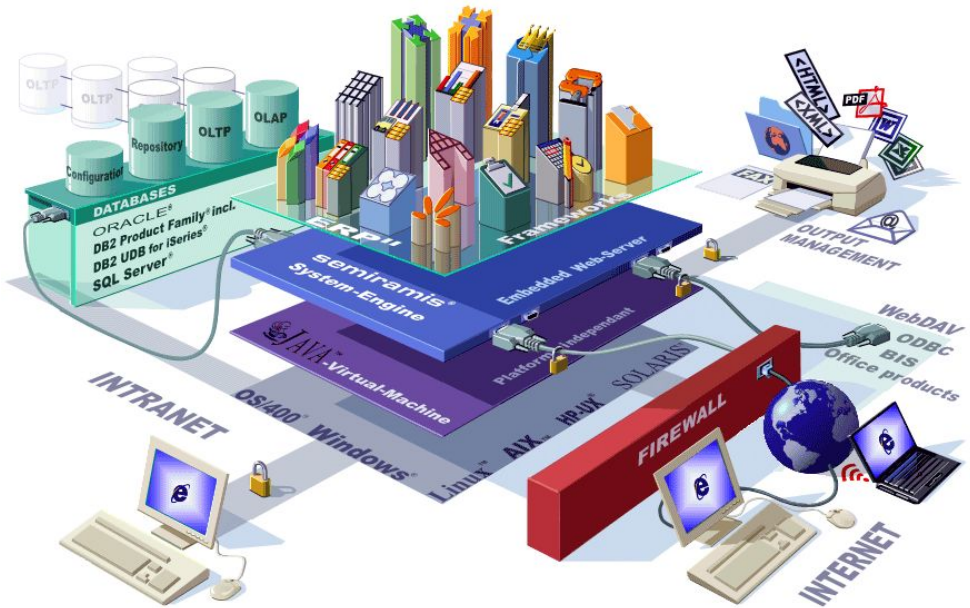
Arquitectura de software

DigitalHouse >
Coding School

Certified Tech Developer
The Ultimate Degree

¿Qué es la arquitectura de software?

La arquitectura de software es un grupo de **principios** y **restricciones** sobre cómo las soluciones de un sistema de software deben ser construidas dentro de un ámbito dado.



Algunos ejemplos de principios y restricciones:

- Solo se pueden utilizar lenguajes de programación basados en JVM (Java Virtual Machine).
- La comunicación sincrónica entre componentes se realizará mediante sus correspondientes APIs REST.
- La comunicación asincrónica se realizará utilizando un bróker de mensajería (ejemplo: RabbitMQ).
- Las trazas (logs, auditoría) que generen los módulos serán centralizados en ELK. Cada mensaje debe tener un formato específico.

Atributos de calidad

El **grado de aplicación** de cada uno de estos atributos debe **ser proporcional** a la solución que se quiere construir. Por lo tanto, definir las estructuras fundamentales, será aplicar los principios, restricciones, requisitos funcionales y no funcionales para construir una solución de software: sus componentes, interfaces, relaciones, estructuras, etc.

¿Por qué arquitectura de software?

A medida que el número de soluciones de software va creciendo, se empieza a requerir un cierto **grado de homogeneización**.

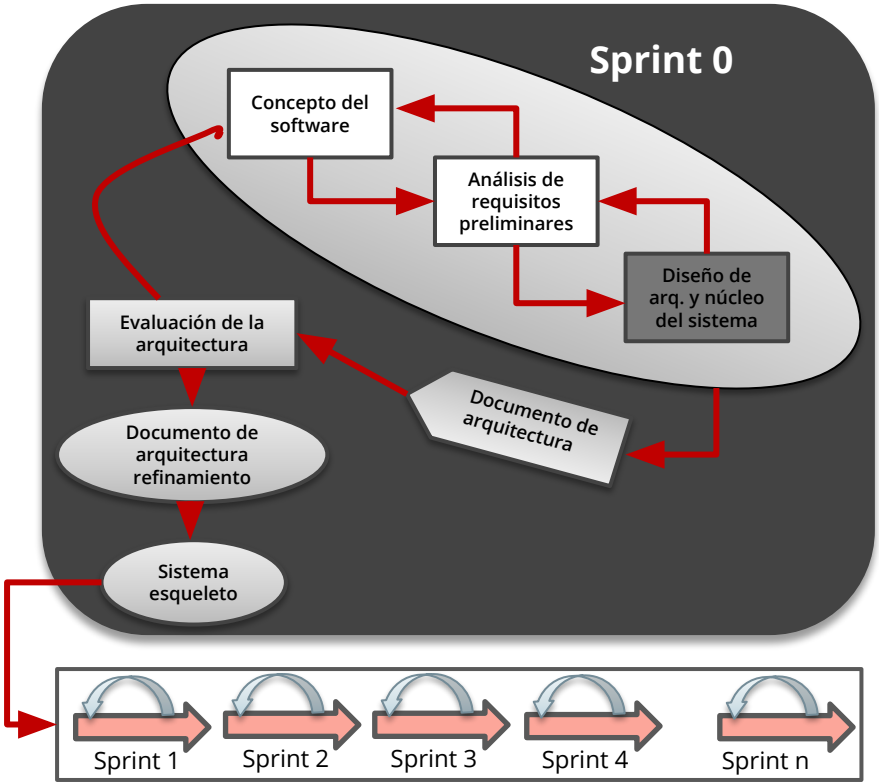
Muchas de las razones tomadas por una compañía en base a las restricciones —por ejemplo, del tipo de lenguaje a utilizar— suelen tener **connotaciones económicas**. Por eso, lo que en un principio podría considerarse como una limitación, puede convertirse en un modo de **ahorro de costos** en el futuro.

"La arquitectura de software es útil si ayuda a mantener los principios y restricciones en los que se basa, al mismo tiempo que sienta las bases para desarrollar las soluciones que demanda el negocio". Miguel Arlandy.

¿Cómo alinear arquitectura de software y equipos ágiles?

La esencia de **agile** gira en torno a la entrega rápida y constante de valor y la aceptación del cambio. Es entregar una solución con un alto grado de calidad en un período corto de tiempo. **Scrum** (uno de los métodos ágiles) se basa en equipos autónomos, autoorganizados y se apoya en un proceso iterativo e incremental.

Como regla general, podría ser interesante hacer algo de diseño de la arquitectura por adelantado, antes del inicio del proyecto (sprint 0), y otra pequeña cantidad antes de cada iteración.


Sprint 0
Concepto del software
Análisis de requisitos preliminares
Diseño de arq. y núcleo del sistema
Evaluación de la arquitectura
Documento de arquitectura
Documento de arquitectura refinamiento
Sistema esqueleto
Sprint 1
Sprint 2
Sprint 3
Sprint 4
Sprint n

- Se puede incluir una revisión de la arquitectura como parte del *Definition of Done* (DoD) de cada historia de usuario.
- Podría ser también muy interesante que al menos un miembro del equipo fuese responsable de asegurar que la arquitectura y el desarrollo del producto se encuentran alineados.

Si simplemente esperamos que la arquitectura de software emerja, podemos acabar en un ciclo de **“sprints de refactorización”** debido a la falta de diseño por adelantado.

Cuando se hace referencia a equipos ágiles, también se refiere a equipos **multidisciplinarios**. Pueden estar compuestos por roles tales como: Scrum master, desarrollador back, desarrollador front, product owner, líder técnico, etc.

El **líder técnico** será el responsable de hacer entender la arquitectura de referencia con el resto del equipo. Estará a cargo de:

- La resolución de cuestiones técnicas.
- Gestionar los atributos de calidad.
- Manejar los requisitos que requieran un cierto grado de innovación.
- Sincronizar los que apliquen en la arquitectura de referencia.
- Y, además, actuará como un mentor que ayude al equipo ágil a mejorar su efectividad, trabajando codo a codo con ellos.

Bibliografía

- Brown, Simon. (2018). *Software architecture for developers*.
- Bass, L., Clements, P., & Kazman, R. (2021). *Software Architecture in Practice*. Addison-Wesley Professional.


DigitalHouse>
Coding School