



 pragma



Agenda

1. Arquitectura de Software y Agilismo
2. ¿Qué desafíos tenemos?
3. ¿Qué buscamos?
4. ¿Cómo resolvemos estos desafíos?



Arquitectura de software

“Es la organización fundamental de un sistema, incorporando sus componentes, sus relaciones entre sí y con el medio ambiente y los principios que guían su diseño y evolución”

IEEE

Pilares del agilismo

Individuos e interacciones sobre procesos y herramientas

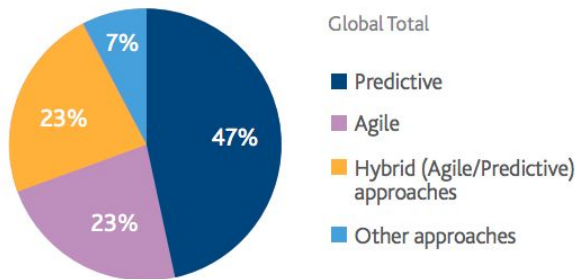
Software funcionando sobre documentación extensiva

Colaboración con el cliente sobre negociación contractual

Respuesta ante el cambio sobre seguir un plan

¿Qué ha cambiado?

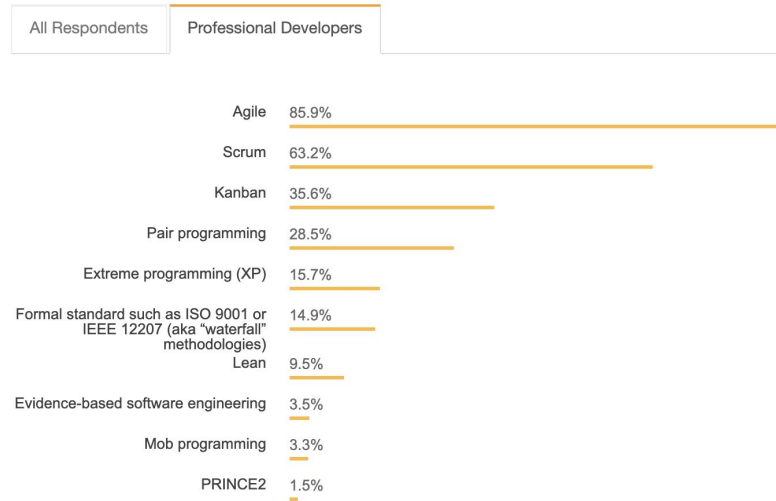
Q: In your estimation, what percentage of the projects completed within your organization in the past 12 months used the following types of approaches?



Note: Numbers may not sum to 100% due to rounding.

Fuente: Project Management Institute 2018

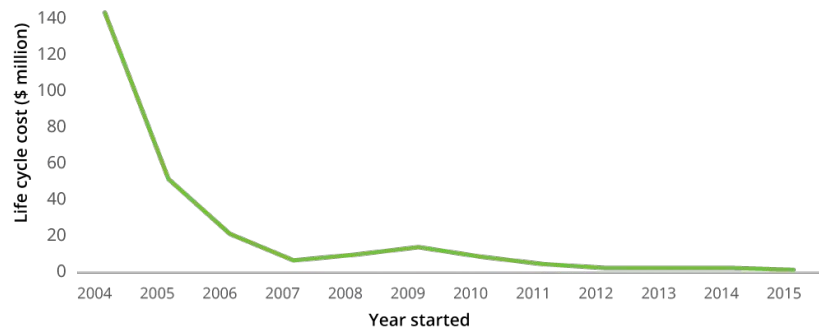
Which Methodologies Do Developers Use?



Fuente: Stack Overflow Developers Survey 2018

¿Qué ha cambiado?

Figure 1. Major federal software projects are getting smaller

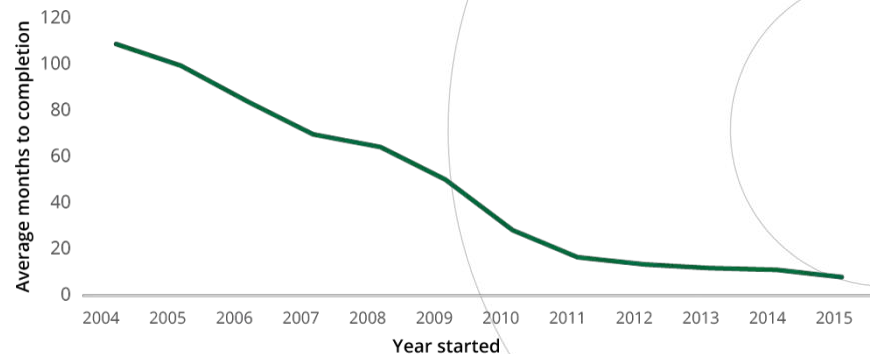


Note: The life cycle cost in 2015 was \$4.09 million.

Source: Deloitte analysis of Office of Management and Budget's ITDashboard.gov.

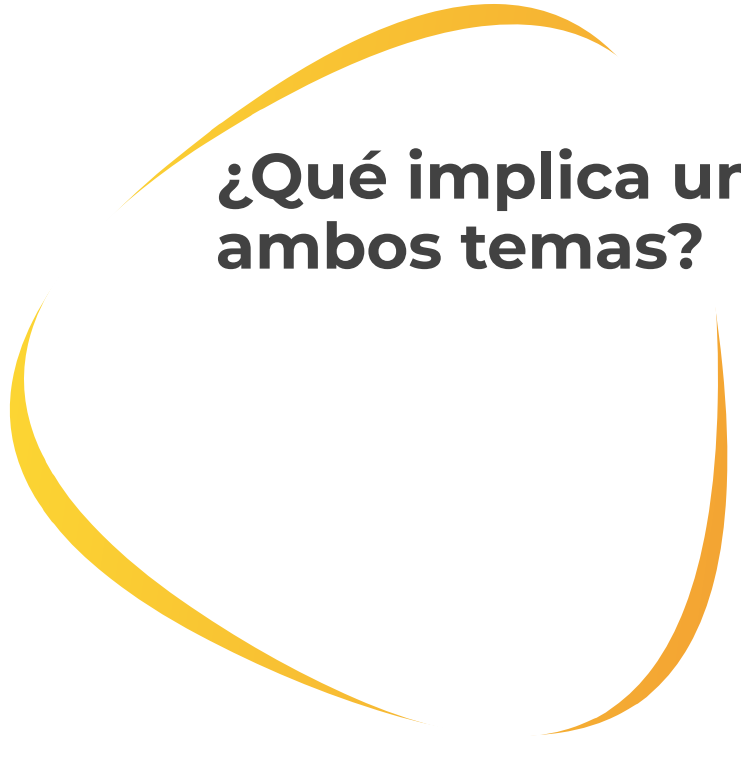
Deloitte University Press | dupress.deloitte.com

Figure 2. Major federal software projects are also getting shorter



Source: Deloitte analysis of Office of Management and Budget's ITDashboard.gov.

Deloitte University Press | dupress.deloitte.com



¿Qué implica unir ambos temas?

La arquitectura debe estar en función del negocio, mediante un esfuerzo mínimo requerido para capitalizar las oportunidades del mercado

Poner la tecnología al servicio del negocio

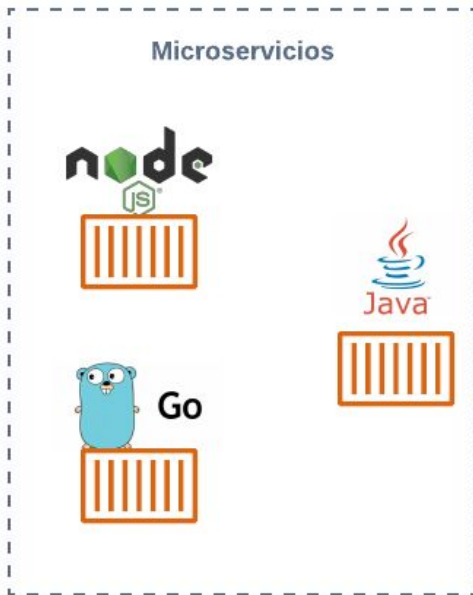




¿Qué priorizamos?

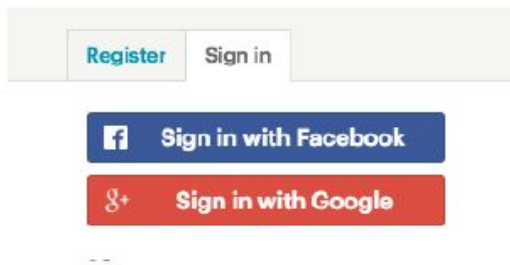
Democratización de la tecnología

Potencializarse con diferentes tecnologías



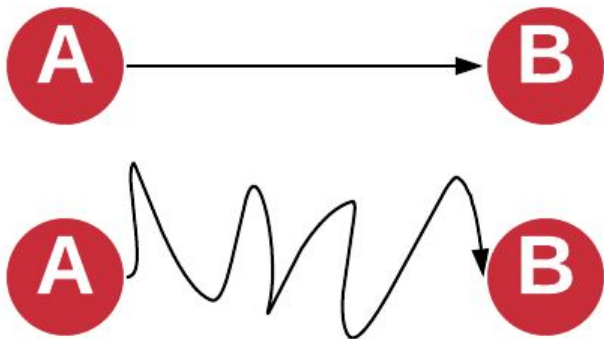
Colaboración

Aprovechar lo que ya existe y comunicarnos de forma estándar



Elasticidad

Realidad + Visión del negocio



Contenedor



Contenedor



Contenedor



Contenedor



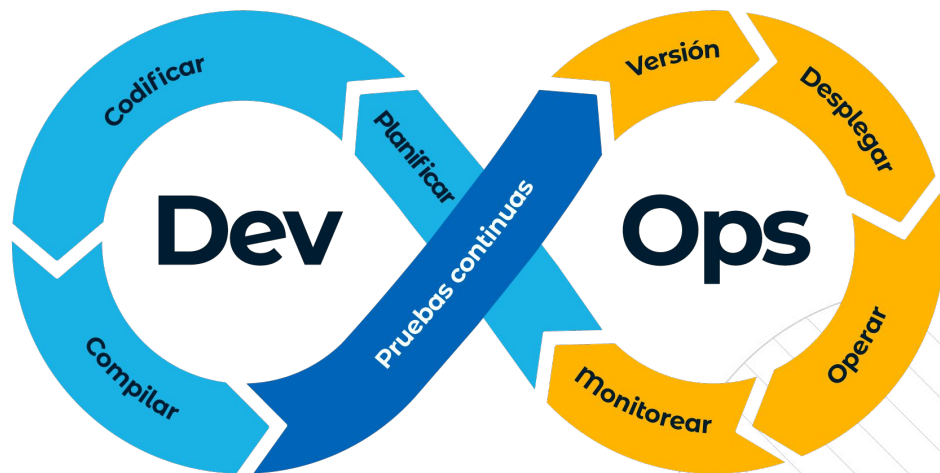
Contenedor



Contenedor

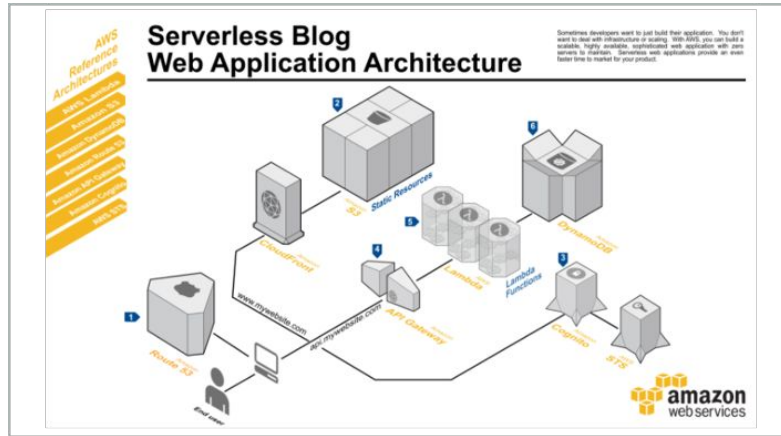
Agilidad

Incrementar la velocidad de implementación

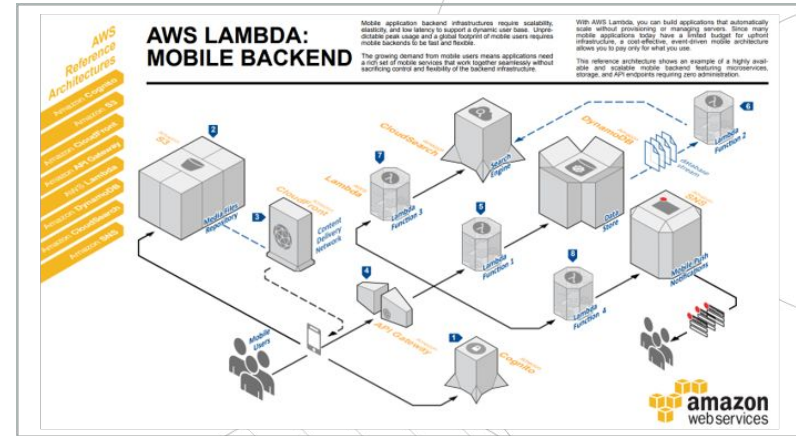


[illegible]

Arquitecturas de referencia



Arquitectura
Aplicación Web



Arquitectura Backend
App Móvil

Arquitecturas de referencia

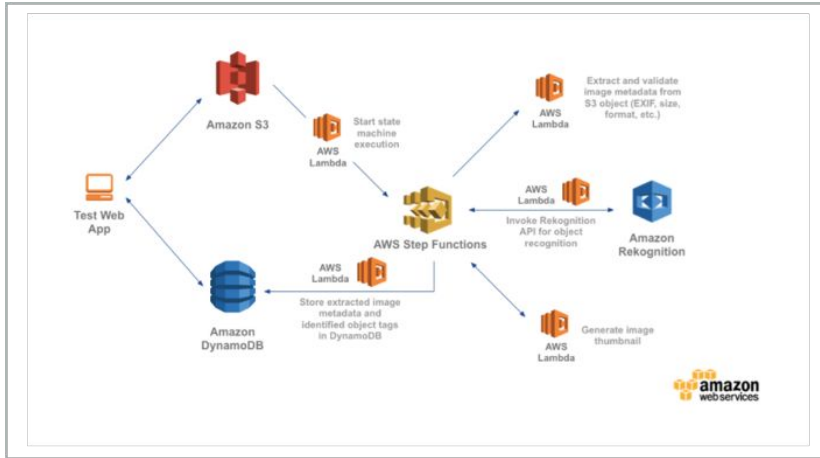
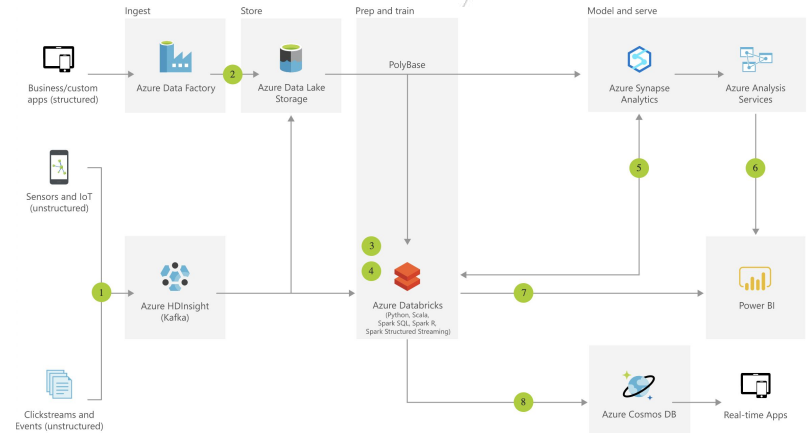


Image recognition & processing

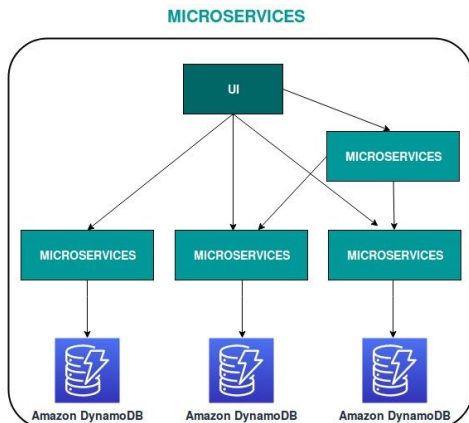


Real Time Analytics on Big Data Architecture

<https://aws.amazon.com/es/lambda/resources/reference-architectures/>

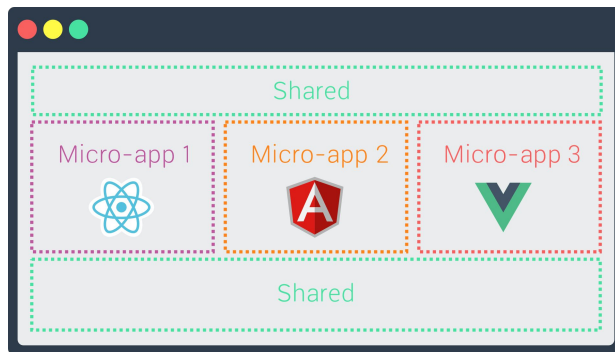
<https://docs.microsoft.com/en-us/azure/architecture/solution-ideas/articles/real-time-analytics>

Estilos de Arquitectura



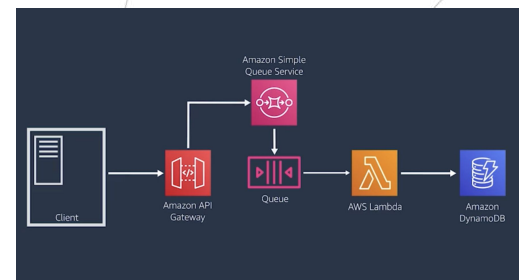
Arquitectura de
Microservicios

Fuente: dzone.com



Arquitectura de
Microfrontends

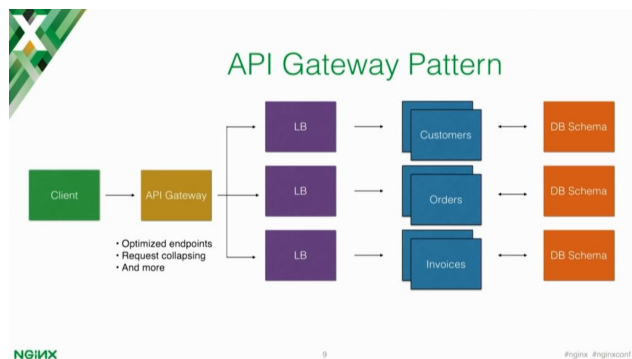
Fuente: medium.com



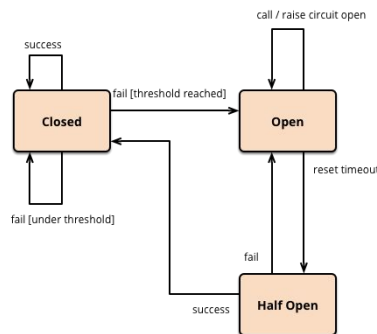
Arquitectura orientada
a eventos

Fuente: aws.com

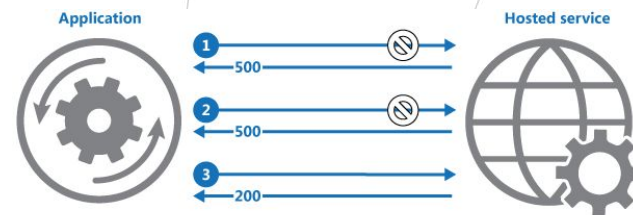
Aplicación de patrones



Patrón Api Gateway



Patrón Circuit-Braker



Patrón de reintentos

<https://github.com/tenshun/api-gateway-spring-security>

<https://martinfowler.com/bliki/CircuitBreaker.html>

Recomendaciones

- Evaluar los temas de impacto técnico con 2 sprints de antelación
- Pruebas de Concepto - Spikes de Arquitectura dentro de los Sprints
- Documentar las decisiones de medio y alto impacto sobre el proyecto de forma detallada



Gracias



2020

Juan Ricardo Herrera
Arquitecto de Software

juan.herrera@pragma.com.co