



HTML5 APIS

Quique Fdez. Guerra
efernandez@plainconcepts.com
@CKGrafico

CLASSLIST

Muy simple, nos permite manipular las clases de los elementos.

Antes

Usamos `className` que es una propiedad con un string y todas las clases.

```
element.className = 'class1 class2';
```

Ahora


Tenemos diferentes métodos para poder interactuar con las clases de un elemento.

```
element.classList.add('class1');
```

```
element.classList.remove('class1');
```

```
element.classList.contains('class1');
```

```
element.classList.toggle('class1', 1 < 2);
```

classList (DOMTokenList)  - LS Global 70.37% + 18.37% = 88.74%

Method of easily manipulating classes on elements, using the DOMTokenList object.

Current aligned **Usage relative** **Show all**

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Chrome for Android
			29						
			45						
			48					1 2 3 4.3	
		45	49			8.4		4.4	
		46	50			9.2		4.4.4	
8									
1 2 3 11	13	47	51	9.1	38	9.3	8	50	50
	14	48	52	10	39				
		49	53	TP	40				
		50	54						

Notes **Known issues (1)** **Resources (7)** **Feedback**

¹ Does not have support for **classList** on SVG or MathML elements.

² Does not support the second parameter for the **toggle** method

³ Does not support multiple parameters for the **add()** & **remove()** methods

DATASET

Nos permite manipular los atributos de tipo data-*

Antes


Tenemos que mirar igual que el resto de atributos.

```
element.attributes['data-name'];
```

Ahora

Podemos trabajar directamente con un diccionario que contiene solo los atributos de tipo data-*

```
element.dataset.name;
```

dataset & data-* attributes  - LS Global 88.06% + 7.31% = 95.37%

Method of applying and accessing custom data to elements.

Current aligned Usage relative Show all

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Chrome for Android
			<div>29</div> <div>45</div> <div>48</div>					<div>4.3</div> <div>4.4</div> <div>4.4.4</div>	
		<div>45</div> <div>46</div>	<div>49</div> <div>50</div>			<div>8.4</div> <div>9.2</div>			
<div>8</div>									
<div>11</div>	<div>13</div>	<div>47</div>	<div>51</div>	<div>9.1</div> <div>10</div> <div>TP</div>	<div>38</div> <div>39</div> <div>40</div>	<div>9.3</div>	<div>8</div>	<div>50</div>	<div>50</div>
	<div>14</div>	<div>48</div> <div>49</div> <div>50</div>	<div>52</div> <div>53</div> <div>54</div>						

Notes Known issues (1) Resources (7) Feedback

Partial support refers to being able to use **data-*** attributes and access them using `getAttribute`.

"Supported" refers to accessing the values using the **dataset** property. Current spec only refers to support on HTML elements, only some browsers also have support for SVG/MathML elements.

¹ While the HTML spec doesn't require it, these browsers also support **dataset** and **data-*** attributes on SVG elements, in compliance with [current plans for SVG2](#)

DRAG AND DROP

Podemos arrastrar y soltar elementos en contenedores.

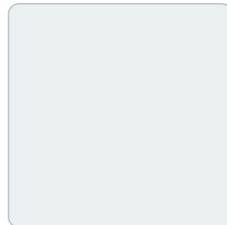
```
<div draggable="true"></div>
<div></div>
```

*ondragover -> e.preventDefault();

```
element.addEventListener('dragstart', fn);
element2.addEventListener('dragover', fn);
element2.addEventListener('drop', fn);
```

```
event.dataTransfer.setData("text/plain", val);
event.dataTransfer.getData("text/plain"); // 1 !!
event.dataTransfer.clearData("text/plain");
```

event
yOffsi



ffset,



GEOLOCATION

Da información sobre la geolocalización de un usuario.

En algunos navegadores solo funciona con https.

```
var options = {  
  enableHighAccuracy: true,  
  timeout: 5000,  
  maximumAge: 0  
};
```

```
navigator  
  .geolocation  
  .getCurrentPosition(success, error, options);
```

```
navigator  
  .geolocation  
  .watchPosition(success, error, options);
```

Geolocation - REC									
Global									
89.28% + 0.01% = 89.28%									
Method of informing a website of the user's geographical location									
Current aligned Usage relative Show all									
IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Chrome for Android
			29						
			45						
			48					4.3	
		45	49			8.4		4.4	
8		46	50			9.2		4.4.4	
11	13	47	51	9.1	38	9.3	8	50	50
	14	48	52	10	39				
		49	53	TP	40				
		50	54						
Notes Known issues (4) Resources (6) Feedback									
No notes									

HISTORY

Sirve para leer y manipular el historial del navegador respecto nuestro site.

Antes

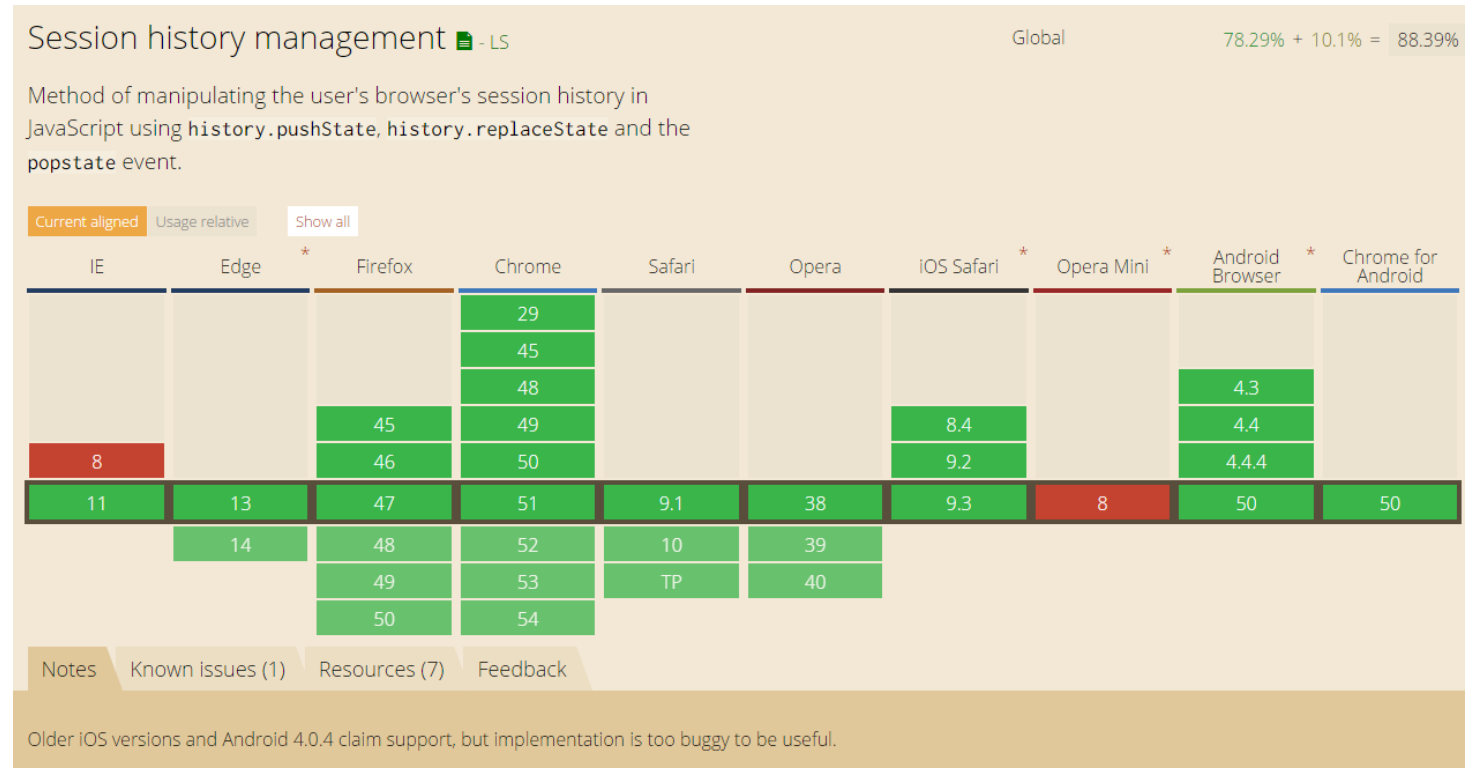
Utilizábamos los anchors '#'
`element.attributes['data-name'];`

Ahora

Somos capaces de navegar en el histórico (de la página), incluso añadir nuevos elementos.

```
var data = { foo: "bar" };  
history.pushState(data, "title", "url");  
history.replaceState(data, "title", "url");  
Evento: 'popstate'
```

```
history.back();  
history.forward();  
history.go(n);
```



INDEXED DB

Para guardar grandes cantidades de datos en la parte de frontend.

Antes

Cookies o nada, más bien nada.

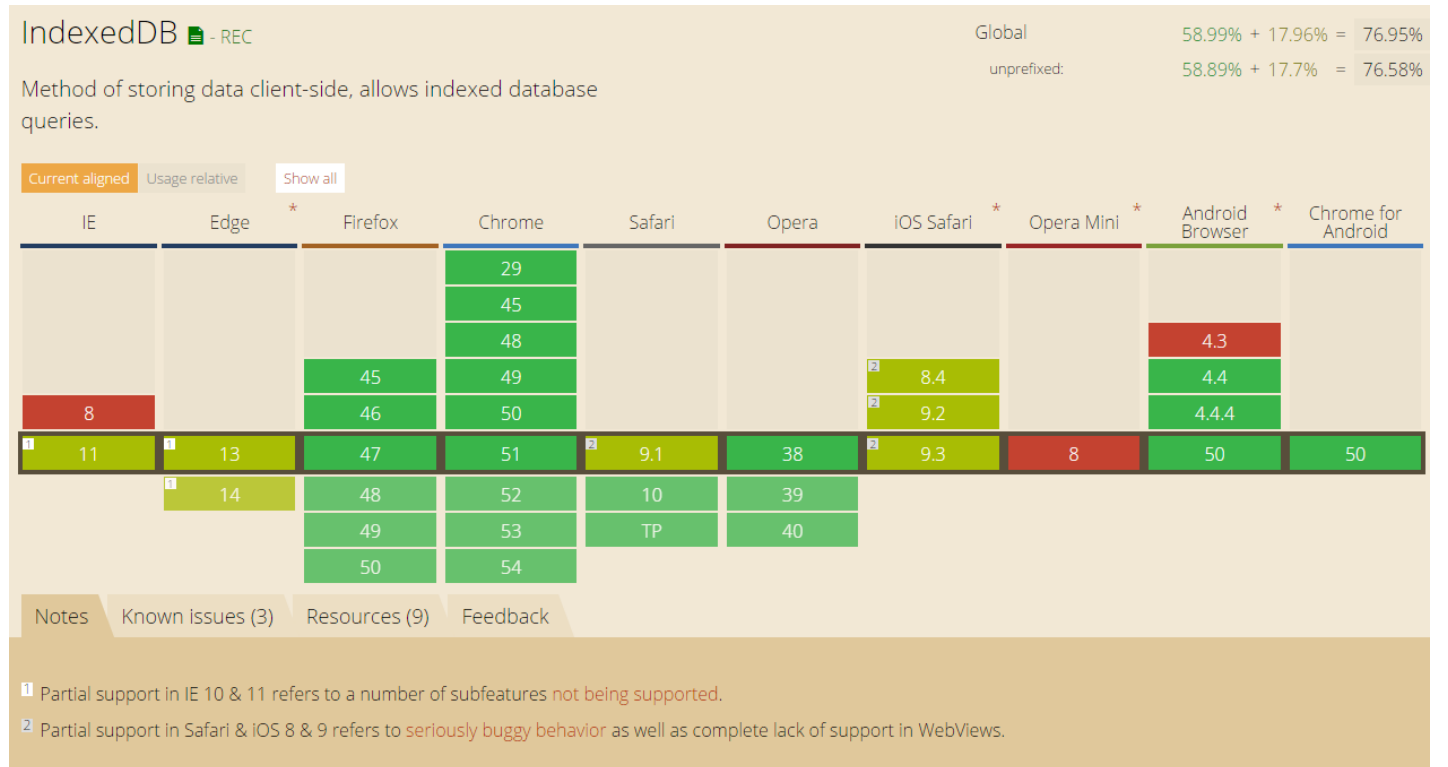
Ahora

Trabajamos de forma parecida a si tuviéramos una base de datos en cliente.

```
var req = indexedDB.open("DB", v);
var db;
req.onsuccess = function(e) {
  db = e.target.result;
}
```

```
db.createObjectStore("table");
var trans = db.transaction(["table"], "readwrite");
var store = trans.objectStore("table");
var req = store.add({a:1}, key);
```

```
db.createObjectStore("table", { keyPath: "a" });
db.createObjectStore("table", { autoIncrement: true });
```



WEBSTORAGE

Para guardar pequeñas cantidades de datos en la parte de frontend.

Antes

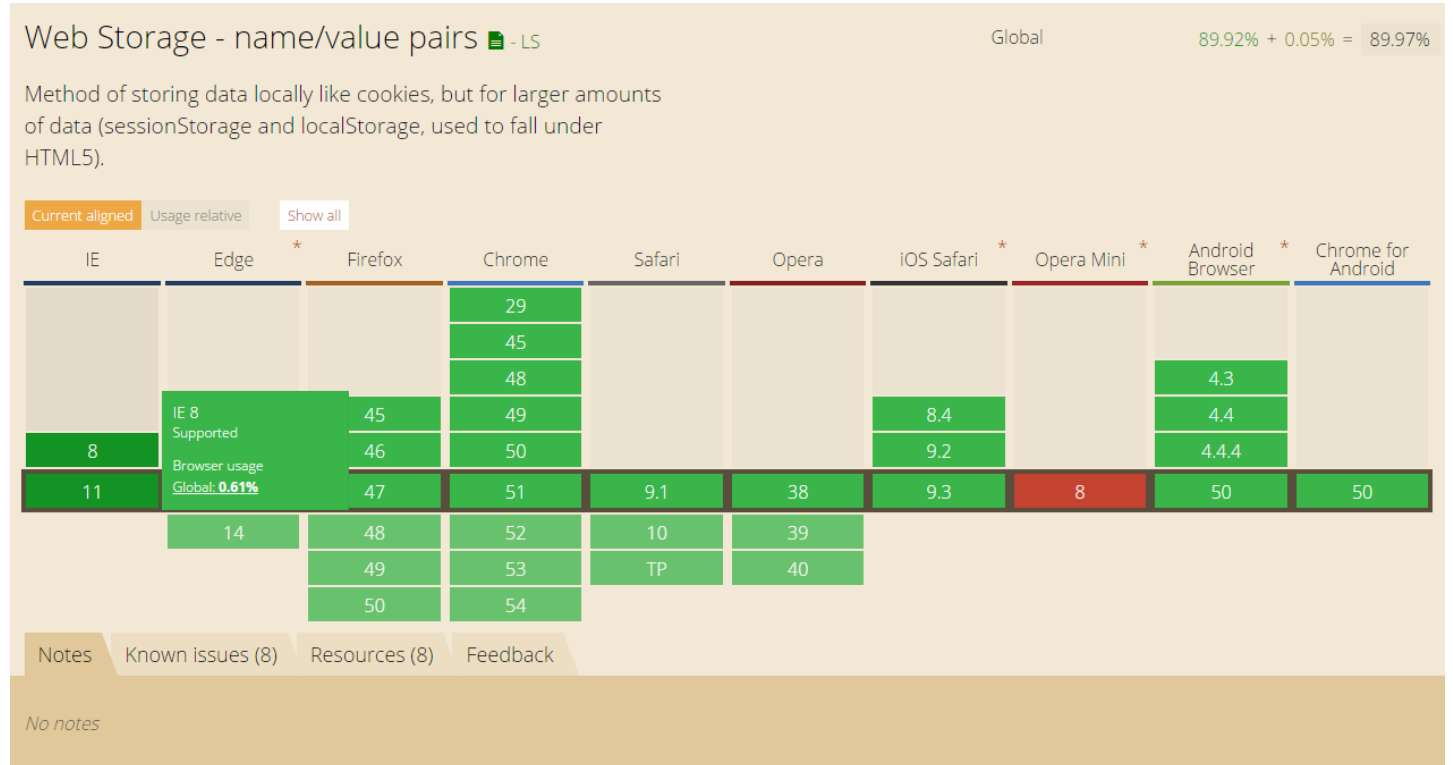
Cookies.

Ahora

Podemos guardar información por sesión o de forma local.

```
localStorage.setItem('item', value);  
localStorage.getItem('item');
```

```
sessionStorage.setItem('item', value);  
sessionStorage.getItem('item');
```



MATCH MEDIA

Mediaqueries en JavaScript

Antes

¿Mirando tamaño de la ventana?

Ahora

Podemos mirar cualquier mediaquery en cualquier momento.

```
if (matchMedia("(min-width: 400px)").matches) {  
  // Más de 400px de ancho  
}  
matchMedia("(min-width: 400px)").onchange  
// Se lanza al cambiar el estado de la mediaquery
```

matchMedia - WD Global 88.59%

API for finding out whether or not a media query applies to the document.

Current aligned Usage relative Show all

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Chrome for Android
			29						
			45						
			48						
		45	49			8.4		4.3	
8		46	50			9.2		4.4	
11	13	47	51	9.1	38	9.3	8	50	50
	14	48	52	10	39				
		49	53	TP	40				
		50	54						

Notes Known issues (0) Resources (7) Feedback

No notes

REQUEST ANIMATION FRAME

La API definitiva para intervalos sin congelar la UI (por fin)

Antes

setInterval

Ahora

Podemos hacer intervalos utilizando la gráfica y el refresco vertical de la pantalla.

```
function fn() {  
  requestAnimationFrame(fn);  
}  
requestAnimationFrame(fn);
```

requestAnimationFrame - LS

API allowing a more efficient way of running script-based animation, compared to traditional methods using timeouts. Also covers support for `cancelAnimationFrame`

Global 87.02% + 0.24% = 87.27%
unprefixed: 86.73%

Current aligned	Usage relative	Show all							
IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Chrome for Android
			29						
			45						
			48					4.3	
		45	49			8.4		4.4	
		46	50			9.2		4.4.4	
8									
11	13	47	51	9.1	38	9.3	8	50	50
	14	48	52	10	39				
		49	53	TP	40				
		50	54						

Notes Known issues (1) Resources (6) Feedback

¹ Partial support refers to lacking `cancelAnimationFrame` support.

² Supports `webkitCancelRequestAnimationFrame` rather than ``webkitCancelAnimationFrame`.

WEB WORKERS

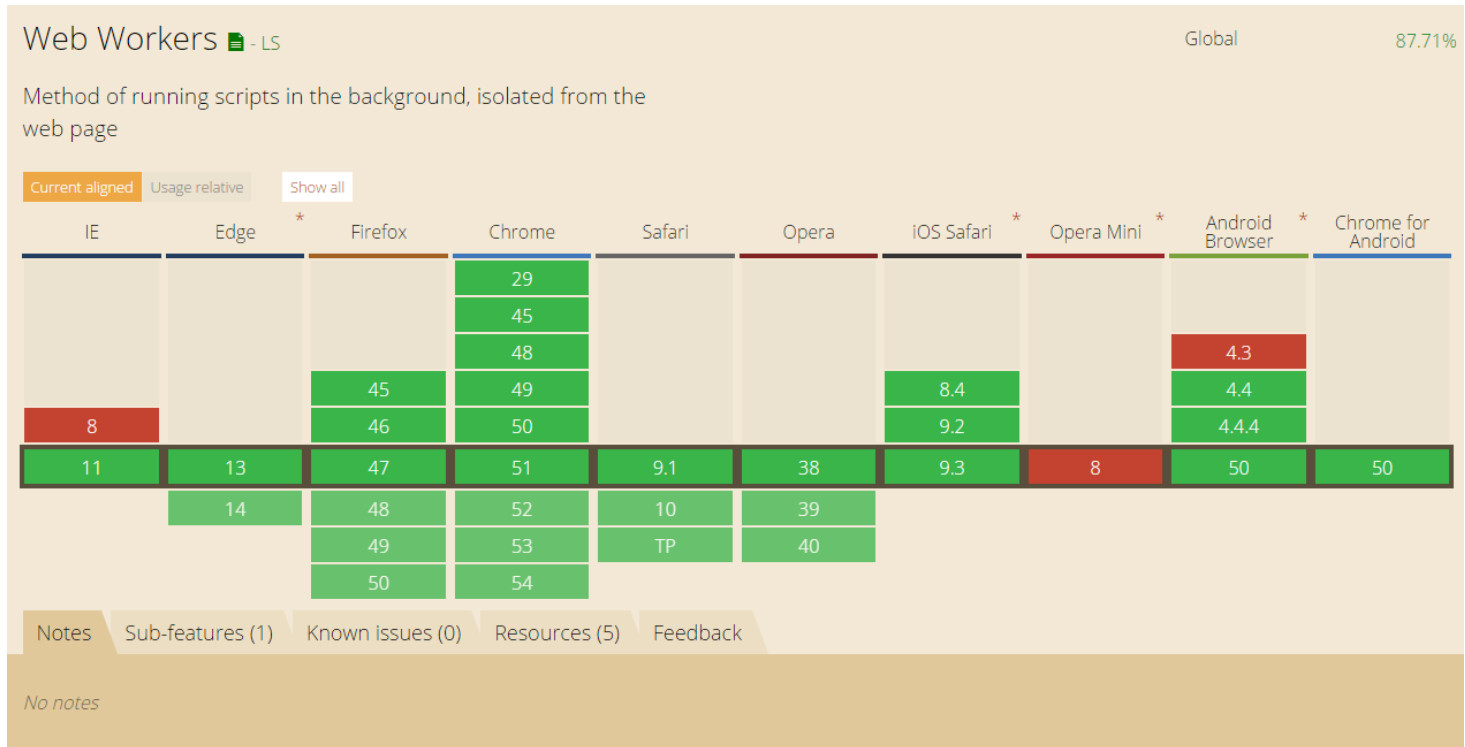
Permite trabajar de forma simultánea con otros subprocesos.

script.js

```
var worker = new Worker('worker.js');
worker.addEventListener('message', function(e) {
  console.log(e.data);
});
worker.postMessage("Hi!");
```

worker.js

```
self.addEventListener('message', function(e) {
  self.postMessage(e.data);
});
```



WEB SOCKETS

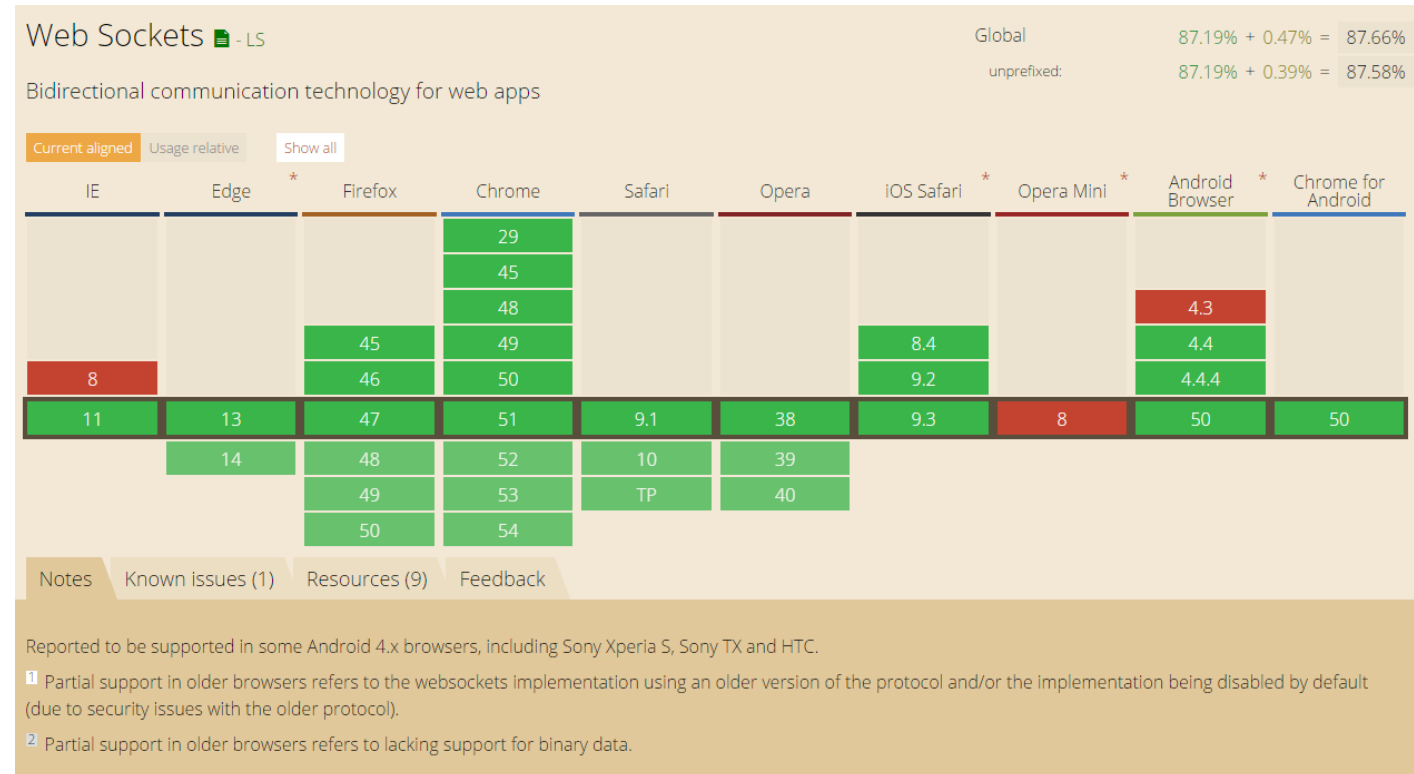
Establecemos conexiones persistentes entre cliente y servidor.

<http://socket.io/>

<http://www.asp.net/signalr>

script.js

```
var socket = new WebSocket(websocketurl);
socket.send(message);
socket.close();
```



CANVAS

Para dibujar gráficos utilizando scripts, desde gráficos de barras hasta juegos o mapas.

script.js

```
<canvas id="tutorial" width="150" height="150"></canvas>
```

Canvas (basic support) - LS

Global

89.3% + 5.39% = 94.69%

Method of generating fast, dynamic graphics using JavaScript.

Current aligned	Usage relative	Show all							
IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
			29						
			45						
			48					4.3	
		45	49			8.4		4.4	
		46	50			9.2		4.4.4	
8									
11	13	47	51	9.1	38	9.3	8	50	50
	14	48	52	10	39				
		49	53	TP	40				
		50	54						

Notes Sub-features (4) Known issues (6) Resources (9) Feedback

For screen readers, IE, Chrome & Firefox support the [accessible canvas element sub-DOM](#).
Firefox & Chrome also support the [drawfocus ring](#).

¹ Does not support [toDataURL\(\)](#)

² Opera Mini supports the canvas element, but is unable to play animations or run other more complex applications.