title: Clase 11 Proyecto 2014
Author: Einar Lanfranco, Claudia Banchoff
description: Clase de repaso general 2014
keywords: Repaso general
css: proyecto.css

Proyecto de Software

Cursada 2014

¿Qué vimos?

- URL
- HTTP
- HTML
- CSS

URL/URI - RFC 2396/3986

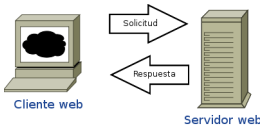
- Una **URI -"Uniform Resource Identifier"-** es un mecanismo por el cual se identifica todo recurso accesible en la web.
- Una **URL -"Uniform Resource Locator"-** permite ubicar un recurso a través de su ubicación.
- Ejemplos

```
http://www.servidor.com.ar/especificacion#parte3
http://www.taller.com.ar/info.php?id=12&qq=11
../cursada2009/mejores/junio.htm
mailto:proyecto@info.unlp.edu.ar
```

---

class: destacado

Protocolo HTTP



- Una transacción HTTP consta de 4 pasos:

inicio conexión - solicitud - respuesta - cierre conexión

- **Protocolo sin estado**
- Clientes web: Firefox, IE, Chrome, Opera, ...
- Servidores web: Apache, IIS, Nginx, etc, ...

Lenguaje HTML

- HTML - **“HyperText Markup Language”**- especifica el formato de las páginas web, separando el contenido de las páginas de su formato de presentación.
- Fue creado en los laboratorios CERN por **Tim Berners-Lee**.
- Define un conjunto de símbolos (etiquetas o tags) que **especifican la estructura** lógica de un documento y de todos sus componentes.
- Es independiente de la plataforma.
- **Su código es interpretado por los clientes web**.

Hojas de Estilo

- **Describen el formato de un documento HTML. Cómo se visualizarán en los distintos medios, por ejemplo en la pantalla o en la impresora.**
    - Permiten separar el contenido de un documento HTML de la forma en que se lo visualizará
- El estándar usado: CSS (Cascading Style Sheets)■ (CSS 2.1 estándar css3 aún no)
- Es posible incluir las definiciones dentro de la página o en un archivo separado (a partir del HTML 4.0).

¿Qué vimos?

- RFC
- MetaDatos
- Web vs Web 2 vs Web Semántica

Internet y la web

RFCs – Request for Comments

- http://www.faqs.org

Campos Meta

- Se usan para identificar meta-información sobre el documento.
- Son usados por buscadores para mejorar la calidad de los resultados en las búsquedas.

Ejemplo:

```
<meta name="description" content="Proyecto de software" />
<meta name="author" content="Claudia Banchoff-Einar Lanfranco" />
<meta name="keywords" content="meteorología, clima">
```

La web 2

- En 2004, por primera vez mencionado por Tim O'Reilly
- Los **usuarios como productores de contenidos**
- Herramientas típicas: blogs, wikis, redes sociales...

La web semántica

- Incorporar metadatos para agregar **significado** a la información del documento HTML.
- Se debe seguir un **formalismo adecuado** para que se lo pueda procesar en forma adecuada.
- En la materia, sólo veremos algunos aspectos sobre HTML semántico...

¿Qué vimos?

- W3c
- Validadores
- Web Responsive
- WAI
- WCAG
- Ley Nacional 26.653

---

class: destacado

Repaso - W3C – El consorcio de la web

- **http://www.w3c.org**
- Desarrollo de estándares y guías.

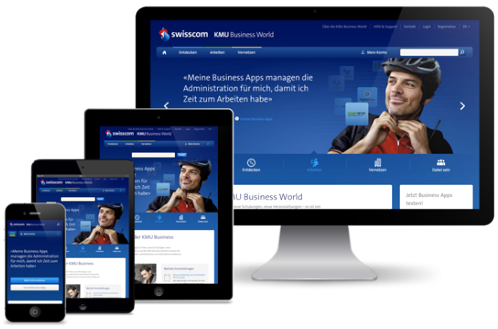
La misión del W3C es:

Guiar la Web hacia su máximo potencial a través del desarrollo de protocolos y pautas que aseguren el crecimiento futuro de la Web.

Validadores

- Permiten verificar el cumplimientos de los estándares.
- La W3C provee algunos:
- Validador HTML: http://validator.w3.org/
- Validador de Hojas de Estilos: http://jigsaw.w3.org/css-validator/
- Unicorn http://code.w3.org/unicorn

Web responsive - Gráficamente



---

class: destacado

WAI - Web Accessibility Initiative

Objetivos:

Desarrollar estrategias, pautas, recursos para hacer la Web accesible a personas con discapacidad. Pero también será accesible en otros entornos y aplicaciones, como navegador de voz, teléfono móvil, PC de automóvil. Y ante limitaciones bajo las que opere, como entornos ruidosos, habitaciones infra o supra iluminadas, entorno de manos libres.

Pautas WCAG

- Definen **principios de diseño** Web.
- Cada principio tiene **pautas**.
- Cada pauta tiene **criterios testeables**.

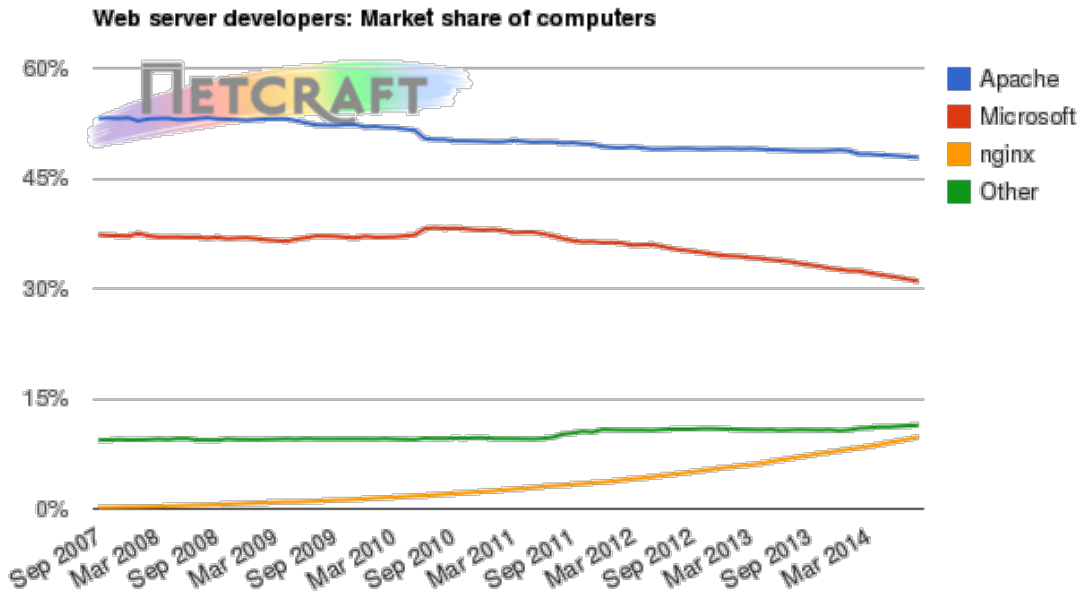
Ley Nacional 26.653

- **Ley Nacional 26.653: Acceso a la Información pública**
- Sancionada: Noviembre de 2010
- Reglamentada: Abril de 2013
- la ley
- **Algunas referencias:**
    - Sobre la reglamentación
    - El INADI

¿Qué vimos?

- Apache
- Nginx
- IIs
- LAMP/WAMP

Apache, Nginx,Iis



Developer	July 2014	Percent	August 2014	Percent	Change
Apache	2,321,141	47.92%	2,338,927	47.83%	-0.09
Microsoft	1,512,933	31.24%	1,515,674	31.00%	-0.24
nginx	460,795	9.51%	478,793	9.79%	0.28

http://news.netcraft.com/archives/category/web-server-survey/

Soluciones LAMP/WAMP: Linux/Windows - Apache - MySQL - PHP/Perl/Python....

¿Qué vimos?

- PHP
- php.ini
- $_POST, $_GET, $_SERVER
- Sesiones
- $_SESSION
- Cookies

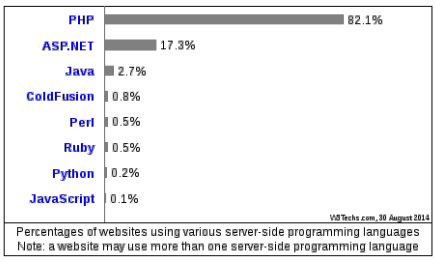
PHP Características Generales

- PHP es un lenguaje de scripting de propósito general que tiene una gran comunidad de usuarios.
- Se utiliza especialmente para aplicaciones Webs pero puede utilizarse para desarrollar cualquier tipo de aplicación (ver ejemplosClase3/comocli.php)
- Es interpretado.
- Es open source distribuido bajo una licencia libre similar a la de BSD, la PHP License v3.01.
- Website: http://php.net/

PHP en Aplicaciones Web

- Es **server-side**.
- Los scripts están embebidos en el código HTML.
- Permite construir páginas dinámicas según la solicitud del cliente y según la información disponible en el servidor.
- Se puede correr con la mayoría de los servidores web conocidos (como CGI/FastCGI/FPM/módulo del servidor).
- Es independiente de la plataforma donde corre.
- Tiene un soporte muy amplio para base de datos.
- Provee soporte para programación orientada a objetos.

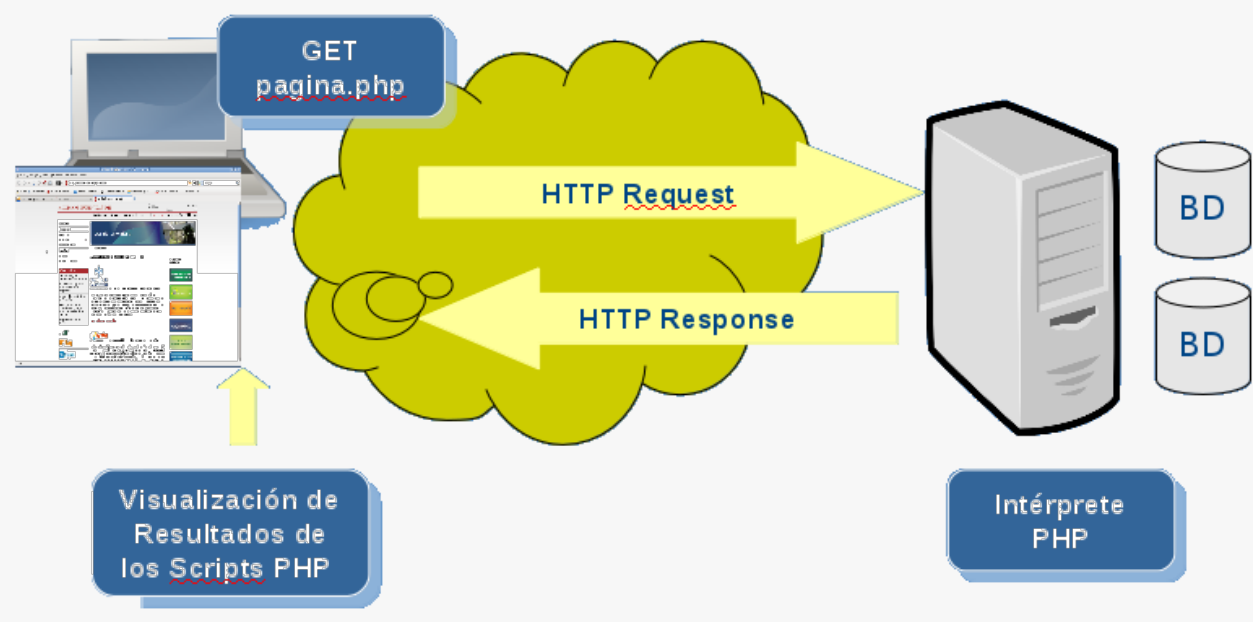
PHP en Aplicaciones Web



http://w3techs.com/technologies/overview/programming_language/all

PHP en Aplicaciones Web

Modelo de ejecución



PHP: Lo básico

- Sintaxis basada en C
- php.ini : archivo de configuracion general
- **Existen constantes predefinidas:**
    - PHP_VERSION: la versión de PHP utilizada
    - PHP_OS: el nombre del sistema operativo sobre el cual está ejecutándose PHP
    - etc.
- **Variables predefinidas (superglobals)**
    - $GLOBALS, $_SERVER, $_GET, $_POST, $_COOKIES, $_REQUEST, $_SESSION, etc.
    - Ejemplo: Para obtener DOCUMENT_ROOT se usará **$_SERVER['DOCUMENT_ROOT']**

Cookies

- Básicamente, son “tokens” en el requerimiento HTTP que permite identificar de alguna manera al cliente en el servidor.
- Se almacenan en el cliente.
- Muy usado por ser HTTP un protocolo sin estado.
- Formato: **nombreCookie=valor;expires=fecha;**

- PHP las considera variables externas: Usa **$_COOKIE** (un arreglo con las cookies generadas).
- Mediante la función setcookie() es posible grabar cookies en el cliente.
- Veamos un ejemplo de uso de cookies

---

class: destacado

Sesiones

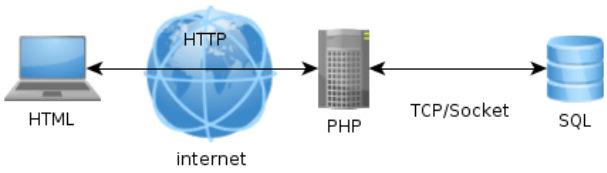
Es un mecanismo para conservar ciertos datos a lo largo de varios accesos.

- Permite registrar un número arbitrario de variables que se conservarán en las siguientes peticiones.
- Identificador: A cada visitante se le asigna un identificador único, llamado **session id** (identificador de sesión).
- **Hay dos formas de propagar un identificador de sesión:**
    - Mediante cookies
    - A través de la URL.

¿Qué vimos?

- SQL
- MySQL
- PHPmyAdmin

Accediendo a Bases de Datos



Lenguaje SQL (Structured Query Language)

- Sentencias insert, update, select, etc....
- **Ejemplos:**
    - select * from tabla where condición
    - insert into tabla (campos) values (valores)
    - update tabla set campo1='valor1' where condición

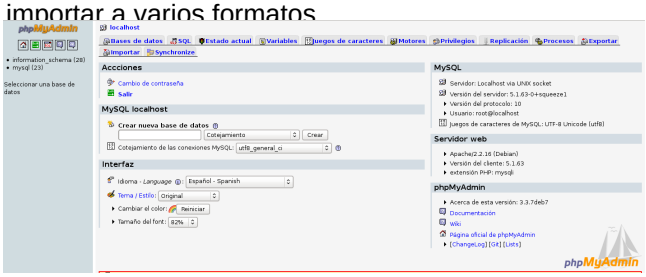
Importante

MySQL: Motor de Base de Datos distribuido por Oracle

SQL: Lenguaje de Consulta

PhpMyAdmin

- Interfaz de Administración de la Base de Datos MySQL
- Podemos exportar e importar a varios formatos



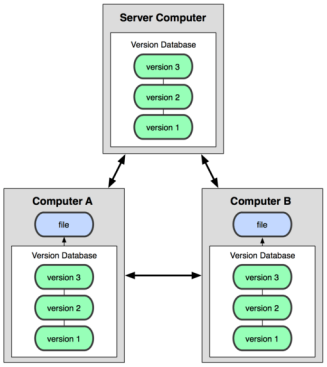
http://www.phpmyadmin.net

¿Qué vimos?

- Git
- GitLab

Git

- Git es un sistemas de control de versiones distribuido libre dieñado para manejar proyectos con velocidad y eficiencia.



GitLab

- GitLab es una aplicación opensource que nos permite administrar repositorios en git mediante una interfaz web.
- Es un clon de http://github.com y es una herramienta muy potente para el desarrollo.

¿Qué vimos?

- mysqli
- PDO
- ORM

Acceso a BBDD – MySQL

- En PHP 5, el soporte para MySQL no se encuentra habilitado por defecto.
- La extensión mysqli es la recomendad que nos permite acceder a la funcionalidad provista por MySQL 4.1.2 o superior.
- Ver formas de utilizarlo en http://www.php.net/manual/es/mysqli.construct.php

---

Abstrayéndonos → Otra opción PDO

- La extensión Objetos de Datos de PHP (PDO por sus siglás en inglés) define un interfaz ligera para poder acceder a bases de datos en PHP.
- Hay que usar un controlador de PDO específico para cada servidor de base de datos.
- PDO proporciona una capa de abstracción de acceso a datos, lo que significa que, independientemente de la base de datos que se esté utilizando, se usan las mismas funciones para realizar consultas y obtener datos → Permitiría cambiar de motor de base de datos.
- Ver drivers

ORM - Object-Relational Mapping

- Mapeo de Objetos a Base de Datos Relacionales.
- Permite acceder a una base de datos relacional como si fuera orientada a objetos.
- Transforma las llamadas a los objetos en consultas SQL, que permiten independizar el motor de BD utilizado en la aplicación.
- De acuerdo a la implementación se utiliza una sintaxis diferente.

¿Qué vimos?

- Twig
- Composer/Pear/Pecl
- Modelo Cliente Servidor
- Modelo MVC

Twig

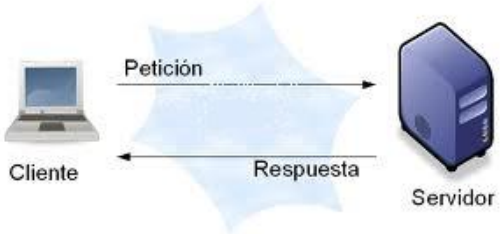
- Twig es un motor de templates en PHP promocionado como un motor de plantilla flexible, rápido, y seguro.
- Desarrollado y distribuido bajo licencia BSD. Documentación bajo licencia Creative Commons.
- **¿Por qué lo elegimos en la cátedra?**
    - Porque es la alternativa que apoya Fabien Potencier, el creador del framework Symfony. Y es la opción por defecto en Symfony 2.
    - Muchos Frameworks como Laravel o Yii lo pueden utilizar
    - Cuestiones de seguridad embebidas
    - Es muy similar a otros motores con lo cual el traspaso es inmediato

---

Dependency Manager for PHP

- https://getcomposer.org/
- https://packagist.org

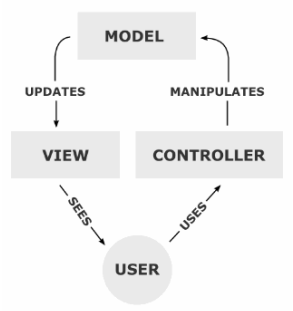
Model cliente servidor



Model – View - Controller

- **Tres componentes:**
    - Modelo
    - Vista
    - Control
- El principio más importante de la arquitectura MVC es la separación del código del programa en tres capas, dependiendo de su naturaleza.
- La lógica relacionada con los datos se incluye en el modelo, el código de la presentación en la vista y la lógica de la aplicación en el controlador.

MVC



- Reduce la complejidad, facilita la reutilización y acelera el proceso de comunicación entre capas.

¿Qué vimos?

- XML
- DTD
- Schema

XML - eXtensible Markup Language

- Es un lenguaje de marcas.
- Es un metalenguaje.
- Surge de la necesidad de contar con un mecanismo para describir información estructurada.
- XML describe semántica.
- **Existe SGML – "Standarized General Markup Language", pero ...**
    - Es complejo de procesar.
- **Existe HTML, pero ...**
    - NO fue pensado para este fin.

XML - Sintaxis Básica

¿Nos suena conocido?

```
<?xml version="1.0"?>
<ficha>
        <nombre>Roland Garros</nombre>
        <lugar>Paris</lugar>
        <estadioPrincipal fotoEstadio="estadio.jpeg"/>
</ficha>
```

- Usamos etiquetas, aunque las definimos nosotros...

DTD – Document Type Definition

- Describe la “gramática” del documento.
- **Define los elementos del documento XML:**
    - Qué elementos.
    - Qué atributos.
- Elementos vs. atributos

¿Cómo se asocia un DTD a un documento XML?

- Si se encuentra en archivo externo:

```
<!DOCTYPE elementoRaiz SYSTEM “http://servidor/DTD/archi.dtd”>
```

- Puede incluirse en la definición del documento:

```
<!DOCTYPE elementoRaiz[
        definiciones
]>
```

Schemas

- También permiten definir la estructura de un documento XML
- A diferencia de los DTD, utiliza sintaxis XML.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<xsd:element name="ficha">
<xsd:complexType>
        <xsd:sequence>
          <xsd:element name="nombre" type="xsd:string"/>
          <xsd:element name="lugar" type="xsd:string"/>
          <xsd:element name="fechaInicio"  type="xsd:date"/>
        </xsd:sequence>
        <xsd:attribute name="tipo" type="xsd:string" use="required" />
  </xsd:complexType>
......
</xsd:schema>
```

¿Qué vimos?

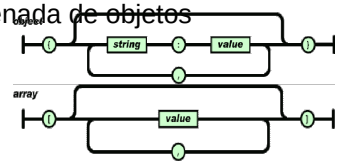
- JSON
- YAML

JSON – JavaScript Object Notation

- Formato alternativo para el envío y recepción de datos.
- **Es un subconjunto de la notación literal de objetos de Javascript.**
    - Si bien aún no vimos JS, veremos cómo es la notación.
- Se lo conoce también como LJS.
- Es un formato ligero de intercambio de datos.
- Muy popular.

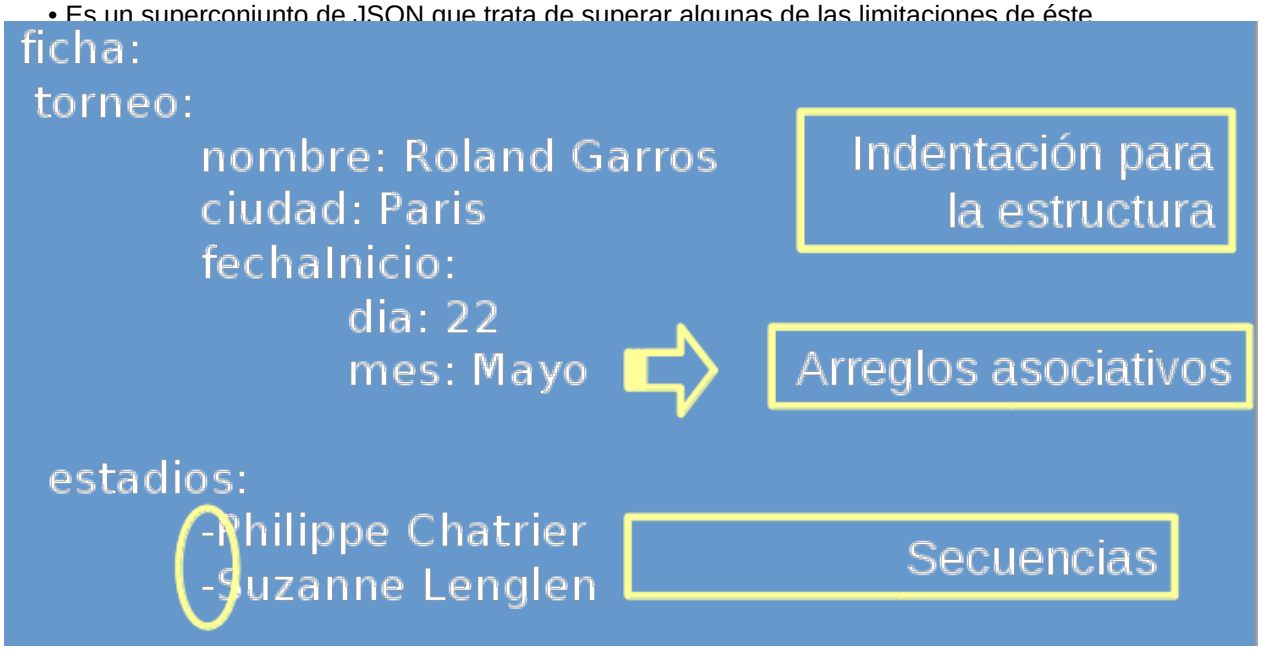
Sintaxis JSON

- **JSON está constituído por dos estructuras:**
    - Objetos: Una colección de pares de nombre/valor
    - Arreglos: Una lista ordenada de objetos



YAML – YAML Ain't Markup Language

- Es un superconjunto de JSON que trata de superar algunas de las limitaciones de éste



YAML - Notación resumida

¿Nos suena conocido?

```
ficha:
        torneo:{nombre: Roland Garros, ciudad: Paris,
        fechaInicio: { dia: 22, mes: Mayo },
        estadios: [Philippe Chatrier, Suzanne Lenglen]
```

¿Qué vimos?

- DOM
- Javascript
- Jquery
- Ajax

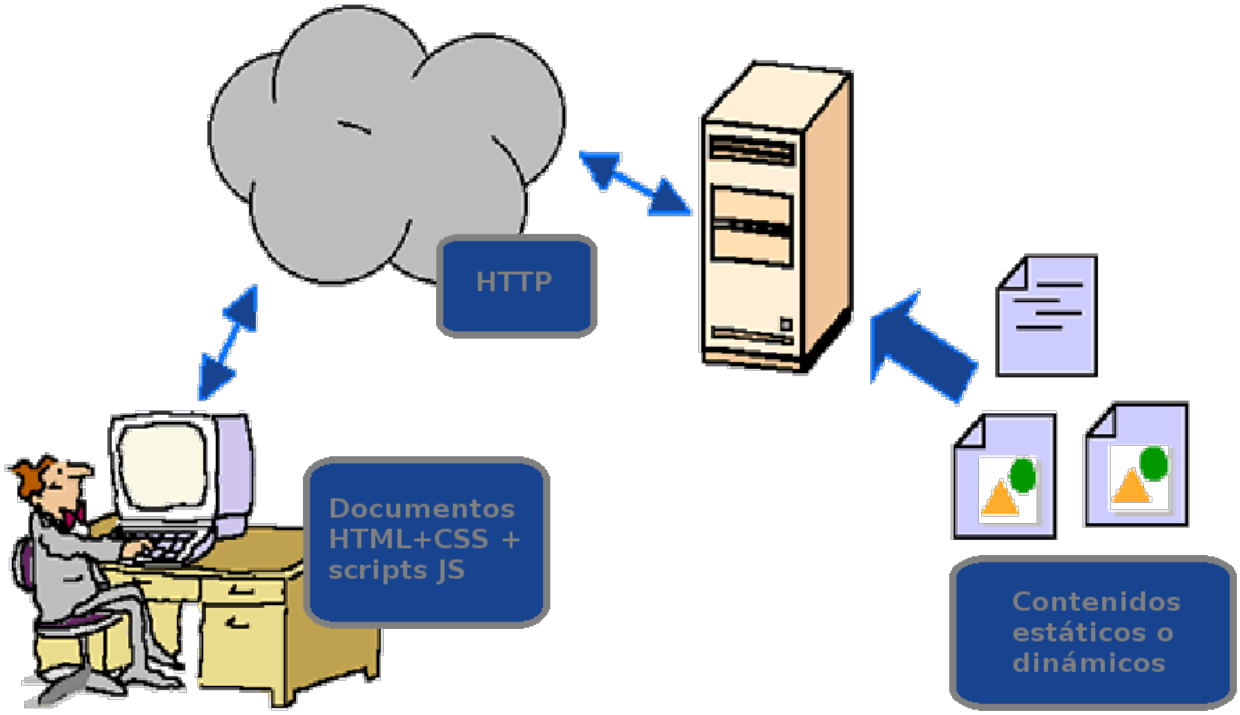
DOM

- Es una **API**, que permite acceder a los contenidos de un documento HTML/XML.
- Proporciona una **interfaz estándar** para trabajar con **eventos**.

- El documento se ve como un **árbol de nodos**.
- **Interfaz Node:** con propiedades y métodos para acceder al árbol de nodos.
- **interfaz Document**: proporciona métodos para acceder y crear otros nodos en el árbol del documento.

Javascript

- Es interpretado: el intérprete de Javascript está contenido en el navegador.



Javascript

- Es multiplataforma.
- Surgió como Livescript (creado por la empresa Netscape) y luego, junto con la empresa Sun, se convirtió en Javascript.
- El estándar es el **ECMA 262**.
- Maneja el concepto de objetos: es un lenguaje basado en prototipos.
- Es case-sensitive. La sintaxis de Javascript es similar a la del Lenguaje C o Java.
- Es un lenguaje de asignación dinámica de tipos.
- V8 (motor JavaScript desarrollado por Google)

---

jQuery

- Una de las tantas ...
- Muy usada.
- Se debe incluir el archivo jquery.js (descargado de http://jquery.com/download/)
- Es código Javascript:

```
<script  src="ruta/jquery.js"> </script>
```

---

jQuery (cont.)

- **Nos provee formas de acceder a los elementos con atajos a la función DOM getElementById.**
    - Con DOM: **document.getElementById("p1")**
    - Con JQuery: **$("#p1")**
- **JQuery usa los selectores CSS para acceder a los elementos:**
    - **$("p"):** todos los elementos <p>.
    - **$("#elem"):** el elemento cuyo id="elem".
    - **$("p.intro"):** todos los elementos <p> con class="intro".
    - **$(".intro"):** todos los elementos con class="intro"
    - **$("p#demo"):** todos los elementos <p> id="demo".
    - **$(this):** el elemento actual
    - **$("ul li:odd"):** Los <li> impares dentro de

AJAX

Asynchronous JAvascript + XML

AJAX

- NO es una tecnología, sino una combinación de varias tecnologías.
- **AJAX incluye:**
    - Presentación basada en estándares usando **XHTML** y **CSS**;
    - Exhibición e interacción dinámicas usando **DOM**;

- Intercambio y manipulación de datos usando **XML** y **XSLT**; (podemos usar otras notaciones también)
- Recuperación de datos asincrónica usando **XMLHttpRequest**;
- **JavaScript** como lenguaje de programación.

¿Qué vimos?

- API
- Oauth
- OpenID

Qué es una API?

- Interfaz de programación de aplicaciones (IPA) o API (del inglés Application Programming Interface) es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.
- Ejemplo: librerías del sistema operativo

¿Qué nos permiten las API?

- **En general:**
  - Intercambiar datos con un tercero.
  - Aprovechar el software y/o capacidad de procesamiento y almacenamiento de terceros para utilizarlo en nuestro sistema pero sin necesariamente incluirlo en nuestro desarrollo sino sólo invocándolo.
  - Los cambios en lo que está en la capa de atrás de la API no nos afectan.
  - Por ej. si la API define una función **listarDatos**, desde nuestro desarrollo no nos afecta que la implementación de esa función cambie de usar un while a un for siempre que devuelva lo que esperamos.

Oauth

- El protocolo OAuth, es un protocolo de autorización, más exactamente, de delegación de acceso.
- Es decir, permite definir cómo un tercero va a acceder a los recursos propios.

¿Qué es OpenID Connect?

- OpenID Connect 1.0 es una capa de identificación construida sobre OAuth 2.0.
- Permite al cliente verificar la identidad del usuario final basándose en la autenticación realizada por el servidor de autorizacion,
- Facilita además obtener información básica del perfil del usuario final.
- OpenID Connect permite cliente de todo tipo web, mobile, y clientes JavaScript clients.
- Opcionalmente se puede utilizar encriptación, discovery de proveedores OpenID, o manejo de sesión.

¿Qué vimos?

- SQLi
- XSS

SQL Inyección

Obtener acceso a una aplicación:

- Suponiendo que la consulta de autenticación de una pagina que pide id y pass es:

```
select * from users where id='". **$id** ."'
              and pass='". **$pass** ." ' ;
```

- Suponiendo **$id='admin'** y **$pass='admin'** el sql quedaría:

```
select * from users where **id='admin'** and **pass='admin'**;
```

SQL Inyección

- ¿Qué sucede si usamos **$id=$pass= 1' or '1=1**?

```
select * from users where id='". **"1' or '1=1"** ."' and
                          pass='". **"1' or '1=1"** ."'";
```

- Lo que se se resuelve en:

```
select * from users where **id='1' or '1=1'**
                  and **pass='1' or '1=1'** ;
```

- **(Cualquier cosa OR True)** es siempre **TRUE**

XSS - Cross Site Scripting

- Se lo conoce como XSS para que no sea confundido con CSS.
- En general ocurren cuando una aplicación toma datos de un usuario, no los filtra en forma adecuada y los retorna sin validarlos ni codificarlos.
- Puede insertarse HTML, Javascript, entre otros, a través de los formularios o la URL.
- Con esta vulnerabilidad es posible robar el acceso de los usuarios y violar la integridad y confiabilidad de sus datos. Por ejemplo robando nombres de usuarios, claves, cookies. También es posible ejecutar código en forma remota inyectando código a través de la URL.
- **Existen tres tipos conocidos de fallas XSS:**

    1. Almacenados,
    2. Reflejados,
    3. XSS basado en DOM.