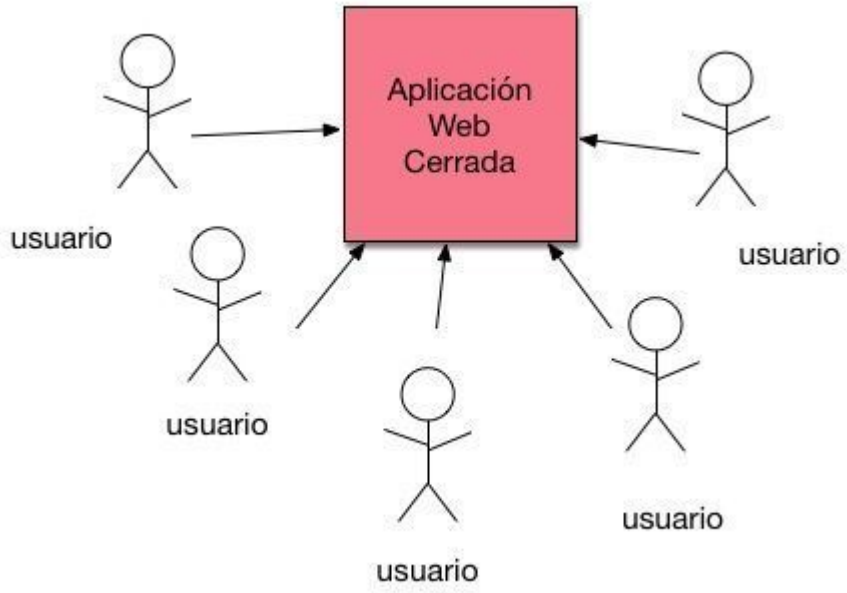
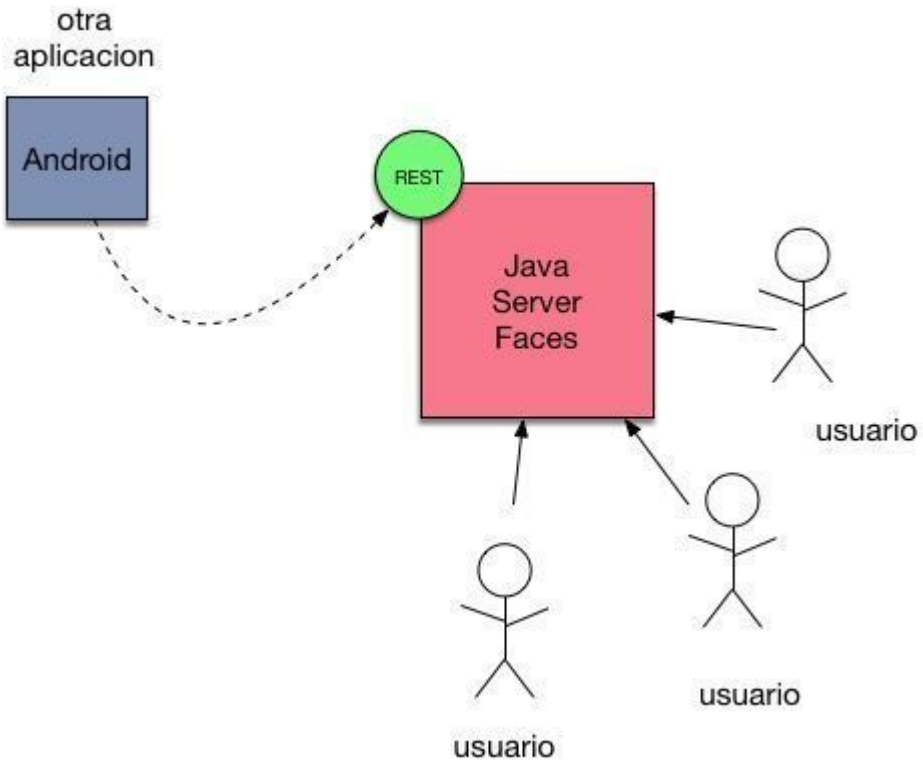
¿ Arquitecturas Web Abiertas o Cerradas? . Esta es una pregunta interesante en nuestros días. Para contestarla debemos saber en primer lugar que consideramos una arquitectura web cerrada. Para mí y este es un tema muy personal es una arquitectura web que no dispone de servicios publicados hacía otras aplicaciones. Por decirlo de otra forma nuestra web es una web orientada al consumo por parte de otros usuarios y no por otras aplicaciones.



Arquitecturas Web Cerradas y sus ventajas

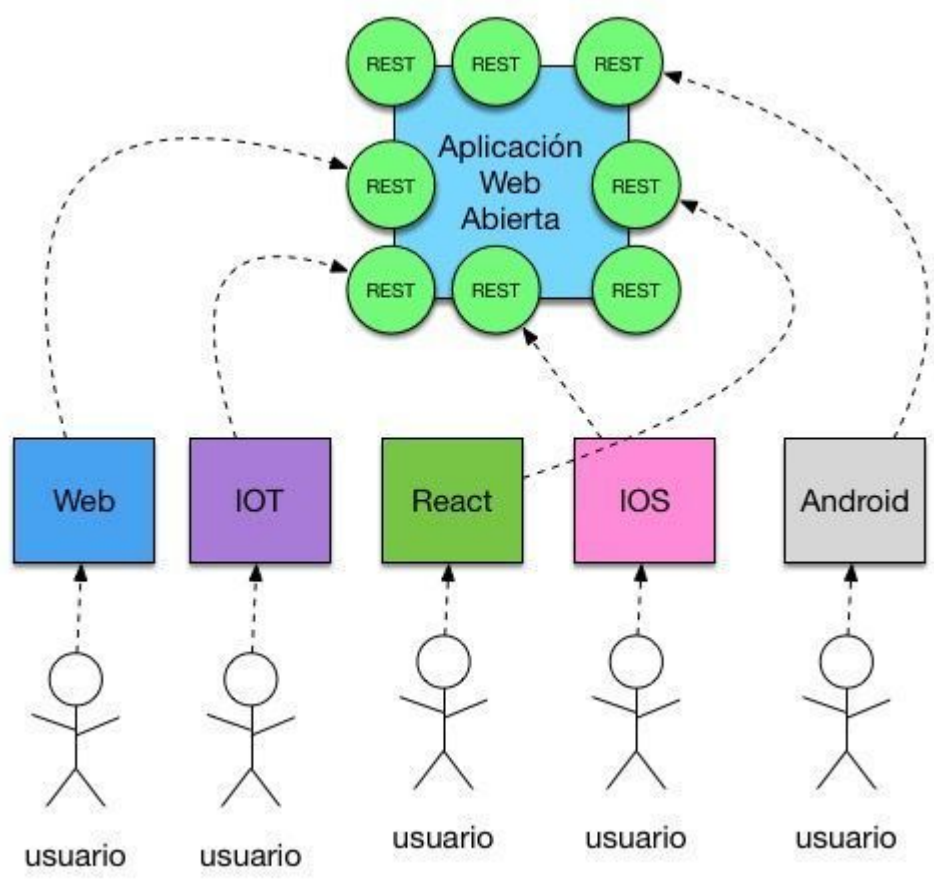
Estas arquitecturas muchas veces están desarrolladas en tecnologías como JSF , o ASP.NET ,PHP o incluso en Spring MVC o ASP.NET MVC , pero no recurren prácticamente en ningún momento a servicios WEB ya sean SOAP o REST.



Tienen sus ventajas ,los desarrolladores están muy acostumbrados a usarlas y los frameworks existentes que las soportan llevan muchos años en el mercado . Son arquitecturas maduras , podemos decir que todo el mundo tiene una fuerte confianza en ellas. Esto tambien implica **que mucha gente no quiera cambiarlas de ninguna manera** . Recordemos que si ninguna aplicación se conecta a la nuestra no hay demasiados problemas en mantener una arquitectura web cerrada.

Arquitecturas Web Abiertas

Las arquitecturas Web Abiertas funcionan al revés y se encargan de publicar gran parte de su información vía servicios usando formatos JSON o similar, dejando a los programadores la construcción de los diferentes clientes ya sea en Angular ,React, Android , NET etc .

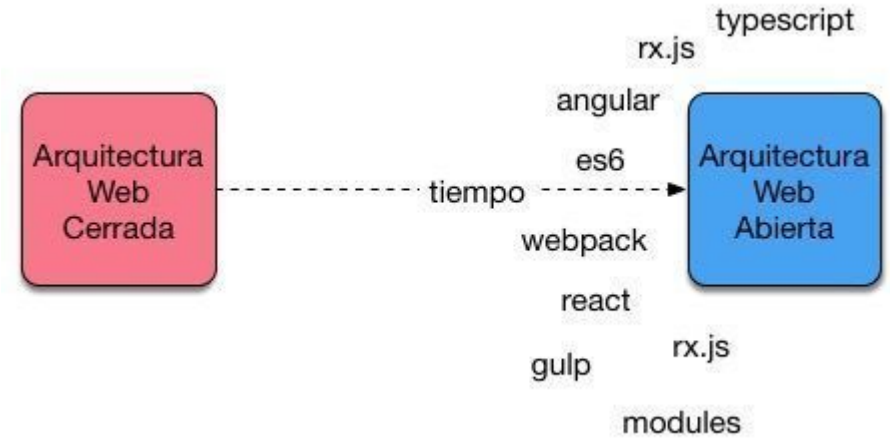


Estas arquitecturas son más flexibles y permiten la integración de todo tipo de clientes. Ahora bien también son arquitecturas más complejas de construir. Hay que abordar nuevas tecnologías , nuevos conceptos de arquitectura , nuevos patrones y nuevas herramientas . Entran en juego muchos más factores que en las arquitecturas cerradas.

Arquitecturas y madurez

Varias personas me han comentado que se encuentran contentos con la arquitectura cerrada que tienen en estos momentos y por ahora no van a evolucionar. Es algo razonable, sin embargo hay un problema importante de fondo. Hoy por hoy existe una tendencia hacia las arquitecturas abiertas. Muchos frameworks y soluciones han aparecido con ese enfoque. Esto de entrada no nos obliga a migrar pero nos obliga a hacernos una pregunta.

¿ Cómo de sencillo es evolucionar de una arquitectura cerrada a una abierta?. La realidad es que no es SENCILLO. Hay que elegir nuevos frameworks , nuevas herramientas de despliegue y sobre todo NUEVOS CONCEPTOS. No es algo que con una formación o un buen libro podamos aprender de golpe. Necesitaremos tiempo para adaptarnos y poder el salto.



Por lo tanto aunque hoy por hoy uno decida no saltar a las arquitecturas web abiertas, es importante ir abordando este tipo de arquitecturas en algunos proyectos . De tal forma que cuando el momento de saltar llegue no nos encontremos superados.

Otros artículos relacionados:

1. React vs Angular 2 , frameworks vs librerias
2. Angular 2 y el futuro de las arquitecturas web
3. Vuejs una alternativa a React y Angular