

# PokemonGO Statistics report

**AUTHOR:** BOZHENOV OLEG

**SUBMISSION DATE:** JUNE 13TH, 2022

## Problem statement

Create an application using QT Framework which loads a dataset with Pokemon characteristics and represent them in Master-Detail way. MVC Pattern was used to display the data from the dataset.

## Individual project specification

Can be found by following [this](#) link.

## Implementation details

### General

Repo can be found by following [this](#) link.

I have tried to follow the given specification, however, there are some differences that makes app more user-friendly. It was asked to make the window with information about project creator, which was accessed by three buttons. This idea seemed uncomfortable, so after clicking “help” button the dialog window shows up with buttons “about”, leading to the *QMessageBox*, and “back”.

Also, there is a change in logo. I have decided to move it to the main window (which seems more logical than putting it in the about window) and make it more recognisable by drawing a poke ball instead of thunder. Locating it in the middle of the screen in the back of the buttons seemed like a good design solution to me.

In general, there was no problems related to the representation and processing of the data.

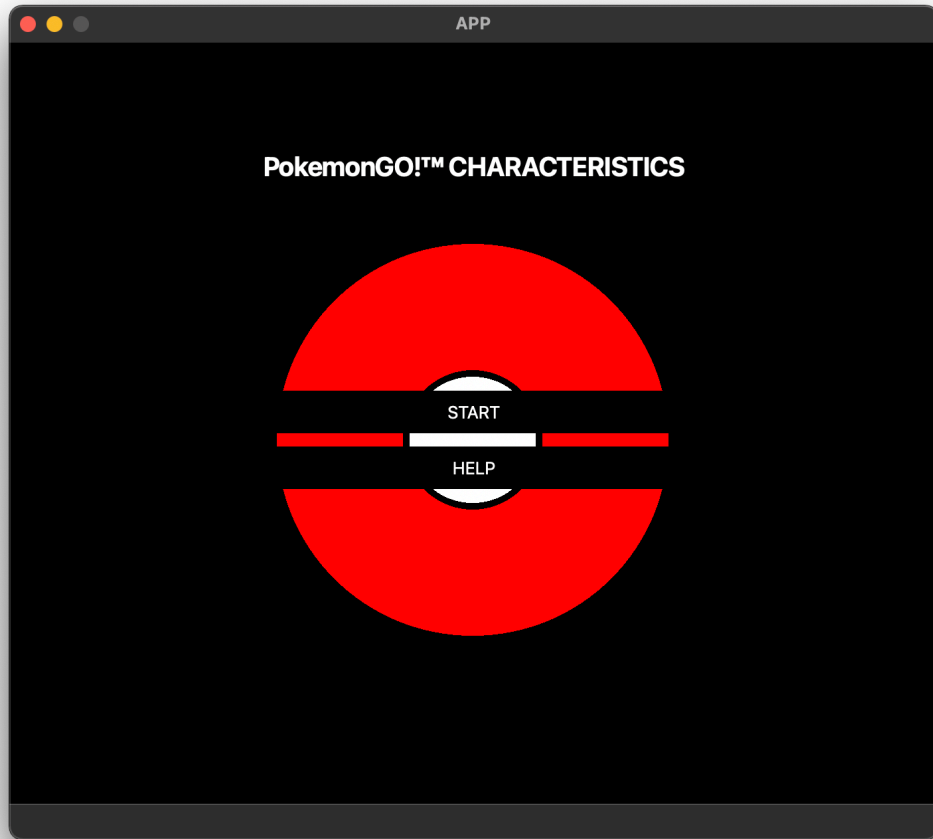
## Project Structure

- PokemonGO.csv — used dataset
- updData.csv — saved dataset
- UIs:
  - addwindow.ui — the UI of the “Add Window”
  - info.ui — the UI of the “Info” Dialog Window
  - mainwindow.ui — the UI of the “Main Window”
  - pokechar.ui — the UI of the Pokemon Characteristics Window
- pal.xml — used palette
- Header Files:
  - addwindow.h
  - info.h
  - mainwindow.h
  - pokechar.h
  - pokemon.h — the struct of the Pokemon (Number, Name, Types, CP, HP, Link)
  - proxymodel.h
- Source Files:
  - addwindow.h
  - info.h
  - mainwindow.h

- pokechar.h
- proxymodel.cpp
- main.cpp — runs the app

## How it works?

### Main Window (mainwindow.cpp/h/ui)



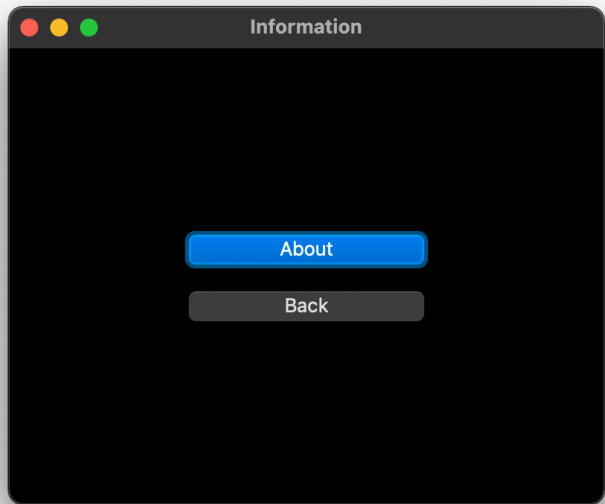
Main Window

The main window contains two buttons: “Help” and “Start”. The first one leads to the dialog window (see [Info Window](#)) and the second one to the Pokemon Characteristics (see [Pokemon Characteristics Window](#)). As I have already mentioned, making logo in the back of the buttons is the best design solution. Also, I decided to make the whole app in the dark theme to save battery power.

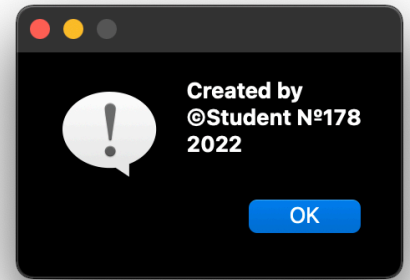
### Info Window (info.cpp/h/ui)

I have decided to make the “About” button the default one (you can access it by tapping Enter/Return button on the keyboard). By following this button, the Message Box (used QMessageBox library) shows up with information about creator. I used built-in Information Message Box. This Message Box contains the “OK” button which closes it.

There is “Back” button which closes the Info Window.

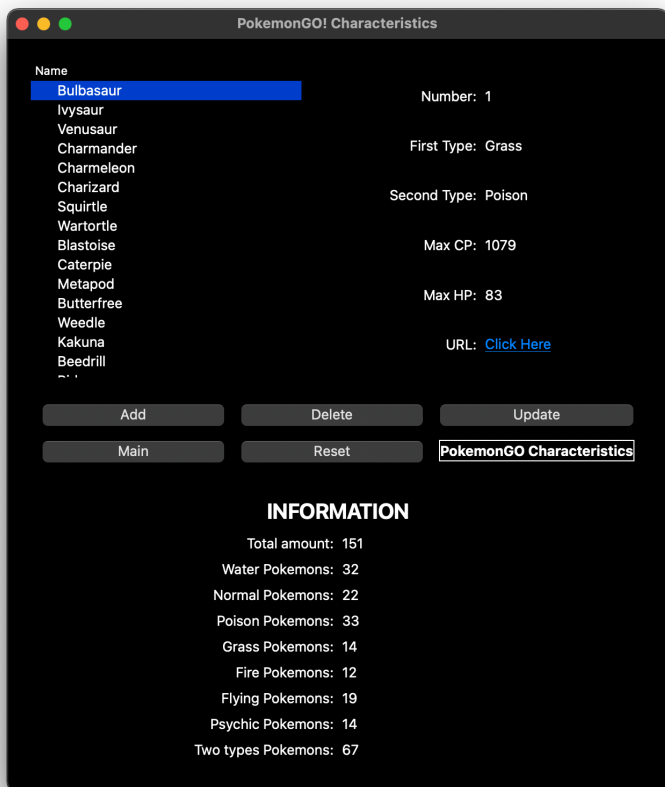


Info Window

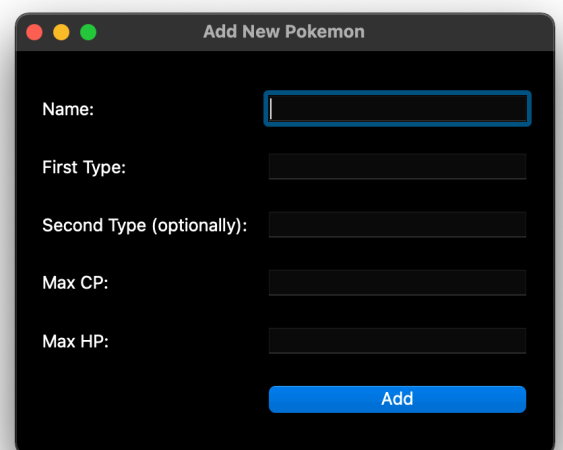


Message Box

## Pokemon Characteristic Window (pokechar.cpp/h/ui, addwindow.cpp/h/ui)



Pokemon Characteristic Window



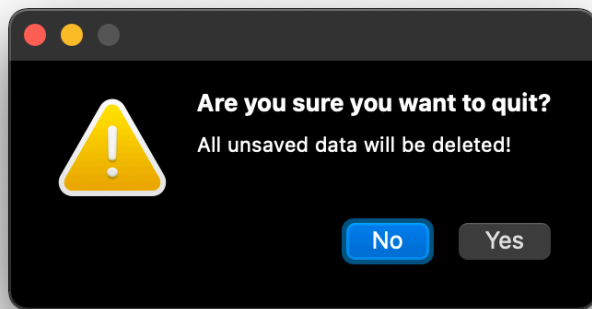
Add Window

I used Master-Detail view for representing the data. On the left there are names of the pokemons, on the right there are the information about it: number in the dataset, first type, second type, max CP, max HP. Names could be sorted lexicographically by clicking the column name.

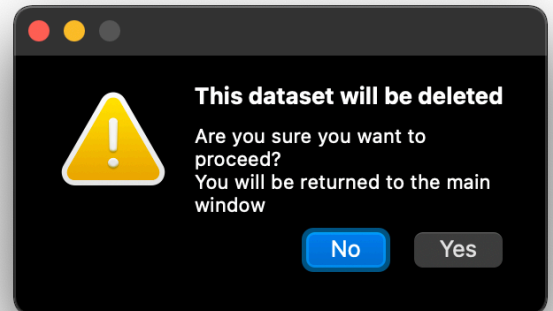
There is the information about all pokemons, which are counted in the vector: amount and types.

There are buttons: Add (see Add Window pic.), Delete, Update, Main (see Warning Message Box No. 1 pic.), Reset (see Warning Message Box No. 2 pic.).

- “Add” button opens the dialog window with QLineEdits. After clicking default button add, all entered data will be pushed back to the vector.
- “Delete” button deletes chosen pokemon from the vector.
- “Update” button creates new dataset “updData.csv” with all added pokemons.
- “Main” warns user about closing the window and, after clicking “Yes”, returns to the main window.
- “Reset” warns user about deleting all new pokemons and returning to the main window. After clicking “Yes”, deletes the created “updData.csv”.



Warning Message Box No. 1



Warning Message Box No. 2

## Possible improvements

As one could mention there are proxymodel files, so the project could be improved by making filters.

## Conclusion

The project requirements were fulfilled with the exception of minor differences in the logo location and making message boxes with warning the user of their steps.