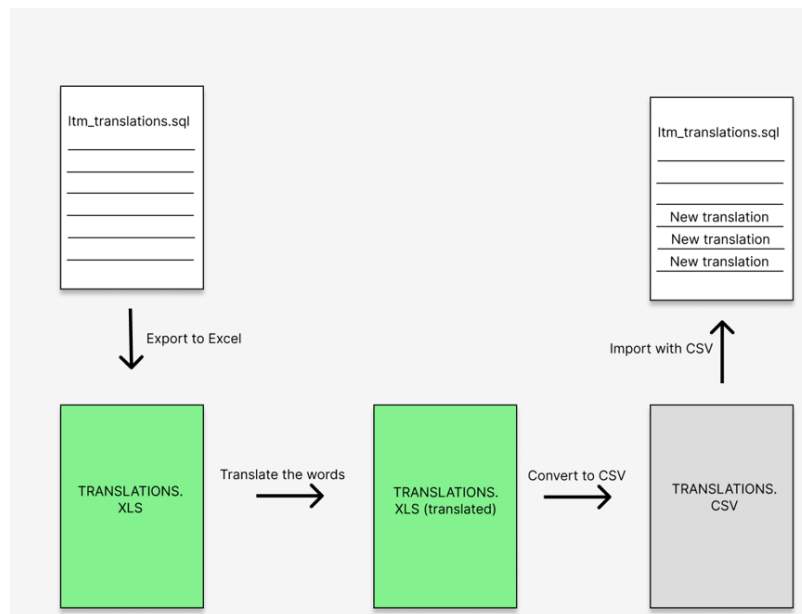


# TRANSLATIONS EVULPO

As we can see in the example of **Itm\_translations.sql**, we have all the words in german and we need to translate them and store them in the same table. First of all, there are several ways to proceed and we need to chose one. In my opinion, I would have an Excel document with all the translations and I would load in the database. The steps are the following:

- 1) We export the table **Itm\_translations.sql** in xls format in order to get all the keys of the different words sorted.
- 2) We have to translate manually all the words, in this case in german, to the language needed. In this step I would do the translation manually because the automatic translations are very tricky and the results are not exactly and in that way you ensure that the translation is accurate.
- 3) We convert the Excel file into a CSV file because is better when you want to import information through a file.
- 4) Finally, we load all the information which we have in the file to the database.



Being more specific and with a UI/UX flow I would put a button in order to choose the file and then and a dropdown menu to choose the language like in the picture:

Type\*

File\*  No file chosen

In our case instead of type, it will be language. The information will be taken like this:

- Id: Created automatically because is AUTO\_INCREMENT.
- Status: 1st column of the CSV file.
- Locale: Inserting the value of the selected dropdown menu.
- Group: 2nd column of the CSV file.
- Key: 3rd column of the CSV file.
- Value: 4th column of the CSV file.
- Created At: Inserting the current time using `LocalDate.now()` for instance.
- Updated At: Inserting the current time using `LocalDate.now()` for instance.
- Language\_Id: Using an internal converter in our code with the input language. For example, german is 5 so if it is german, we will insert a number 5.