

# Sistemas y Tecnologías Web (2014-2015)

Casiano Rodriguez Leon

15 de diciembre de 2014

## 1. Horarios

### Tutorías:

- Localización: Despacho #96, 4ta Planta Edificio Física/Matemáticas.
- Reserva: Tfno. 922 31 81 87 Fax. 922 31 81 70
- e-mail: `casiano@ull.es`

### Horario:

1. Teoría y Problemas los tenemos en el Aula 1.2 los Jueves de 10:30 a 12:20.
2. Grupo L2 los Viernes de 8:30 a 10:10 en la sala 2.3.
3. Grupo L1 los Viernes de 10:30 a 12:10 en la sala 2.3.
4. Tutorías de Lunes a Jueves de 09:00 a 10:30 en el despacho 96 de la 4ª planta del edificio de Física/Matemáticas (crguezl@ull.edu.es, 922 31 81 87)

Para la tutoría virtual contacta al profesor en google plus usando una "hangout". Mi identificador g+ de ull.edu.es es 101973567008975662199.

También puede ir a la página <http://gplus.to/crguezl>.

Véase también el calendario oficial de la ULL en

[http://www.ull.es/Private/folder/institucional/ull/wull/estudios\\_docencia/calendario/calendario](http://www.ull.es/Private/folder/institucional/ull/wull/estudios_docencia/calendario/calendario)  
”

## 2. Programa

1. Diseño, desarrollo e implementación Web
2. Medios y librerías digitales

3. Arquitecturas orientadas a servicios
4. Fundamentos, sistemas, servicios y aplicaciones basados en contenidos multimedia
5. Software social y colaborativo
6. Comercio electrónico

Los temas anteriores se desarrollan en este conjunto de tópicos:

1. Software colaborativo: GitHub, Cloud9 (c9.io), G+, etc.
2. El protocolo HTTP
3. Servidores HTTP y Servidores de Aplicación
4. Sesiones y Cookies
5. Lenguajes de hipertexto: HTML5
6. Lenguajes y Librerías de estilo: CSS3, SASS, CSS3, BootStrap, etc.
7. Framework webs en Ruby: Rack, Sinatra, Padrino, Rails
8. Despliegue de aplicaciones Web: Heroku, Amazon AWS, Capistrano
9. Pruebas Unitarias, TDD y BDD en la web, Integración Continua, Despliegue Continuo: Cucumber, Capybara, Rack-Test, Selenium, Poltergeist (PhantomJS), Travis, Coveralls, etc.
- 10.
11. Autorización y Autenticación: OAuth, Omniauth, OmniContacts, OpenId, ...
12. Templates: Haml, ERB, Slim, etc.
13. Asignadores de Objetos a Bases de Datos Relacionales (ORM): DataMapper, Sequel, Active Record, etc.
14. Programación en el Cliente: JavaScript, CoffeeScript
15. Programación Orientada a Eventos. El modelo de Eventos de JS
16. El Modelo del Objeto Documento: DOM
17. Ajax
18. Librerías en JS: jQuery, Underscore
19. AngularJS

20. Streaming: COMET, Server Sent Events, WebSockets
21. Almacenamiento en el cliente: localStorage, sessionStorage, Application Storage, Bases de Datos en el lado del Cliente, etc.
22. Gráficos: SVG, Rafäel, Canvas, WebGL
23. Juegos en JS
24. JavaScript en el lado del Servidor: Node.JS
25. Framework Webs para Node.JS: Express
26. Geolocalización
27. Arquitecturas Orientadas a Servicios:
  - La Arquitectura de Transferencia de Estado Representacional (Representational State Transfer) REST
28. Multimedia: Programando Audio y Vídeo
- 29.
30. E-commerce: Spree

### **3. Micro-Exámenes**

Aproximadamente cada cuatro semanas se realiza una prueba objetiva para la valoración de las actividades prácticas, trabajos y proyectos que se califican de 0 a 10. Los alumnos deberán superar los micro-exámenes con una nota mínima de 5.

Las fechas previstas son:

1. Jueves 9 de Octubre
2. Jueves 6 de Noviembre
3. Jueves 18 de Diciembre

### **4. Presentaciones**

Todos los alumnos hacen al menos una presentación de un tema durante el curso. Los alumnos deberán superar esta parte con una nota mínima de 5.

El alumno realiza no mas de tres de estas exposiciones por curso. En la mayoría de los casos sobre el mismo tema en diferentes niveles de profundidad: básico, intermedio y avanzado. No obstante, es posible cambiar de tema.

Sigue una lista de temas. Pueden proponer cualquier tema que encaje en el perfil de la asignatura. He evitado poner software propietario pero si alguien quiere proponerlo porque tiene acceso a el mismo, es bienvenido.

## 1. JavaScript, HTML y CSS

- a)* Objetos,
- b)* Funciones
- c)* Herencia
- d)* Arrays
- e)* Expresiones Regulares
- f)* Métodos
- g)* El DOM (Document Object Model)
- h)* Eventos (Propagación, manejadores, contexto, etc.)
- i)* Canvas
- j)* SVG: Scalable Vector Graphics
- k)* JavaScript en el lado del servidor: Node.JS
- l)* Audio y Vídeo
- m)* Ajax
- n)* JSON
- ñ)* JQuery
- o)* APIs HTML5 en JavaScript (p.ej. Geolocation, gestión de la historia, Web Sockets, etc.)
- p)* BootStrap
- q)* Testing/Pruebas en JS
- r)* CSS
- s)* Responsive Web Design
- t)* SASS
- u)* Web Design
- v)* API de JavaScript de Google
- w)* API de JavaScript de Google Maps
- x)* AngularJS
- y)* Raphael (JavaScript library to work with vector graphics on the web)
- z)* Gráficos 3D

## 5. Proyectos

Plantearémos proyectos en los que trabajarán en grupos de no mas de 3 personas.

El proyecto usará un sistema de control de versiones (git, hg, svn) y residirá en un repositorio público como GitHub, Google-Code o Bitbucket.

Cada semana habrá una exposición por el grupo de los avances logrados en el proyecto.

Para la primer semana tendremos que hacer una propuesta inicial de grupos y proyectos y ver que tecnologías necesitamos para poder instalarlas, si es posible en el Centro de Cálculo. En caso contrario podremos trabajar con nuestros portátiles.

## 6. Charlas

Tendremos charlas de ex-alumnos y expertos en sistemas y Tecnologías Web a lo largo del curso que serán convenientemente anunciadas. Si conoces de un ponente interesante no dudes en proponerlo.

También usaremos cursos y charlas en la web (Coursera, Udacity, etc.). En particular:

1. Coursera. Software Engineering for SaaS por Armando Fox y David Patterson
2. Udacity. Web Development (CS253). How to Build a Blog por Steve Huffman

## 7. Curso de Sinatra

A lo largo del curso iremos desarrollando un curso de uso de Rack y Sinatra.

## 8. Programa de Prácticas

Se llevarán a cabo micro-proyectos/prácticas individuales y en grupo cuya evaluación se hace mediante un taller/workshop. Los alumnos deberán superar los micro-proyectos con una nota mínima de 5.

## 9. Convocatorias de Exámenes

Véase Convocatorias de Exámenes del Grado para el curso 2014/2015 en <http://www.ull.es/view/centros/etsii/>

**Enero**

DÍA: 12 de enero  
HORA: 9:00  
AULA: 2.1

DÍA: 20 de enero  
HORA: 9:00  
AULA: 2.1

### **Junio**

DÍA: 27 de mayo  
HORA: 16:00  
AULA: 2.4

### **Julio**

DÍA: 10 de julio  
HORA: 9:30  
AULA: 2.6

## **Referencias**

- [1] Flanagan, David. *JavaScript: The Definitive Guide*. O'Reilly. 2011.
- [2] Douglas Crockford. *JavaScript: the Good Parts*. O'Reilly. 2008.
- [3] Ara Pehlivanian and Don Nguyen. *Jump Start JavaScript*. SitePoint. 2013.
- [4] Louis Lazaris. *Jump Start CSS*. SitePoint. 2013.
- [5] Don Nguyen. *Jump Start Node.js*. SitePoint. 2012.
- [6] Fox, Armando; Patterson, David. *“Engineering Long-Lasting Software. An Agile Approach Using SaaS & Cloud Computing”*. Strawberry Canyon LLC. 2012.
- [7] Alan Harris; Konstantin Haase. *Sinatra: Up and Running*. O'Reilly Media, Inc. 2011
- [8] Darren Jones. *Jump Start Sinatra*. SitePoint. 2013.
- [9] Fowler, Chad. *“Rails Recipes”*. The pragmatic Bookshelf. 2011.
- [10] Marshall, Kevin; Pytel, Chad; Yurek, Jon. *“Pro Active Record: Databases with Ruby and Rails.”*. Apress. 2007
- [11] Griffiths, David, *“Head First Rails: A Learner’s Companion To Ruby On Rails”*. O'Reilly. 2009.

- [12] Al Barazi, Rida; Carneiro Jr., Cloves; Carneiro, Cloves. *“Beginning Rails 3”*. Apress. 2010.
- [13] Hartl, Michael; Prochazka, Aurelius. *“RailsSpace: Building a Social Networking Website with Ruby on Rails”*. Addison-Wesley. 2007.
- [14] Bell, Gavin; *“Building Social Web Applications”*. O’Reilly. 2009.
- [15] Follansbee, Joe. *“Hands-On Guide to Streaming Media”*. Elsevier. 2006.
- [16] Christian; Laine, Jarkko, *“Beginning Ruby on Rails E-Commerce: From Novice to Professional”*. Apress. 2006.
- [17] Dix, Paul, *Service-Oriented Design With Ruby And Rails*. Addison-Wesley. 2010.
- [18] Richardson, Leonard; Ruby, Sam, *“RESTful Web Services”*. O’Reilly.
- [19] Rodríguez C. *Apuntes de la Asignatura Lenguajes y Paradigmas de Programación*  
HTML en <http://nereida.deioc.ull.es/~lpp/perlexamples/perlexamples.html>. 2003.
- [20] Vogel, Lars. *Git Tutorial* <http://www.vogella.de/articles/Git/article.html> 2012.
- [21] The Git Community. *Git Community Book* <http://book.git-scm.com/> 2012.
- [22] Scott Chacon. *Pro Git* <http://progit.org/book/> 2010.