

# Progetto Municipio 3

Dicembre 2017

Divisione del lavoro in gruppi, stilata nella giornata del 14/12/2017

- [Gruppo Api Google Maps](#)
  - [Studying](#)
  - [Coding](#)
- [Gruppo Api Google Charts](#)
  - [Studying](#)
  - [Coding](#)
- [Gruppo Web Design](#)
  - [Studying](#)
  - [Coding](#)

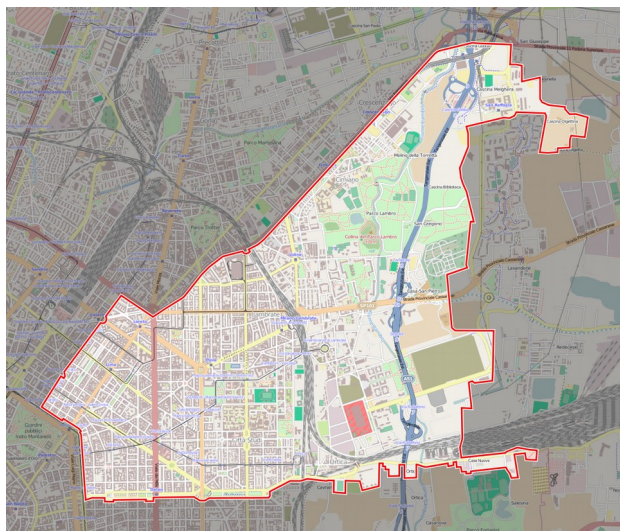
## IMPORTANTE:

Lavorate su una branch di GitHub diversa dalla master, e fate una pull request quando il vostro codice è funzionante, testato e commentato estensivamente.

Per lavorare su GitHub, scaricate [GitHub Desktop](#)

## Schema di [progetto](#):

```
Home>
  index>
  About>
  Dati>
    Pagina1>
    Pagina2>
    ...
    PaginaN>
      Map>
      Charts>
```



# Gruppo API Google Maps

## Studying:

Per correttamente comprendere le meccaniche di interazione tra le fusion table e la API di Google Maps per JavaScript, è necessario studiare la documentazione di entrambe le tecnologie.

[Documentazione JavaScript API Google Maps](#)

[Documentazione Fusion Table](#)

In particolare, è necessario sapere bene la parte di interazione tra le due tecnologie, [Google Maps API > Fusion Table](#).

In caso di dubbi sul linguaggio JavaScript, consiglio il corso su [CodeAcademy](#) o su [Sololearn](#), completabile interamente in 10 ore, oppure in meno se avete dubbi su un argomento in particolare.

## Coding:

È necessario creare un sistema di visualizzazione su mappa di diversi dati. Inclusi sulla repository GitHub del progetto, sotto la sezione dati, vi sono diversi link contenenti le Fusion Tables con:

- Le sezioni di censimento
- Dei dati generati randomicamente per test
- Le due tabelle precedenti fuse con il valore SEZ2011

Le tabelle includono tutto il comune di Milano, ma sarà poi semplice scalarlo soltanto alla zona 3.

I sistemi di visualizzazione da creare sono i seguenti:

- [Market Clustering](#)
- [Polygon based visualization and Styling](#)
- [Heatmap visualization](#)

Questi sistemi vanno creati all'interno di una div HTML chiamata "map", per facilitare il lavoro dei Web Designer.

Il codice JavaScript deve essere scritto in un file esterno, ed incluso da HTML, il tutto deve essere estensivamente documentato.

# Gruppo API Google Charts

## Studying:

Per realizzare grafici e tabelle con le API di Google Charts, sarà necessario studiare ed imparare ad utilizzare le API stesse e anche quelle delle Fusion Tables.

[Google Charts API](#)

[Fusion Tables API](#)

Bisogna quindi acquisire le competenze necessarie ad interagire con entrambe le API, in particolare sono disponibili ai seguenti link, per riferimento e studio.

[Fusion Table and Charts interaction](#)

[Codice esempio di tabelle e grafici](#)

In caso di dubbi sul linguaggio JavaScript, consiglio il corso su [CodeAcademy](#) o su [Sololearn](#), completabile interamente in 10 ore, oppure in meno se avete dubbi su un argomento in particolare.

## Coding:

È necessario realizzare un sistema in grado di visualizzare tabelle e grafici, a discrezione del gruppo. L'importante è rendere la visualizzazione migliore per ciascun tipo di dato che ci verrà fornito. Inclusi sulla repository GitHub del progetto, sotto la sezione dati, vi sono diversi link contenenti le Fusion Tables con:

- Le sezioni di censimento
- Dei dati generati randomicamente per test
- Le due tabelle precedenti fuse con il valore SEZ2011

Le tabelle includono tutto il comune di Milano, ma sarà poi semplice scolarlo soltanto alla zona 3.

Il codice JavaScript (che deve essere scritto in un file esterno) e la div della tabella, devono essere estensivamente documentati per facilitare il lavoro dei Web Designer.

# Gruppo Web Designer

## Studying:

Sarà necessario acquisire elevata confidenza con i linguaggi HTML, CSS e JavaScript. Per migliorare le proprie conoscenze su queste tecnologie, consiglio i seguenti corsi.

[CodeAcademy learn JavaScript](#)

[Sololearn HTML corso](#)

[Sololearn CSS corso](#)

[Sololearn JavaScript corso](#)

Sarà anche necessario andare a dare un'occhiata alla documentazione studiata dagli altri gruppi, o richiedere del codice campione per capire meglio come integrare le tecnologie sviluppate.

## Coding:

Sarà necessario creare un sito contenente diverse pagine: una per tipo di dato, una home, ed una pagina "about" con informazioni sul progetto.

Ogni pagina deve seguire uno stile coerente, a discrezione del gruppo.

Importante è anche integrare i file Javascript e le "div" sviluppate dagli altri gruppi all'interno delle pagine.

Va seguito lo schema di cartelle del progetto, definito nella prima pagina di questo documento.