

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Seminario de Sistemas 2
Sección N



Investigación D3JS

Edward Danilo Gómez Hernández
201602909

Introducción

Este es una breve investigación cuyo fin es brindarle al lector el conocimiento básico mediante el cual pueda entender ¿Qué es D3JS?, ¿Por qué es utilizado?, Ejemplos de Graficas generadas haciendo uso del framework, así como también servir de una breve guiar de como puede utilizarlo y aprovechar las ventajas que este ofrece en algún proyecto Educativo Personal e incluso en un ambiente profesional.



¿Qué es D3js?

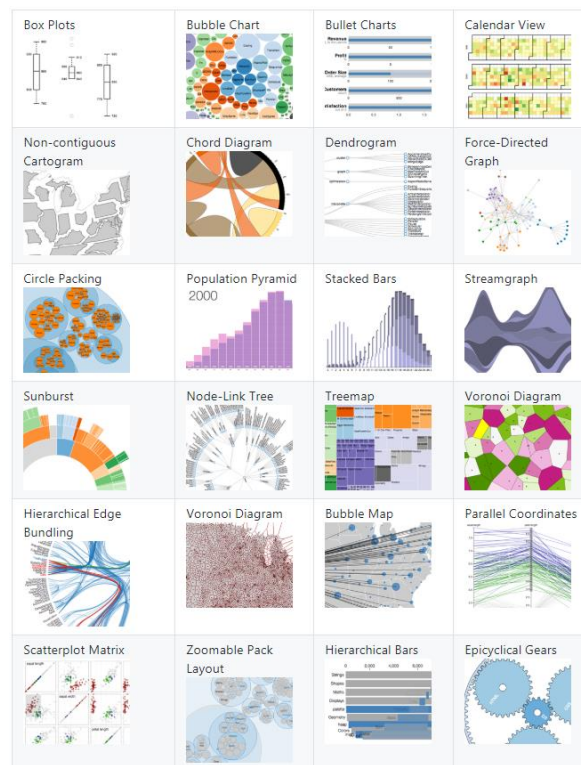
D3 representa **Data-Driven Document** (Documentos Orientados a Datos). Este es una librería de JavaScript la cual permite manipular documentos con base en datos.

Este framework se especializa en la visualización de datos mediante una gran variedad de gráficos los cuales son bastante dinámicas las cuales cambian con las interacciones del usuario mostrando información más específica.

Para la generación de estas graficas se pueden utilizar varios lenguajes entre los cuales se pueden destacar HTML, SVG, CSS. La utilización de estos resulta de gran utilidad ya que permite que se puedan visualizar sin problemas desde cualquier navegador.

Un previo conocimiento de estos lenguajes resultara de gran ayuda para los usuarios que quieran generar distintas graficas. Ya que les permitirá personalizarlos a un mayor nivel. Otra ventaja acerca de la utilización de estos lenguajes en el framework es la inmensa comunidad que existe en ellos con lo cual se pueden encontrar una gran cantidad de tutoriales, ejemplos, foros. En los cuales es posible apoyarse y aprender la utilización.

Una de las razones por la cual es bastante utilizada este framework es debido a la gran cantidad de graficas que se pueden generar haciendo uso del lo cual es de bastante utilidad para que cualquier individuo encuentre alguna que necesite, entre las principales podemos encontrar las siguientes.



¿Para qué se utiliza 3Djs?

Debido a su gran variedad de graficas disponibles es bastante fácil poder aprovechar el framework para casi cualquier sitio en el que se requiera mostrar a los usuarios información mediante graficas.

Debido a esto el framework puede llegar a utilizarse en una gran variedad de situaciones desde usos personales o educativos para la visualización de forma atractiva de cierta información. Hasta en ambientes mas profesionales como pueden ser estudios de información con Business Intelligence o incluso con volúmenes grandes de información como puede ser con Big Data.

¿Como se usa?

Para iniciar con la utilización de 3Djs primero es necesario realizar la instalación de la librería en el ambiente en el cual se va a utilizar esto lo podemos realizar mediante la utilización del siguiente comando:

```
npm install d3
```

Selección

Esta es una función bastante útil y que es bastante empleada al momento de genera graficas ya que permite la selección de uno o múltiples nodos según lo necesite el usuario para realizar esto puede hacer de la función de la siguiente manera.

```
// Selección de un nodo
var a = d3.select("a");
console.log(a);

// Selección de todos
var lis = d3.selectAll("li");

// Selección anidada
var as = d3.selectAll("li")
    .select("a");
```

Ejes

Esta función permite dar uso a los distintos ejes los cuales son **d3.axisTop([scale])**, **d3.axisRight([scale])**, **d3.axisBottom([scale])** y **d3.axisLeft([scale])**. Lo cual permite alinearlos en escena y mostrarlos. Incluso generar escalas como las del siguiente ejemplo:

```

<!-- Creamos una figura SVG con dimensiones -->
<svg width="100%" height="40">
  <g class="eje"> <!-- Dentro ubicamos un grupo -->
</svg>

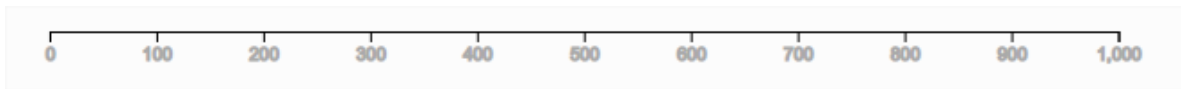
<style>
.eje {
  fill: none;
  stroke: #aaa;
}
</style>

<script>
// Creamos una escala
var scale = d3.scaleLinear()
  .domain([0, 1000])
  .range([0, 600]);

// Creamos un axis pasándole la escala
var axis = d3.axisBottom(scale)

// Seleccionamos el grupo dentro del svg
d3.select('.eje') // Lo movemos a la derecha
  .attr("transform", "translate(40, 0)")
  .call(axis); // Llamamos al eje para insertarlo
</script>

```



Formas

Estas son múltiples funciones las cuales tal y como su nombre lo indica permite agregar al diseño distintas formas las utilizar para generar una forma mas compleja. Entre las cuales se podrían mencionar **d3.arc()**, **d3.area()**, **d3.linkVertical()** y **d3.linkHorizontal()**. Y pueden emplearse tal y como se muestra en el siguiente ejemplo:

```

<div id="arc-1"></div>
<script>
var arco_d = d3.arc()
  .innerRadius(0)
  .outerRadius(100)
  .startAngle(0)
  .endAngle(Math.PI);

arco_d(); // M6.123233995736766e-15,-100A100,100,0,1,1,6.123233995736766e-15,100L0,0Z

d3.select("#arc-1")
  .append("svg")
  .attr('width', 250)
  .attr('height', 250)
  .append("g")
  .attr("transform", "translate(125,125)")
  .append("path")
  .attr("d", arco_d)
  .attr("fill", "red");
</script>

```








Para el caso de graficas existen otras funciones propias de 3Djs las cuales junto a algunas de las funciones explicadas con anterioridad se pueden utilizar para generar la grafica deseadas.

Ya que el procedimiento cambiara dependiendo del tipo de grafica que se necesite.

Ejemplo Implementación 3Djs

Para el caso de este ejemplo se utilizará además de 3Djs, el framework de Angular para ejecutar la aplicación.

Para la elaboración del ejemplo se utilizará el siguiente dataset:

Basic Cars Characteristics						
Data Card Code (1) Discussion (0)						
wagon, minivan, pickup ...), type of traction system (RWD or 4x4, if not is FWD), msrp, dealer_cost, horse power, ...						
A name	✓ sports_car	✓ suv	✓ wagon	✓ minivan	✓ pickup	
Vehicle name and model	Is a sport car?	Is a SUV?	Is a wagon?	Is a minivan?	Is a pickup	
425 unique values	 true 49 11% false 379 89%	 true 60 14% false 368 86%	 true 30 7% false 398 93%	 true 20 5% false 408 95%	 true 0 0% false 425 100%	
Chevrolet Aveo 4dr	FALSE	FALSE	FALSE	FALSE	FALSE	
Chevrolet Aveo LS 4dr hatch	FALSE	FALSE	FALSE	FALSE	FALSE	
Chevrolet Cavalier 2dr	FALSE	FALSE	FALSE	FALSE	FALSE	
Chevrolet Cavalier 4dr	FALSE	FALSE	FALSE	FALSE	FALSE	
Chevrolet Cavalier LS 2dr	FALSE	FALSE	FALSE	FALSE	FALSE	
Dodge Neon SE 4dr	FALSE	FALSE	FALSE	FALSE	FALSE	
Dodge Neon SXT 4dr	FALSE	FALSE	FALSE	FALSE	FALSE	
Ford Focus ZX3 2dr hatch	FALSE	FALSE	FALSE	FALSE	FALSE	
Ford Focus LX 4dr	FALSE	FALSE	FALSE	FALSE	FALSE	
						Summary
						1 file
						19 columns

La carga de los datos en la variable **data** se realizo de la siguiente forma:

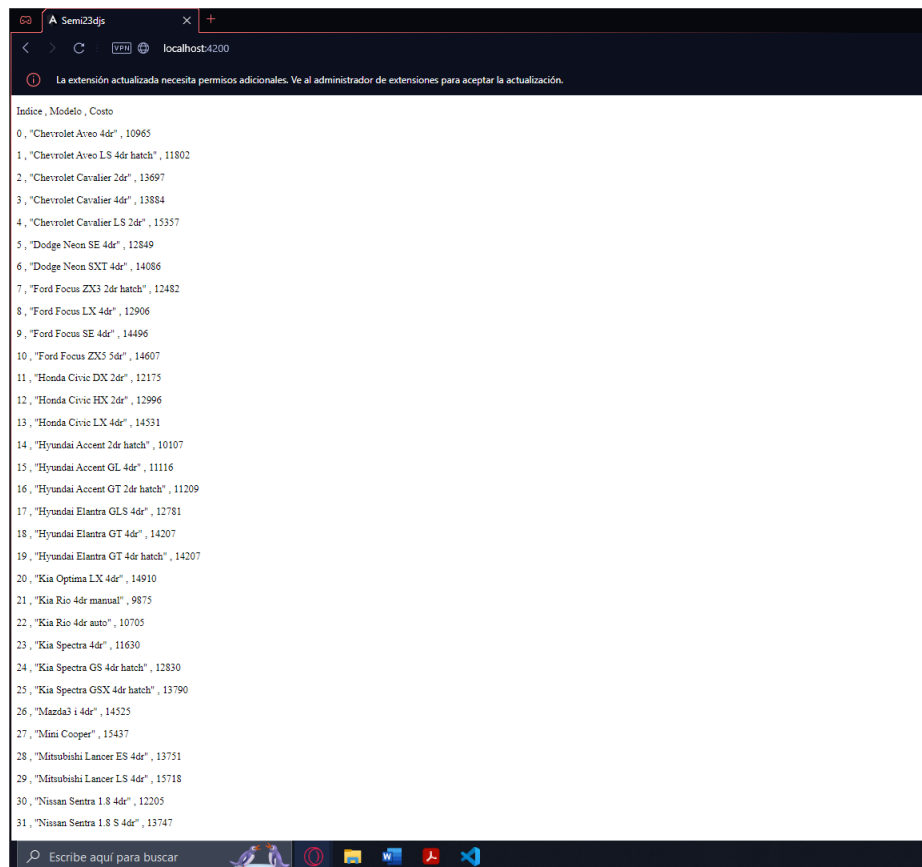
```
export class AppComponent {
  title = 'Grafica Barras con 3Djs y Angular';
  data: any[] = [];

  constructor(private http: HttpClient) {
    this.http.get('assets/Cars.csv', { responseType: 'text' }).subscribe(
      data => {
        let csvToArray = data.split("\n");
        for (let index = 1; index < csvToArray.length; index++) {
          let row = csvToArray[index].split(",");
          this.data.push({
            name: row[0],
            sports_car: row[1],
            suv: row[2],
            wagon: row[3],
            minivan: row[4],
            pickup: row[5],
            all_wheel: row[6],
            rear_wheel: row[7],
            msrp: row[8],
            dealer_cost: row[9],
            eng_size: row[10],
            ncity: row[11],
            horsepower: row[12],
            city_mpg: row[13],
            hwy_mpg: row[14],
            weight: row[15],
            wheel_base: row[16],
            length: row[17],
            width: row[18]
          });
        }
        console.log(this.data);
      },
      error => {
        console.log(error);
      }
    );
  }
}
```

Para mostrar los datos cargados en la variable **data** se verifico de la siguiente forma:

```
TS app.component.ts M app.component.html M X
src > app > app.component.html > ...
1 <p>Indice , Modelo , Costo</p>
2 <div *ngFor="let car of data; let i=index">
3   <p>{{i}} , {{car.name}} , {{car.msrp}}</p>
4 </div>
5 <router-outlet></router-outlet>
6
```

Lo cual dio como resultado lo siguiente:



Una vez cargado los datos, se procede a utilizar 3Djs para utilizar la información dentro del proyecto.

Para la realización de la grafica de barras se empleó el siguiente código:

Para los imports se utilizo el del 3Djs y HttpClient para cargar la información del Dataset.

```
import { Component } from '@angular/core';
import * as d3 from 'd3';
import { HttpClient } from '@angular/common/http';
```

Se crea el SVG que contendrá la grafica de barras con los datos del Dataset.

```
data: any[] = [];  
private svg:any;  
private margin = 50;  
private width = 1000 - (this.margin * 2);  
private height = 1000 - (this.margin * 2);  
  
> constructor(private http: HttpClient) { ...  
}  
  
private createSvg(): void {  
  this.svg = d3.select("figure#bar")  
    .append("svg")  
    .attr("width", this.width + (this.margin * 2))  
    .attr("height", this.height + (this.margin * 2))  
    .append("g")  
    .attr("transform", "translate(" + this.margin + "," + this.margin + ")");  
}
```

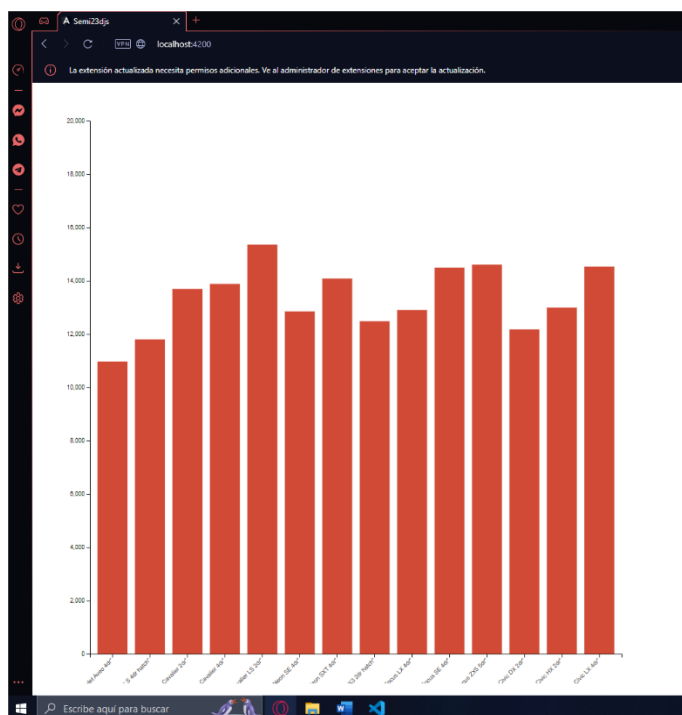
Se define función que genera la gráfica, en esta se configuran los límites de las escalas, así como también los atributos del dataset que se emplearan para crear la gráfica.

```
private drawBars(data: any[]): void {  
  const x = d3.scaleBand()  
    .range([0, this.width])  
    .domain(data.map(d => d.name))  
    .padding(0.2);  
  
  this.svg.append("g")  
    .attr("transform", "translate(0," + this.height + ")")  
    .call(d3.axisBottom(x))  
    .selectAll("text")  
    .attr("transform", "translate(-10,0)rotate(-45)")  
    .style("text-anchor", "end");  
  
  const y = d3.scaleLinear()  
    .domain([0, 20000])  
    .range([this.height, 0]);  
  
  this.svg.append("g")  
    .call(d3.axisLeft(y));  
  
  this.svg.selectAll("bars")  
    .data(data)  
    .enter()  
    .append("rect")  
    .attr("x", (d: any) => x(d.name))  
    .attr("y", (d: any) => y(d.msrp))  
    .attr("width", x.bandwidth())  
    .attr("height", (d: any) => this.height - y(d.msrp))  
    .attr("fill", "#d04a35");  
}  
}
```

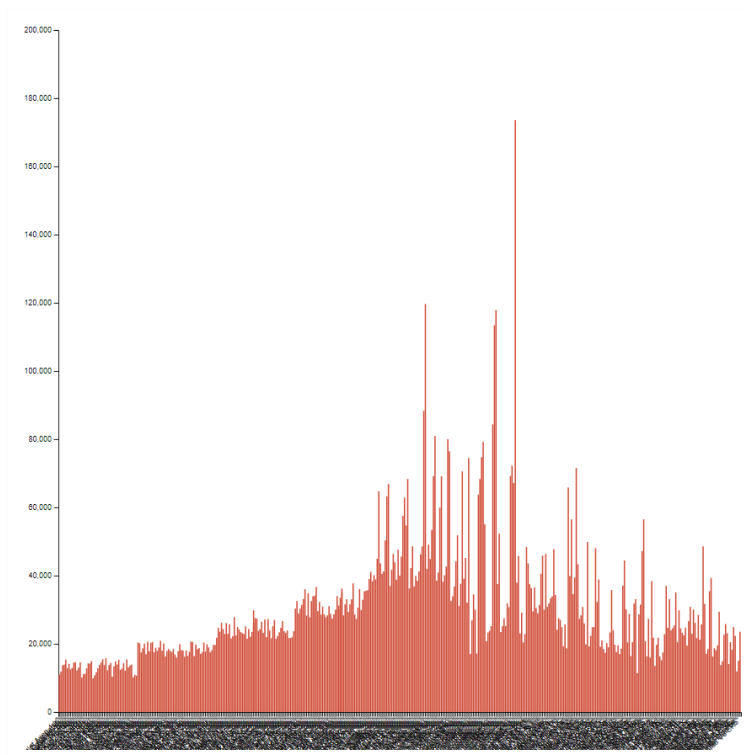

Por último, se hace uso de las funciones creadas con anterioridad y se pasa como parámetro para la grafica la variable con los datos cargados desde el dataset.

```
constructor(private http: HttpClient) {
  this.http.get('assets/Cars.csv', {responseType: 'text'}).subscribe(
    data => {
      let csvToArray = data.split("\n");
      for (let index = 1; index < csvToArray.length; index++) {
        let row = csvToArray[index].split(",");
        this.data.push({
          name: row[0],
          sports_car: row[1],
          suv: row[2],
          wagon: row[3],
          minivan: row[4],
          pickup: row[5],
          all_wheel: row[6],
          rear_wheel: row[7],
          msrp: row[9],
          dealer_cost: row[10],
          eng_size: row[11],
          ncyl: row[12],
          horsepower: row[13],
          city_mpg: row[14],
          hwy_mpg: row[15],
          weight: row[16],
          wheel_base: row[17],
          length: row[18],
          width: row[19]
        });
      }
      const svg = this.createSvg();
      this.drawBars(this.data);
    },
    error => {
      console.log(error);
    }
  );
}
```

Resultados de Grafica de Barras con Aplicación Compilando:



Grafica de Barras con todos los datos del Dataset:



Conclusiones

- Se espera que mediante la lectura de esta breve investigación se cuente con una perspectiva más clara acerca de que es 3Djs y lo importante que puede llegar a resultar para mejorar la visualización de datos de una manera grafica.
- También es importante resaltar que para alguien que no posee muchos conocimientos en lenguajes tales como JavaScript, HTML, CSS, SVG puede resultar bastante complejo adentrarse y hacer uso de esta librería.
- Tal y como se pudo apreciar en el ejemplo brindado, una gran ventaja de 3Djs es que puede integrarse con otros frameworks para el desarrollo web con los cuales los usuarios también pueden respaldarse y crear aplicaciones más complejas.

Bibliografías

- <https://d3js.org>
- <https://www.bigdata.uma.es/visualizacion-de-datos-con-d3-js/>
- <https://d3-spanish.readthedocs.io/es/latest/basico/introduccion.html>