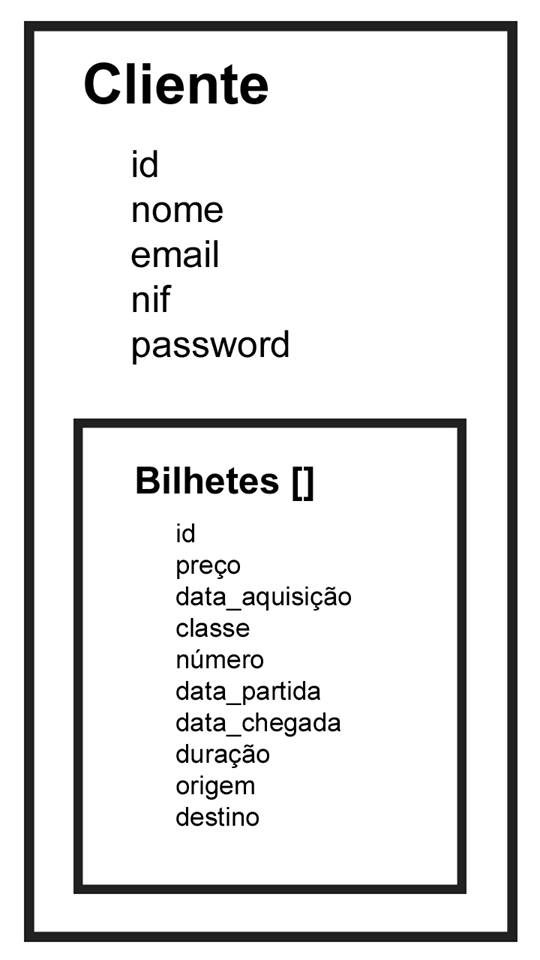
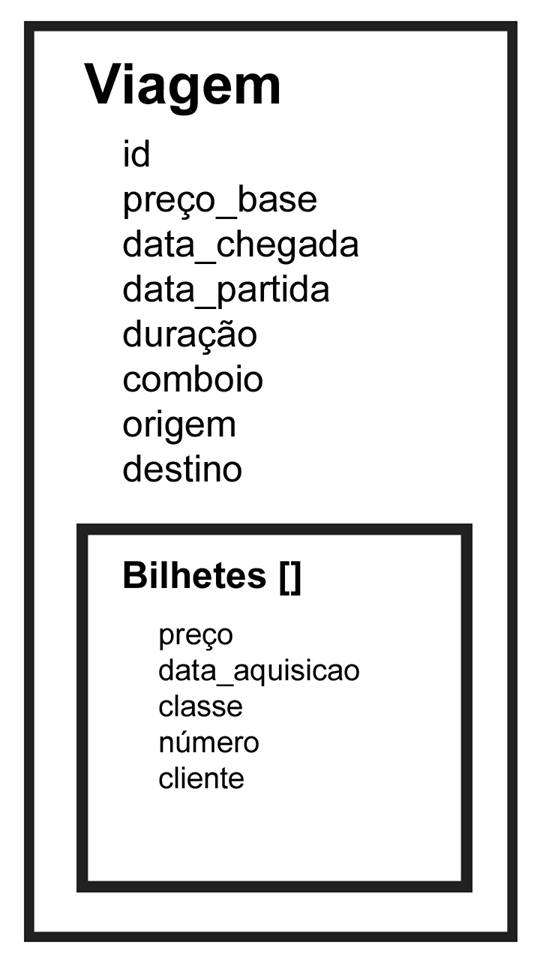
# Definir a estrutura base para o sistema de dados NoSQL que satisfaça os requisitos e as questões apresentadas anteriormente

Perante os requisitos apresentados e tendo em conta o modelo lógico da base de dados relacional, decidimos definir a estrutura base para o sistema de dados NoSQL da seguinte maneira:

* Coleção “Cliente”, cujos documentos contêm as informações relativas a cada um dos clientes (nome, email,NIF e password), apresentando ainda uma lista de documentos “embedded” relativos aos bilhetes desse mesmo cliente. Cada documento “Bilhete” contêm o seu preço, data de aquisição, classe, número do lugar, data de partida, data de chegada, duração, origem e destino. Ambos os tipos de documentos contêm ainda o seu id.
* Coleção “Viagem”, cujos documentos apresentam o preço base da viagem, a data de partida, a data de chegada, a duração, o id do comboio e as estações de origem e de destino. Além disso, os documentos apresentam também o seu id e documentos “embedded” relativos a todos os bilhetes emitidos para essa viagem, quer já tenham sido adquiridos ou não.

Desta forma, conseguimos resolver a maior parte das questões sem recorrer a “joins”, ou seja, neste caso, a “aggregates”. Isto porque anexamos todos os bilhetes de um cliente ao documento respetivo e todos os bilhetes relativos aos lugares do comboio na dita viagem. Não utilizamos portanto nenhuma coleção para os bilhetes isoladamente, pois estes existem sempre no contexto de uma viagem e de um cliente (caso este o tenha adquirido). Não utilizamos também nenhuma coleção ou documento para as estações e para os comboios, pois no MongoDB apenas precisamos de saber o nome das estações de origem e destino de uma viagem (não necessitando de criar um documento apenas para podermos guardar o seu “id”) e, no que toca aos comboios, tendo também todos os bilhetes associados a cada viagem, torna-se também desnecessário guardar propositadamente o seu “id”.

 Os esquemas que descrevemos acima são então os seguintes:

(por imagens q tão na pasta)