



Presentación Experto Java

Materiales, metodología, contenidos

Índice

- Materiales y recursos
- Objetivos y metodología
- Módulos y contenidos



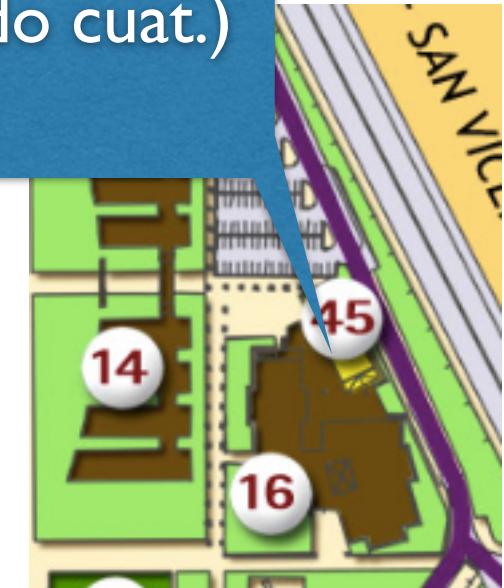
Presentación del Experto

- **Título Propio de la Universidad de Alicante** de 20 créditos ECTS
- **Fechas**
 - 17 octubre de 2015 - 23 de mayo de 2016
- **Horario**
 - Viernes: 16:00 - 21:00
 - Sábado: 9:00 - 14:00
- **Información del curso**
 - <http://web.ua.es/expertojava>
- **Recursos y materiales** de las asignaturas
 - Apuntes Experto Java (<http://expertojava.ua.es>)
 - Moodle de Campus Virtual (<http://www.ua.es>)
- **Redes sociales**
 - <https://twitter.com/expertojavaua>



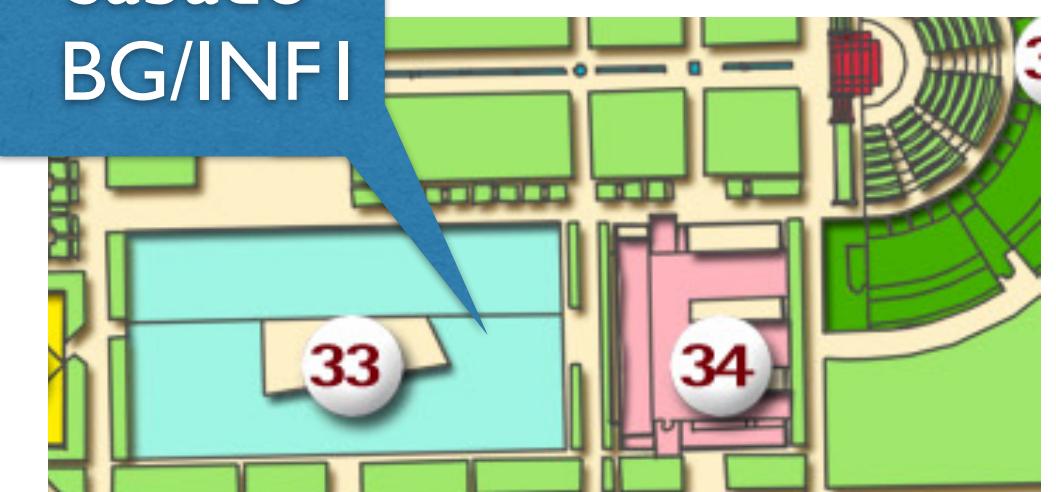
Calendario detallado
en Moodle y en la web
del experto

Viernes
L17 (primer cuat.)
L22 (segundo cuat.)



EPS I

Sábado
BG/INF1



Biblioteca General

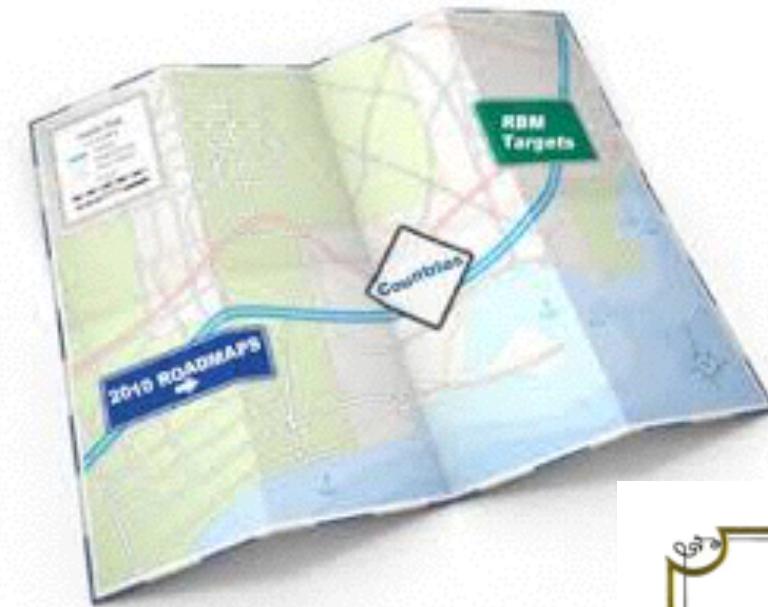


Objetivos del curso

- Ofrecer una **formación básica y sólida** en las principales tecnologías de desarrollo de aplicaciones web con Java EE, JavaScript y otras tecnologías alternativas cercanas al mundo Java
- Proporcionar un **roadmap** para acometer con garantías de éxito el aprendizaje de Java EE, frameworks JavaScript y otras tecnologías relacionadas
- Proporcionar guías, ejemplos y modelos de desarrollo de estas aplicaciones

A flowchart titled "Overview the_post() function" showing the execution of the `the_post()` function. It starts with "Set post variable", then "Set in loop to true", followed by "Check if loop started". If the loop has just started, it executes `do_action_ref_array('loop_start', array(&$this))`. Then it sets `$post = $this->next_post();` and calls `setup_postdata($post);`. Finally, it ends with "Setup the_post function".

```
graph TD; Start[Set post variable] --> SetInLoop[Set in loop to true]; SetInLoop --> CheckLoop[Check if loop started]; CheckLoop --> LoopStart[if ($this->current_post == -1) // loop has just started  
do_action_ref_array('loop_start', array(&$this));]; LoopStart --> NextPost[$post = $this->next_post();]; NextPost --> SetupPostdata[setup_postdata($post)]; SetupPostdata --> End[Setup the_post function];
```





Certificaciones Oracle

- Puedes encontrar más información en la [página de Oracle sobre registro para certificación](#).
- En las certificaciones Java EE los exámenes suelen durar entre 90 y 130 minutos y en la actualidad cuestan todos ellos 190 €.
- La certificación de *Java SE Programmer* es requisito previo para cualquier certificación Java EE:
 - Oracle Certified Professional, Java SE 6 Programmer
- Existen cinco certificaciones Java EE 6. Son independientes y es posible acceder a cualquiera de ellas por separado.

Oracle Certified Expert,
Java EE 6 Web Component
Developer

Oracle Certified Expert,
Java Platform, Enterprise
Edition 6 Enterprise
JavaBeans Developer

Oracle Certified Expert,
Java EE 6 Web Services
Developer

Oracle Certified Expert,
Java EE 6 JavaServer
Faces Developer

Oracle Certified Expert,
Java EE 6 Java
Persistence API Developer

Java EE 6 Web Component Developer Certified Expert [New & Upcoming Releases](#) : [Print this Exam](#) :

Exam Number :	1Z0-899	Duration:	140 minutes
Associated Certifications:	Oracle Certified Expert, Java EE 6 Web Component Developer	Number of Questions:	57
Exam Product Version:	Java EE,	Passing Score	64% View passing score policy
Exam Price:	€ 193 More on exam pricing	Validated Against:	This exam has been validated against EE 6.
		format:	Multiple Choice



Formación en tecnologías



JavaScript



BACKBONE.JS



React





Materiales

- Apuntes de cada módulo y guías de ejercicios
- Web de apuntes con el contenido completo del curso
- Máquina virtual Ubuntu Linux
- Disco externo SSD USB 3.0 de 120 GB



The screenshot shows a web browser window for the URL expertojava.ua.es. The page title is "Componentes Web". On the left, there's a sidebar with "Duración" (20 horas (8 sesiones)) and "Profesor" (Miguel Ángel Lozano Ortega). The main content area has a section titled "Contenidos" which describes Java EE components and their functionality. Below this, there's a "Libro de apuntes" section with PDF, HTML, and ePub download links. The bottom part of the page lists five sessions with "Apuntes y ejercicios" and "Diapositivas" buttons.

Sesiones	Apuntes y ejercicios	Diapositivas
1. Introducción a las aplicaciones web	Apuntes y ejercicios	Diapositivas
2. Procesamiento de peticiones	Apuntes y ejercicios	Diapositivas
3. Manejo de cookies y sesiones	Apuntes y ejercicios	Diapositivas
4. Contexto global de la aplicación web	Apuntes y ejercicios	Diapositivas
5. Websocket	Apuntes y ejercicios	Diapositivas



Web apuntes expertojava

- <http://expertojava.ua.es>
- Apuntes (html, pdf, ePub)
- Ejercicios (html, pdf)
- Diapositivas (pdf, html)
- Acceso restringido



The screenshot shows the homepage of the 'EXPERTO JAVA' website. At the top right is the logo 'EXPERTO JAVA' with 'UNIVERSIDAD DE ALICANTE' underneath. Below the logo are three main sections: 'Otras webs' (Moodle, Web UA), 'Curso 2015-16' (Java EE 7, JavaScript, Tecnologías alternativas, Proyecto de aplicación Web), and 'Cursos pasados' (with a list of past courses like Lenguaje Java Avanzado, JavaServer Faces, etc.). At the bottom is a copyright notice: © 2014-16 Departamento de Ciencia de la Computación e Inteligencia Artificial, Universidad de Alicante.



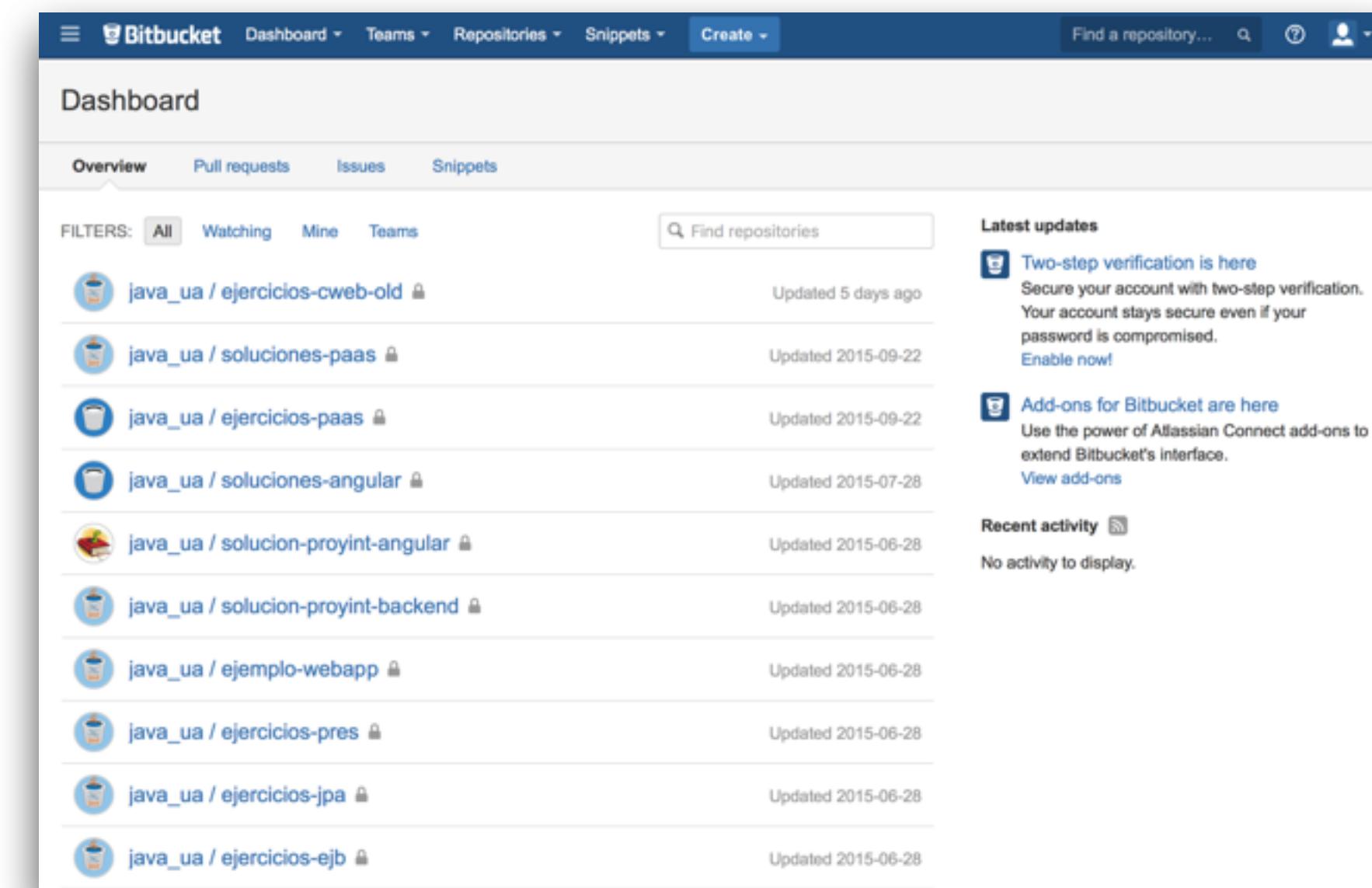
Moodle CV

- <http://www.ua.es>
- Foro de dudas
- Entrega de ejercicios
- Calificaciones y evaluación

The screenshot shows the Moodle course interface for the '2015-16_EXPERTO EN DESARROLLO DE APLICACIONES WEB CON JAVA EE Y JAVASCRIPT' course. The top navigation bar includes links to 'Página Principal', 'Mis cursos', 'Expertos Universitarios', and 'JAVA_EE_JS'. The left sidebar contains sections for 'Personas' (Participants), 'Usuarios en línea' (Users online), 'Administración' (Administration) with options for 'Calificaciones' (Grades), 'Cambiar rol a...' (Change role to...), and 'Ajustes de mi perfil' (My profile settings); it also has a 'Navegación' (Navigation) section. The main content area features a 'Repaso lenguaje Java' (Java language review) section with text about the course and a link to 'Módulo Lenguaje Java Avanzado curso 2012-12'. On the right, there is a 'Calendario' (Calendar) showing the month of October 2015, a 'CLAVE DE EVENTOS' (Event key) for filtering calendar items, and a 'Buscar en los foros' (Search forums) search bar.

Bitbucket

- <https://bitbucket.org/>
- Cuenta de bitbucket académica con repositorios git para cada módulo
 - Proyectos iniciales y plantillas
 - Soluciones
- Cuentas de alumnos donde se irán subiendo los ejercicios realizados



The screenshot shows the Bitbucket dashboard for the user 'java_ue'. The top navigation bar includes 'Bitbucket', 'Dashboard', 'Teams', 'Repositories', 'Snippets', 'Create', and search/filter options. The main area is titled 'Dashboard' with tabs for 'Overview', 'Pull requests', 'Issues', and 'Snippets'. A 'FILTERS' dropdown is set to 'All'. Below is a list of repositories, each with a small icon, the repository name, and the last update date:

Repository	Last Update
java_ue / ejercicios-cweb-old	Updated 5 days ago
java_ue / soluciones-paas	Updated 2015-09-22
java_ue / ejercicios-paas	Updated 2015-09-22
java_ue / soluciones-angular	Updated 2015-07-28
java_ue / solucion-proyint-angular	Updated 2015-06-28
java_ue / solucion-proyint-backend	Updated 2015-06-28
java_ue / ejemplo-webapp	Updated 2015-06-28
java_ue / ejercicios-pres	Updated 2015-06-28
java_ue / ejercicios-jpa	Updated 2015-06-28
java_ue / ejercicios-ejb	Updated 2015-06-28

On the right side, there are sections for 'Latest updates' (including 'Two-step verification is here' and 'Add-ons for Bitbucket are here') and 'Recent activity' (which is currently empty).

Metodología y desarrollo del curso

- 10 asignaturas que explican en profundidad tecnologías Java EE, JavaScript y alternativas
 - Se aprende haciendo
 - Ejercicios guiados
 - Ejercicios más abiertos
 - Mini-proyectos
 - Integración en el módulo Proyecto web: biblioteca
- **Sesiones teórico-prácticas**
 - Se intercalan explicaciones teóricas y ejercicios
 - Muchos ejercicios guiados
 - Abundante código fuente de ejemplo
- **Proyecto de aplicación web**
 - Se aplican los contenidos de la asignatura
 - El proyecto se construye de forma incremental

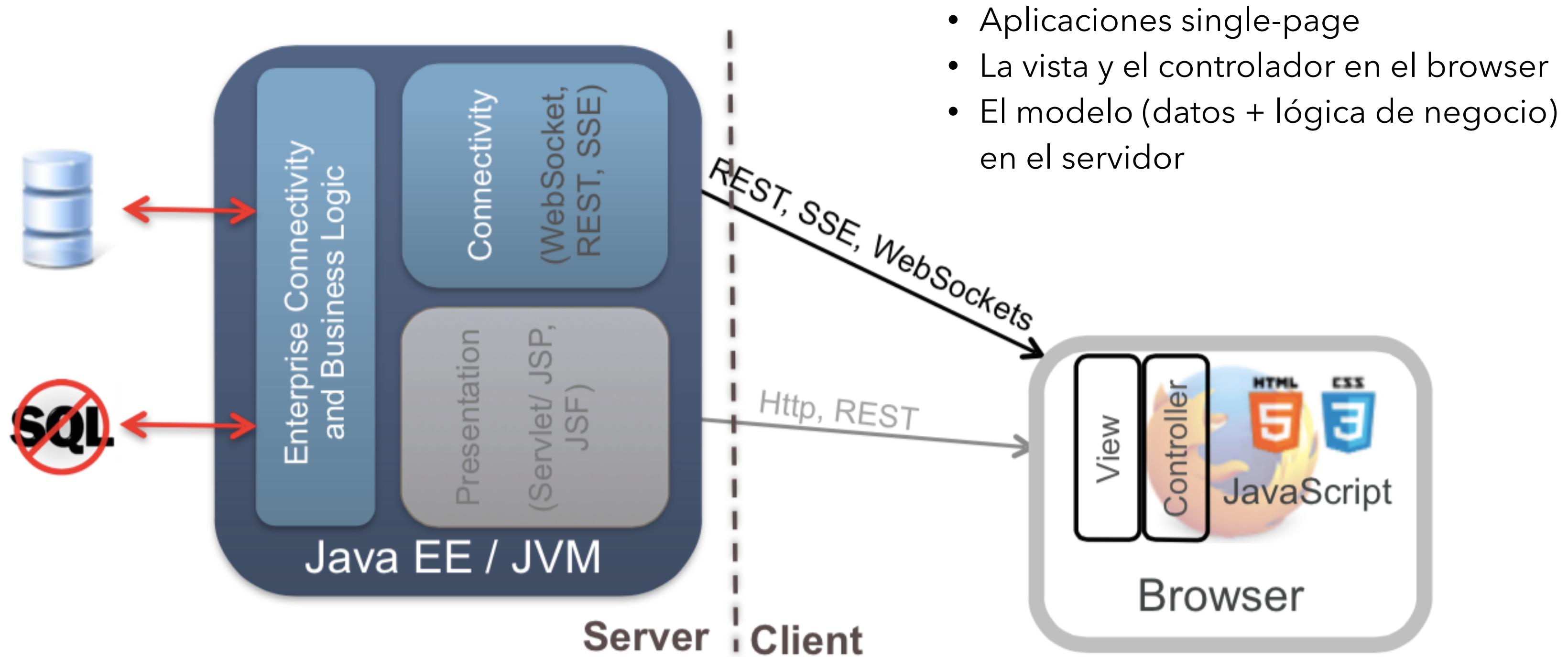
Calendario de entregas de ejercicios y evaluación

- Asistencia **obligatoria** (se permiten 25% de faltas)
- **Asignaturas cuatrimestrales** con actas independientes, como en un título oficial, que se firmarán en el cuatrimestre de la asignatura
- Proyecto de aplicación web: asignatura **anual**
- Intentaremos dejar disponibles en Moodle las calificaciones de cada asignatura conforme van terminando
- Nota de cada asignatura basada en los ejercicios de las sesiones
 - Cada ejercicio tiene una puntuación asignada, todas las puntuaciones suman 10
 - Desde el principio se sube a bitbucket el repositorio con los ejercicios y se le da permiso de lectura al profesor
 - Se habilitará en Moodle una tarea para confirmar la entrega y entregar la corrección
- Cuando sea posible, se realizará un descanso el sábado previo a las entregas de ejercicios y se habilitará una sesión de tutorías.

De forma excepcional se pueden entregar los ejercicios en un plazo extraordinario al final del cuatrimestre. En este caso se deberán realizar ejercicios por una puntuación de 7 y como máximo se podrá conseguir una calificación de 5.



¿Cómo es una aplicación web con Java EE y JavaScript?





Bloques

- Componentes Web
- JPA - Frameworks de persistencia
- EJB - Componentes enterprise
- Servicios REST
- Interfaz de usuario en dispositivos móviles
- Servidores Web y PaaS

Java EE
80 horas

- Lenguaje JavaScript
- Frameworks JavaScript - Backbone y React
- AngularJS - Frameworks JavaScript(II)

JavaScript
60 horas

- Framework Grails
- Bases de datos NoSQL

Alternativas
30 horas

- Proyecto de Aplicación Web

PAW
30 horas



Profesorado

- Profesores depto. DCCIA:
 - Isabel Alfonso Galipienso - eli@dccia.ua.es
 - Otto Colomina Pardo - otto@dccia.ua.es
 - Domingo Gallardo López - domingo@dccia.ua.es
 - Miguel A. Lozano Ortega - malozano@dccia.ua.es
- Profesores externos:
 - Francisco García Rico - fgarcia@ua.es
 - Aitor Medrano Escrig - aitormedrano@gmail.com
 - Alejandro Such Berenguer - alejandro.such@gmail.com
 - José Luis Zamora Sánchez - joseluiszamora.jlz@gmail.com

Java EE 7 - Web Profile

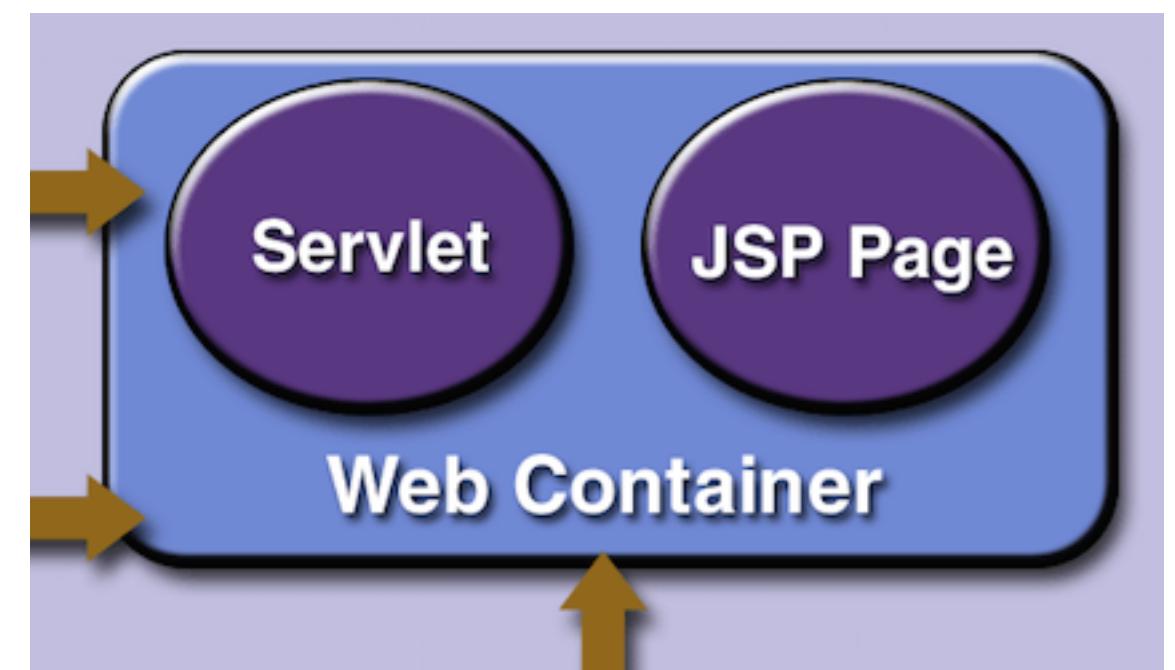
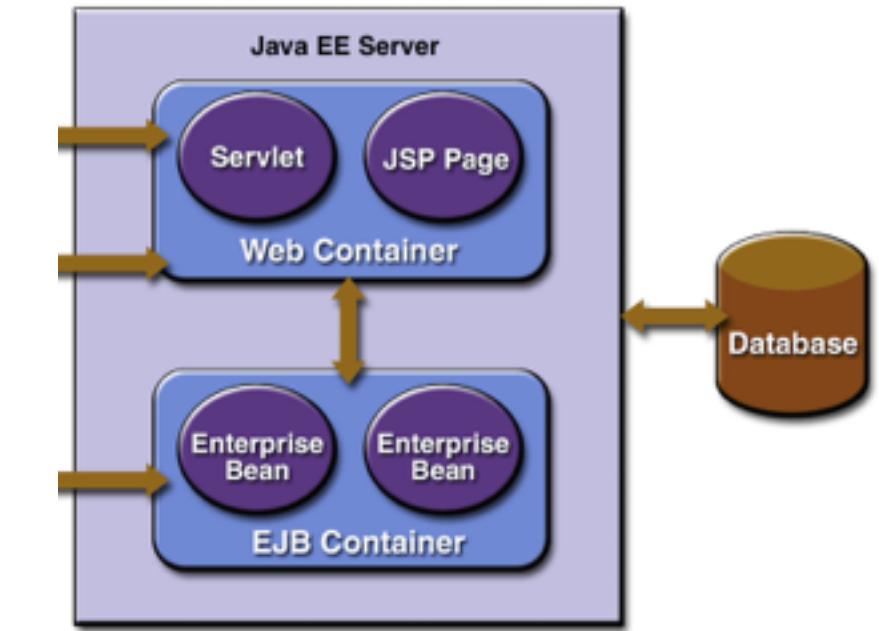


Asignatura **Componentes Web**

- Profesor: Miguel Ángel Lozano
- Calendario: del 17 al 30 de octubre

Aplicaciones web Java EE

- Aplicaciones a las que se accede mediante **protocolo HTTP**
- Tienen una **estructura estándar** que podrá utilizarse en cualquier servidor de aplicaciones Java EE
- Contienen **componentes Java EE**
- Servlets
 - Son **clases Java** que se ejecutarán cuando el servidor reciba una determinada petición HTTP
 - Pueden ejecutar cualquier código Java (por ejemplo para establecer **conexiones a bases de datos**)
- Facelets
 - Páginas **XHTML** que pueden incluir **librerías de etiquetas y lenguaje de expresiones**
 - Permiten separar la **vista** del modelo
 - El modelo y la lógica de negocio se llevará a otros componentes: beans, taglibs, etc
- WebSocket endpoints
 - Establecen un **canal de comunicación bidireccional** entre cliente y servidor
- Filtros
 - **Interceptan** las peticiones HTTP realizadas al servidor de aplicaciones



Asignatura **Componentes Web**

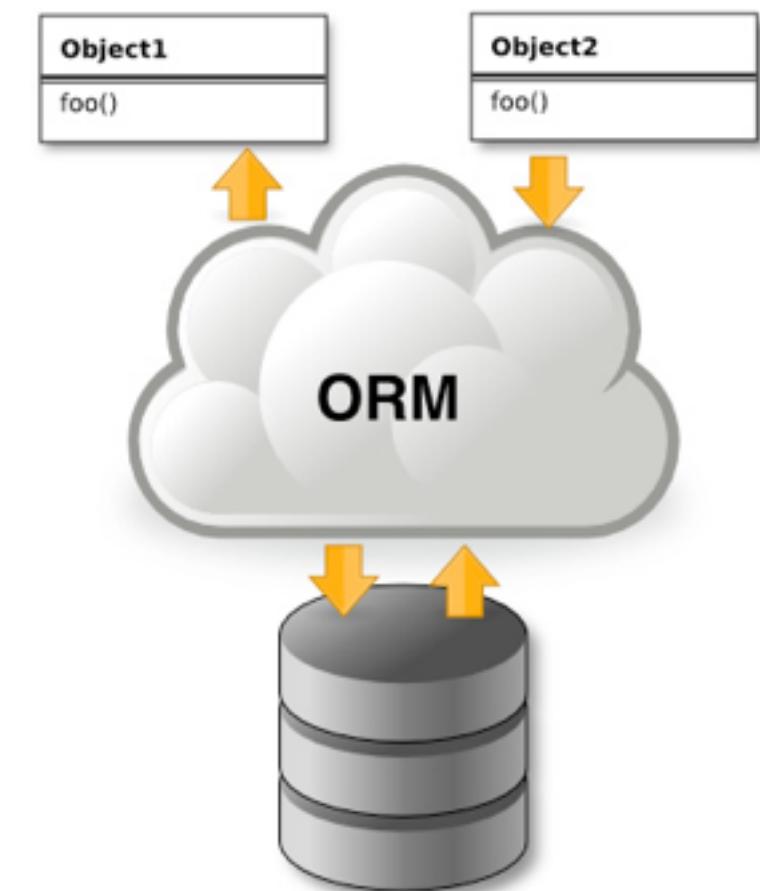
- 8 sesiones (20 horas, 2 créditos ECTS):
 1. Introducción a las aplicaciones web
 2. Procesamiento de peticiones
 3. Manejo de cookies y sesiones
 4. Contexto global de la aplicación
 5. *WebSocket*
 6. Seguridad en aplicaciones web
 7. Filtros y wrappers
 8. *Facelets*, JSTL y lenguaje de expresiones

Asignatura *JPA - Frameworks de Persistencia*

- Profesor: Domingo Gallardo
- Calendario: del 13 al 20 de noviembre

¿Qué es JPA?

- Java Persistence API: API estándar de Java para la gestión de objetos persistentes
- Evoluciona a partir de Hibernate (proyecto open source) y se introduce en Java EE 5 (2006)
- Clases y objetos persistentes (entidades)
- Los objetos persistentes se mapean con la base de datos:
 - Las clases se mapean con tablas
 - Los objetos se mapean con filas de las tablas
- Las operaciones habituales de una BD (creación, actualización y búsqueda) se realizan mediante la creación y actualización de objetos de clases persistentes
- Lenguaje JPQL para la realización de consultas
- Mapeo objetos-relacional (ORM): el API gestiona una BD relacional “de toda la vida”. Todas las operaciones sobre los objetos se transforman en sentencias SQL.



Asignatura *JPA - Frameworks de Persistencia*

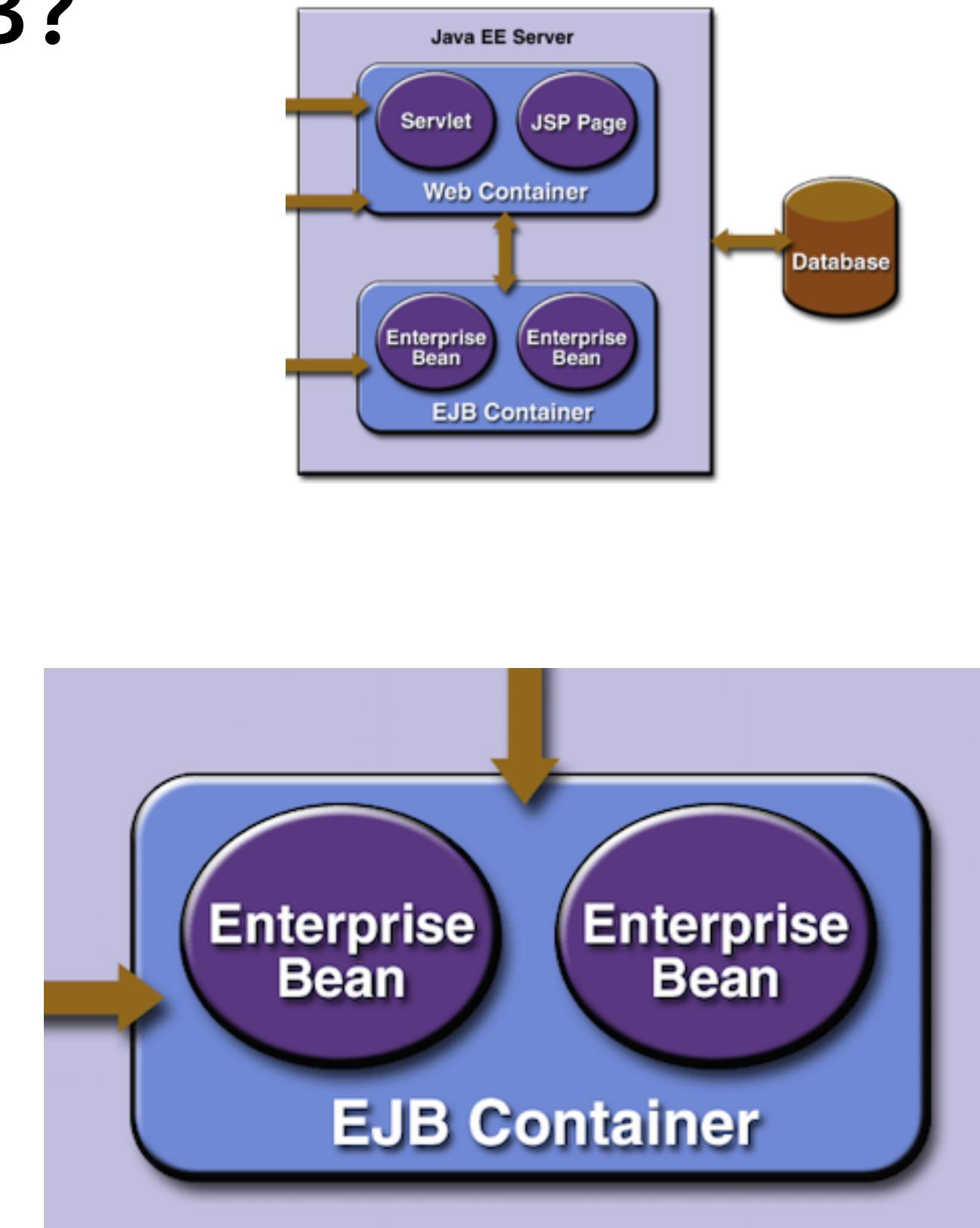
- 6 sesiones (15 horas, 1,5 créditos ECTS):
 1. Introducción a JPA
 2. Entity Manager y contexto de persistencia. Pruebas con DB-Unit.
 3. Mapeado entidad-relación: tablas. Bean validation.
 4. Mapeado entidad-relación: relaciones
 5. Consultas JPQL, API criteria
 6. Transacciones y concurrencia

Asignatura *Componentes Enterprise*

- Profesores: Domingo Gallardo
- Calendario: del 21 al 28 de noviembre

¿Para qué sirven los componentes EJB?

- Un componente EJB (o Enterprise JavaBean) es un **componente gestionado** por el servidor de aplicaciones que puede ser utilizado de forma local o remota desde otros componentes de la aplicación (servlets u otros EJB)
- Implementa la **lógica de los servicios** y de la aplicación, proporcionando una capa separada de la capa que responde a las peticiones HTTP
- Proporciona servicios relacionados con el servidor de aplicaciones:
 - Seguridad
 - Transaccionalidad
 - Conurrencia
 - Escalabilidad



Asignatura **Componentes Enterprise**

- 5 sesiones (12,5 horas, 1 crédito ECTS):
 1. Introducción a los enterprise beans. Beans de sesión sin estado. Pruebas.
 2. Tipos de beans y acceso concurrente.
 3. Enterprise beans y JPA. Transacciones gestionadas por el contenedor.
 4. Temporizadores. Seguridad.
 5. Interceptores. Invocación asíncrona de métodos.

Asignatura *Servicios REST*

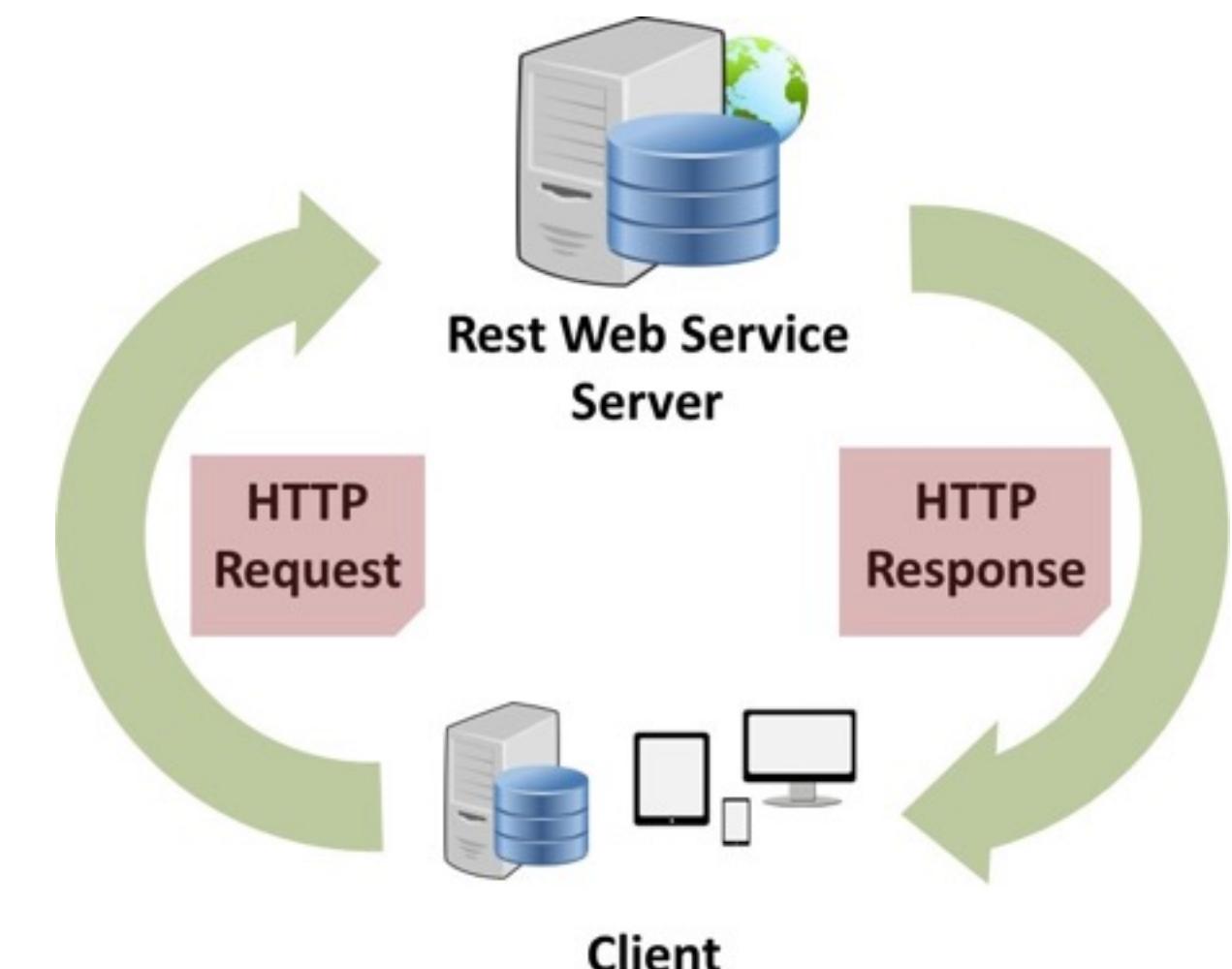
- Profesora: María Isabel Alfonso
- Calendario: del 11 al 18 de diciembre



¿Qué son los servicios web RESTful?

- Los servicios Web fueron “inventados” para solucionar el problema de la **INTEROPERABILIDAD** de las aplicaciones
 - La Web y el protocolo HTTP pueden utilizarse para implementar los servicios
- Los servicios web RESTful se adhieren a un conjunto de **restricciones** “arquitectónicas” y utilizan estándares Web, como URI, HTTP, XML y JSON
- Restricciones **REST** (**REST** = **R**Epresentational **S**tate **T**ransfer)
 - Sistema cliente-servidor
 - Sistema por capas
 - Sin estado
 - Caché
 - Interfaz uniformemente accesible
 - Recursos, Representaciones, URIs, Mensajes auto-descriptivos, HATEOAS

INTEROPERABILIDAD



Asignatura *Servicios REST*

- 5 sesiones (12,5 horas, 1,5 créditos ECTS):
 1. Introducción a REST. Diseño y creación de servicios RESTful
 2. Anotaciones básicas JAX-RS. El modelo de despliegue
 3. Manejadores de contenidos. Respuestas del servidor y manejo de excepciones
 4. HATEOAS. Seguridad
 5. Procesamiento JSON y pruebas

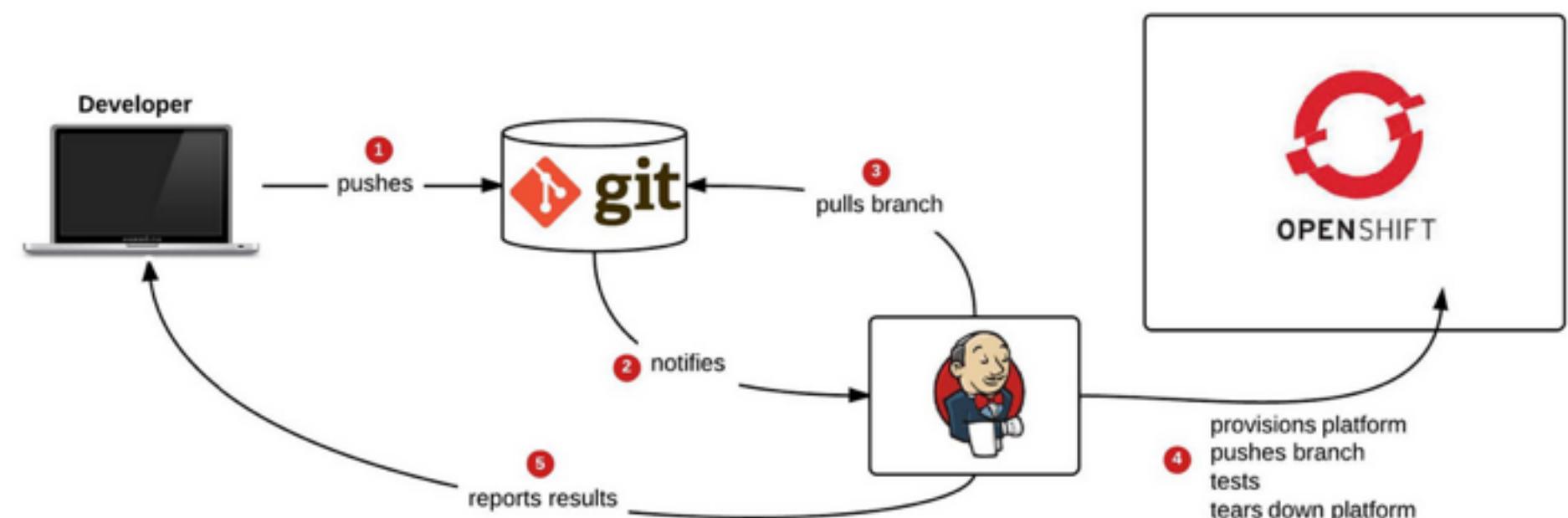
Asignatura *Servidores Web y PaaS*

- Profesor: José Luis Zamora
- Calendario: del 8 al 16 de enero



Servidores de aplicaciones, desde el desarrollo hasta el despliegue en cloud

- Arquitecturas Java EE clásicas y actuales
 - Microservicios, Docker
- Aspectos básicos de administración de Red Hat WildFly
 - Conceptos básicos, herramientas, despliegue de recursos (Datasources, EJB's)
- Seguridad declarativa (JAAS)
 - Security domains, login modules y Realms.
 - Seguridad basada en ficheros, BD y LDAP.
- Certificados y encriptación SSL
- Introducción a Cloud Computing
 - SaaS, PaaS, IaaS
- Despliegue de aplicaciones en OpenShift
 - Creación y modificación de aplicaciones Java EE.
 - Variables de entorno, markers, action hook scripts, SSH
 - Copias de seguridad
 - Monitorización



- Escalabilidad
- DevOps
- CI/CD Integración continua y despliegue continuo

Asignatura *Servidores Web y PaaS*

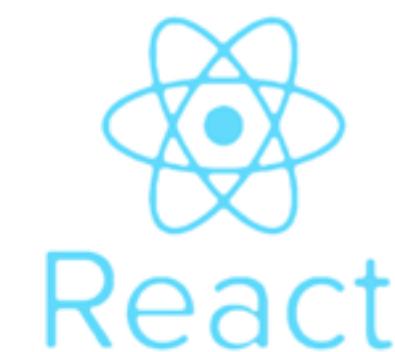
- 8 sesiones (20 horas, 2 créditos ECTS)
 1. Introducción a los servidores de aplicaciones y a WildFly
 2. Administración de recursos
 3. Seguridad declarativa
 4. Certificados digitales y SSL
 5. Introducción a Cloud Computing
 6. Plataforma como servicio
 7. Características IaaS+ de OpenShift
 8. Escalabilidad e Integración Continua



Tecnologías JavaScript



JavaScript



Asignatura *Lenguaje JavaScript*

- Profesor: Aitor Medrano
- Calendario: del 26 de Febrero al 9 de Marzo



¿Por qué JavaScript?

- Lenguaje de scripting disponible en todos los navegadores.
 - Muy buen “amigo” de HTML y CSS.
 - Sencillo pero potente, con soporte para orientación a objetos y *closures*.
 - Las funciones son clases, con propiedades y métodos.
 - Herencia mediante uso de prototipos.
- Pese a partir como lenguaje de cliente, se está extendiendo a otros ámbitos (servidor con Node.js, desarrollo de juegos o aplicaciones de escritorio).
- *Testable* mediante herramientas de terceros (*QUnit*, *Jasmine*).
- Existen multitud de librerías que extienden su funcionalidad.
- jQuery es la librería web estándar *de facto* (implementada en JavaScript)
 - Uso de selectores y *method chaining* para simplificar el manejo del DOM
- *jQuery UI*: widgets para enriquecer la web
- *jQuery Deferreds*: promesas



JavaScript



Asignatura *Lenguaje JavaScript*

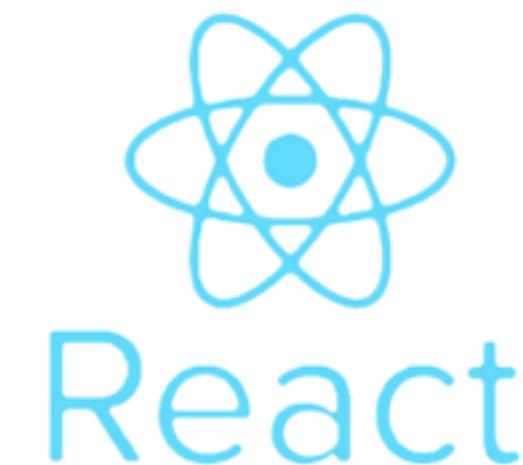
- 8 sesiones (20 horas, 2 créditos ECTS):
 1. JavaScript. El lenguaje.
 2. JavaScript Orientado a Objetos
 3. JavaScript con DOM
 4. JavaScript Avanzado
 5. JavaScript y el Navegador
 6. jQuery
 7. Promesas y jQuery Deferreds
 8. jQuery Avanzado

Asignatura Frameworks Javascript I

- Profesor: Otto Colomina Pardo
- Calendario: del 15 al 23 de abril

Frameworks Javascript

- Las **librerías JS** al estilo *jQuery* nos hacen más productivos y nos ayudan a tratar con las incompatibilidades entre navegadores
- Los **frameworks** además nos ayudan a **estructurar el código** de nuestra aplicación
- La **arquitectura** predominante en la actualidad es **MVC (Modelo/Vista/Controlador)**, aunque hay otras alternativas
- Veremos varios frameworks
 - **BackboneJS** (MVC) : framework sencillo y fácil de aprender, como contrapartida no implementa demasiadas funcionalidades
 - **Marionette** (La Vista del MVC). Se combina con backbone para permitirnos implementar vistas más complejas
 - **ReactJS**: para diseñar la interfaz. No es MVC, usa una arquitectura propia denominada *Flux*



Asignatura Frameworks Javascript I

- 8 sesiones (20 horas, 2 créditos ECTS):
 1. Introducción a MVC en Javascript con Backbone.js
 2. Backbone básico: Modelos, colecciones, vistas y routers
 3. Persistencia con APIs REST
 4. Vistas y subvistas con el framework Marionette
 5. Routers y Testing
 6. Miniproyecto de Backbone+Marionette
 7. React (I): componentes de interfaz
 8. React (II): la arquitectura Flux

Asignatura *AngularJS - Frameworks JavaScript (II)*

- Profesor: Alejandro Such
- Calendario: del 6 al 14 de mayo

AngularJS

- Mejora HTML para la creación de aplicaciones web SPA
- Probablemente, el framework más popular hoy en día
- Cliente rico = menos carga en el servidor
 - Nada de montar JSPs
 - Nada de mantener el estado
 - Mejor sensación (nada de “cargando”)
- Fácilmente integrable con jQuery (si queremos)
- Orientado a testing
- MVW (Model-View-Whatever)
- Two-way data binding
- Nos permite extender HTML, creando nuestros propios tags de una manera relativamente sencilla.
- Templating en HTML, con lo que puede existir una separación real entre el desarrollador y el maquetador/diseñador.



Superheroic MVW Framework

Asignatura *AngularJS - Frameworks JavaScript (II)*

- 8 sesiones (20 horas, 2 créditos ECTS) donde se abordarán:
 1. MVVM
 2. Scopes
 3. Módulos y servicios
 4. Filtros
 5. Routing con ngRoute y ui-router
 6. Formularios y validación
 7. Directivas para extender HTML y validar datos
 8. Promesas de resultados
 9. Comunicación con el servidor
 10. Automatización
 11. Testing



Tecnologías alternativas





Asignatura *Grails - Frameworks de Aplicaciones Web*

- Profesor: Fran García
- Calendario: del 5 al 13 de febrero

¿Qué es Groovy&Grails?

- Groovy es lenguaje de la máquina virtual de Java con características similares a otros lenguajes como Ruby y Perl.
- Grails es un framework completo para el desarrollo rápido de aplicaciones web.
- Utiliza varias soluciones open-source ampliamente aceptadas como Spring, Hibernate o Tomcat, entre otras.
- Basado en el patrón Modelo Vista Controlador.
- El scaffolding nos permitirá prototipar rápidamente la aplicación.
- Spock se ha impuesto como framework de tests en Grails.



Asignatura *Grails - Frameworks de Aplicaciones Web*

- 8 sesiones (20 horas, 2 créditos ECTS):
 1. Introducción a Groovy
 2. Aspectos avanzados del Lenguaje Groovy. Metaprogramación.
 3. Introducción a Grails. Scaffolding.
 4. Patrón MVC: Vistas y controladores.
 5. Patrón MVC: Dominios y servicios.
 6. Framework de test Spock. TDD.
 7. Seguridad con Spring Security Plugin
 8. Configuración de aplicaciones. Plugins interesantes.

Asignatura *Bases de Datos NoSQL*

- Profesor: Aitor Medrano
- Calendario: 11 y 12 de marzo

¿Por qué NoSQL?

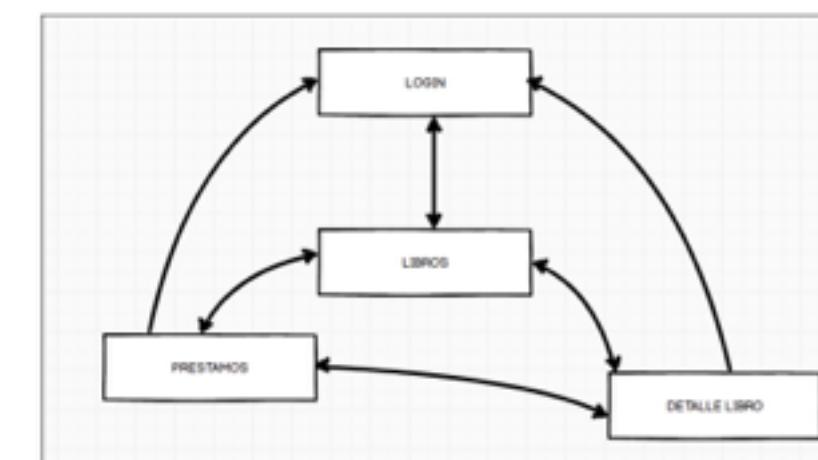
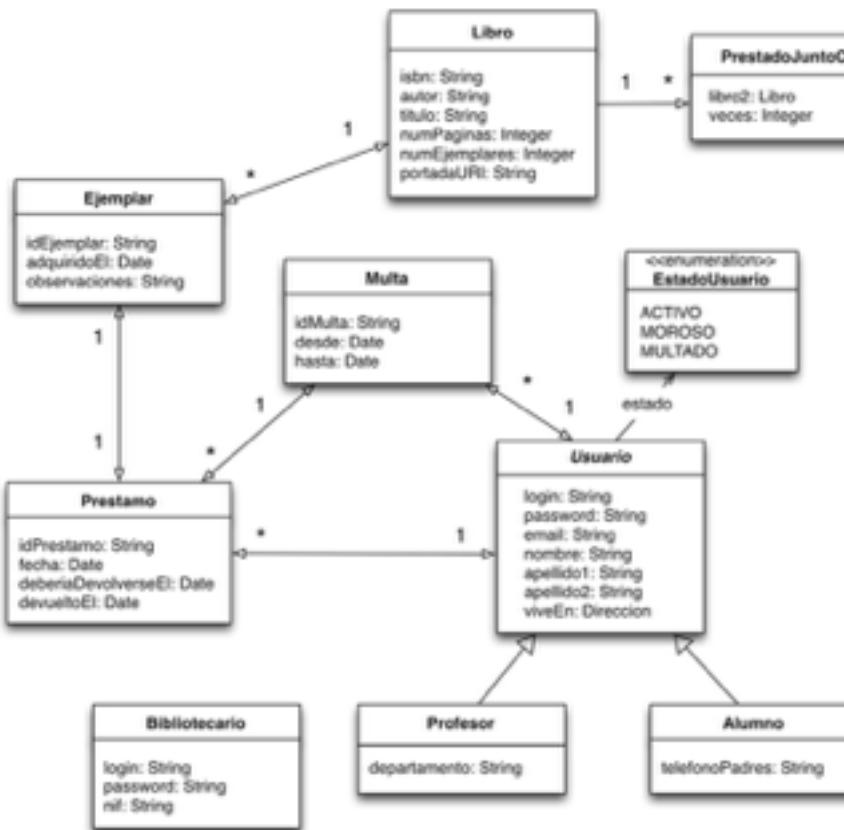
- Tecnologías alternativas a SQL que se centran en el rendimiento
- Permiten esquemas dinámicos
- Pensados para la escalabilidad horizontal (*sharding*)
- Soporta diferentes tipos de BBDD: documentales, grafos, clave-valor, basada en columnas
- MongoDB como estándar de facto como SGBD documental
 - Uso de JSON
 - Soporte REST
 - Soporte de *sharding* y grids (escalado y replicación)
 - Soporte de consultas agrupadas con alto rendimiento



Asignatura *Bases de Datos NoSQL*

- 4 sesiones (10 horas, 1 crédito ECTS):
 1. No Sólo SQL
 2. MongoDB
 3. Java y MongoDB. Rendimiento
 4. Agregaciones y Escalabilidad

Proyecto de aplicación web



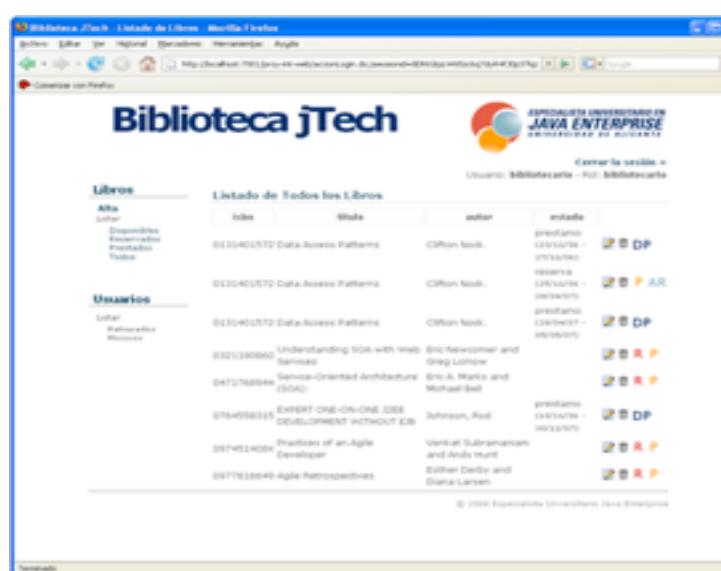
Listado de libros

Biblioteca jTech	Mastering Java MartiÁo Dickinson Ver detalle	Thinking in Java Winchester McFly Ver detalle	Wildfly in action Arun Gupta Ver detalle

Head First Design Patterns
Freeman & Freeman

Biblioteca jTech	1234567890 Páginas: 180 Ejemplares: 10

Biblioteca jTech	Reservar



Asignatura *Proyecto Java EE*

- Profesores: Domingo Gallardo, Isabel Alfonso, Jose Luis Zamora, Alejandro Such
- Calendario: a lo largo de todo el curso, al terminar el módulo correspondiente
- 6 sesiones (30 horas, 3 créditos ECTS):
 1. Caso de estudio, introducción a Maven, Git y Bitbucket
 2. Capa de persistencia y lógica de negocio (JPA y EJB)
 3. API REST
 4. Despliegue en PaaS
 5. Cliente JavaScript con AngularJS (I)
 6. Cliente JavaScript con AngularJS (II)

Ahora os toca a vosotros

- ¿Qué has estudiado? ¿Experiencia con Java?
- Ocupación en la actualidad
- ¿Qué esperas del curso?



Concluyendo

- Esperamos que el experto sea de vuestro agrado y que salgáis de él con una idea clara sobre:
 - Qué tecnologías existen en Java EE y cómo puedo usarlas
 - Cómo afrontar un proyecto Java EE + JavaScript
 - Qué tecnologías escoger para un proyecto concreto
 - Cuál es el camino a seguir (roadmap) en cada tecnología



¿Preguntas?