Medi_Vizz User Guide
(Developed based on the Windows 11)

If you struggle with any part of the user guide or running the application, please contact Ms Uscinnia Dyn'ko at this address: uscinnia.dynko@ut.ee

Medi_Vizz is an application for the concurrent visualisation of medical data. It can visualise diagnoses encoded by ICD-10 classification and numerical laboratory measurements.
Medi_Vizz is not designed for data protection. Please don't leave personal identification in the names of the files or the data. It runs without access to the Internet, so there is no possibility of a data leak. However, the application's creator is not responsible for who sees your computer screen with the data while you run the application.

Installing Python to run the Medi_Vizz
(original guide https://phoenixnap.com/kb/how-to-install-python-3-windows)

Medi_Vizz is coded in Python, and to start the application, you need to have Python version 3.12 or later versions.

1. Go to https://www.python.org/downloads/windows/ and select the Python version and a suitable computer architecture (Fig. 1).

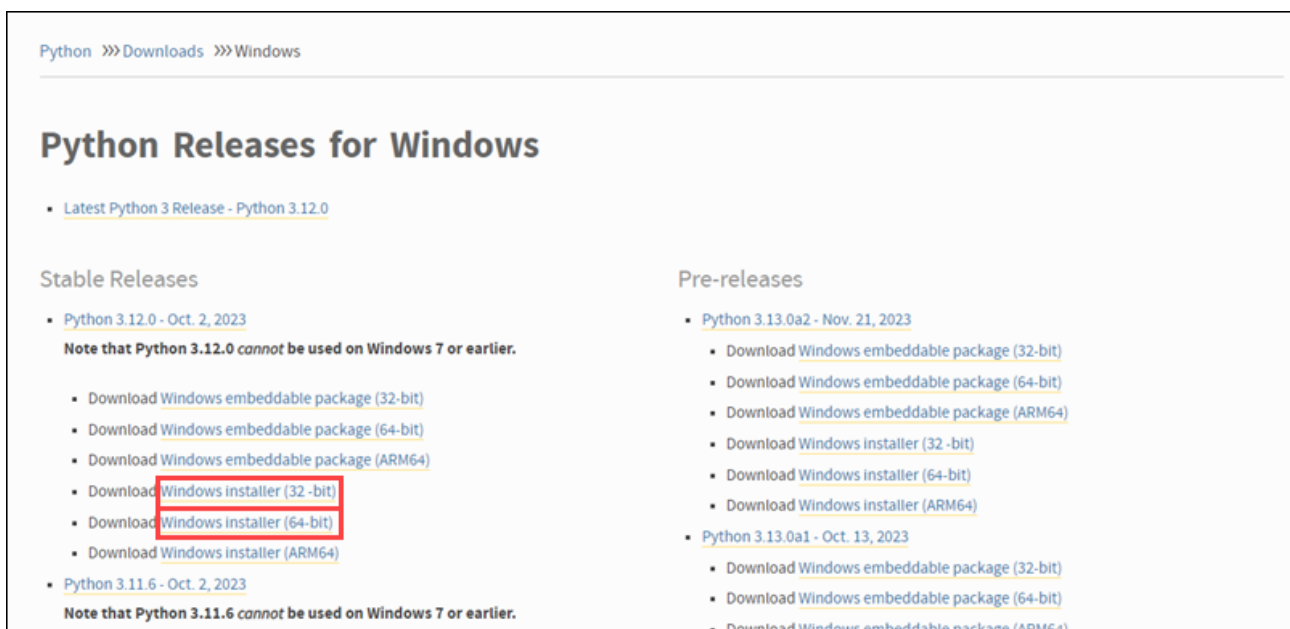**Fig. 1.** Python version selection of the python.org website. Use the version for either 32-bit or 64-bit computer architecture indicated by red box.

2. Find the installer in the Downloads folder and run it. Check all the boxes indicated by red arrows (Fig. 2).
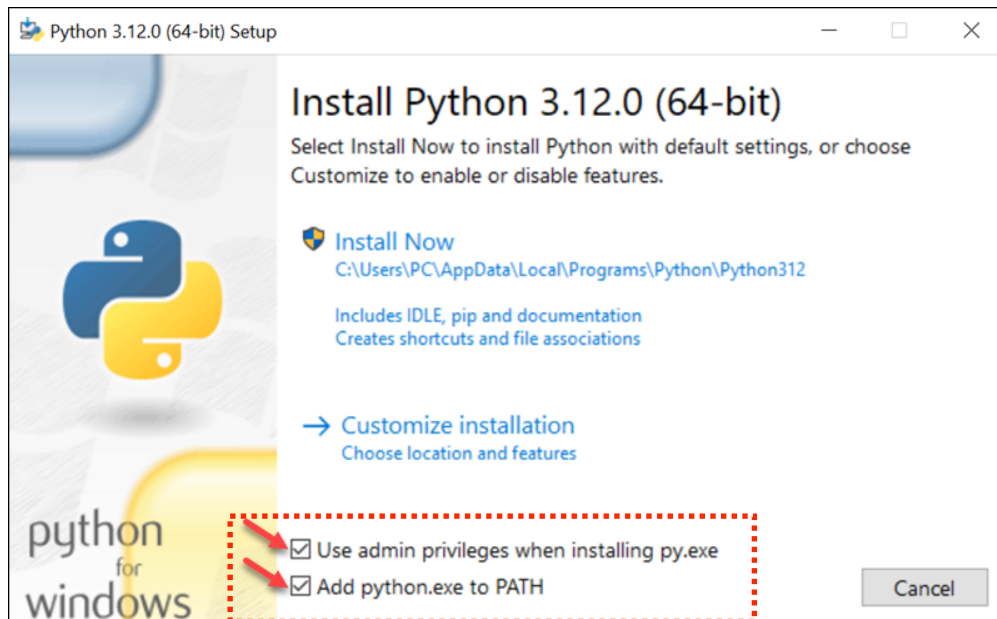

Fig. 2. Python installer started. Tick the boxes highlighted by the red rectangle and arrows.

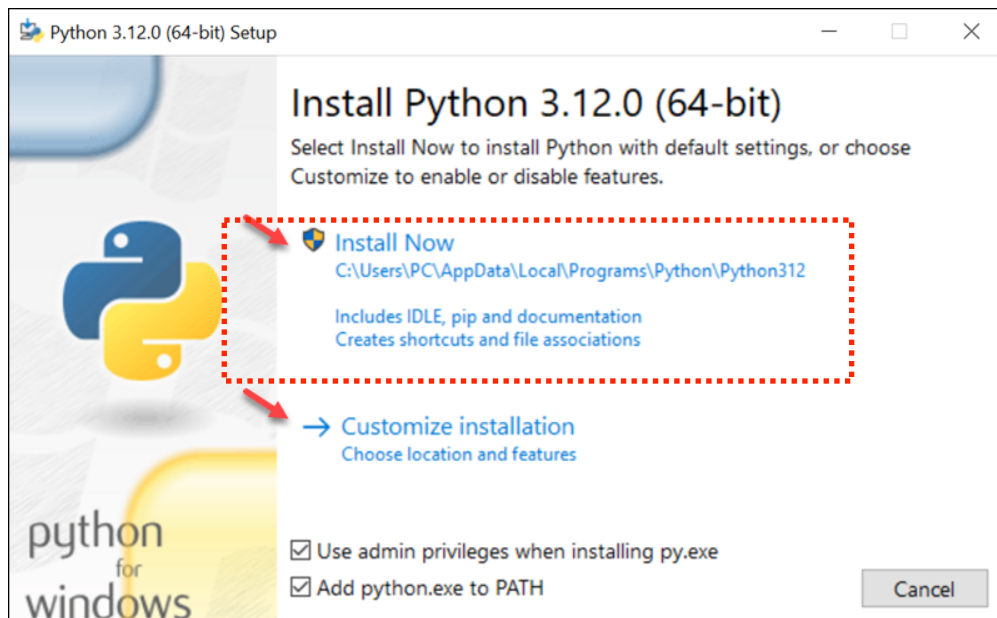3. Select the "Install Now" option (Fig. 3, red box).

Fig. 3. Installation process. Press on the "Install Now" highlighted by red box.

4. Check the installation in the command line:
- Press the buttons combination "Windows" + "R".
- Write **cmd** command to start the command line and press "Enter".
- Type in **python --version**. The expected output is shown in Fig. 4.

Fig. 4. Expected output after running the command **python --version** in the command line. The blue arrow points at the entered command, and the red arrow points at the output after running the command.

Installing required libraries for Medi_Vizz
(based on https://ehmatthes.github.io/pcc/chapter_12/installing_pip.html#pip-on-windows)

You will use the pip manager to install Python libraries required by the Medi_Vizz tool.

1. Check that the pip manager is installed using the Windows command line:
-   Press the buttons combination "Windows" + "R".
-   Write **cmd** command to start the command line and press "Enter".
-   Type in **python -m pip --version**.

2. If pip is not found during the previous step:
First, try to install it using the command **python3 -m ensurepip --default-pip**
If that doesn't work, try these steps:

-   Go to the website https://bootstrap.pypa.io/get-pip.py
-   Select the entire text using Ctrl+A and copy it into any text editor.
-   Save the copied text into your Downloads folder as a .txt file named "get-pip.py".
-   Go to your command line and navigate into your Downloads folder using the command **cd "path"**
The path here can be found by opening File Explorer, going to the Downloads folder and copying the path from the top search bar (Fig. 5).
The overall command in this case would look like this:

cd "C:\Users\Steve\Desktop"

-   Run the command **python get-pip.py**
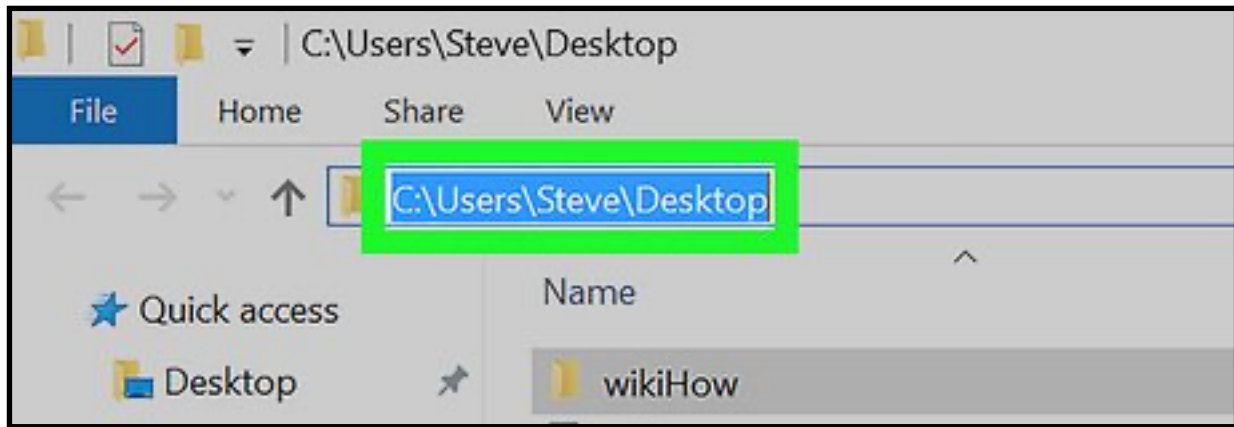-   Check that the pip is installed using the command **pip3 --version**

Fig. 5. Find the path to the file in Windows File Explorer.

3. After pip is installed, in the terminal, run two commands:

py -m pip install dash plotly pandas dash-bootstrap-components python-dateutil
py -m pip install kaleido numpy
This command installs the list of libraries required by Medi_Vizz. It will take a minute to run and gives a large output (part of the output is shown in Fig. 6).

Fig. 6. Installing the list of libraries required by Medi_Vizz. Red arrows indicate the command entered.

Starting the Medi_Vizz application

Download the zip folder "Medi_Vizz.zip" with the code and example filed from the GitHub link https://github.com/PlanetWyh/Medi_Vizz
To start the application, you must extract filed from the folder, navigate to it through the command line, and start the application using Python.
You can try to start the script through file explorer by double-clicking on the "dash_app.py" file in the Medi_Vizz folder. If that doesn't work for you, follow the steps below.

1. In the command line, navigate into the Medi_Vizz folder saved in Downloads (Fig. 7, upper red arrow):
-   Press the buttons combination "Windows" + "R".
-   Write **cmd** command to start the command line and press "Enter".
-   Find the path to the Medi_Vizz folder using Windows File Explorer. Go to the Downloads folder and then to the Medi_Vizz folder (can be downloaded as a zip folder from GitHub https://github.com/PlanetWyh/Medi_Vizz and should be unpacked by right clicking on it and choosing "Extract All…")
-   Find the path to the folder on top of the File Explorer and copy it (Fig. 5).
In your case, it should be something like this:
cd "C:\Users\Your_Username\Downloads\Medi_Vizz"
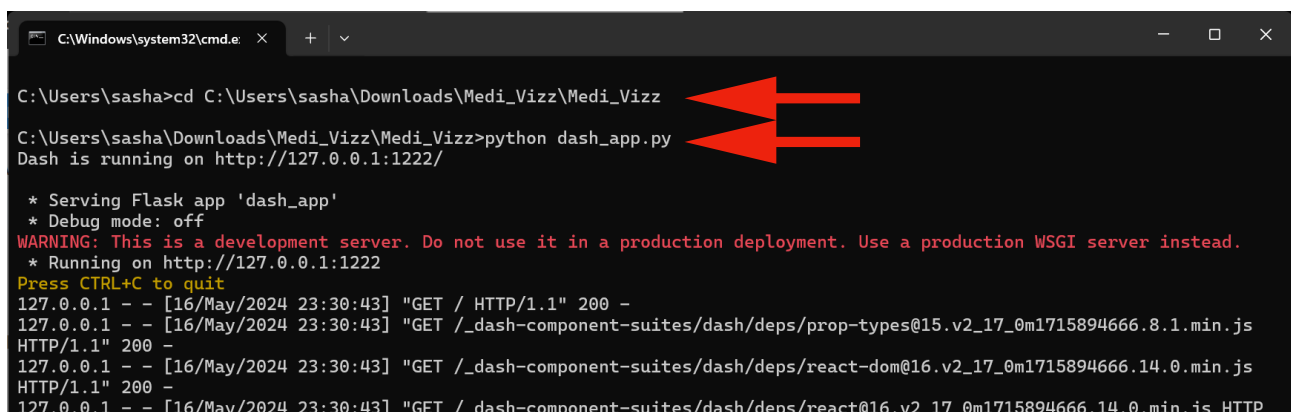-   Write the modified command from the previous step with your username into the command line (Fig. 6).

2. Check that you are in the correct folder by running the command:
dir
This short command displays the contents of your folder.

3. To start the application, run the command (Fig. 7):

python dash_app.py

Fig. 7. Starting the Medi_Vizz application from the command line. Red arrows point at the commands.

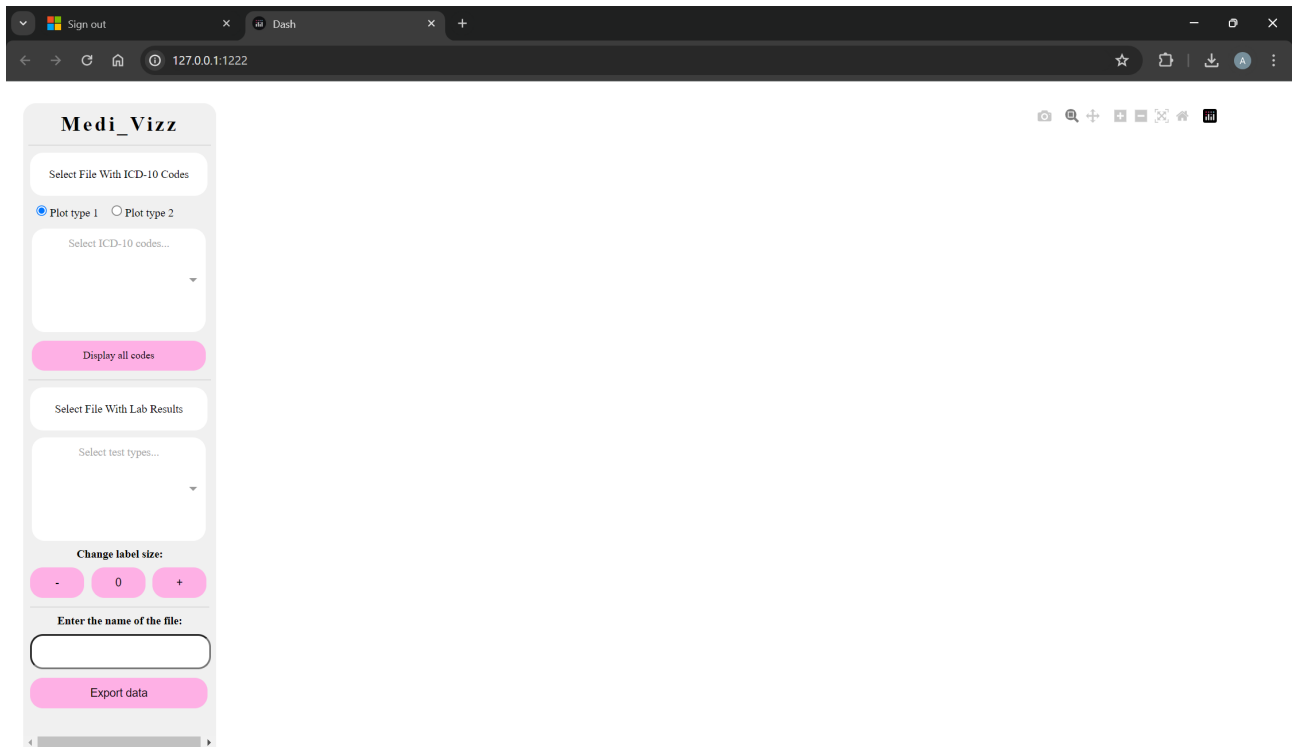4. Step 3 should open your default browser e.g. Google Chrome (Fig. 8).

Fig. 8. Medi_Vizz in the Google Chrome browser.

Data preprocessing

To use the application, you must format your data in a specific way. There are two example files in the zip folder.

ICD-10 file ("ICD-10-example-file.txt" in a zip folder)

The ICD-10 diagnoses file should include three columns:

- Numbers
- Dates: the column header should be **_Dates,_** while the column with codes should be labelled **_Codes_**. Dates should be written only in the format YYYY-MM-DD without time (e.g. 1978-12-19).
- Codes. The column header should be **_Codes_**. Codes should be written as in the ICD-10 classification.

Fig. 9A shows the data format example from the ICD-10-example file.txt, which you can download from GitHub.
Save the file in the **tab-delimited .txt format** (e.g. using Excel software).

ICD-10-example-file		
Numbers	**Dates**	**Codes**
1	2000-11-08	R36.4
2	2001-01-09	W45
3	2001-01-09	L74.80
4	2001-01-09	S82.4
5	2002-04-20	J80.3

LAB-MES-example-file			
Numbers	**Dates**	**TestTypes**	**TestResultNumeric**
1	2018-10-07	Test-1	0.1
2	2010-06-21	Test-1	0.5
3	2014-09-17	Test-1	0.02
4	2015-05-06	Test-1	0.9
5	2016-02-29	Test-2	6.7

Fig. 9. A) ICD-10 file example. **B)** Laboratory measurements file example.

Laboratory measurements file ("LAB-MES-example-file.txt" in a zip folder)

To make the file with laboratory results, you need to have four columns:

- Numbers
- Dates written in the format YYYY-MM-DD without time (e.g. 1978-12-19), the column labelled **_Dates_**
- A column with measurement types that can include text data labelled **_TestTypes_**
- A column with numeric test results labelled **_TestResultNumeric_**

Fig. 9B shows the data format example from the LAB-MES-example file.txt, which you can download from GitHub.
Save the file in the **tab-delimited .txt format** (e.g. using Excel software).

Medi_Vizz functionality

Fig. 10 shows the Medi_Vizz menu and functionalities associated with each element.
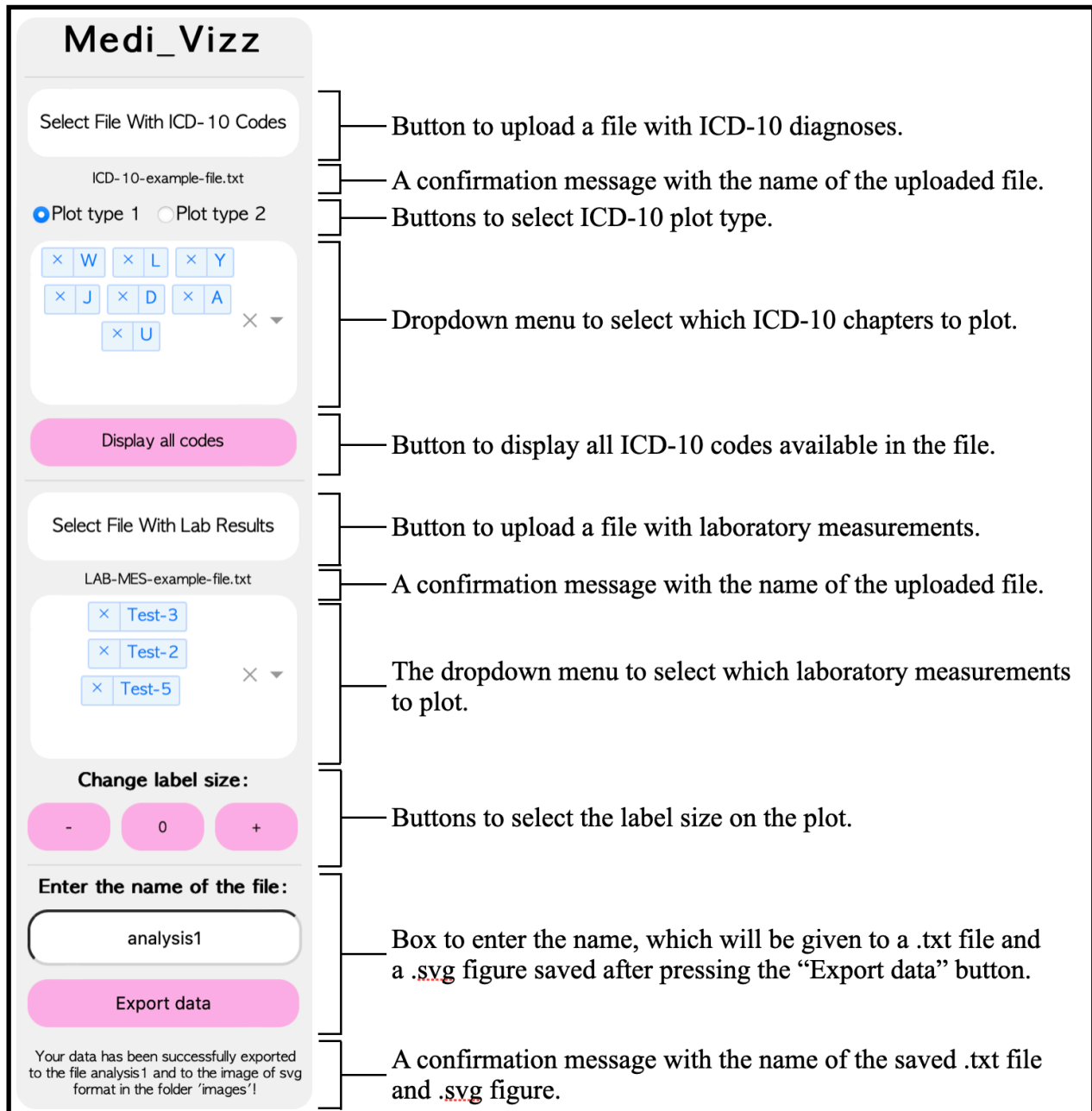

Fig. 10. Medi_Vizz menu layout with corresponding functionalities.

Medi_Vizz can generate two types of panels which only differ by how the ICD-10 diagnoses are represented. The first type of plot (Plot type 1) includes diagnoses as bubbles on the lines (Fig. 11). Line thickness shows how many diagnoses were assigned in one day, and diagnoses are displayed alphabetically.
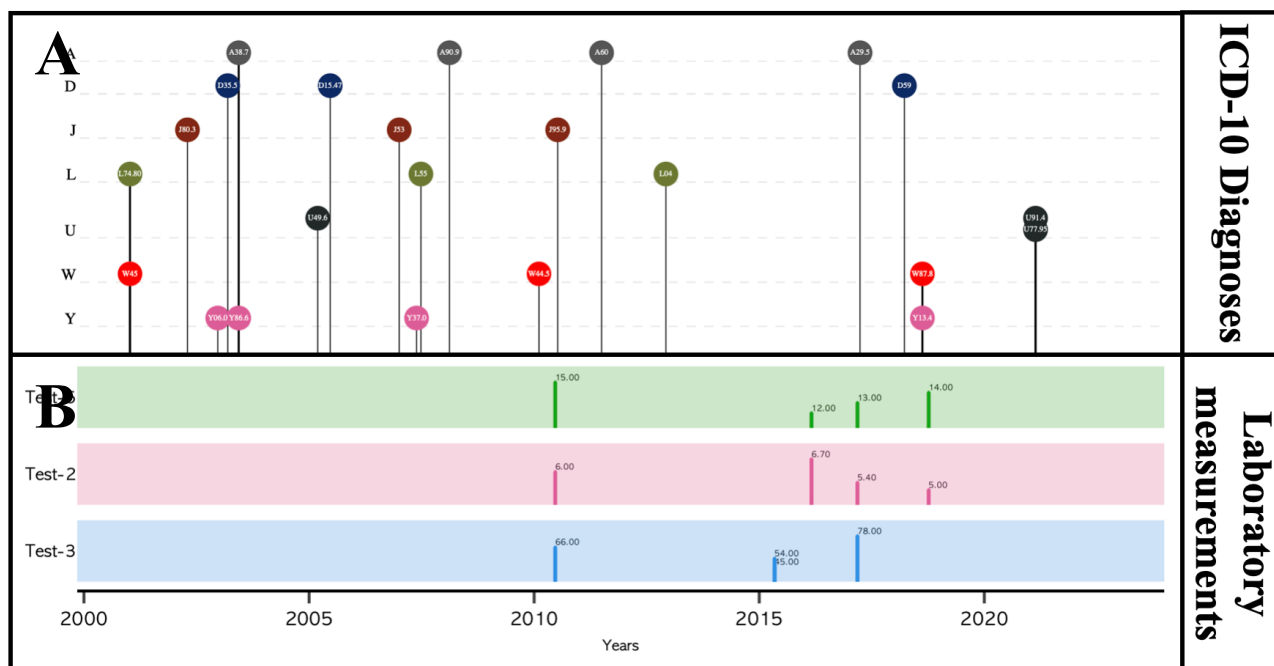
Plot type 2 shows diagnoses plotted in groups over lines (Fig. 12). The laboratory measurements panel stays the same between Plot Type 1 and Plot Type 2. Medi_Vizz considers the maximum and minimum results and distributes all the other values between them.

Fig. 11. Plot type 1. **A)** ICD-10 diagnoses part. **B)** Laboratory measurements part.
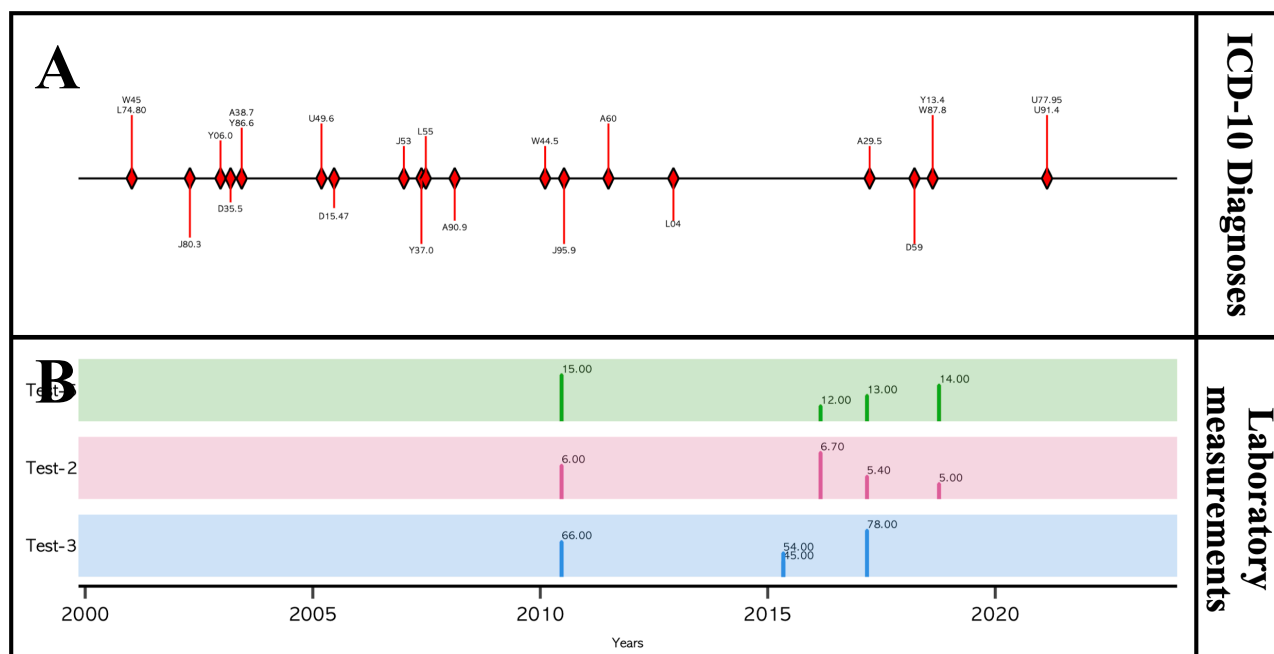

Fig. 12. Plot type 2. **A)** ICD-10 diagnoses part. **B)** Laboratory measurements part.

You can find a video example of how to use the application using this link: https://drive.google.com/file/d/1OoZ-JSpeAznnTq8mTncb2Wjbv_-UzwSE/view?usp=sharing