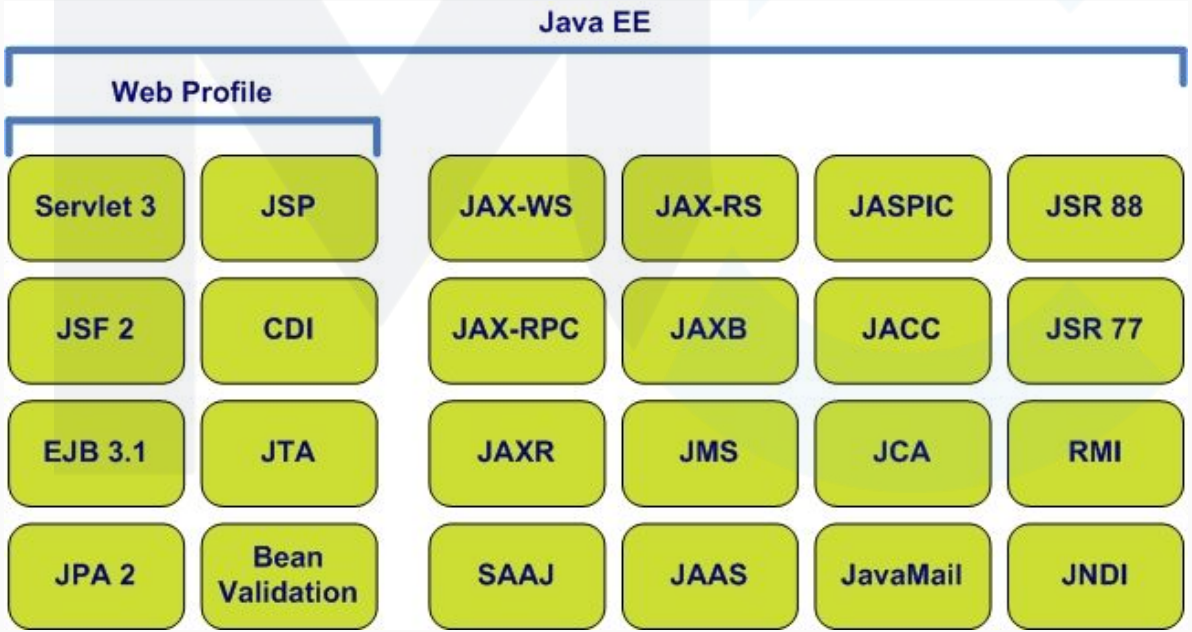
JAVA FULL STACK DEVELOPER

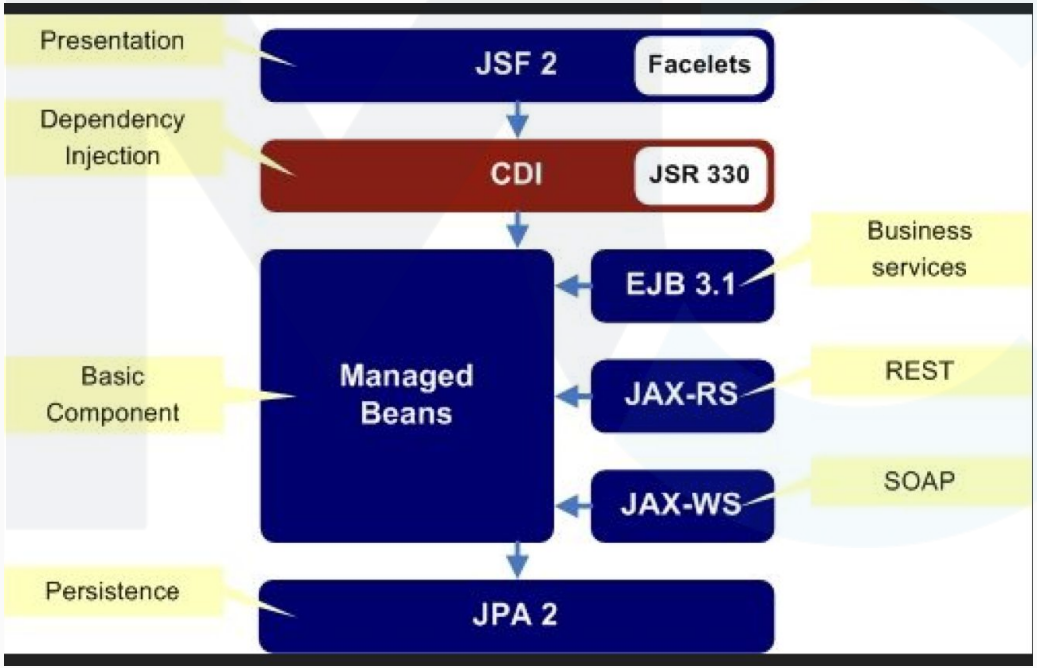
MitoCode Network

¿Back-end vs Front-end?

Hablemos de Java (Java EE | Jakarta)

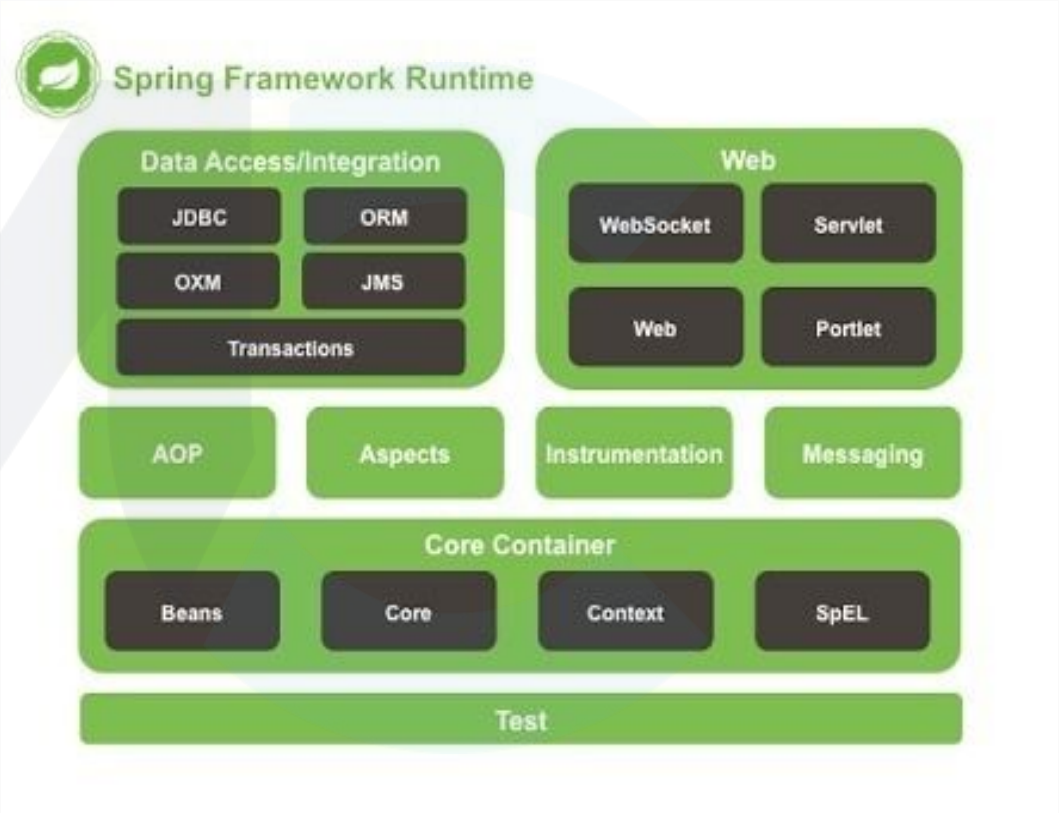


Hablemos de Java (Java EE | Jakarta)

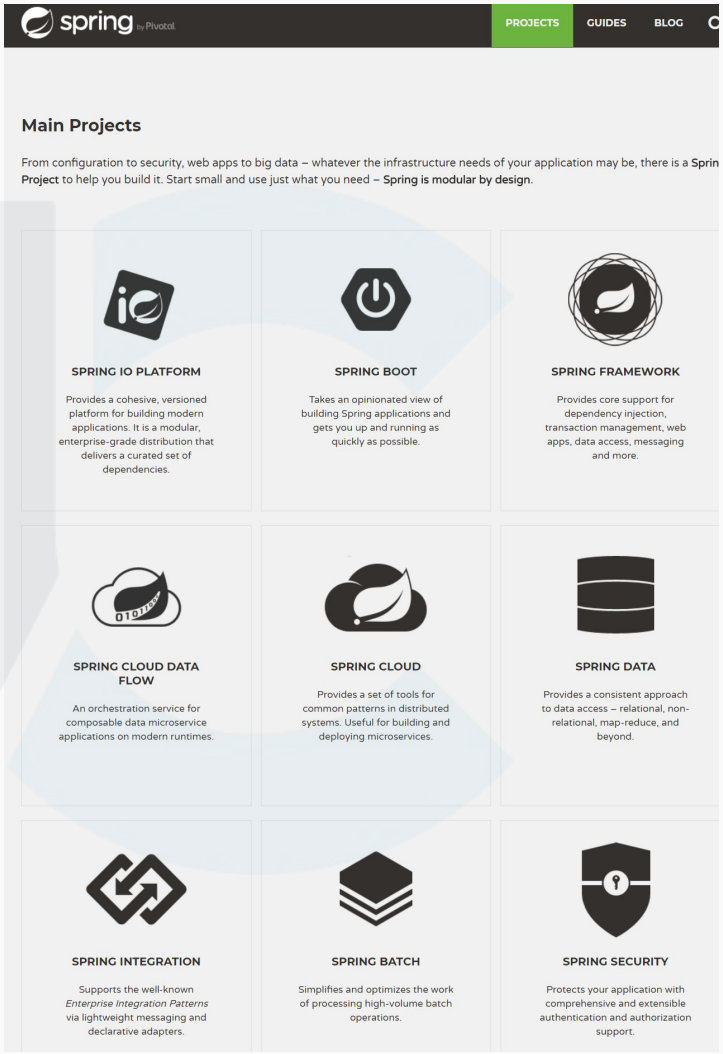


Hablemos de Spring

Spring Framework



Spring Projects

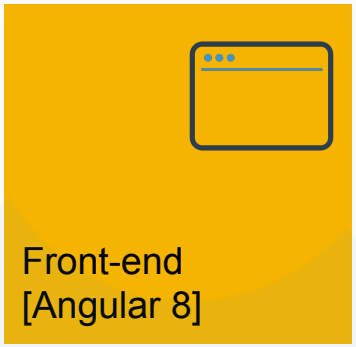


Comparativa

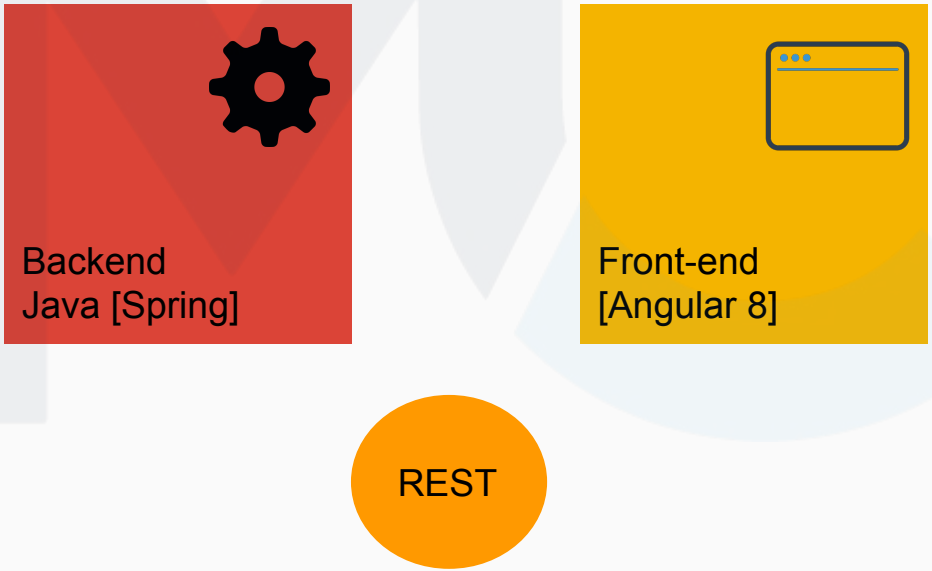
Java EE vs. Spring Framework Features/APIs

Fuente: https://www.slideshare.net/kelapure/2012-0406v2tdp1163java-eevsspringshootoutfinal.

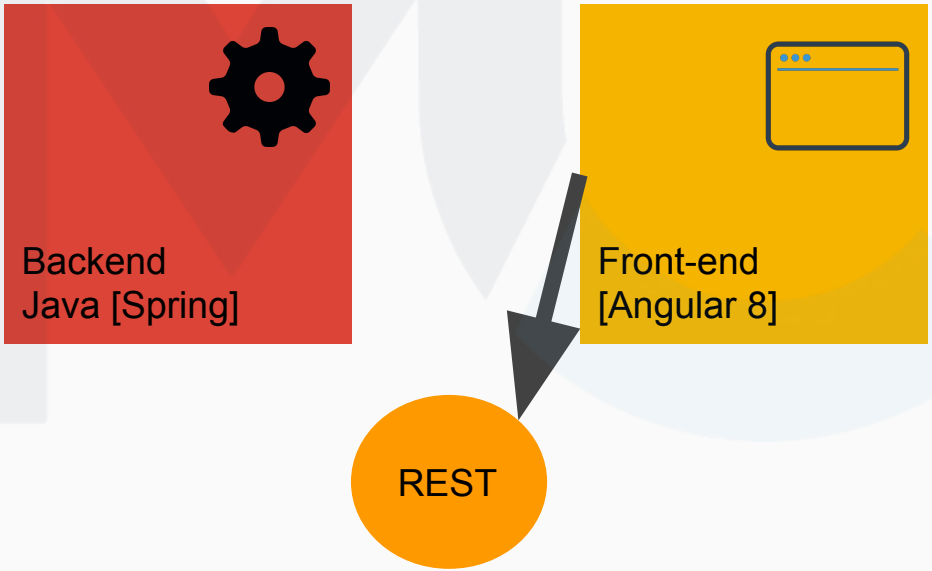
¿Dónde nos centraremos en el curso?



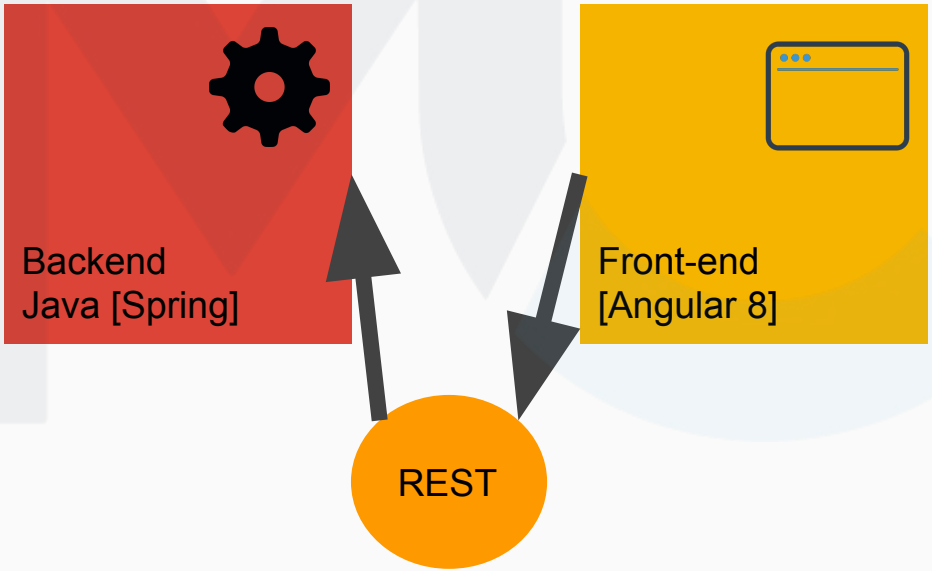
¿Dónde nos centraremos en el curso?



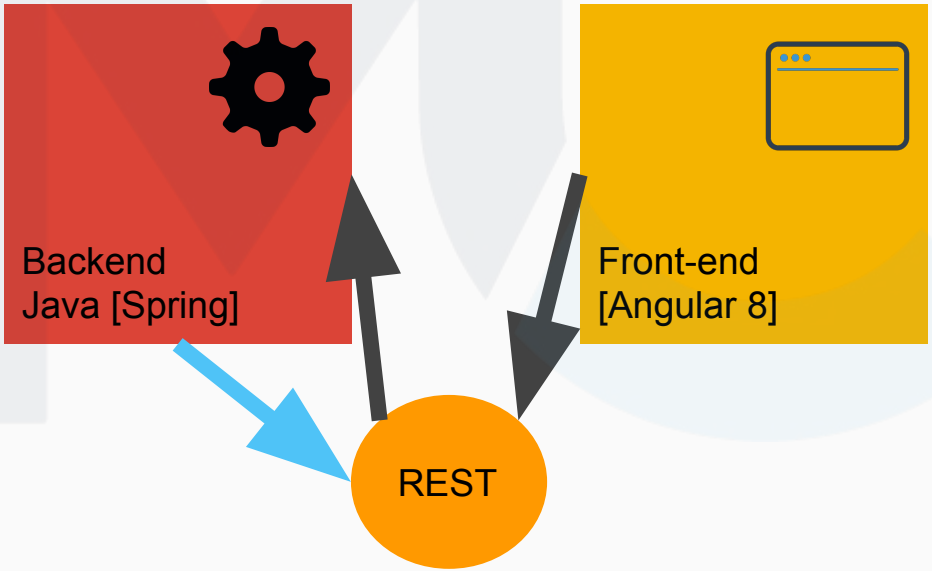
¿Dónde nos centraremos en el curso?



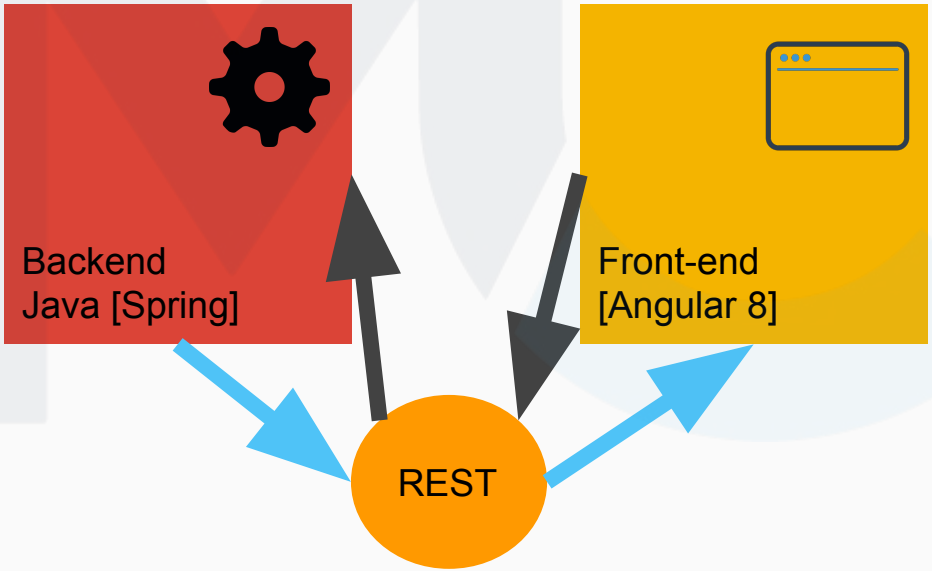
¿Dónde nos centraremos en el curso?



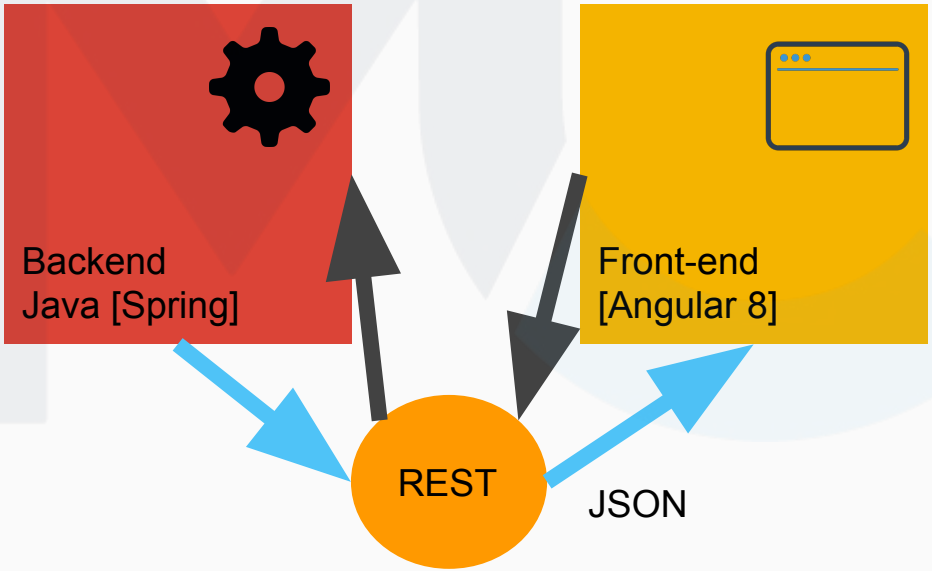
¿Dónde nos centraremos en el curso?



¿Dónde nos centraremos en el curso?



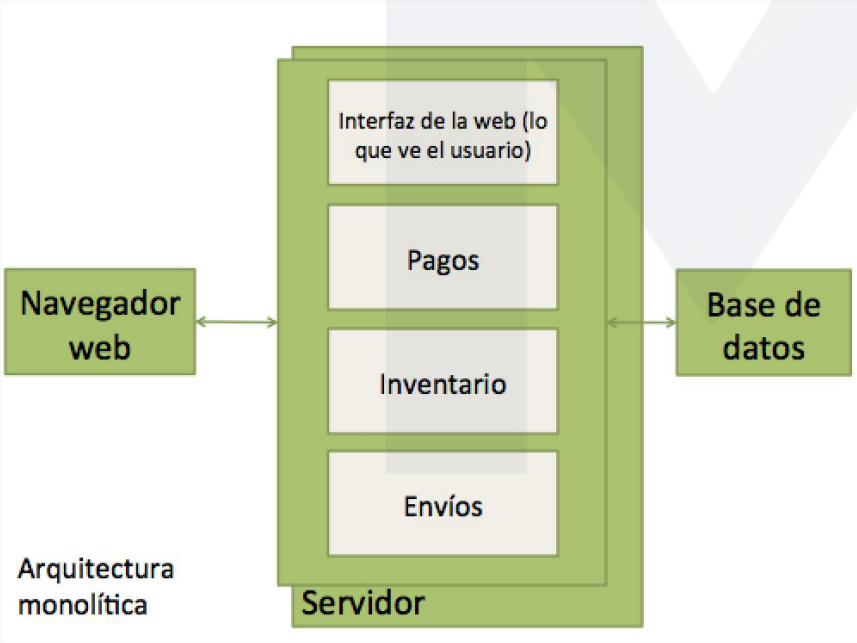
¿Dónde nos centraremos en el curso?



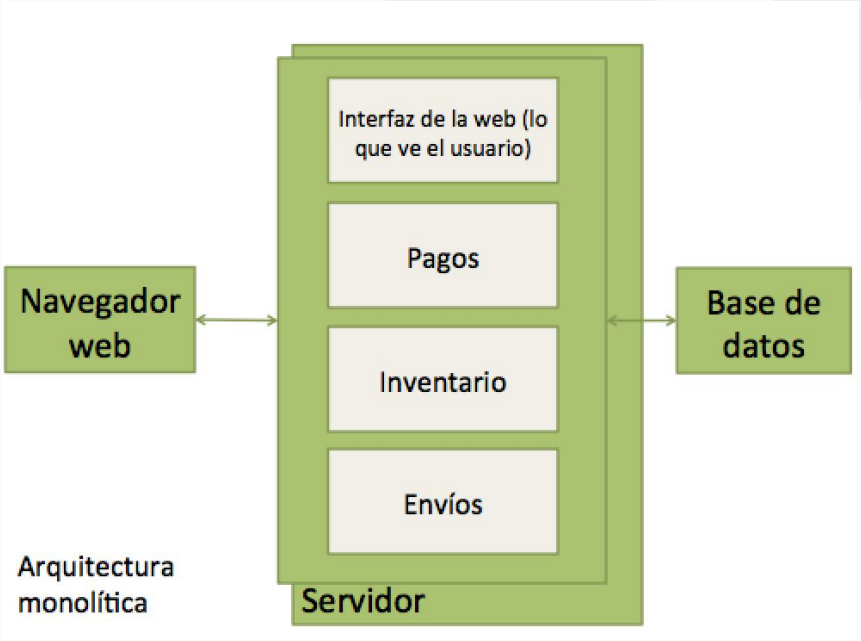
Objetivos del curso

- Crear un backend con Spring [Boot]
  - Servicios REST
  - Protección por JWT
  - Spring Data JPA Repository
  - JSON
  - Enfoque Monolítico
  - Enfoque Microservicios
- Crear un frontend con Angular 8
  - Angular Material
  - Seguridad y comunicación con servicios
- Puesta en producción., despliegue en DigitalOcean

Arquitectura 1 [Monolito]

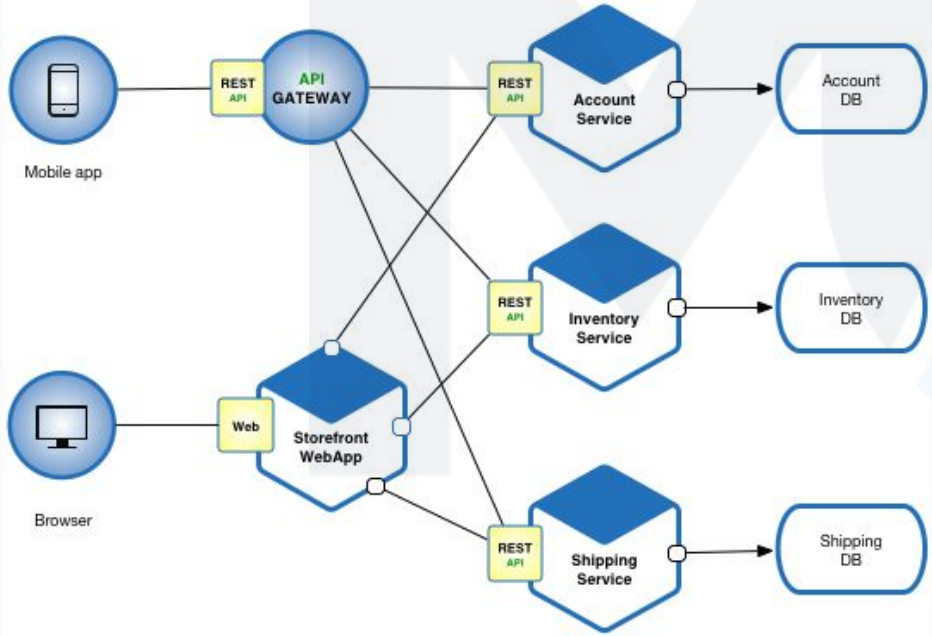


Arquitectura 1 [Monolito]

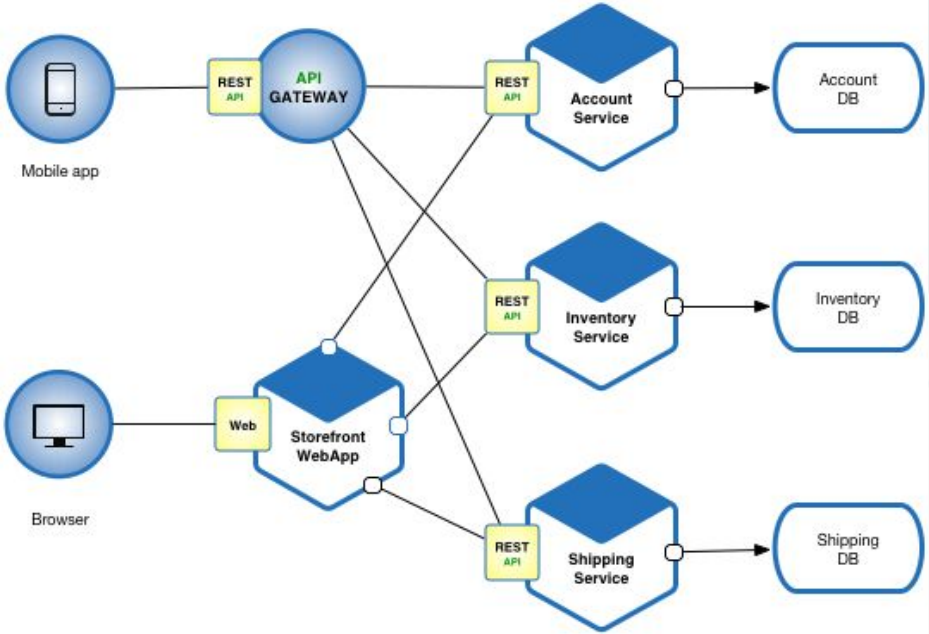


- Simple desarrollo
- Simple de probar
- Simple de desplegar
- Simple de escalar

Arquitectura 2 [Microservicios]

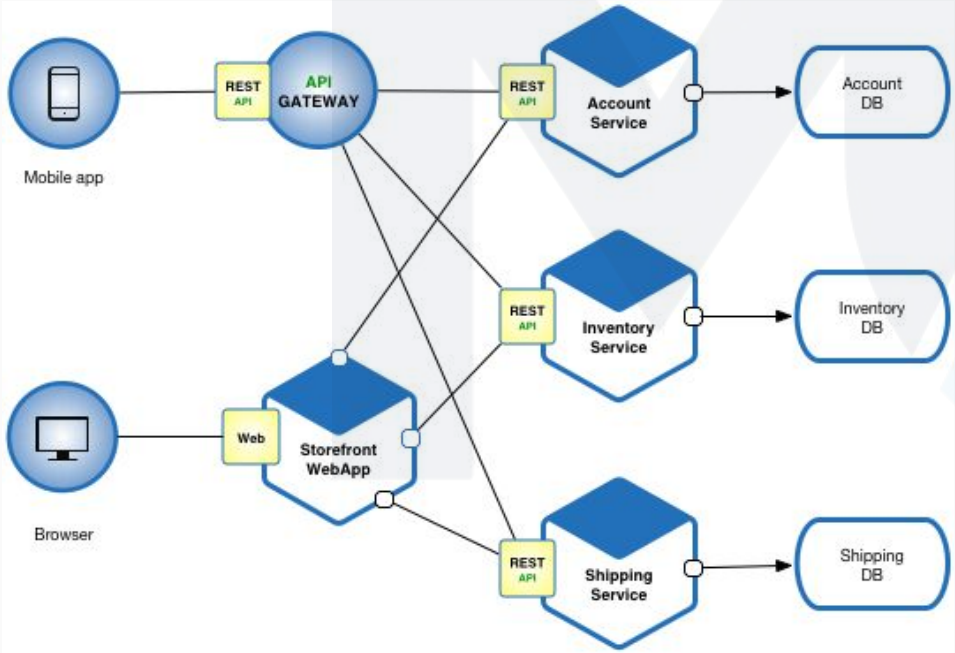


Arquitectura 2 [Microservicios] - Ventajas



- Simple mantenimiento
- Equipo multi-lenguaje (diversidad tecnológica)
- Desarrollos en paralelo

Arquitectura 2 [Microservicios] - Desventajas



- Complejidad de configuración
- Complejidad de seguridad
- Complejidad de gestionar errores