Documentação ETL

A extração de dados ocorreu através de um arquivo disponível em link (ttps://s3.amazonaws.com/dev.etl.python/datasets/data_points.tar.gz) onde se encontra zipado. O arquivo continha três arquivos cujo nome era (data_points_20180101, data_points_20180102, data_points_20180103), esses arquivos textos tinha informação de coordenadas conforma imagem abaixo:

```
Latitude: 30°02'59"S   -30.04982864
Longitude: 51°12'05"W   -51.20150245
Distance: 2.2959 km  Bearing: 137.352°
Latitude: 30°04'03"S   -30.06761588
Longitude: 51°14'23"W   -51.23976111
Distance: 4.2397 km  Bearing: 210.121°
Latitude: 30°03'21"S   -30.05596474
Longitude: 51°10'22"W   -51.17286827
Distance: 4.9213 km  Bearing: 118.814°
Latitude: 30°02'18"S   -30.03841576
Longitude: 51°14'58"W   -51.24943145
Distance: 3.088 km  Bearing: 262.19°
Latitude: 30°00'22"S   -30.00613726
Longitude: 51°14'19"W   -51.23864809
Distance: 3.7605 km  Bearing: 327.479°
Latitude: 30°04'02"S   -30.06713593
```

Essas informações não eram completas as vezes onde faltava alguns campos onde foi considerado NULO como por exemplo a figura abaixo:

```
Longitude: 51°12'58"W   -51.21613898
Distance: 1.1348 km  Bearing: 7.408°

Latitude: 30°01'43"S   -30.02862333
Longitude: 51°13'07"W   -51.21870748

Latitude: 30°03'36"S   -30.0599525
Longitude: 51°10'44"W   -51.17884852
Distance: 4.678 km  Bearing: 126.999°
```

Conforme a figura faltava campo “Distance” e “Bearing”, esses campos foram considerados nulos como por exemplo um dicionário de dados conforme a figura abaixo:

```
{'Latitude': -30.02862333, 'Longitude': -51.21870748, 'Distance': None, 'Bearing': None}
```

Conforme a figura foi considerado apenas o número decimal da latitude e longitude onde se vinha que não precisava guardar essas informações em banco. Conforme numero decimal é de melhor modo de trabalhar com geolocalização

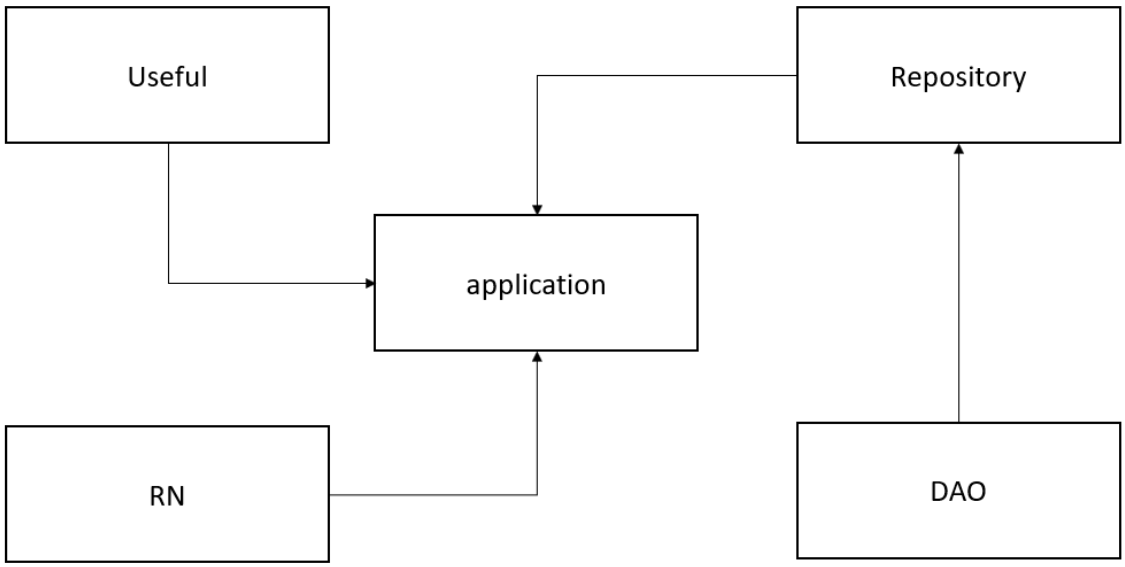
Tecnologia Usada

- GoogleMaps API - foi usado para extração das coordenadas para ter o endereçamento;
- PyCharm – Essa IDE foi usada para o controle do código onde o projeto foi criado;
- SQL Server – Foi usado para bancos de dados relacionais (RDBMS – Relational Database Management Systems)

Arquitetura

Conforme a imagem abaixo:

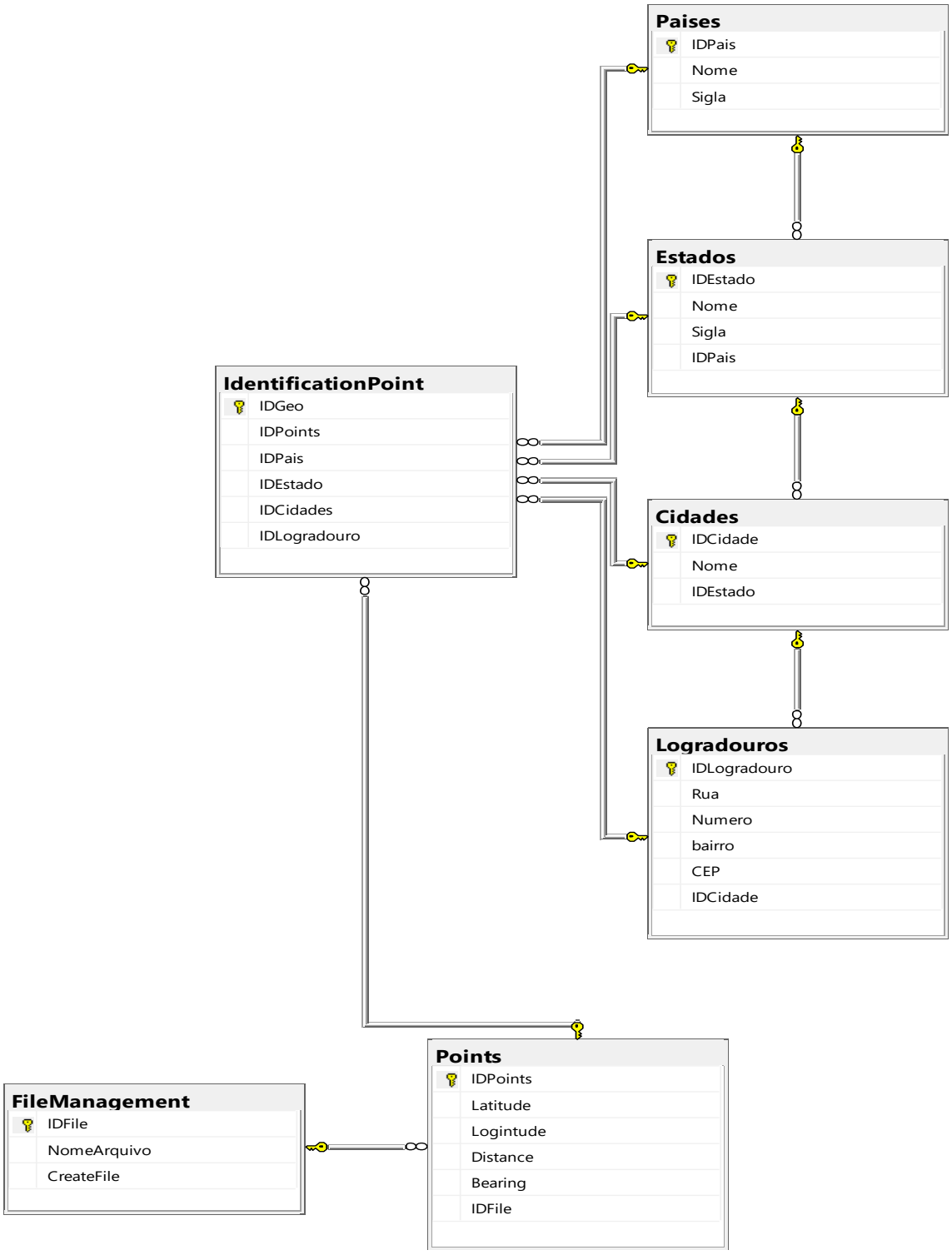
- DAO (Data Access Object) é onde o banco de dados onde está implementada;
- Repository - É as entidades onde tem toda regra para cada tabela do banco de dados; que for consultada, alterada ou atualizada;
- Useful – Essa solução onde estão itens uteis a todo projeto como funções de alteração de arquivos, GoogleMaps e etc.
- RN (Regra do Negócio) - Essa solução está é onde está toda regra da extração e transformação dos dados da aplicação
- Application – Essa solução tem como função de salva, verificar



Diagrama

Conforme o Diagrama abaixo:

- FileManagement – Guarda os arquivos que foram executados;
- Points - Leitura dos arquivos de extração relacionado com FileManagement;
- Países – informações dos países capturado do GoogleMaps;
- Estados - informações dos estados capturado do GoogleMaps e relacionado com pais;
- Cidades - informações das cidades capturado do GoogleMaps e relacionado com estados;
- Logradouros - informações das cidades capturado do GoogleMaps e relacionado com cidades;
- IdentificationPoint – identificação dos pontos capturados onde tem a função interligar os pontos de geolocalização que está relacionado com points, países, estado, cidades, logradouros.


Paises
IDPais
Nome
Sigla
Estados
IDEstado
Nome
Sigla
IDPais
IdentificationPoint
IDGeo
IDPoints
IDPais
IDEstado
IDCidades
IDLogradouro
Cidades
IDCidade
Nome
IDEstado
Logradouros
IDLogradouro
Rua
Numero
bairro
CEP
IDCidade
Points
IDPoints
Latitude
Logintude
Distance
Bearing
IDFile
FileManagement
IDFile
NomeArquivo
CreateFile