I. RESULTS

The implementation of all the programs contained in the project result in obtaining the data from the web scrapper, then transforming it to build the Graphical User Interface (GUI). This will help the user to get information on the lausanne housing market, see price listings, historical listings and navigate easily between available properties.

A. Dataset

The dataset we used for the building the GUI is reduced compared to all the data we initially have access to. We chose to keep this reduced version of the dataset, to keep the GUI easy to use and read for the user. Figure 1 and 2 show the the first few rows of the two cvs files: `property_codes.csv` and `property_details.csv` returned from the webscrapping. Figure 3 shows the data we used to feed the GUI. Figure 4 shows the data we used for the average price by zip code graph. Figure 5 shows the data we used for the average price by number of rooms graph.

Fig. 1: First rows of `property_codes.csv`

Fig. 2: First rows of `property_details.csv`

Fig. 3: First rows of `data.xlsx`

Fig. 4: First rows of data used for the average price by zip code graph

B. GUI

The GUI presents the scrapped data in an orderly manner.

Fig. 5: First rows of data used for the average price by number of rooms graph

1) Main Window: The main window contains the four available tabs namely, price range, Rooms, Zip Code and Graphs and the empty treeview. (see 6)
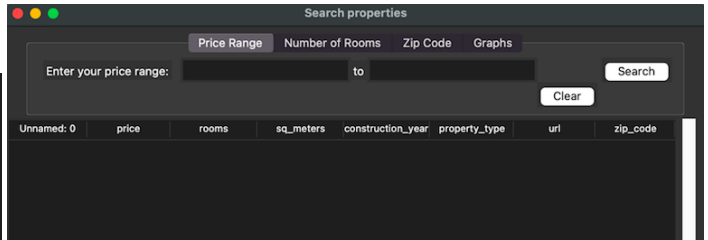

Fig. 6: View of the main window

2) Frame 1: The first frame allows the user to search within a specific price range. The user enters a minimum value and maximum value within each of the boxes and presses the search button. The treeview returns all the properties within the chosen price range. (see 7) To restart the process, the user can either press the clear button or enter new values.
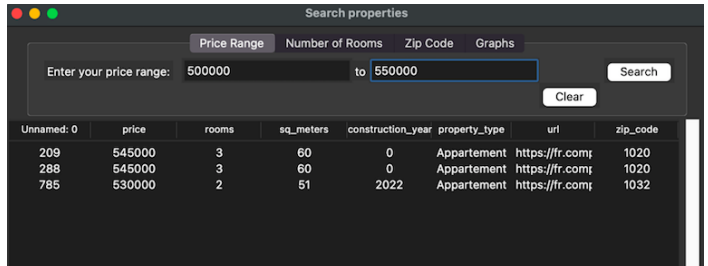

Fig. 7: View of the first frame

3) Frame 2: The second frame allows the user to search within a specific number of rooms. The user enters a minimum value and maximum value within each of the boxes and presses the search button. The treeview returns all the properties within the chosen number of rooms. (see 8) To restart the process, the user can either press the clear button or enter new values.

4) Frame 3: The third frame allows the user to search for properties in a specific zip code. The user enters the wished zip code in the box and presses the search button. The treeview returns all the properties within the chosen zip code.(see 9) To

Fig. 8: View of the second frame

restart the process, the user can either press the clear button or enter a new value.

Fig. 9: View of the third frame

5) Frame 4: The fourth frame displays two buttons corresponding to two different bar graphs, namely the average price by zip code and the average price by number of rooms. The user must simply press on the corresponding button and the graph is displayed. The figure 10 displays the average price per zip code in millions of CHF. At the time when we run the program we can see that the zip code 1094 correspnding to

Belmont sur Lausanne has the highest average price, namely of 3.7 million CHF. In general, the zip codes corresponding to the outskirts of Lausanne have a higher average price. This could be due to those areas being less built up, hence the properties there tend to be houses and villas rather than apartments. has a much higher price.
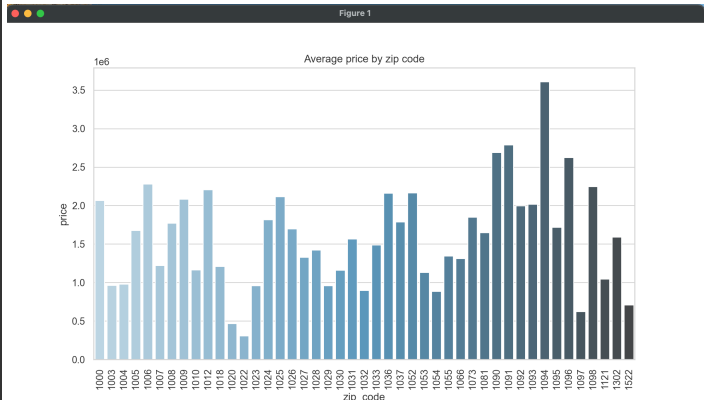

Fig. 10: Average price by zip code

The figure 11 displays the average price per number of rooms by a base of 10 million CHF . At the time when we run the program we can see that the price per number of rooms does not grow exponentially as the number of rooms increases. This could be due to the dataset containing many types of properties such as: commercial properties, apartments, houses and hotels, which messes with overall average. Moreover a lack of data can also explain the lack of exponential growth as the number of rooms increases.
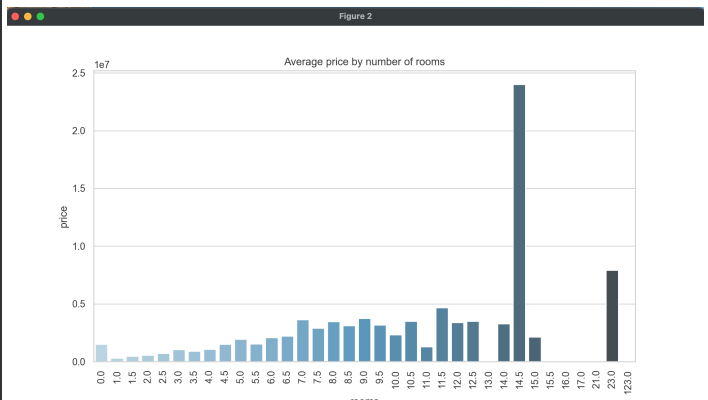

Fig. 11: Average price by zip code

6) Heatmap: Iniatially we wanted to display a heatmap of Lausanne based on the average price by zip code. This map is built with the help of the geopandas and folium packages. To build it we merge the MeanPriceZip file with the geolocations of the Lausanne's zip codes. Nonetheless, due to a merging and geolocation issue we were unable to display the map.