This little program is capable of managing words and icons, so the user can add words, match them with icons, and later delete the words and their icons’ relation. All of this while making sure there’s no duplicate words or that a word only has one icon matched.

**How to start and use the program**

In order to start the program, you only need to:

* Open a Terminal on the program’s root folder
* Execute the following line: php -S localhost:8000

These steps will run the program on the url **localhost:8000**, so in order to use the program you only need to access that URL (without closing the Terminal).

**Directory structure:**

database

|\_ db.json

icons

|\_ (65 icons in .svg format)  
src

|\_functions.php

index.php

**db.json**

This is the database file, which uses a JSON format. It has three “tables”:

* words: table for storing the added words. It has two columns:
  + id
  + word
* icons: table for storing the imported icons. It has two columns:
  + id
  + icon : this saves the name of the icon file
* words\_icons: table for storing the relationship between words and icons. When the user matches a word with an icon, this relation is stored here. It has two columns:
  + id\_word
  + id\_icon

**icons folder**

The icon folder contains the 65 icons provided in the test task email, which were decompressed from the provided .zip file.

**index.php**

In this file is where all of the functions are called. There’s some interactions included as an example.

First, the program starts calling cleanDB(), a function that will clean and start a new DB from scratch. Then, the function importIcons() is executed, which will import all of the icons from the folder /icons.

**functions.php**

**readDB()**

This function reads the db.json file and returns it as an array for easier data manipulation.

**saveDB($db)**

This function saves the data to the JSON file with a “pretty print” for easier reading. If there’s an error writing the file, it lets you know in the browser.

**importIcons()**

This function reads all the icons from the /icons directory and saves them to the database. It will also assign an ID to each of them. It is mandatory to run this function at the start of the program, or there won’t be any icons to work with.

**addWord($new\_word)**

Adds a new word to the database and assigns an ID to it. Before adding the word, the program checks if it doesn’t exist in the database already. In case it does, it won’t add a duplicate. Notice that two words with lowercase/uppercase letters are considered different to the system (“cat” and “Cat” are two different words).

**removeWord($word)**

Deletes a word from the database. First it checks if the word actually exists in the DB, if it does, it deletes it. Then, it checks if the word has a related icon. If it does, it also deletes the relation with the icon.

**checkDuplicateWord($new\_word)**

Checks if the inserted word already exists in the database.

**matchWordIcon($word, $icon)**

Saves a new relation between a word and a selected icon in the “word\_icon” table. It also checks if the inserted word doesn’t have a matched icon already, because the logic is “One word can have only one icon, but an icon can have many words”.

**getID($data, $table, $column)**

This function searches for a word or icon ID in the database. Since it’s a generic function, it’s mandatory to specify the “data” (is it a word or an icon?), the “table” (table ‘words’ or table ‘icons’?) and the column (the table ‘words’ has a ‘word’ column and the table ‘icons’ has a ‘name’ column).

**checkMatchedWord($id\_word)**

Checks if a word already has a matched icon.

**cleanDB()**

Starts a new database from scratch and gives it the following format:

{"words":[],

"icons":[]

,"words\_icons":[]}

This function was specifically created for testing.