

Instituto Federal Goiano - Campus Ceres
Bacharelado em Sistemas de Informação
Prof. Me. Ronneesley Moura Teles

Adallberto Lucena Moura
Andrey Silva Ribeiro
Anny Karoliny Moraes Ribeiro
Daniel Moreira Cardoso
Davi Ildeu de Faria

Relatório Chart.js

Outubro
2017

Sumário

1	Introdução	2
2	Tipos de gráficos	2
3	Exemplos de gráficos implementados.	4
3.1	Gráfico de linhas	4
3.2	Gráfico Polar	5
3.3	Gráfico Radar	5
4	Vantagens	6
5	Conclusão	7

Relatório Chart.js

1 Introdução

Chart.js é uma biblioteca desenvolvida por Nick Dowine sob a linguagem JavaScript, renderiza os gráficos utilizando o elemento canvas do HTML5. O Chart.js framework utiliza uma biblioteca para construção de maneira mais performática e fácil, ela é open source e disponível sob licença MIT no Github <https://github.com/chartjs/Chart.js>. Por ser open source, qualquer pessoa pode contribuir com a comunidade e manter sempre ativo, atualizado com novas features.

O Chart.js possui 8 tipos diferentes de gráficos, entre os tradicionais como barra e pizza. Há um suporte responsivo, possui legenda para os gráficos e opções de gráficos interativos e modulares.

2 Tipos de gráficos

Alguns exemplos de gráficos que podem ser gerados:

Red Orange Yellow Green Blue



Figura 1: Pie Chart / Gráfico de Torta. Fonte: <http://www.chartjs.org/samples/latest/charts/pie.html>; Acesso em 20/10/2017.

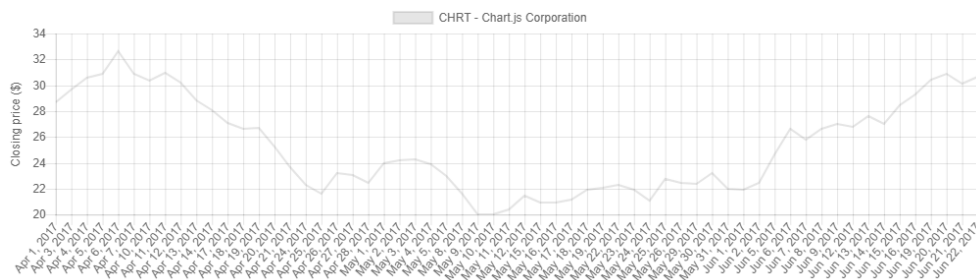


Figura 2: Line Chart / Gráfico de linha. Fonte: <http://www.chartjs.org/samples/latest/scales/time/financial.html>; Acesso em 20/10/2017.

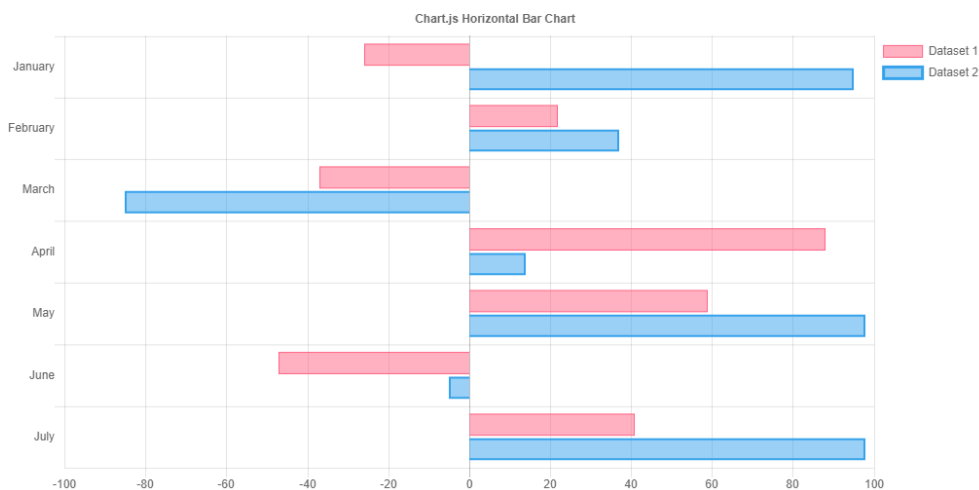


Figura 3: Horizontal Bar Chart / Gráfico de barra horizontal. Fonte: <http://www.chartjs.org/samples/latest/charts/bar/horizontal.html>; Acesso em 20/10/2017.

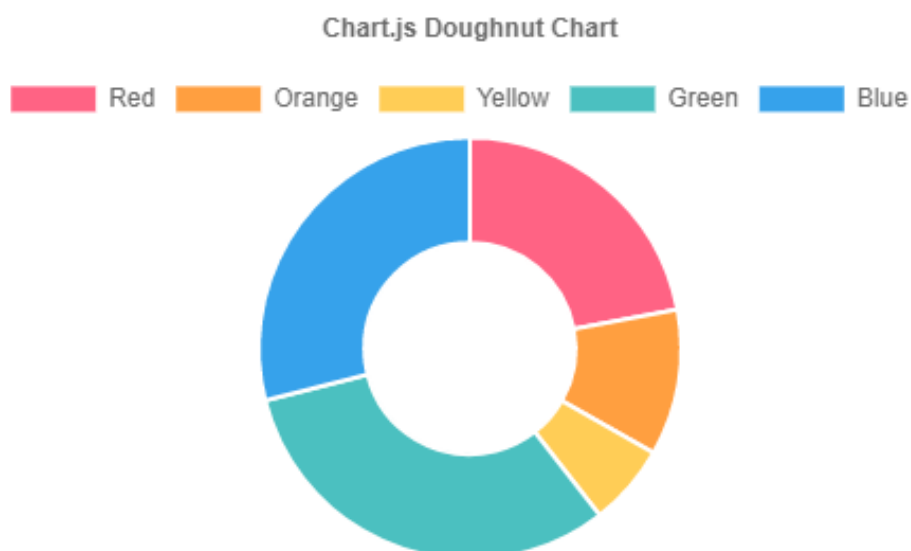


Figura 4: Doughnut Chart / Gráfico de Rosquinha. Fonte: <http://www.chartjs.org/samples/latest/charts/doughnut.html>; Acesso em 20/10/2017.

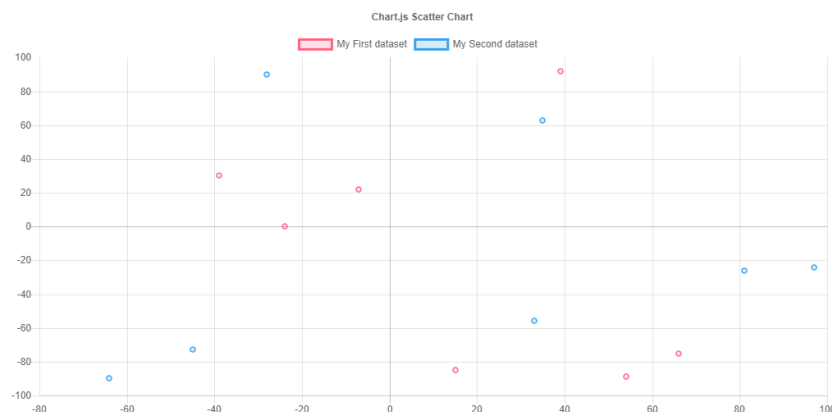


Figura 5: Scatter Chart / Gráfico de Dispersão. Fonte: <http://www.chartjs.org/samples/latest/charts/scatter/basic.html>; Acesso em 20/10/2017.

3 Exemplos de gráficos implementados.

3.1 Gráfico de linhas

```

1 <html>
2 <head>
3   <meta charset="UTF-8">
4   <meta name="viewport" content="width=device-width, initial-scale=1.0">
5   <title>Chart Js - Linhas</title>
6 </head>
7 <body>
8   <!-- Inicia Canvas -> desenhar elementos graficos usando JS -->
9   <canvas class="line-chart"></canvas>
10
11  <!-- Include Chart Js -->
12  <script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.0/Chart.min.js"></script>
13
14  <script>
15    var ctx = document.getElementsByClassName("line-chart");
16
17    //type, data e option
18    var chartGraph = new Chart(ctx, {
19      type: 'line',
20      data: {
21        labels: ["Jan", "Fev", "Mar", "Abr", "Mai", "Jun", "Jul", "Ago", "Set", "Out", "Nov", "Dez"],
22        datasets: [{
23          label: "Taxa de cliques - 2017",
24          data: [5,10,5,14,20,15,6,14,8,12,15,5,10],
25          borderWidth: 5,
26          borderColor: 'rgba(77,166,253,0.85)',
27          backgroundColor: 'transparent',
28        },
29        {
30          label: "Taxa de clique em propagandas - 2017",
31          data: [9,14,11,6,5,3,8,4,7,13,8,8,19],
32          borderWidth: 5,
33          borderColor: 'rgba(6,204,6,0.85)',
34          backgroundColor: 'transparent',

```

```

35         },
36     ]
37 }
38 });
39 </script>
41
42 </body>
43 </html>

```

recursos/linhas.html

3.2 Gráfico Polar

```

1 <html>
2 <head>
3   <meta charset="UTF-8">
4   <meta name="viewport" content="width=device-width, initial-scale=1.0">
5   <title>Chart Js - Polar</title>
6 </head>
7 <body>
8   <!-- Inicia Canvas -> desenhar elementos graficos usando JS -->
9   <canvas class="polar-chart" width="600" height="200"></canvas>
10
11   <!-- Include Chart Js -->
12   <script src="https://cdn.jsdelivr.net/npm/chart.js@2.7.0/dist/chart.min.js"></script>
13
14   <script>
15     var ctx = document.getElementsByClassName("polar-chart")[0].getContext("2d");
16
17     //type, data e option
18     var myRadarChart = new Chart(ctx, {
19       type: 'polarArea',
20       data: {
21         labels: ['Vermelho', 'Verde', 'Amarelo', 'Cinza', 'Azul'],
22         datasets: [{
23           label: "Taxa de cliques - 2017",
24           data: [20, 10, 40, 60, 70],
25           backgroundColor: ["rgb(255, 99, 132)", "rgb(75, 192, 192)", "rgb(255, 205, 86)", "rgb(201, 203, 207)", "rgb(54, 162, 235)"],
26           borderWidth: 0,
27           borderColor: 'rgba(22,143,122,0.85)',
28         },
29       ],
30     });
31   </script>
32
33 </body>
34 </html>

```

recursos/polar.html

3.3 Gráfico Radar

```

1 <html>
2 <head>
3   <meta charset="UTF-8">
4   <meta name="viewport" content="width=device-width, initial-scale=1.0"
5     ">
6   <title>Chart Js - Radar</title>
7 </head>
8 <body>
9   <!-- Inicia Canvas -> desenhar elementos graficos usando JS -->
10  <canvas class="radar-chart"></canvas>
11
12  <!-- Include Chart Js -->
13  <script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.0/
14    Chart.min.js"></script>
15
16  <script>
17    var ctx = document.getElementsByClassName("radar-chart");
18
19    //type, data e option
20    var myRadarChart = new Chart(ctx, {
21      type: 'radar',
22      data: {
23        labels: ['Facebook', 'Twitter', 'YouTube', 'GitHub', 'Vimeo', '
24          Orkut'],
25        datasets: [{
26          label: "Taxa de cliques - 2017",
27          data: [20, 10, 4, 2, 9, -4],
28          borderWidth: 2,
29          borderColor: 'rgba(77,166,253,0.85)',
30        },
31
32        {
33          label: "Taxa de clique em propagandas - 2017",
34          data: [35, 7, 1, 7, -10, -6],
35          borderWidth: 2,
36          borderColor: 'rgba(11,56,193,0.99)',
37        },
38      ]
39    });
40  </script>
41
42 </body>
43 </html>

```

recursos/radar.html

4 Vantagens

- O código é open source e está disponível na página do Github;
- Tem continuidade;
- Tem compatibilidade com outros tipos de navegadores;

- A biblioteca possui uma documentação oficial dentro da página do projeto;
- Existe uma comunidade ativa no Github;
- Possui exemplos de código na página do projeto.

5 Conclusão

Para aqueles que desejam adicionar gráficos na web, sendo para pequenos projetos ou em um dashboard de uma página de adm back end, por exemplo, o Chart.js é uma perfeita escolha, já que os gráficos são todos responsivos e personalizáveis, podendo assim se adaptar a qualquer tamanho de tela, ter suas cores mudadas entre outras coisas que são todas passíveis de personalização.

Outro ponto bastante interessante é que o Chart.js é extremamente leve se comparado com outras bibliotecas e não possui dependências.

Também podemos combinar o código da biblioteca com PHP, Java e outras linguagens, para incluir os dados a partir de um banco de dados e automatizar a atualização dos gráficos.