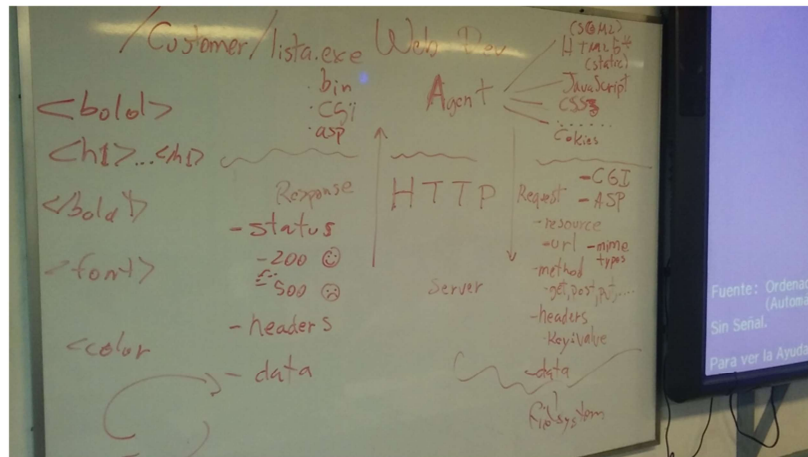


Desarrollo Web

Cliente



Desarrollo Web



Referencia

https://www.ntu.edu.sg/home/ehchua/programming/webprogramming/HTTP_Basics.html

<https://github.com/jkbrzt/httpie>

HTTP, RFCs (7230-7237)

<http://shop.oreilly.com/product/9781565925090.do>

HTML

- La etiqueta.
 - DOM.
 - Metadata (atributos).
- HTML5*.

CSS

- Lenguaje para consultas sobre el DOM (Document Object Model).
- Selector (query).
- Estilos.
- Graceful Degradation vs Progressive Enhancement.
 - Responsive Design.
 - Mobile First.
- Bibliotecas y frameworks.
 - Bootstrap.
 - jQuery Mobile.
 - ...

Bootstrap

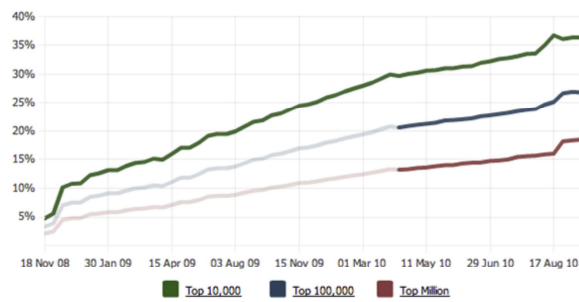
- Fuertemente posicionado en la comunidad.
- Framework de CSS.
- Colección de sectores y estilos.
- Excelente para prototipos.
- Se recomienda personalizar.

JavaScript

- El lenguaje de programación de interpretados por los navegadores.
- El navegador como un Sistema Operativo.
- Diversidad de APIs.
- Estándar ECMAScript.
- Bibliotecas y frameworks.
 - jQuery.
 - Knockout.
 - Angular.
 - ...

jQuery

Biblioteca de JavaScript más popular



jQuery

- Fácil de aprender
- Permite cargar plugins
- Soporte para selección DOM al estilo de CSS
- Pequeña
- Soportado por la comunidad y Microsoft

jQuery, Obtener

- Obtener la última versión
 - [jQuery.com](https://jquery.com)
- Incluirla como parte del sitio
 - Incluir en su proyecto
- CDN (Content Delivery Network)
 - Microsoft, jQuery, Google
- Formas de empaquetado
 - Source
 - Minified (utilizar esta en producción)

jQuery, Fundamentos

- Siempre primero se busca algo (QUERY)
 - Se utilizan los “selectores” disponibles en CSS, más métodos de selección propios de la biblioteca
- Luego se hace algo con lo encontrado

```
$("#contenido").hide();
```

```
<div id="contenido"></div>
```

jQuery, Eventos

```
function() myClickHandler {  
    $(this).css('color', 'red');  
};
```

```
$('#a.tab').click(myClickHandler);
```

jQuery, Eventos

```
$('#a.tab').click( function() {  
    $(this).css('color', 'red');  
});
```

jQuery, Encadenamiento

```
$('tr:odd')  
  .addClass('odd')  
  .click( function () {  
    alert('you clicked me!');  
  });
```

Knockout

- MVVM.
- Bindings declarativos.
- Plantillas.



Knockout es una biblioteca JavaScript que ayuda a crear interfaces de usuario enriquecidas, donde la interface de usuario (HTML) y el modelo de datos subyacente (ViewModel) permanece enlazado de forma clara y limpia. Cada vez que tenga secciones de interfaz de usuario que se actualizan de forma dinámica (por ejemplo, cambiando en función de las acciones del usuario o cuando una fuente de datos externa cambia), Knockout puede ayudarle a implementar esa funcionalidad de forma más simple.

Características

- Seguimiento de Dependencias: Actualiza automáticamente las partes correctas de la interfaz de usuario cada vez que hay cambios en el modelo de datos.
- Enlaces declarativos: Una manera simple para conectar partes de la interfaz de usuario con el modelo de datos. Es posible construir interfaces de usuario dinámicas fácilmente utilizando contextos de enlace, que pueden ser anidados.
- Extensible.

Beneficios

- Biblioteca de JavaScript: Funciona con cualquier tecnología de servidor o en el cliente.
- Puede ser añadido sobre una aplicación web existente sin necesidad de grandes cambios en la arquitectura.
- Compacto: Alrededor de 13kb después gzipping.
- Funciona en cualquier navegador convencional (IE 6+, Firefox 2+, Chrome, Safari, Edge, otros).

Si se tiene experiencia en el patrón MVC es más clara su comprensión y aprovechamiento inicial.

Herramientas Nodejs

Nodejs

- Plataforma para el desarrollo de aplicaciones utilizando JavaScript.
- Interpreta JavaScript.
- Arquitectura de I/O asíncrona.



Node.js es una plataforma de código abierto para el desarrollo de aplicaciones web (entre otros) a nivel de servidor. Aunque Node.js no es un framework de JavaScript, muchos de sus módulos básicos están escritos en JavaScript, y los desarrolladores pueden escribir nuevos módulos en JavaScript.

El entorno de ejecución de Nodejs interpreta JavaScript usando el motor V8 de JavaScript de Google.

Node.js tiene una arquitectura orientada a eventos capaz de I/O asíncrona. Este diseño tiene como objetivo optimizar el rendimiento y la escalabilidad de aplicaciones Web con muchas operaciones de I/O, así como para las aplicaciones Web en tiempo real (por ejemplo, programas de comunicación en tiempo real y juegos de navegador).

NPM

- NPM es el gestor de paquetes de JavaScript basado en Nodejs.
- Sirve como punto de entrada para <https://www.npmjs.com/>.
- Es como el Nuget.org para .NET Core.



El npm ya viene instalado como parte del nodejs, pero puede actualizado a su última versión ejecutando desde la línea de comandos (con derechos de administración):
`npm install npm -g`. npm es un módulo de los muchos existentes para node.

Bower

- El reto de administrar archivos de tecnologías cliente.
- Bower es el gestor de paquetes de tecnologías web cliente basado en Nodejs.



Los sitios web están hechos de un montón de cosas (frameworks, bibliotecas, utilidades, entre otros). Bower gestiona todas estas cosas.

Hacer un seguimiento de todos los paquetes y permite la actualización de versiones al día (o conjunto de las versiones específicas/dependencias que se necesitan).

Bower puede administrar los componentes que contienen HTML, CSS, JavaScript, fuentes e incluso archivos de imagen. Bower no concatena o minifica código o cualquier otra cosa, sólo instala las versiones correctas de los paquetes que necesitan y sus dependencias.

Bower ofrece una solución genérica al problema de la gestión de paquetes de front-end, mientras expone el modelo de dependencias de paquetes a través de una API que puede ser reutilizada por otras aplicaciones.

Bower corre sobre Git, y es independiente del paquete. Un paquete de Bower puede estar compuesto de cualquier tipo de archivo, y el uso de cualquier tipo de transporte (por ejemplo, AMD, CommonJS, etc.).

Bower puede ser instalado/actualizado a su última versión ejecutando desde la línea de comandos (con derechos de administración): `npm install bower -g`

Para las definiciones de versiones de paquetes en Bower ver:

<http://developer.telerik.com/featured/mystical-magical-semver-ranges-used-npm-bower/>

Algunas combinaciones frecuentes al utilizar Bower:

- El archivo `.bowerrc` se debe crear manualmente. El siguiente ejemplo define la carpeta donde localmente quedarán los paquetes. `{ "directory": "app/components/" }`
- `bower -?`, ayuda sobre bower
- `bower init`, para crear el archivo `bower.json`
- `bower search <package>`, buscar un paquete en bower
- `bower info <package>`, para obtener información acerca de un paquete, en particular las versiones disponibles
- `bower install <package> --save`, instalar un paquete y dejarlo registrado en archivo de `bower.json`
- `bower install <package>#<versión> --save`, instalar un paquete para una versión específica
- `bower list`, listar los paquetes instalados
- `bower update`, actualizar paquete de acuerdo a la política de versión definida para el paquete en `bower.json`
- `bower uninstall <package>`, desinstalar un paquete

