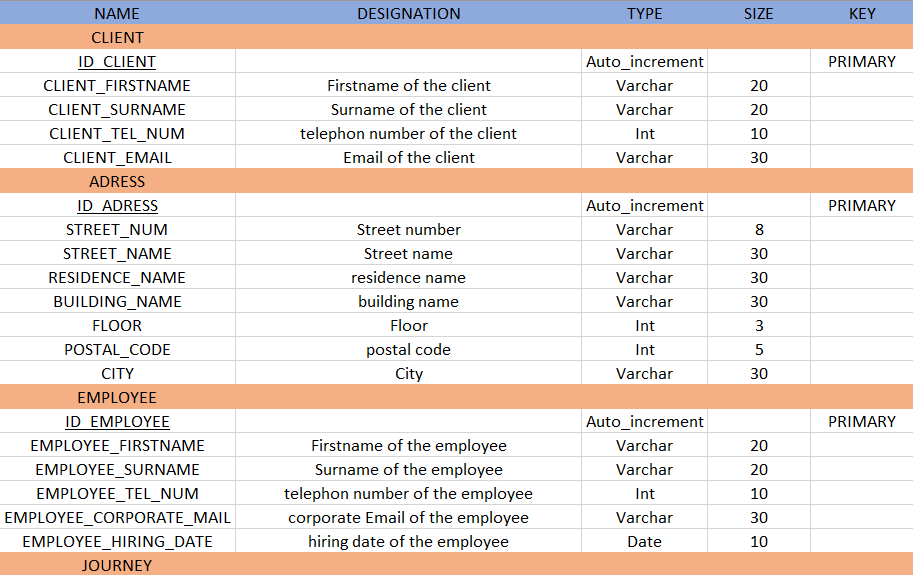
**DELIVERABLE 1**

In This deliverable we had to do firstable a data dictionary on the base of the context of our project, then we had to create our Conceptual Data Model,

1. Data Dictionary:



*(see attached document “dictionary.xlsx”)*

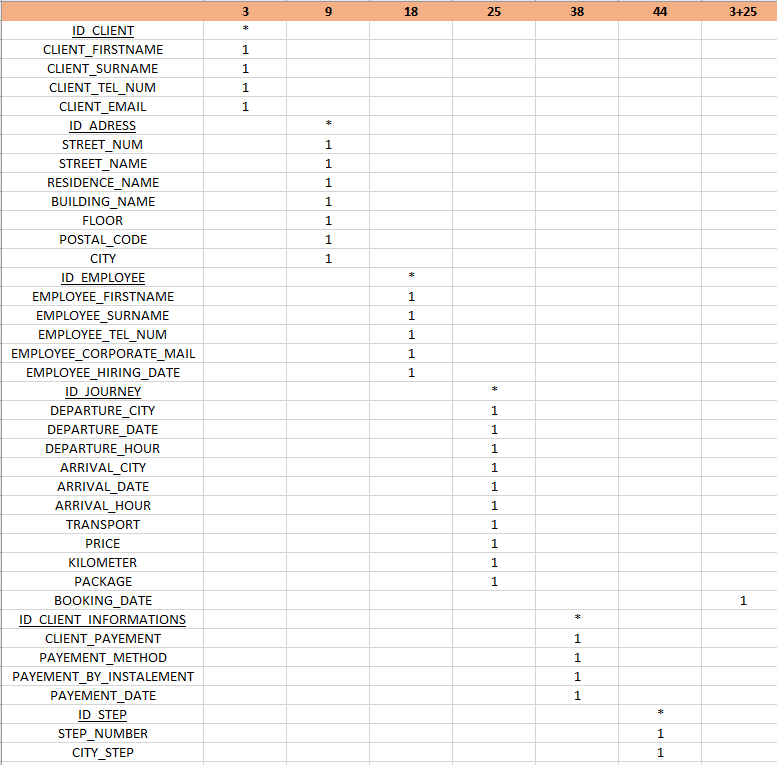
Here is a part of our dictionary. We can see that we have 5 columns, Name, Designation, Type, Size and Key.

* Name: The name of our table (highlighted in orange), the name of our primary keys (underlined) and the name of our attributes.
* Designation: The description of our attributes.
* Type: The type of our attributes and keys
* Size: The size of our attributes and keys
* Key: If the attribute is a key or not

In this dictionary we created 6 tables named Client, Address, Employee, Journey, Client Information, and step. In these we put all the attributes we need to meet customer expectations. We also created our primary keys to link the tables together.

1. Conceptual Data Model:

Before creating the CDM we did a functional dependency based on our dictionary and our tables.

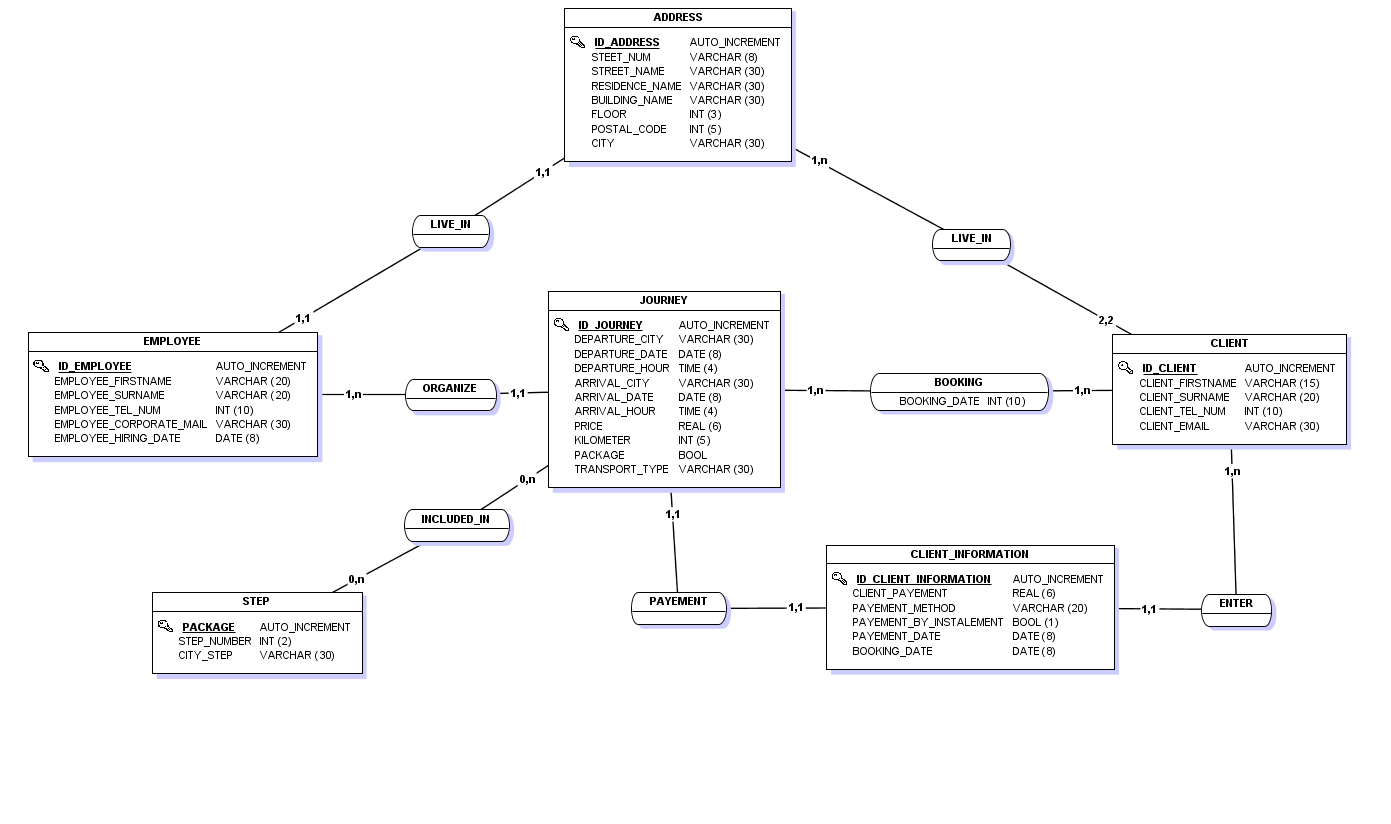


*(see attached document “dictionary.xlsx”)*

We matched all our attributes to the key they are linked.

The last column attached to “BOOKING\_DATE” is a link between two tables “client” and “journey”.

After that we done the CDM.



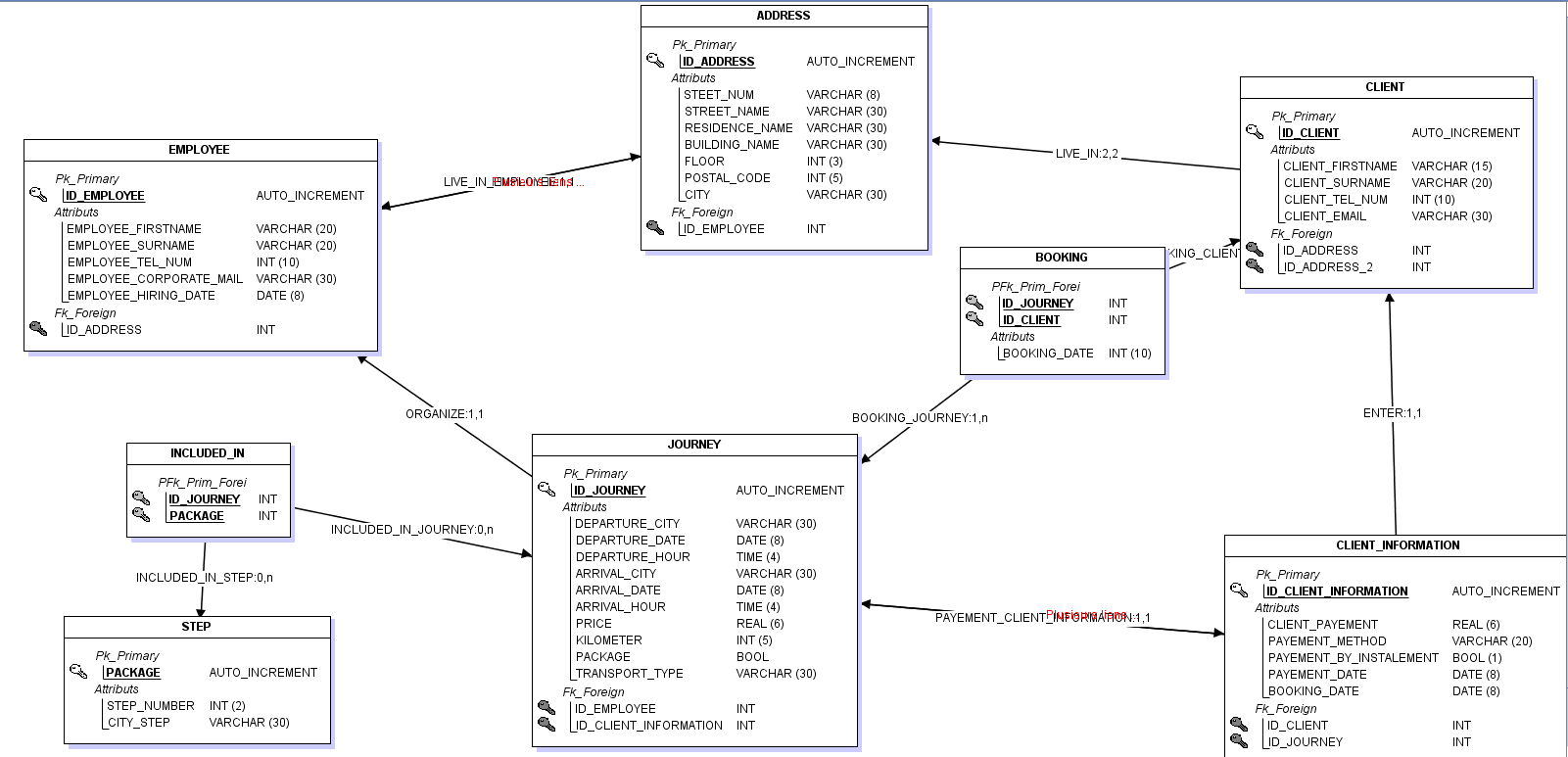
*(see attached document “CDM.mcd”)*

We first linked “ADDRESS” with “EMPLOYEE” and “CLIENT” then we linked them to “JOURNEY” because the employee organizes a journey, and a client books it. After that we linked “CLIENT” to “CLIENT INFORMATION” and “CLIENT INFORMATION” to “JOURNEY” to have a table with just the client payment information and the booking date. It is useful if you want to have a better security for your client payment information. Finally, we added the table “STEP”, it is useful when a journey has steps and by adding it in another table than “JOURNEY” it does not flood our principal table.

1. Logical Data Model:

After finishing our CDM we where able to create our LDM.

Our CDM was supposed to be correct to generate a LDM so we worked on it to be as correct as it should be.



*(see attached document “CDM.mcd”)*

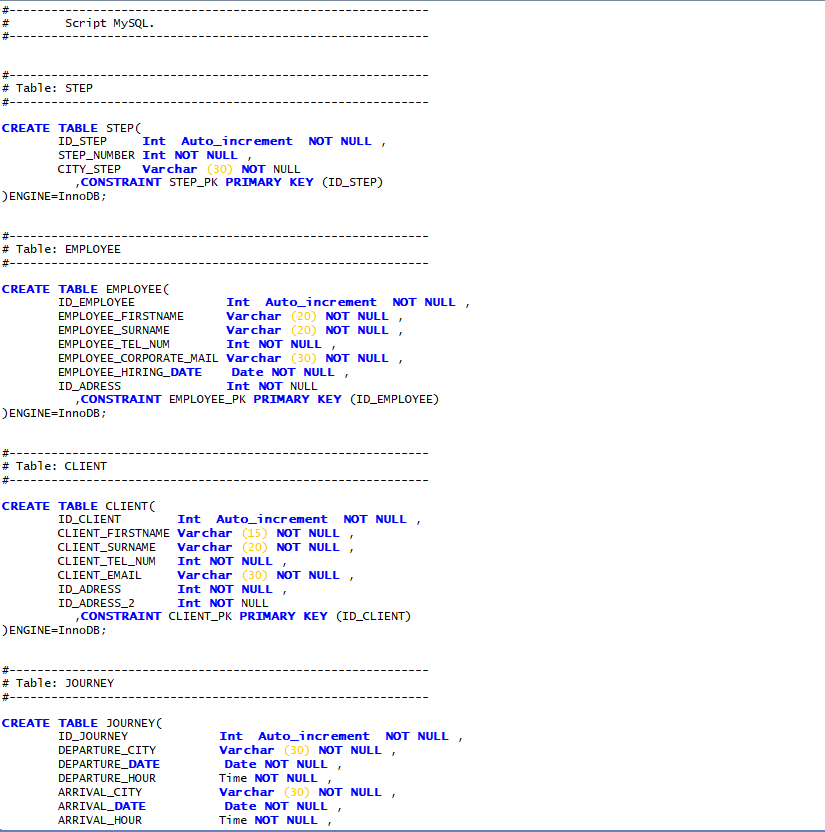
This LDM allows us to clearly see our primary keys and foreign keys and cardinalities between our tables.

It’s very useful to understand better relations between our tables.

1. Data Definition Language:

We generated this script with the CDM using JMersie.

Here is a part of our SQL script.



*(see attached document “CDM.mcd”)*

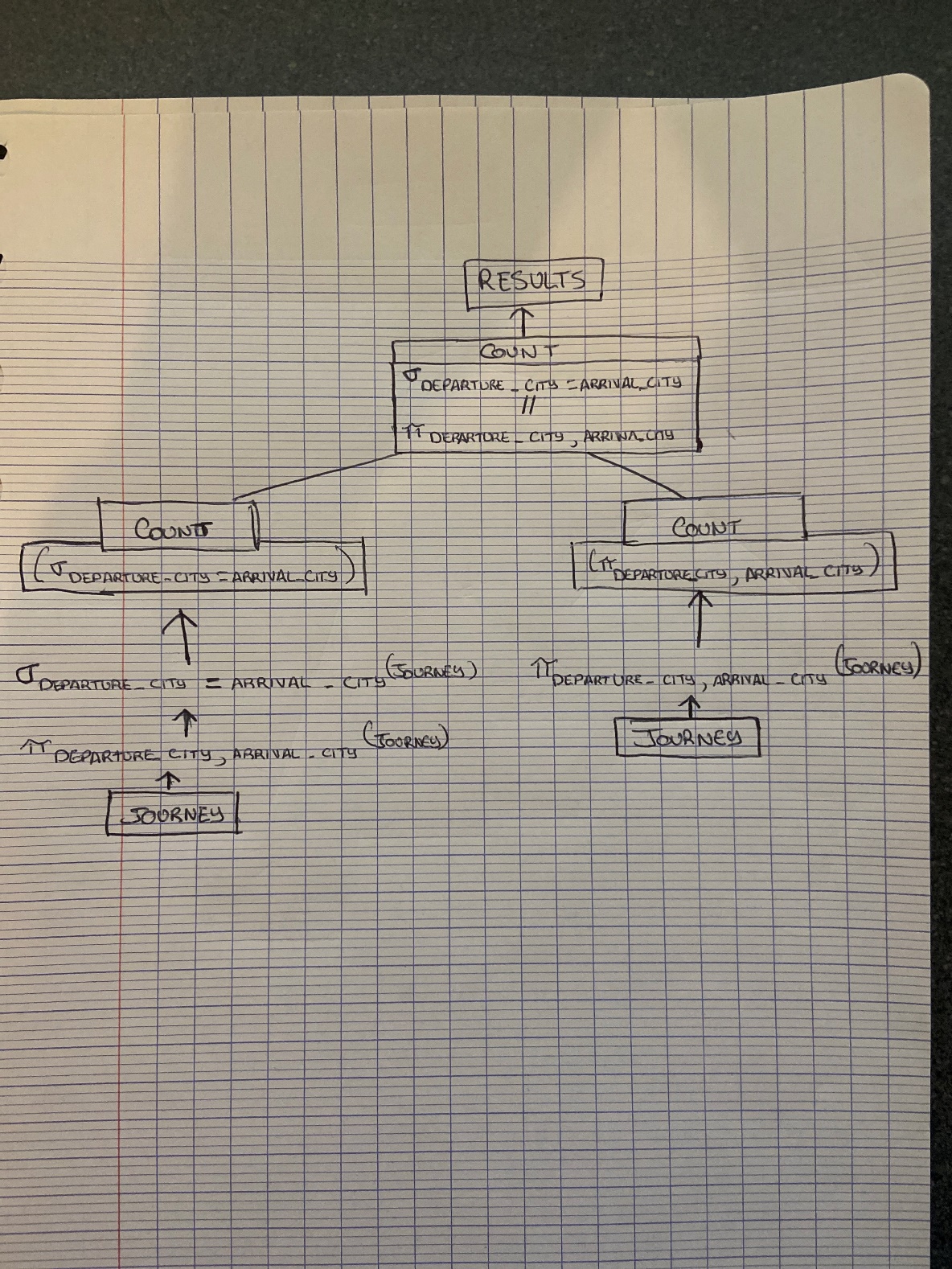
In this script we can find the creation of the tables, the joins between the tables, and all the foreign and primary keys.

If we run this script, it will create our database and we will just have to enter data to finish the DB.

1. SQL Request:

Request N°5:

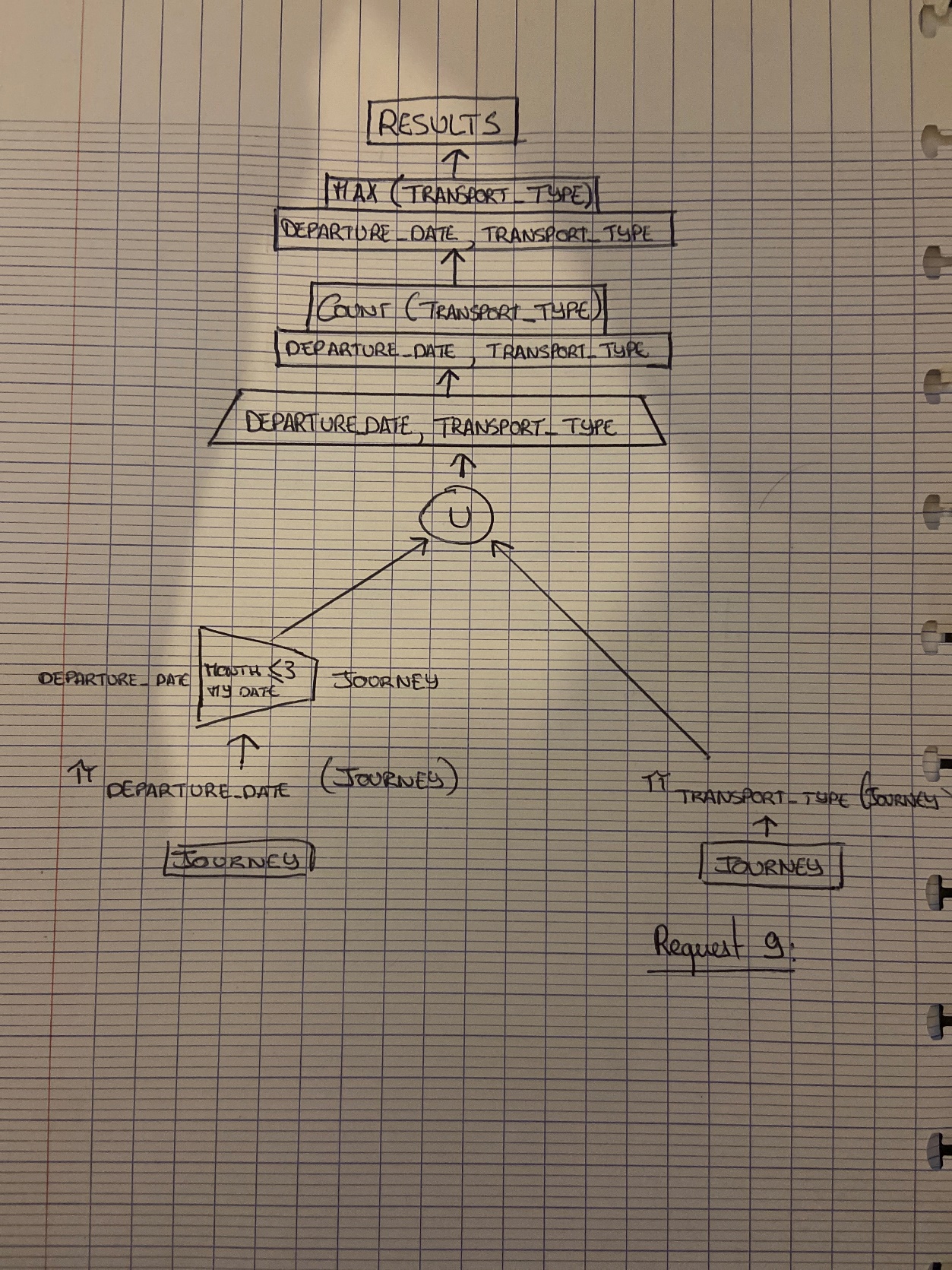
Know the proportion of intra-city and inter-city trips.



We select only the intra-city trips and we dived them by the entire attribute “DEPARTURE\_CITY” and “ARRIVAL\_CITY”. Then we divided it to obtain the proportion of each trips.

Request N°9:

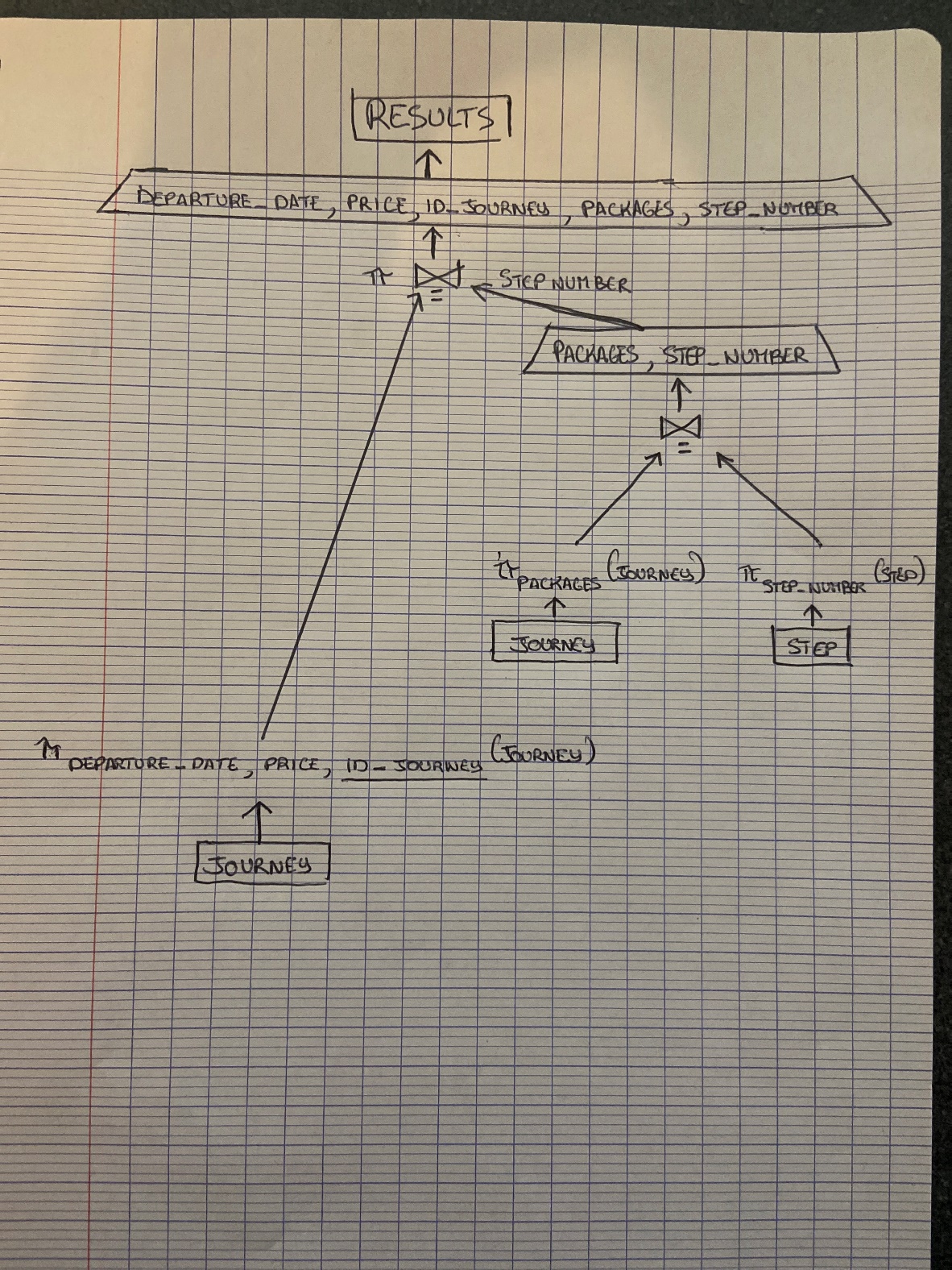
Know the most used type of transport over the last 3 months.



Firstable user must enter the date of the day after that we select “DEPARTURE\_DATE” then we put a condition that we took the last 3 months before user date information of the attribute. We select “TRANSPORT\_TYPE” in the table “JOURNEY” and we count each transport type, and we isolate the most popular type of transport.

Request N°16:

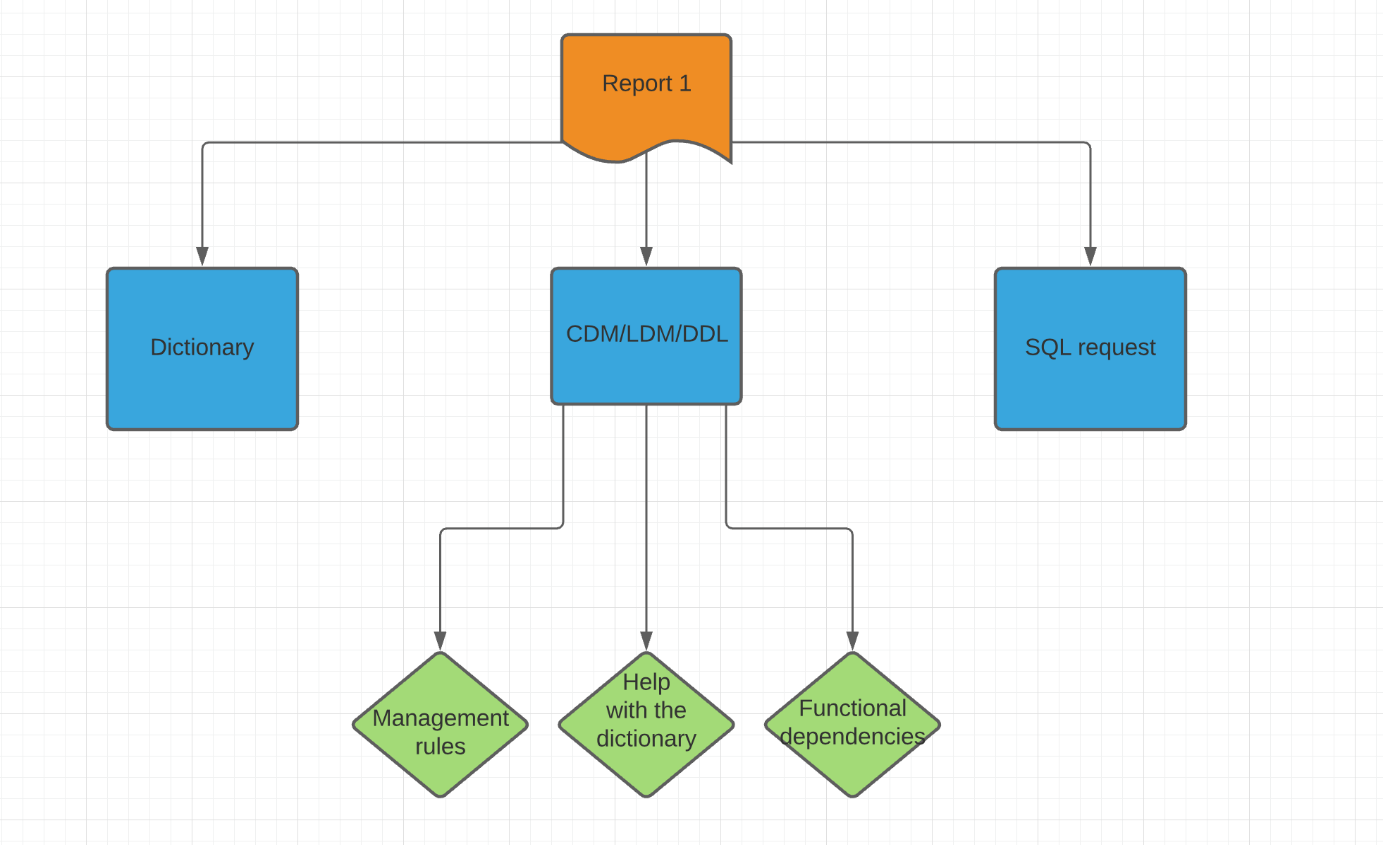
Know the different information for a trip: Trip number, its price, date of departure, number of cities crossed.



We only pick in the table “JOURNEY” the attributes “DEPARTURE\_DATE”, “PRICE” and the primary key “ID\_JOURNEY”. We select “PACKAGE” and “STEP\_NUMBER” and we joint it to have on table and after we joint the first information, we pick with our new table to make another table.

1. WBS:

For the WBS, we done one on the deliverable 0 but we refined it for this deliverable to be more precise for the expectations of this deliverable.

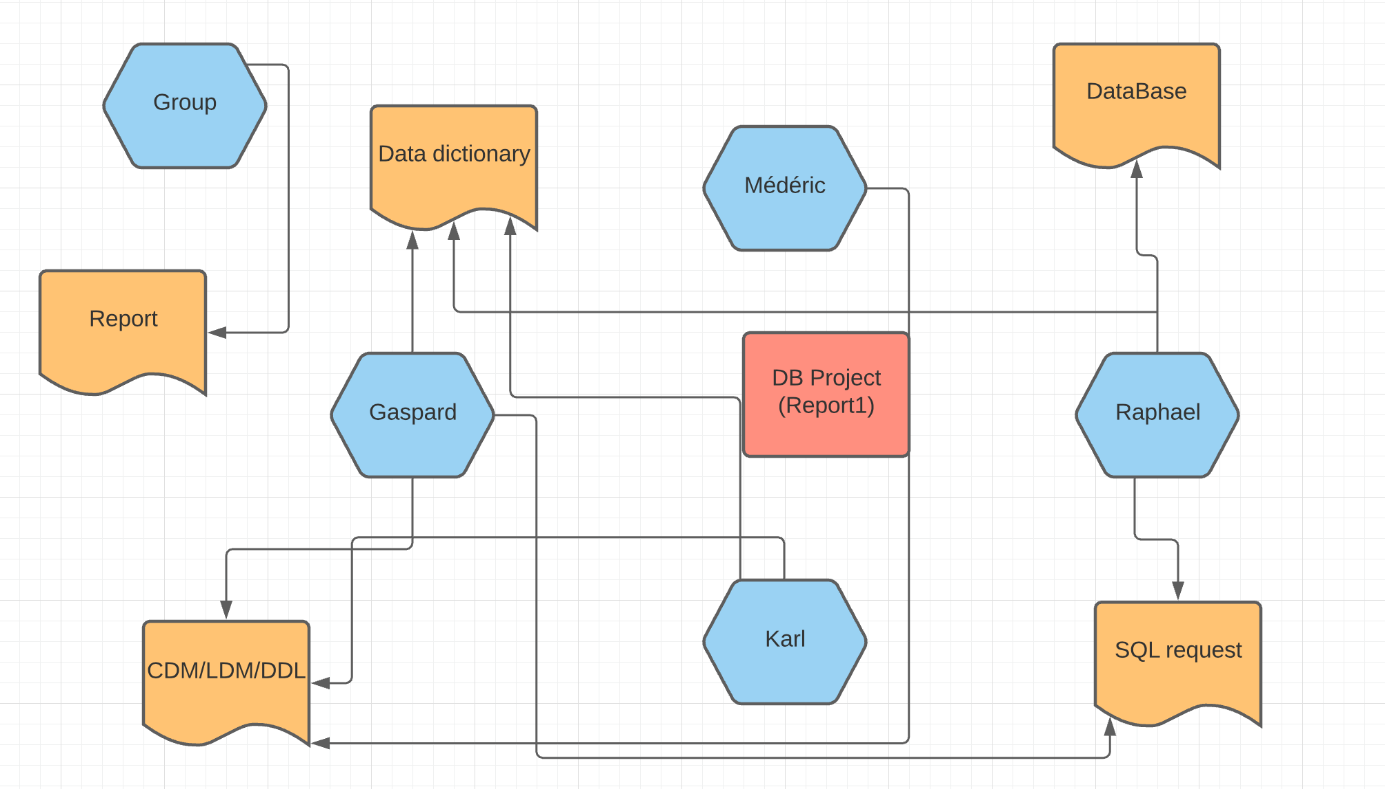


We dispatched the work into 3 groups “Dictionary”, “”, and “SQL request” so we could better understand the expectations of this deliverable and also better understand what we had to do what to start with and in which order we have to do the things we have to do!

1. OBS:

During all the course of the deliverable we noted all we were doing and so with this document we could organize an OBS.

Thanks to this document and the upstream organization we could have a good workplan and being very efficient.



We wanted to have the most skills we could, so we divided the work note by tasks, but everybody was doing all the things together. We worked all together on all the things we had to do during this deliverable. We were constantly on call to confront our ideas and find the best way to answer the problematic.

1. Conclusion:

To conclude we done a blank functional database which will be able to contain the future society’s information. We also done some of the SQL request to allow the company to filter, research and product analytics of their clients, journeys, and other things. The only last thing we can do is improving and optimising the database to be as easy as “abc” to use for our client!