

## Mãos na massa: Aplicação de busca de aeroportos

Chegou a hora de implementarmos o que está nos vídeos.

---

1) Faça o download do projeto [aqui \(https://caelum-online-public.s3.amazonaws.com/712-dotnet-mongodb-parte2/05/aspnetmvc-gmaps-inicial-para-importar.zip\)](https://caelum-online-public.s3.amazonaws.com/712-dotnet-mongodb-parte2/05/aspnetmvc-gmaps-inicial-para-importar.zip), extraia-o na pasta dos seus projetos e abra a solução **Alura.GoogleMaps**.

2) Instale o **MongoDB.Driver** ao projeto.

3) Em **Models**, na classe **conectandoMongoDBGeo**, utilize as referências **MongoDB.Bson** e **MongoDB.Driver**:

```
using MongoDB.Bson;
using MongoDB.Driver;
```

4) Copie o conteúdo da classe **conectandoMongoDBGeo** que você fez na aula anterior, e cole no conteúdo da classe **conectandoMongoDBGeo** do projeto atual.

5) Faça a mesma coisa com a classe **Aeroporto**, copiando o conteúdo da classe de mesmo nome implementada na aula anterior, e colando na classe do projeto atual, dentro de **Models**. Copie também as seguintes referências:

```
using MongoDB.Bson;
using MongoDB.Bson.Serialization.Attributes;
using MongoDB.Driver.GeoJsonObjectModel;
```

6) Em **Controllers**, na classe **HomeController**, adicione as seguintes referências:

```
using MongoDB.Bson;
using MongoDB.Driver;
using MongoDB.Driver.GeoJsonObjectModel;
```

7) Procure pelos comentários, e faça o que é pedido, conectando ao MongoDB, configurando o ponto atual no mapa, implementando o filtro, capturando a lista e escrevendo os pontos no mapa:

```
//Conecta MongoDB
var conexaoAeroporto = new conectandoMongoDBGeo();

//Configura o ponto atual no mapa
var ponto = new GeoJson2DGeographicCoordinates(lon, lat);
var localizacao = new GeoJsonPoint<GeoJson2DGeographicCoordinates>(ponto);

// filtro

var construtor = Builders<Aeroporto>.Filter;
FilterDefinition<Aeroporto> filtro_builder;

if (tipoAero == "")
```

```
{
    filtro_builder = construtor.NearSphere(x => x.loc, localizacao, distancia);
}
else
{
    filtro_builder = construtor.NearSphere(x => x.loc, localizacao, distancia) & construtor.Eq(

};

// Captura a lista
var listaAeroportos = await conexaoAeroporto.Airports.Find(filtro_builder).ToListAsync();

// Escreve os pontos
foreach (var doc in listaAeroportos)
{
    var aero = new Coordenada(doc.name,
        Convert.ToString(doc.loc.Coordinates.Latitude).Replace(",", "."),
        Convert.ToString(doc.loc.Coordinates.Longitude).Replace(",", "."));
    aeroportosProximos.Add(aero);
}
```