

ChartBool

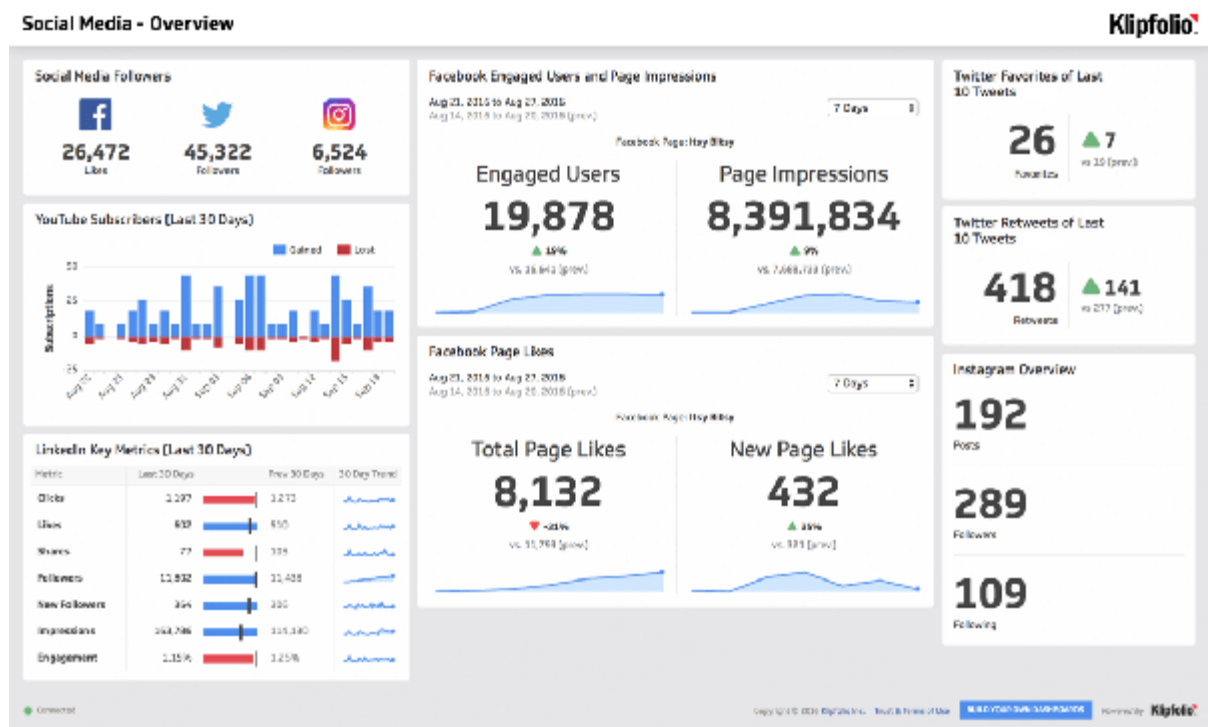
(repo: rest-chartbool)

In questo esercizio vogliamo replicare una richiesta comune in varie aziende: la creazione di una dashboard (o cruscotto) che riassume in una unica schermata tutti gli indicatori essenziali per valutare l'andamento dell'azienda (altrimenti detti KPI, *Key Performance Indicator*).

Gli indicatori sono spesso rappresentati con grafici (a linea, a barre, a torta, etc) per evidenziarne l'andamento temporale e permettere una lettura immediata.

In questo esercizio, utilizziamo la libreria ChartJs (<http://www.chartjs.org/>) che ci permette di creare semplicemente grafici per la Dashboard aziendale mentre i dati verranno presi da una API.

Esempio di dashboard:



Milestone 1:

Facendo una chiamata GET all'endpoint /sales, l'API ci ritornerà la lista di tutte le vendite fatte dai venditori dell'azienda:

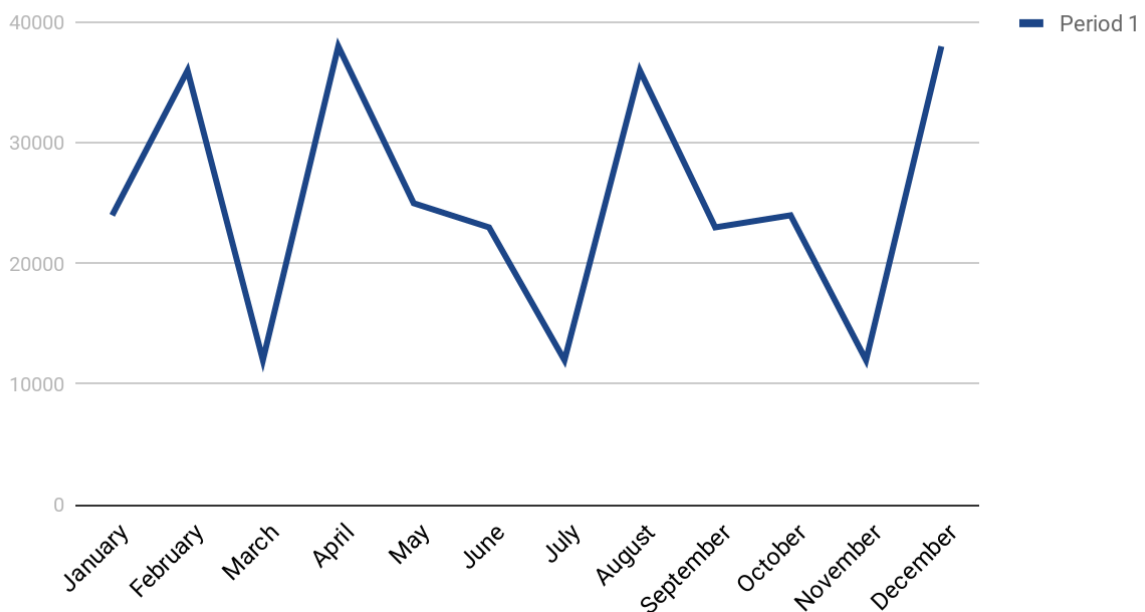
```
{
  "sales":
  [
    {
      "id": 1,
      "salesman" : "Marco",
      "amount": 9000,
      "date": "12/02/2017"
    }
  ]
}
```

```
},
{
  "id": 2,
  "salesman" : "Giuseppe",
  "amount": 1000,
  "date": "12/04/2017"
}
]
```

Da questi dati dobbiamo creare due grafici:

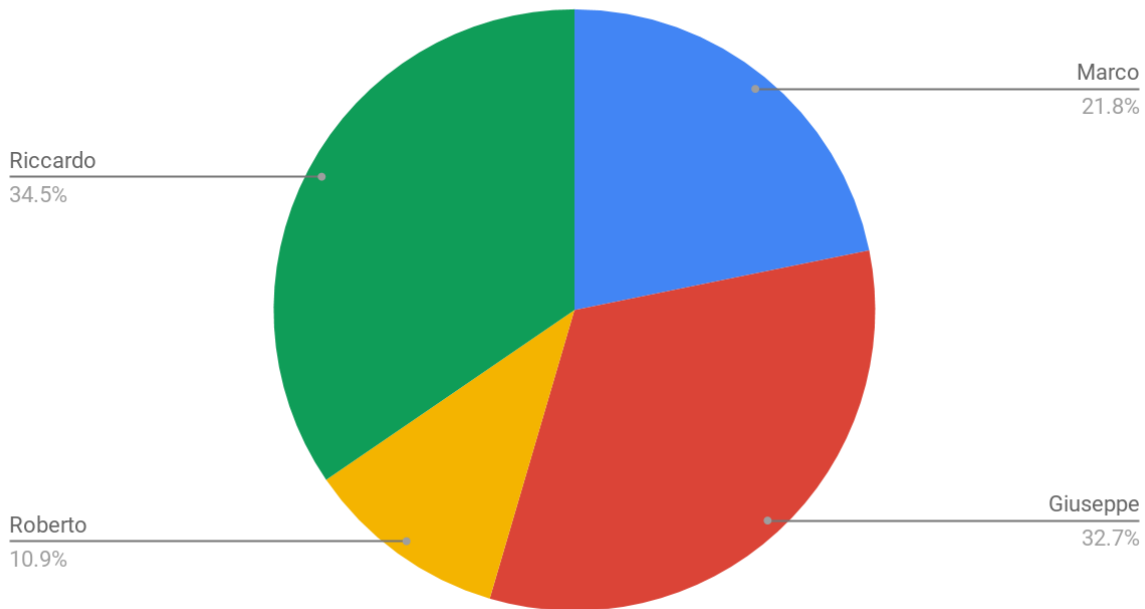
1. Andamento delle vendite totali della nostra azienda con un grafico di tipo Line (<http://www.chartjs.org/docs/latest/charts/line.html>) con un unico dataset che conterrà il numero di vendite totali mese per mese nel 2017.

Fatturato mensile



2. Il secondo grafico è quello a torta (<http://www.chartjs.org/docs/latest/charts/doughnut.html>) che evidenzierà il contributo di ogni venditore per l'anno 2017. Il valore dovrà essere la percentuale di vendite effettuate da quel venditore ($\text{fatturato_del venditore} / \text{fatturato_totale}$)

Vendite per venditore



Milestone 2:

Ora vogliamo dare la possibilità di aggiungere vendite.

Creiamo due select, una contenente i nostri venditori e l'altra contenente il mese dell'anno e una input per inserire il valore della vendita.

Aggiungiamo un bottone e onClick dovremo aggiungere una vendita del valore inserito al venditore selezionato per il mese selezionato, facendo una chiamata POST a /sales.

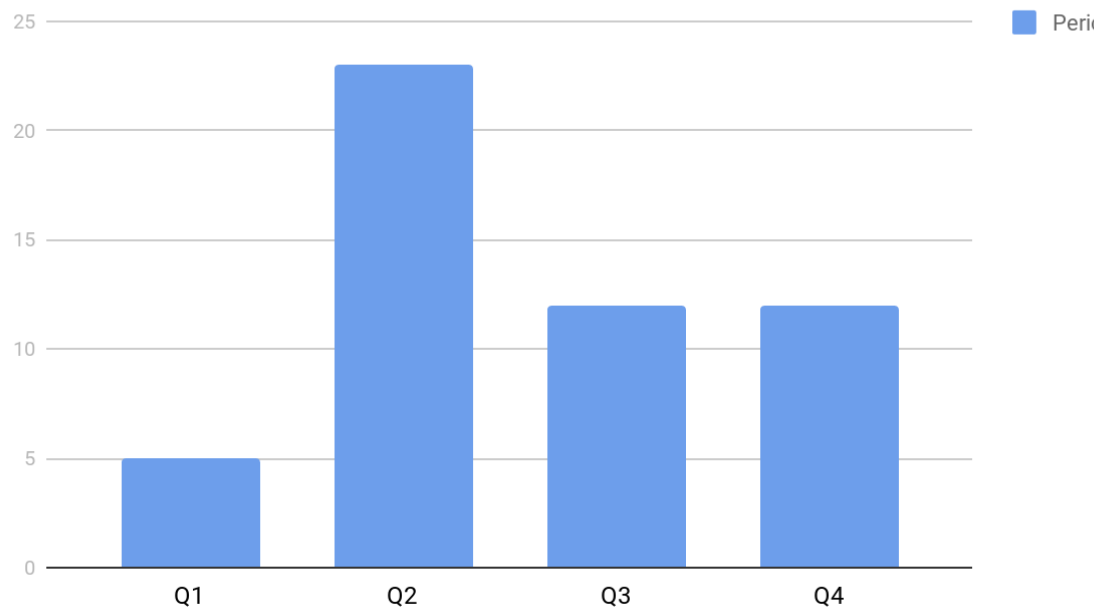
I grafici andranno modificati mostrando i nuovi dati.

Milestone 3 (Opzionale):

La terza Milestone, opzionale, ci aiuta a capire meglio l'andamento della nostra azienda visualizzando l'andamento durante i quarter. Creeremo un grafico a barre (<http://www.chartjs.org/docs/latest/charts/bar.html>) con il numero di vendite generate durante i nostri 4 quarter, sapendo che il primo andrà da Gennaio a Marzo compresi, il secondo da Aprile a Giugno compresi ecc.

Il grafico quindi dovrà essere così:

Vendite per Quarter



Consigli:

- Installazione di ChartJS: come jQuery o Moment basterà scaricare da una CDN o dal sito ufficiale il file js e importarlo nel nostro html.
- I grafici sono stati messi per facilitare la comprensione della consegna, non sono un esempio di come devono apparire esteticamente che, in questo esercizio, non ci interessa
- Visto che ci sono varie date, potrebbe essere utile importare anche moment. Potrebbe, non è obbligatorio.

Esempio base con grafico Line:

```
var config = {
  type: 'line',
  data: {
    labels: ['Gennaio', 'Febbraio', 'Marzo', 'Aprile'],
    datasets: [
      {
        label: 'Vendite',
        data: [1, 2, 3, 4],
        borderColor: 'red',
        backgroundColor: "green"
      }
    ]
  }
};

var myLineChart = new Chart($('#chart'), config);
```

Per poter aggiornare quando si vuole il nostro grafico, prima si cambia il valore di data, e poi si chiama update();

```
window.myLineChart.update();
```