

Creando aplicaciones Web con AngularJS y JavaEE 7

Víctor Orozco

Nabenik

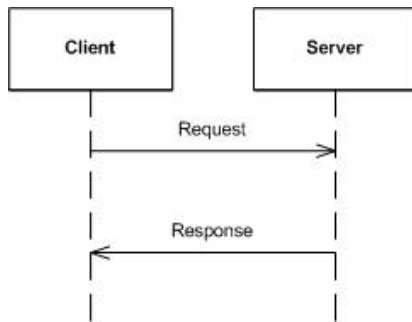
October 13, 2015



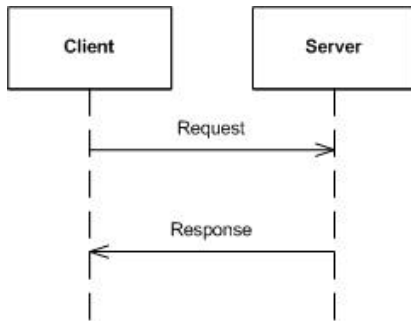
- ▶ Developer (JVM/Open Source Advocate)
- ▶ Consultor independiente (Nabenik)
- ▶ GuateJUG
- ▶ @tuxtor
- ▶ The J*



Cliente/servidor

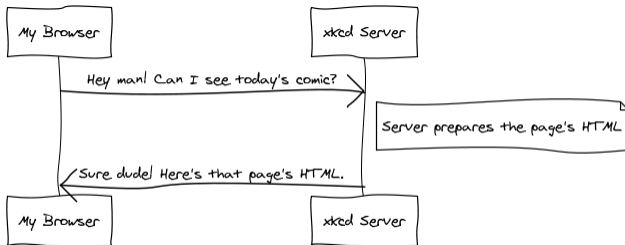


Cliente/servidor



HTTP/1.1 = protocolo asíncrono y sin estado para transmitir texto

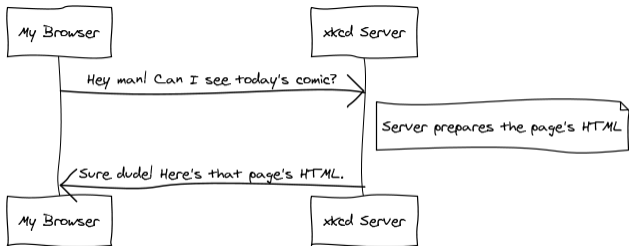
Cliente/servidor



www.websequencediagrams.com



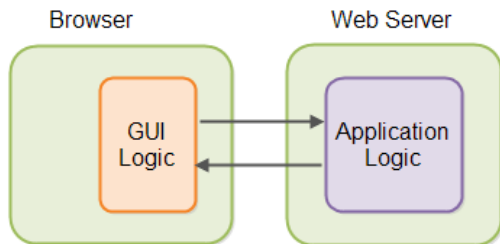
Cliente/servidor



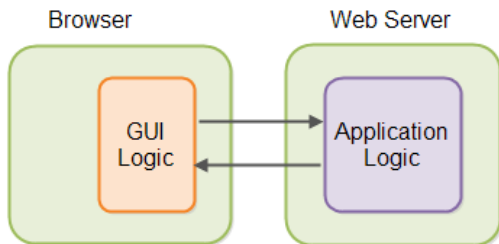
www.websequencediagrams.com

- ▶ Request -> (HTML) -> Response
- ▶ Servidor: PHP, JSP, ASP
- ▶ Servidor Java: JSP/Servlets, JSF, Struts, Spring MVC

Cliente/servidor



Cliente/servidor



- ▶ Rich clients/RIA = obsolescencia?
- ▶ Request -> (App) -> Response
- ▶ Cliente: ActiveX, Applets, Flash, Silverlight, JavaFX

Clientes JavaScript

- ▶ AJAX
- ▶ jQuery, YUI, Dojo

...

- ▶ GWT, Icefaces/Primefaces, Vaadin
- ▶ HTML5, CSS3, WebSockets, WebRTC, HTML Components



Cientes JavaScript

1995-2012: JavaScript SUCKS! - Developer Foo con conocimientos de otro lenguaje que no sea JS.

- ▶ Orientado a hacks
- ▶ Imperativo (manipulación DOM)
- ▶ CoffeeScript, Dart, Kotlin, RapydScript, TypeScript, AtScript
- ▶ MVVM (su buen vecino MS)

2012-2015: JavaScript SUCKS . . . less

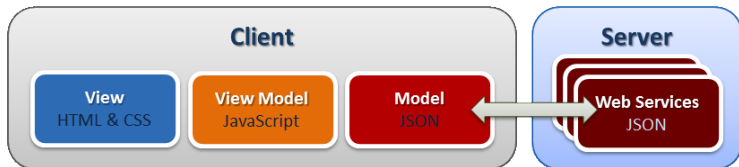


Cientes JavaScript/HTML5

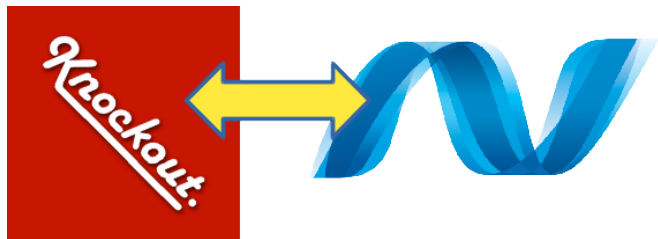
- ▶ Rich clients = HTML+JS+CSS3
- ▶ MVVM +- MVC del lado del cliente
- ▶ JSON/XML
- ▶ Rest - Request-response
- ▶ Websockets - Full duplex

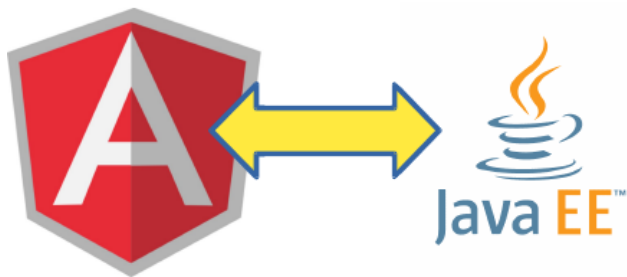


Arquitectura 2015









- ▶ AngularJS fue creado por desarrolladores Java, estamos en familia :) ¹
- ▶ Inyección de dependencias
- ▶ Data binding
- ▶ Directives, partial layouts
- ▶ SPI
- ▶ JS puro (AngularJS 1)
- ▶ Clientes híbridos (móviles) - Cordova + AngularJS

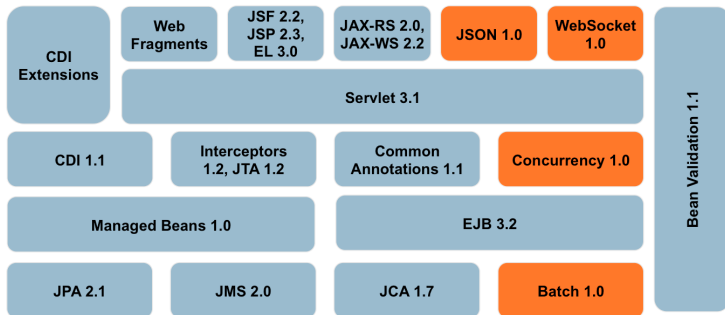


JavaEE 7

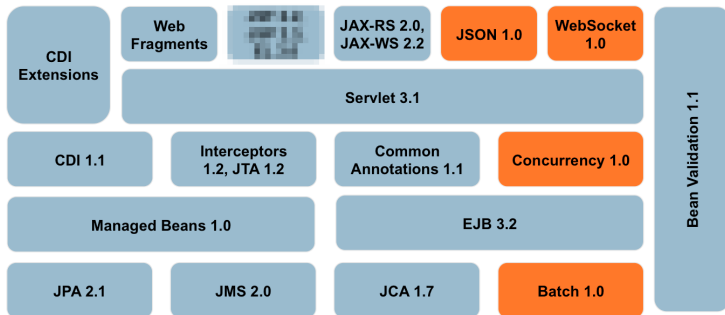
- ▶ API Rest - JAX-RS 2.0
- ▶ WebSocket - WebSocket 1.0, Servlet 3.1
- ▶ JSON - JSON API 1.0
- ▶ SOA, Microservices



JavaEE 7



JavaEE 7



Ventajas

- ▶ Existen n cantidad de bibliotecas JavaScript
- ▶ Independencia de backend
- ▶ Escalabilidad (stateless)
- ▶ Thin server apps
- ▶ Mejor tiempo de respuesta en comparación a JSF/SpringMVC



Desventajas

- ▶ Existen n cantidad de bibliotecas JavaScript
- ▶ Complejidad y restricciones de REST
- ▶ AngularJS no sera compatible hacia atrás



- ▶ Call for papers
- ▶ H2 + WildFly
- ▶ Bean Validation, JPA, JAX-RS, JSON
- ▶ AngularJS vanilla
- ▶ Forge
- ▶ <http://github.com/tuxtor/cfp-angularjs-demo>



- ▶ AngularJS - <https://angularjs.org/>
- ▶ JavaEE - <http://docs.oracle.com/javaee/7/index.html>
- ▶ Libros recomendados:
 - ▶ Java EE 7 Essentials - Arun Gupta
 - ▶ Developing RESTful Services with JAX-RS 2.0 - Masoud Kalali, Bhakti Mehta
 - ▶ Eloquent JavaScript - Marijn Haverbeke



Gracias

- ▶ tuxtor@shekalug.org
- ▶ <http://tuxtor.shekalug.org>
- ▶ <http://github.com/tuxtor/slides>



This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 Guatemala License.

