

# Frikiminutos 2016 (enero–abril), serie B

## ETSIT – URJC

Jesús M. González Barahona, Gregorio Robles Martínez

<http://gsyc.es/~jgb> <http://gsyc.es/~grex/>  
GSyC, Universidad Rey Juan Carlos

15 de febrero de 2016





©2015-2016 Gregorio Robles, Jesús M. González Barahona.  
Algunos derechos reservados. Este artículo se distribuye bajo la licencia  
"Reconocimiento-CompartirIgual 3.0 España" de Creative Commons, disponible en  
<http://creativecommons.org/licenses/by-sa/3.0/es/deed.es>  
Este documento (o uno muy similar) está disponible en  
<http://cursosweb.github.io>

- 1 Trabajo top
- 2 Viéndose con gente...
- 3 Servidor web en Producción
- 4 Contenedores por todas partes
- 5 Atom
- 6 Coffeescript

# Trabajo top

# ¿Qué es un trabajo *bueno*?

- Un trabajo que te permita ser creativo
- Un trabajo donde trabajes con últimas tecnologías
- Un trabajo donde puedas ascender sin dejar de ser ingeniero
- Un trabajo donde te paguen bien (y otros beneficios)

Hay muchas empresas donde buscan este tipo de perfil: Google, Apple, Facebook, Microsoft, Yahoo!, Amazon...

# Salarios en las compañías top

Rank	Company Name	Sector	Average Salary
1	Twitter, Inc.	Social Networking	\$120,111.11
2	Apple, Inc.	Computers, Peripherals	\$113,319.21
3	LinkedIn, Corp	Social Networking	\$111,720.00
4	Cisco Systems	Computer Networking	\$107,998.32
5	Qualcomm	Wireless	\$107,632.31
6	IBM	Information Technology	\$106,508.00
7	Facebook, Inc.	Social Networking	\$105,167.62
8	Google, Inc.	Internet, Computers	\$104,594.27
9	Hewlett Packard	Computers	\$104,379.00
10	Oracle	Computers	\$104,058.03

<http://img59.imageshack.us/img59/802/toppaytech.png>

# ¿Qué te piden en estos trabajos?

- Estructuras de datos
- Algoritmia
- Experiencia en programación
- Redes de ordenadores
- Sistemas operativos

# Más lecturas

- Hay varios libros sobre este tema, algunos en la biblioteca:
  - Cracking the coding interview: 150 programming interview questions and solutions
  - The Google Interview
  - Elements of Programming Interviews: The Insiders' Guide
  - Top 10 coding interview problems asked in Google with solutions: Algorithmic Approach
  - Are You Smart Enough to Work at Google?: Fiendish Puzzles And Impossible Interview Questions From The World's Top Companies
  - Get a Job WITHOUT an Interview - Google & Beyond!: "We don't mind to lose a good applicant, but definitely not hire a bad applicant."
  - The Google Resume: How to Prepare for a Career and Land a Job at Apple, Microsoft, Google, or any Top Tech Company

# Viéndose con gente...

# Meetup

WEDNESDAY, FEBRUARY 3

6:30 PM

Madrid BIM Group

## Dynamo y sus aplicaciones en BIM

13 Members going

7:00 PM

PHPMad

## iTutorial de PHP, acabado! ¿Y ahora que?

94 Members going **6 spots left!**

7:00 PM

Madrid Drones Meetup

## Reunión del grupo + taller de cambio de firmware en los microdrones

4 Drone Lovers going

7:00 PM

NSCoder Night Madrid

## NSPresenters

40 NSCoders going

<http://meetup.com>

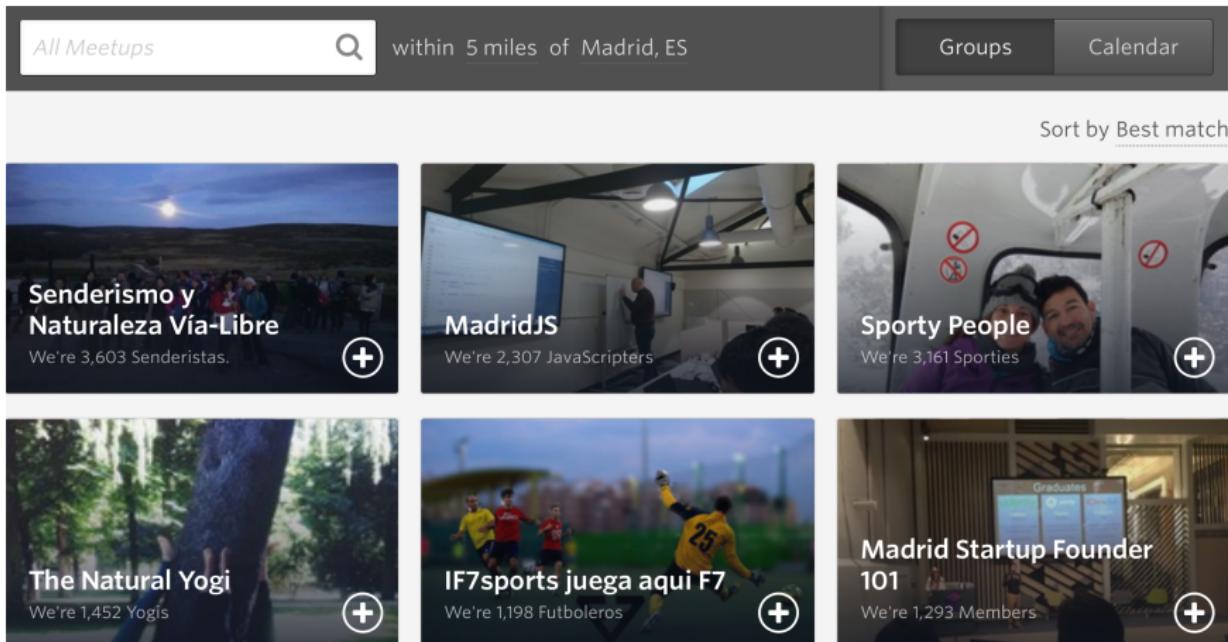


- Información sobre reuniones cercanas
- Mucho contenido técnico
- Y mucho que no

# Grupos

All Meetups  within 5 miles of Madrid, ES Groups Calendar

Sort by Best match



The screenshot shows a grid of six group cards. Each card includes a thumbnail image, the group's name, its member count, and a 'Join' button.

- Senderismo y Naturaleza Vía-Libre**  
We're 3,603 Senderistas. 
- MadridJS**  
We're 2,307 JavaScripters 
- Sporty People**  
We're 3,161 Sporties 
- The Natural Yogi**  
We're 1,452 Yogis 
- IF7sports juega aquí F7**  
We're 1,198 Futboleros 
- Madrid Startup Founder 101**  
We're 1,293 Members 

# Reuniones

**MADRID • JS**

**Madrid, Spain**  
Founded Aug 6, 2011

[About us...](#)  
[+ Invite friends](#)

JavaScripers 2,308  
Group reviews 61  
Upcoming Meetups 2  
Past Meetups 60  
Our calendar

# Keep it simple, use React

Export Tell a friend Share

Thursday, February 11, 2016  
7:00 PM

Impact Hub Next  
C / Alameda, 22, Madrid ([map](#))

Este febrero tendremos **no una charla especial, sino dos**. En este caso vamos a aprovechar la visita de Javi Jiménez desde Tailandia: nos va a contar su experiencia haciendo proyectos grandes con React.

**Keep it simple, use React**

Actualmente React está siendo utilizado por organizaciones como Khan Academy, Netflix, Yahoo, Airbnb, Doist, Facebook, Instagram, Sony, Atlassian y muchas otras. Esto demuestra el

Are you going?

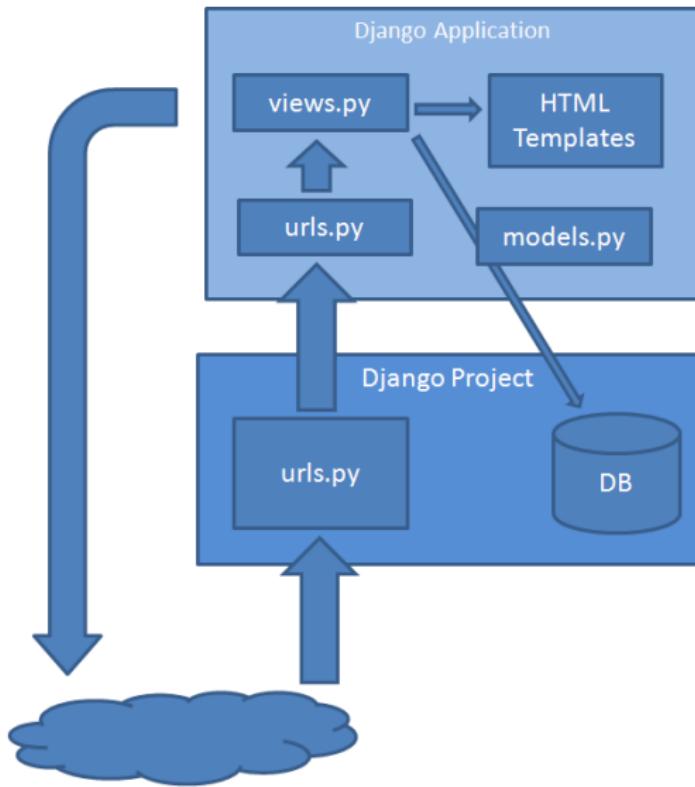
277 going

**soyjavi**  
EVENT HOST  
A self-taught human, in love with creating new stuff.

**Manuel Juan Fosela**  
CO-ORGANIZER  
EVENT HOST  
Programador web o eufimísticamente Ingeniero de Software de Internet XD

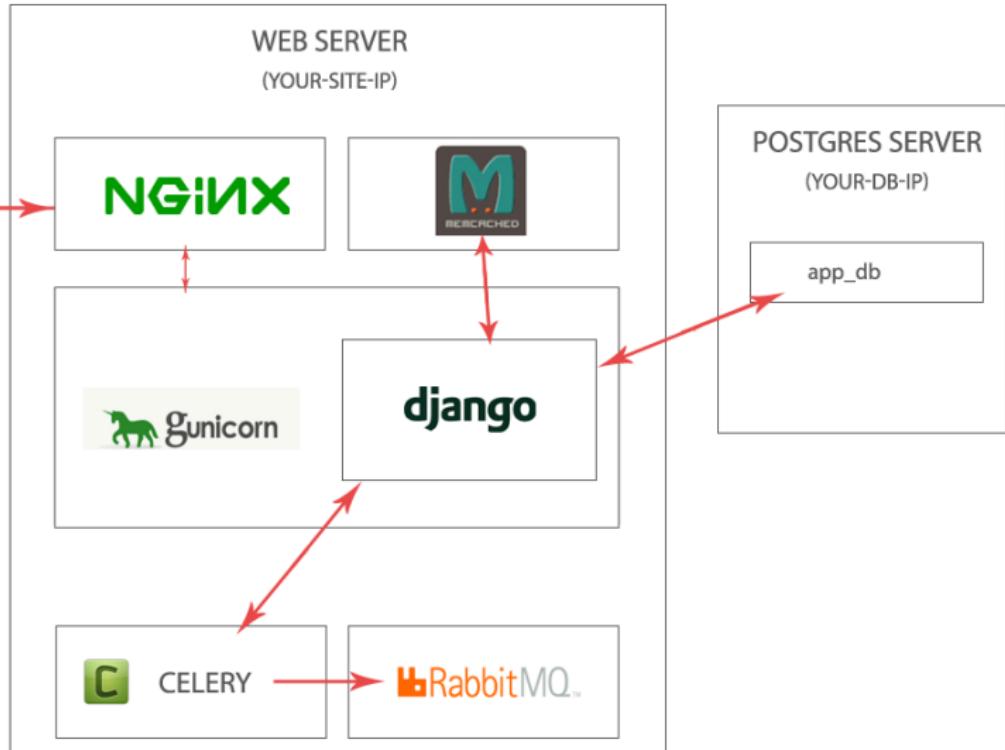
# Servidor web en Producción

# Lo que enseñamos en clase



- Mono-hilo
- Mono-tarea
- Caché básico
- Base de datos limitada (sqlite)
- Pensado para páginas dinámicas
- Sin planificación
- No tiempo real

# Un servidor web en producción



# Tecnologías

- Django: Framework web
- Nginx: Servidor web con balanceo de carga (<http://nginx.org/>)
- Memcached: Caché (<http://memcached.org/>)
- gunicorn: Servidor HTTP (<http://gunicorn.org/>)
- Celery: Tiempo real y planificación de tareas  
(<http://www.celeryproject.org/>)
- RabbitMQ: Mensajería (<https://www.rabbitmq.com/>)
- PostgreSQL: Base de datos (<http://www.postgresql.org/>)

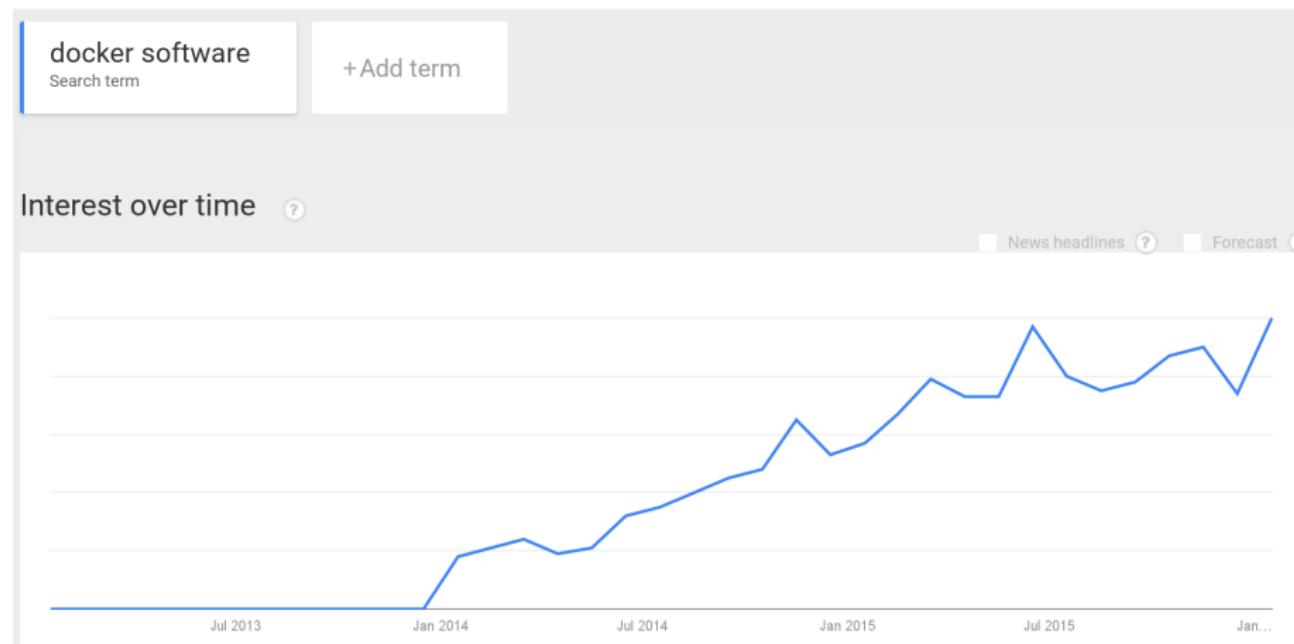
# Contenedores por todas partes



# Contenedores software

- Virtualización sobre sistema operativo
- Evolución de la idea de chroot
- Aislamiento (disco, memoria)
- Gestión de recursos
- Más ligero que máquinas virtuales completas
- Mismo kernel que host
- Docker, LXC, LXD, FreeBSD Jail...

# Docker (search trend)



# Docker



<http://docker.com>

<http://hub.docker.com/>

- Automatización del despliegue de aplicaciones en contenedores software
- Montado sobre cgroups (gestión de recursos), namespaces (separación de recursos), sistema de ficheros con unión

## Referencias y enlaces

Luke Price, "Containers, Port of Rotterdam", CC-by 2.0

<https://www.flickr.com/photos/lukeprice88/9703431992>

# Atom

# Introduciendo Atom

The screenshot shows the Atom code editor interface. On the left, there's a sidebar with a tree view of project files: build, docs, dot-atom, exports, keymaps, menus, node\_modules, resources, script, spec, src (which is selected), static, vendor, .coffeelintignore, .gitattributes, and .gitignore. The main editor area displays the file 'atom.coffee' with the following content:

```
18
19 # Essential: Atom global for dealing with packages, themes, menus, and the window system.
20 #
21 # An instance of this class is always available as the `atom` global.
22 module.exports =
23   class Atom extends Model
24     @version: 1 # Increment this when the serialization format changes
25
26     # Load or create the Atom environment in the given mode.
27     #
28     # Returns an Atom instance, fully initialized.
29     @loadOrCreate: (mode) ->
30       startTime = Date.now()
31       atom = @deserialize(@loadState(mode)) ? new this({mode, @version})
32       atom.deserializeTimings.atom = Date.now() - startTime
33
```

At the bottom of the editor, it says 'src/atom.coffee\*' followed by line numbers 31, 17. To the right, there are status indicators for 'UTF-8', 'CoffeeScript', and 'master'. Above the editor, there are tabs for 'atom.coffee' and 'Settings'.

<http://atom.io> (Hay paquete Debian)

# Características

- Basado HTML, JavaScript, CSS, and Node.js
- Programado en CoffeeScript (compilado a JavaScript)
- Autocompletado
- Paquetes adicionales
- Muchos temas
- Configurable
- Multiplataforma
- ... y es software libre

# Coffeescript

# The basics



[coffeescript.org](http://coffeescript.org)

- Sintaxis más sencilla
- Orientado a ser legible
- Breve
- Indentación
- No hay paréntesis
- En 2012 fue el 11º lenguaje más popular en GitHub
- Se puede probar en línea

# ¿Cómo funciona?

1

**Write code in .coffee file**

2

**Compile into .js file**

3

**Include .js file in web pages**  
`<script src="my.js"></script>`

# Ejemplos

```
# Assignment:
number: 42
opposite_day: true

# Conditions:
number: -42 if opposite_day

# Functions:
square: x => x * x.

# Arrays:
list: [1, 2, 3, 4, 5]

# Objects:
math: {
  root: Math.sqrt
  square: square
  cube: x => x * square(x).
}

# Array comprehensions:
cubed_list: math.cube(num) for num in list.
```



```
var __a, __b, __c, __d, cubed_list, list, math, num,
number, opposite_day, square;
// Assignment:
number = 42;
opposite_day = true;
// Conditions:
if (opposite_day) {
  number = -42;
}
// Functions:
square = function(x) {
  return x * x;
};
// Arrays:
list = [1, 2, 3, 4, 5];
// Objects:
math = {
  root: Math.sqrt,
  square: square,
  cube: function(x) {
    return x * square(x);
}
};
// Array comprehensions:
__a = list;
__d = [];
for (__b=0, __c=__a.length; __b<__c; __b++) {
  num = __a[__b];
  __d[__b] = math.cube(num);
```