

Aproximación a Canvas

Aproximation to Canvas

Autor 1: Camilo E. Muñoz A.

Departamento o Escuela, Universidad, Ciudad, País

Correo-e: Camilo.munoz2@utp.edu.co

Resumen— Se realizará una aproximación acerca del lenguaje canvas en html ligado a css y javascript

Palabras clave— Lienzo, Gráficos, HTML

Abstract— There will be an approximation about the canvas language in html linked to css and javascript.

Key Word — Canvas, Graphics, HTML

I. INTRODUCCIÓN

HTML5 incluye un emocionante elemento llamado CANVAS. Tiene muchos usos, pero para poder utilizarlo, necesitas aprender algo de JavaScript, HTML y a veces CSS. Esto hace que el elemento CANVAS sea un poco desalentador para muchos diseñadores, y de hecho, la mayoría probablemente ignorará el elemento hasta que haya herramientas fiables para crear animaciones y juegos CANVAS sin conocer JavaScript.

II. DESCRIPCION

El contenido debe tener capítulos y subcapítulos enumerados. Canvas es uno de los componentes más novedosos de estándar HTML 5 que sirve para dibujar dinámicamente imágenes en una página web.[1]

Canvas significa en español algo así como lienzo y es básicamente eso, un área donde podemos dibujar como si fuera un lienzo, este elemento canvas crea un área rectangular donde los autores pueden representar gráficos y animaciones programáticamente y en tiempo real. Este es un recurso valorable al diseñar sitios ricos en animación e interactividad, o cuando se desarrollan aplicaciones dentro del documento o videojuegos.[2]

El elemento canvas permite especificar un área de la página donde se puede, a través de scripts, dibujar y renderizar imágenes, lo que amplía notablemente las posibilidades de las páginas dinámicas y permite hacer cosas que hasta ahora estaban reservadas a los desarrolladores en Flash, con la ventaja que para usar canvas no será necesario ningún plugin en el navegador, lo que mejorará la disponibilidad de esta nueva aplicación.[1]

¿Para qué se utiliza el Canvas HTML5?

El elemento HTML5 CANVAS puede ser usado para muchas cosas que antes, había que usar una aplicación incrustada como Flash para generar:

Gráficos dinámicos

Juegos en línea y fuera de línea

Animaciones

Video y audio interactivo

De hecho, la razón principal por la que la gente usa el elemento CANVAS es por lo fácil que es convertir una página web sencilla en una aplicación web dinámica y luego convertirla en una aplicación móvil para su uso en teléfonos inteligentes y tabletas.[3]

¿Cuándo considerar el uso del elemento Canvas?

Su audiencia debe ser su primera consideración a la hora de decidir si utilizar el elemento Canvas.

Si su audiencia está usando principalmente Windows XP e IE 6, 7 u 8, entonces crear una característica de Canvas dinámico no tendrá sentido ya que esos navegadores no lo soportan.

Si está construyendo una aplicación que se utilizará sólo en máquinas Windows, entonces Flash puede ser su mejor opción. Una aplicación para ser usada en ordenadores Windows y Mac podría beneficiarse de una aplicación Silverlight.

Sin embargo, si su aplicación necesita ser visualizada en dispositivos móviles (tanto Android como iOS), así como en ordenadores de sobremesa modernos (actualizados con las últimas versiones de los navegadores), entonces utilizar el elemento Canvas es una buena opción.

Tenga en cuenta que el uso de este elemento le permite tener opciones alternativas como imágenes estáticas para navegadores antiguos que no lo soportan.

Sin embargo, no se recomienda utilizar el Canvas HTML5 para todo. Nunca debes usarlo para cosas como tu logo, titular o navegación (aunque usarlo para animar una parte de cualquiera de ellos estaría bien).

De acuerdo con la especificación, usted debe utilizar los elementos que sean más adecuados para lo que está tratando de construir. Por lo tanto, es preferible utilizar el elemento Header

junto con imágenes y texto que el elemento Canvas para el encabezado y el logotipo.

Además, si está creando una página web o aplicación que está destinada a ser utilizada en un medio no interactivo como la impresión, debe tener en cuenta que el elemento Canvas que ha sido actualizado dinámicamente puede no imprimirse como usted espera. Es posible que obtenga una impresión del contenido actual o del contenido alternativo.[3]

El elemento canvas es sólo un contenedor para gráficos. Debe utilizar JavaScript para dibujar los gráficos.

Canvas tiene varios métodos para dibujar trazados, cuadros, círculos, texto y añadir imágenes.

Un Canvas es un área rectangular en una página HTML. Por defecto, un lienzo no tiene borde ni contenido.

El margen de beneficio se ve así:

```
<canvas id="myCanvas" width="200"
height="100"></canvas>
```

Nota: Especifique siempre un atributo id (al que se hará referencia en un script), y un atributo width y height para definir el tamaño del lienzo. Para añadir un borde, utilice el atributo style.

Aquí hay un ejemplo de un lienzo básico y vacío:



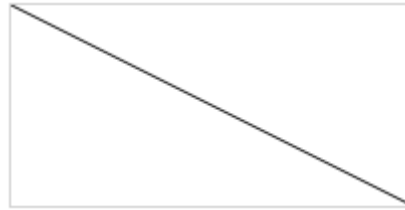
```
<!DOCTYPE html>
<html>
<body>
```

```
<canvas id="myCanvas" width="400" height="200"
style="border:2px solid #000000;">
</canvas>
```

```
</body>
</html>
```

Ejemplos de canvas

1. Dibujar una línea



```
<html>
<body>
```

```
<canvas id="myCanvas" width="600" height="200"
style="border:1px solid #d3d3d3;">
```

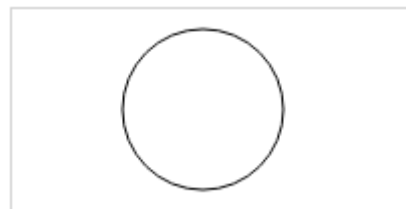
```
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.moveTo(0, 0);
ctx.lineTo(600, 200);
ctx.stroke();
```

```
</script>
```

```
</body>
```

```
</html>
```

2. Dibujar un círculo



```
<!DOCTYPE html>
<html>
<body>
```

```
<canvas id="myCanvas" width="400" height="200" style
="border:2px solid #d3d3d3;">
```

```
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.beginPath();
ctx.arc(200,100,50,0,2*Math.PI);
ctx.stroke();
</script>
```

```
</body>
```

```
</html>
```

3. Escribir texto con trazo



```
<html>
<body>

<canvas id="myCanvas" width="600" height="200"
style="border: 1px solid #d3d3d3;">

<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.font = "30px Arial";
ctx.strokeText("Hello World", 10, 50);

</script>

</body>
</html>
```

4. Dibujar Gradiente Circular



```
<html>
<body>

<canvas id="myCanvas" width="600" height="200"
style="border: 1px solid #d3d3d3;">

<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");

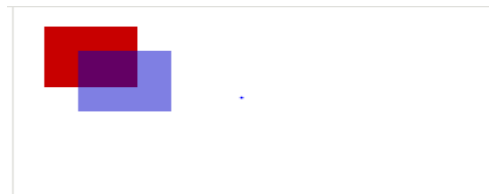
// Create gradient
var grd = ctx.createRadialGradient(75, 50, 5, 90, 60, 100);
grd.addColorStop(0, "red");
grd.addColorStop(1, "white");

// Fill with gradient
ctx.fillStyle = grd;
ctx.fillRect(10, 10, 150, 80);

</script>

</body>
</html>
```

5. Dibujar dos rectángulos que intersecan



```
<html>
<head>
<script type="application/javascript">
function draw() {
var canvas = document.getElementById("canvas");
if (canvas.getContext) {
var ctx = canvas.getContext("2d");

ctx.fillStyle = "rgb(200,0,0)";
ctx.fillRect (10, 10, 55, 50);

ctx.fillStyle = "rgba(0, 0, 200, 0.5)";
ctx.fillRect (30, 30, 55, 50);
}
}
</script>
</head>
<body onload="draw();">
<canvas id="canvas" width="150" height="150"></canvas>
</body>
</html>
```

III. CONCLUSIONES

- El elemento HTML Canvas se usa para dibujar gráficos a través de JavaScript.
- El elemento Canvas es solo un contenedor para gráficos. Se debe usar JavaScript para dibujar los gráficos.
- Los paneles tienen varios métodos para dibujar rutas, cuadros, círculos, texto, y añadiendo imágenes.

REFERENCIAS

- M. Alvarez. (2009, Nov.). Introducción a Canvas del HTML. [Online]. Available: <https://desarrolloweb.com/articulos/introduccion-canvas-html5.html>
- ELEMENTO CANVAS. [Online]. Available: <http://www.htmlquick.com/es/reference/tags/canvas.html>
- J. Kyrnin. (2019, May.). HTML5 Canvas Uses. [Online]. Available: <https://www.lifewire.com/why-use-html5-canvas-3467995>

HTML5 Canvas. [Online]. Available:
https://www.w3schools.com/html/html5_canvas.asp

HTML | Canvas Basics. [Online]. Available:
<https://www.geeksforgeeks.org/html-canvas-basics/>