**Design challenge COSMOS (30’)**

In this exercise, you have to model a Cosmos DB, that suits the following scenario:

*As a wine lover, you couldn’t find a suitable app to manage your cellar. You decided to create your own app, and share it with others, by publishing it to the different app stores. People can download it and manage their own cellar. You designed for potential growth, as you do not know in advance how many users might be interested in using your app. The app responds to the following requirements:*

* *The home screen should show a dashboard with the number of wines, total cellar value, average price per wine, wine count per color.*
* *The second screen should allow you to add/edit/delete wines. You should also be able to add reviews (possibly more than 1 per wine). Appellations, merchants & wineries should be made available as pre-defined lists. Users cannot enter a free text.*
* *The third screen should allow you to search on the wine name, the appellation, the country, the merchant and winery. Here again, refiners should only list values that exist in the current cellar. You should also be able to search all wines based on the reviews (1 star, 2 stars, etc.)*
* *At last, an audit log should be available for all the operations you did on your cellar*

You must respect the following constraints, imposed by the customer cloud team:

You cannot rely on a search service such as Azure Cognitive Search (this would be too easy 😊)

* You can only use Cosmos DB as a data store and must aim at the cost-friendliest approach.

Note that, you initially built something only for your own, which was based on a traditional relational database. Here is the design you came up with:

