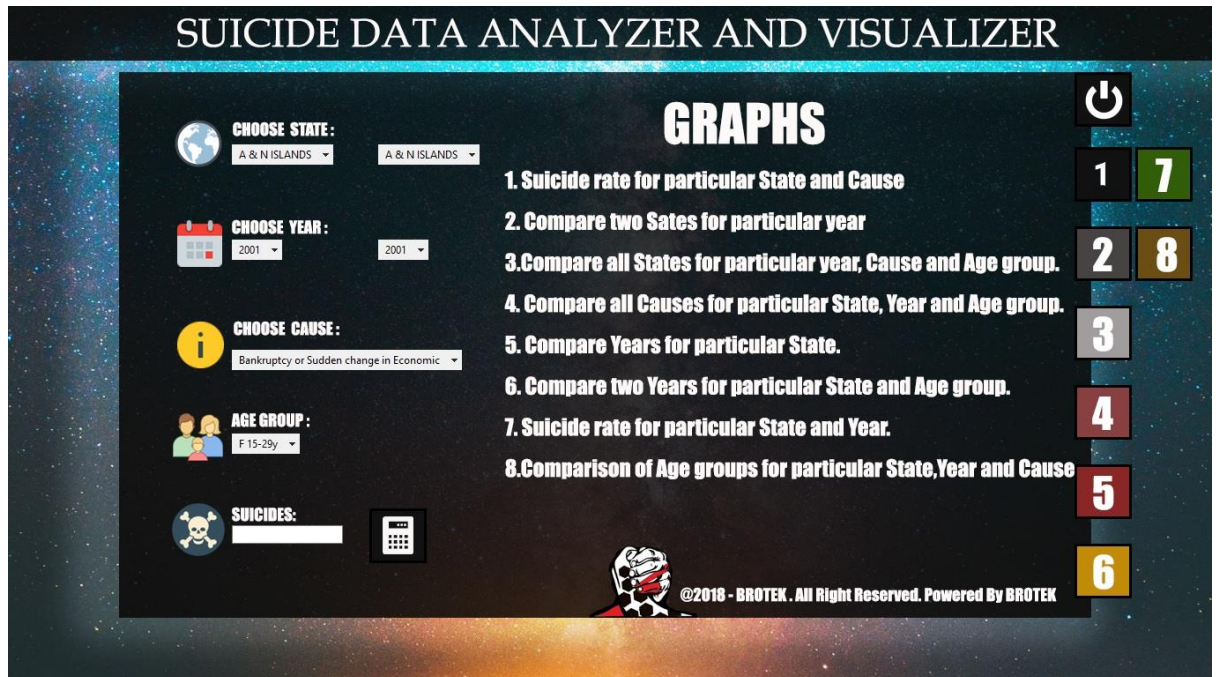# CHAPTER 3

DETAILS ABOUT PROJECT

Our goal about the project was quite simple, using data science to make a software that generates graphs which can help in making a quick report. For this we took Suicide cases in India as our data. By using several techniques, we generate different types of graphs which gives the user an accurate chart to make use of.

3.1 HARDWARE REQUIREMENTS

Computer hardware includes the physical parts or components of a computer, such as the central processing unit, monitor, keyboard, computer data storage, graphic card, sound card, speakers and motherboard. By contrast, software is instructions that can be stored and run by hardware.

Hardware is directed by the software to execute any command or instruction. A combination of hardware and software forms a usable computing system.

The software we have created is quite compact in nature and hence does not require much from hardware. Its requirements are low and can easily be fulfilled by today's computers. Its requirement being as follows: -

- 1 GB RAM

- A Display Screen

- Intel® Core™ i3 processor

- A keyboard and a mouse

This is the minimum requirement for Hardware, to have better performance the requirement can always be matched with higher specs.

Recommend hardware would be: -

- 4 GB RAM

- A Display Screen

- Intel® Core™ i5 processor

- 2 GB Graphics card

- A keyboard and a mouse

Any higher than this is redundant in nature for the software.

3.2 SOFTWARE REQUIREMENTS

As we are developing a small software and it has software requirement as follows: -

- **Python**

Python can be easy to pick up whether you're a first-time programmer or you're experienced with other languages. The following pages are a useful first step to get on your way writing programs with Python!

Python is developed under an OSI-approved open source license, making it freely usable and distributable, even for commercial use. Python's license is administered by the Python Software Foundation.

The Python Package Index (PyPI) hosts thousands of third-party modules for Python. Both Python's standard library and the community-contributed modules allow for endless possibilities.

The community hosts conferences and meetups, collaborates on code, and much more. Python's documentation will help you along the way, and the mailing lists will keep you in touch.

- **Pycharm IDE**

Python **PyCharm** is an integrated development environment (IDE) used in computer programming, specifically for the Python language. It is developed by the Czech company JetBrains.

It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django.

PyCharm is cross-platform, with Windows, macOS and Linux versions. The Community Edition is released under the Apache License, and there is also Professional Edition released under a proprietary license - this has extra features

PyCharm knows everything about your code. Rely on it for intelligent code completion, on-the-fly error checking and quick-fixes, easy project navigation, and much more.

- **ANACONDA**

Anaconda is a free and open source distribution of the Python and R programming languages for data science and machine learning related applications (large-scale data processing, predictive analytics, scientific computing), that aims to simplify package management and deployment. Package versions are managed by the package management system conda. The Anaconda distribution is used by over 6 million users, and it includes more than 250 popular data science packages suitable for Windows, Linux, and MacOS Pycharm IDE.

Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda distribution that allows you to launch applications and manage conda packages, environments and channels without using command-line commands. Navigator can search for packages on Anaconda Cloud or in a local Anaconda Repository, install them in an environment, run the packages and update them. It is available for Windows, macOS and Linux.

Navigator is automatically installed when you install Anaconda version 4.0.0 or higher.

- **Libraries**

The libraries are what we use in python as tools. These tools helps us to complete a project or a program with much easy. As these are only tools but not the programs themselves, we have to learn how to use them to actually implement them on our project and also identifying that if they are what we require to get the job done.

The libraries we have used are as follows:-

- **Tkinter**

Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit and is Python's de facto standard GUI. Tkinter is included with standard Linux, Microsoft Windows and Mac OS X installs of Python. The name Tkinter comes from Tk interface.

Tkinter was written by Fredrik Lundh. Tkinter is free software released under a Python license. Tkinter uses a Parent-child relation to establish one widget with another. For example: If you place a text label inside a frame, the frame is the parent of the label.

As above mentioned we have used this library to create a GUI for the user to actually be able to see what it is that the program provides and then selects what the user actually wants with it.

- **Pandas**

In computer programming, pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals.

We have used to this library to read massive amount of data we have and store it in a readable structure format. By doing so we became actually capable of manipulating the data according to our will and hence making it possible for us to implement the data science to it.

- **Matplotlib**

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, WxPython, Qt, or GTK+. There is also a procedural "Pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged. SciPy makes use of matplotlib.

As our main motive is to visually show the data we have tried our best to graphically show this. To generate the graph of our desired result we have to maintain the data in such form that it is readable by this library and for that we used the above mentioned library but still it is not readable by us hence we converted it into readable data type.

3.3 FEATURES OF PROJECT

The project's features can be described as follows: -

- **Taking User input-**

The central processing unit is the unseen part of a computer system, and users are only dimly aware of it. But users are very much aware of the input and output associated with the computer. They submit input data to the computer to get processed information, the output.

Some input data can go directly to the computer for processing. Input in this category includes bar codes, speech that enters the computer through a microphone, and data entered by means of a device that converts motions to on-screen action.

Some input data, however, goes through a good deal of intermediate handling, such as when it is copied from a source document and translated to a medium that a machine can read, such as a magnetic disk.

In either case the task is to gather data to be processed by the computer sometimes called raw data and convert it into some form the computer can understand.

- **Showing numerical value-**

After the dropdowns have been selected, the User can see the particular suicide victims for given data in numerical form in the main screen of the project.

As there are 2 dropdowns for same values i.e. State and Year only the first one will be considered for generating the numerical value of the victim.

The number is based on state, cause, year and Age group.
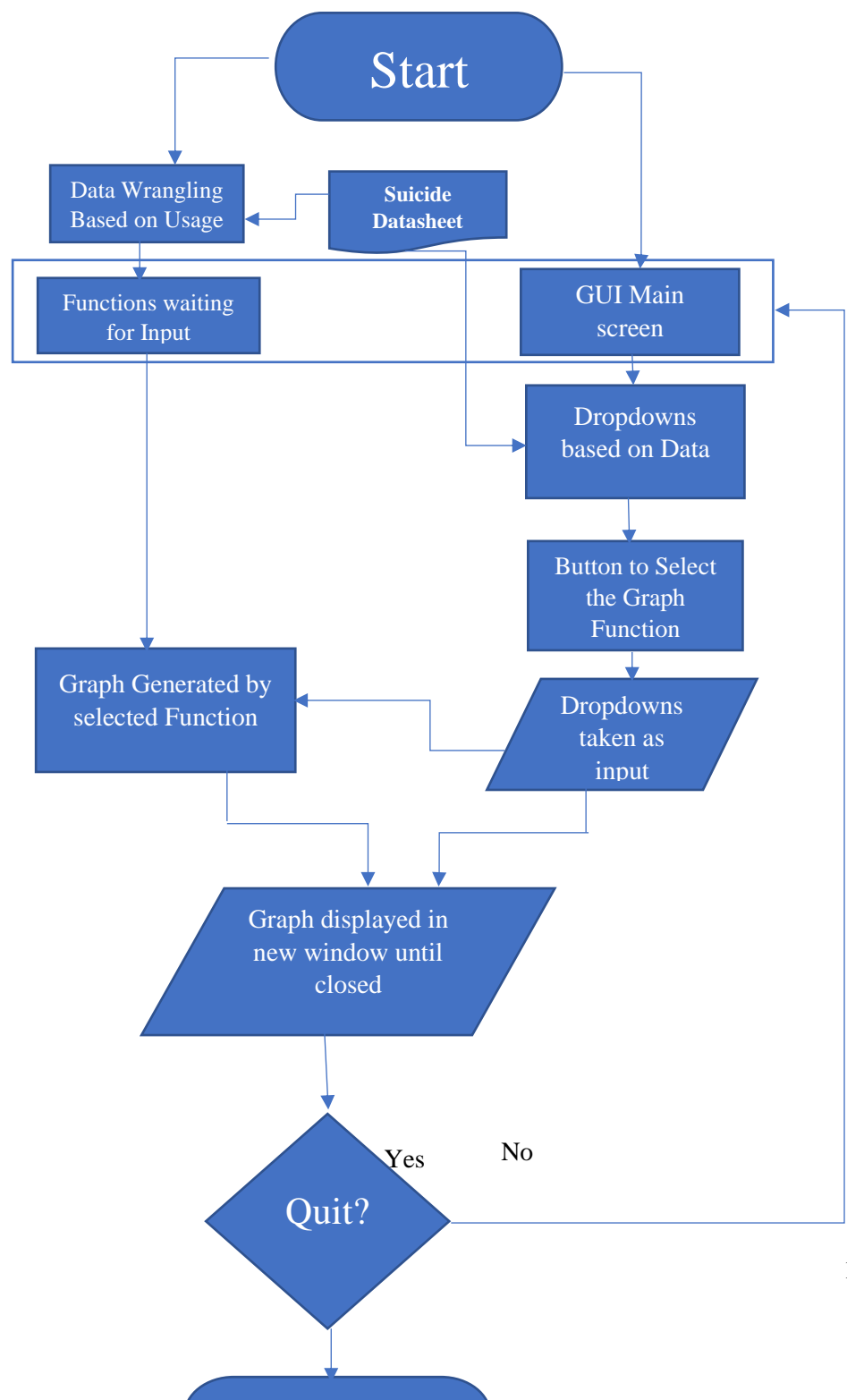
- **Generating different graphs –**

o A line graph showing suicide rates for Total Males and Female and Grand Total for User given State and Cause.

o 2- Bar Graph showing comparison b/w 2 User given states and Year for all Age group regardless of Cause.

o 3-Bar Graph to compare all states based on User given Cause, Age group and Year

o 4-Bar Graph showing victims of all causes for particular State, Year and Age Group

o 5-Bar Graph showing all Victims for a particular Cause and User given State for all Years in list

- 6- Bar Graph comparison b/w 2 years for same cause and same state given by the User

- 7-A pie chart which illustrates which Cause is more prominent in User given Year and State.

- 8- Bar Graph showing comparison of victims divided by Age group for User given State and Cause

CHAPTER 4

WORKING OF PROJECT

4.1 FLOWCHART DEPICTING WORKING OF PROJECT

```
                    ┌─────────────┐
                    │    Start    │
                    └─────────────┘

┌──────────────┐   ┌──────────────┐        ┌──────────────┐
│Data Wrangling│ ← │   Suicide    │        │  GUI Main    │
│Based on Usage│   │  Datasheet   │        │    screen    │
└──────────────┘   └──────────────┘        └──────────────┘

┌──────────────┐                           ┌──────────────┐
│Functions     │                           │  Dropdowns   │
│waiting       │                           │based on Data │
│for Input     │                           └──────────────┘
└──────────────┘
                                           ┌──────────────┐
                                           │Button to     │
                                           │Select the    │
                                           │Graph Function│
                                           └──────────────┘

┌──────────────┐                           ┌──────────────┐
│Graph         │                           │  Dropdowns   │
│Generated by  │ ←                         │  taken as    │
│selected      │                           │   input      │
│Function      │                           └──────────────┘
└──────────────┘

               ┌──────────────┐
               │Graph displayed│
               │in new window  │
               │until closed   │
               └──────────────┘

                    Yes        No
               ┌──────────────┐
               │    Quit?      │
               └──────────────┘
```

4.2 ALGORITHM OF PROJECT
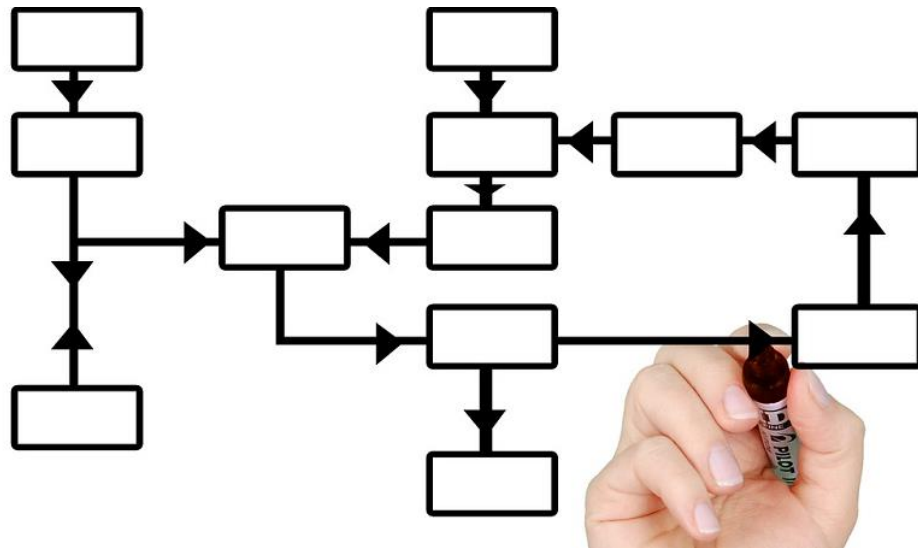
1. On the start of program, call main frame where all widgets are set from Main_screen.py file.

2. User decides which Graph to plot from the list, the list of graph that can be plotted are mentioned above.

3. Set drop downs according to graph that the user wants to generate.

4. The dropdown widget sends selected value as an input of graph functions present in functions.py file

5. Click on the specific side button to plot particular graph

6. On the click of button particular graph function works by taking inputs from step 4.

7. New plot window generated where graph is shown.

8. After completion of work close graph window and control again goes to main frame.

9. If want to continue again start from 2 point or go to next step.

10. To quit program, click on power-off button which calls quit function.

11. Quit function generates yes or no dialog box.

12. If you click on yes it destroys main frame and the program ends but if you select no control again goes to main frame.

4.3 EXPLANATION OF ALGORITHM

On starting of our program, the First thing to happen is generation of the Main Screen of our program. This Main Screen holds all inputs and buttons. The Dropdown created with help of data frame generated. Hence the program reads the csv file and with some techniques puts them into dropdowns.

User can select this Dropdown to give input based on the graph User wants to generate. After the User has selected the Dropdown to be input. User can click the on Button in Main Screen to generate graph. The input is taken by the function associated with particular button. This generates a customized graph.

A New Window is created by the program to display the graph. This New Window has complete control now and hence the main screen can't access by the User until this window is closed. After closing of the New Window, the User can once again use the Main Screen to generate graphs. If the User wants to quits. It can simply click on Quit insignia on the upper right corner of program which will pop up a message box that asks for confirmation of quitting.

CHAPTER 5

FIGURES OF PROJECT
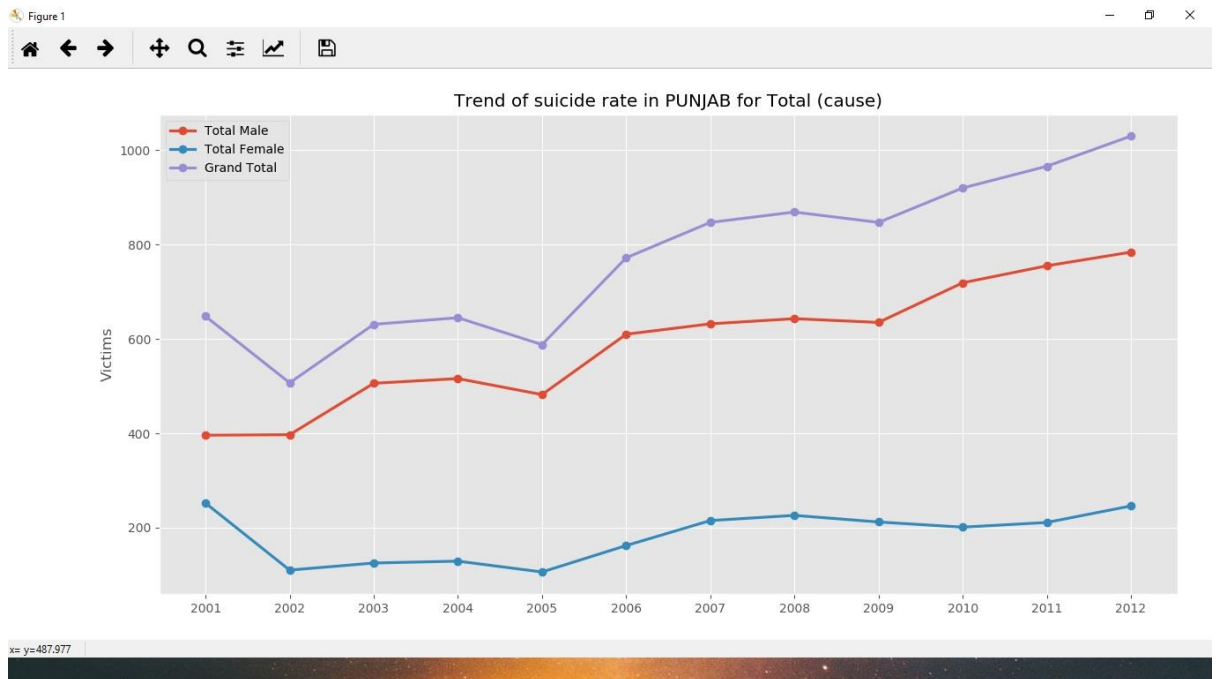
Fig 1. Line Graph for Punjab for all Causes

This graph indicates that the suicides rates have gone up significantly in females in Punjab from 2001-2012.and the suicides rate for males have not been affected much since 2001.and thus the total suicide rate have gone up in Punjab.
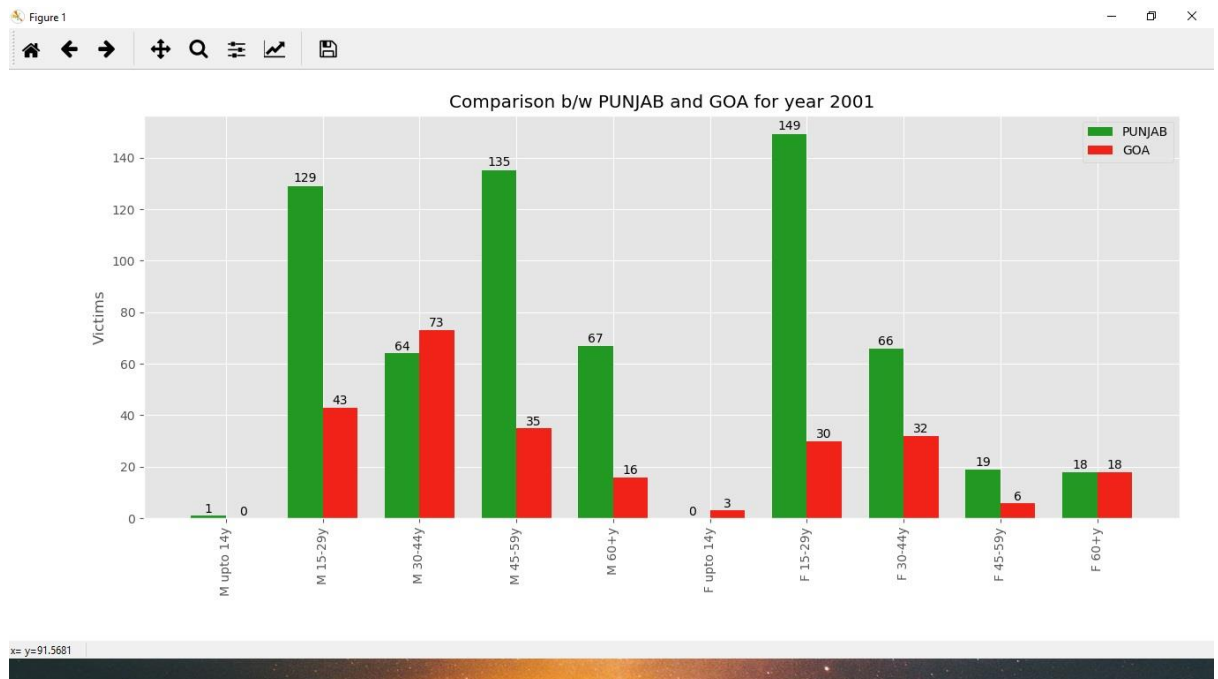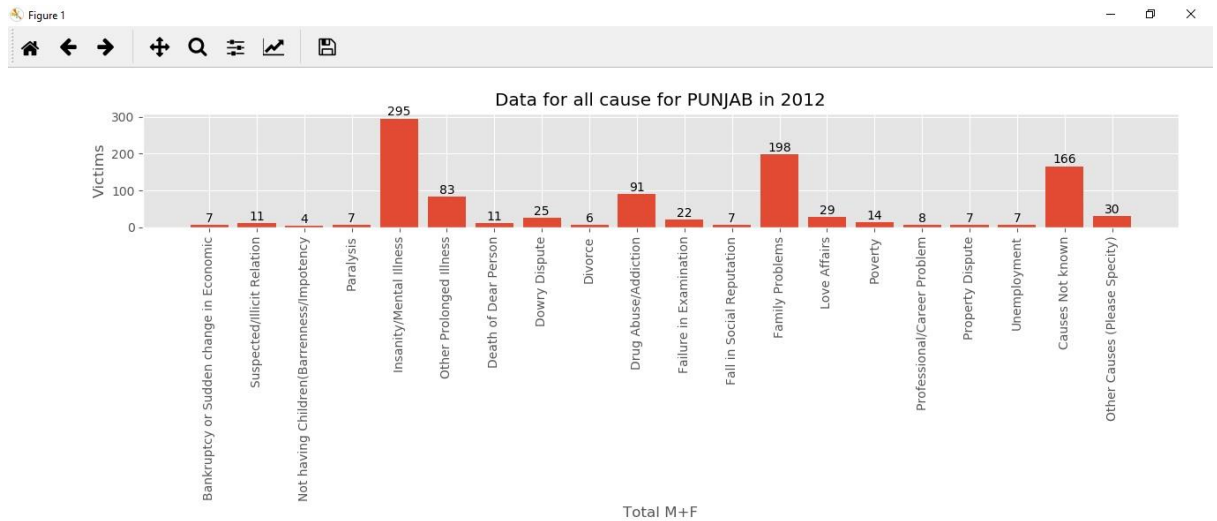
Fig 2. Bar Graph comparing b/w Punjab and Goa in Year 2001

This graph indicates that the Punjab have more suicides rate than Goa. It also correlates to the higher population in Punjab and then the total suicide rate has gone up in Punjab.
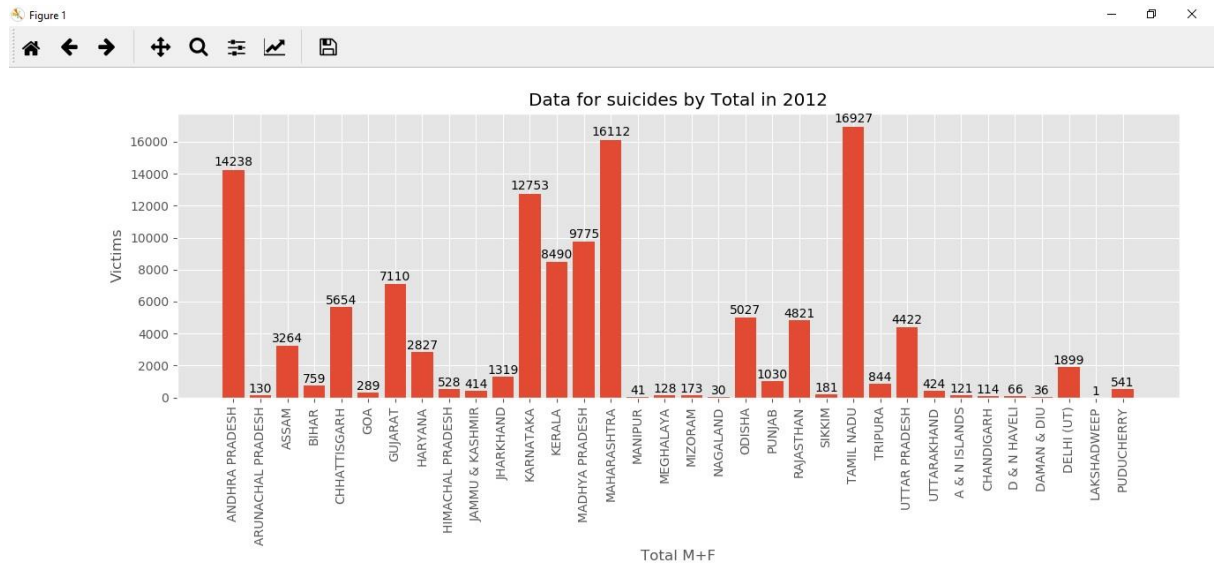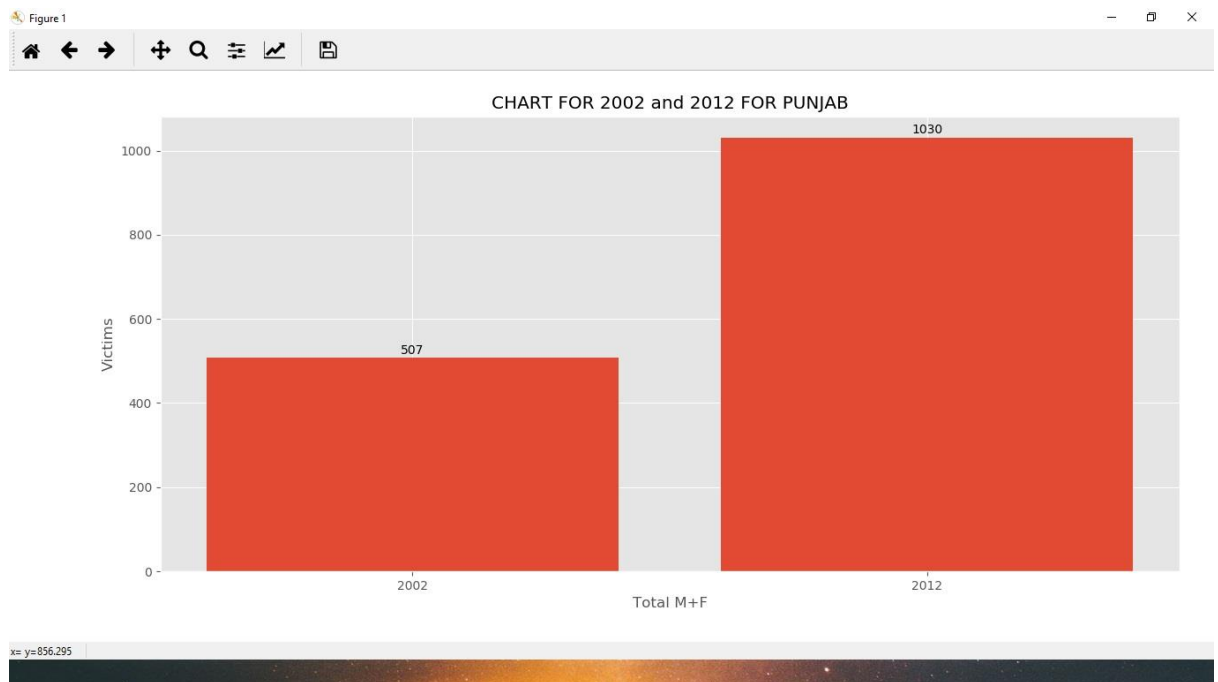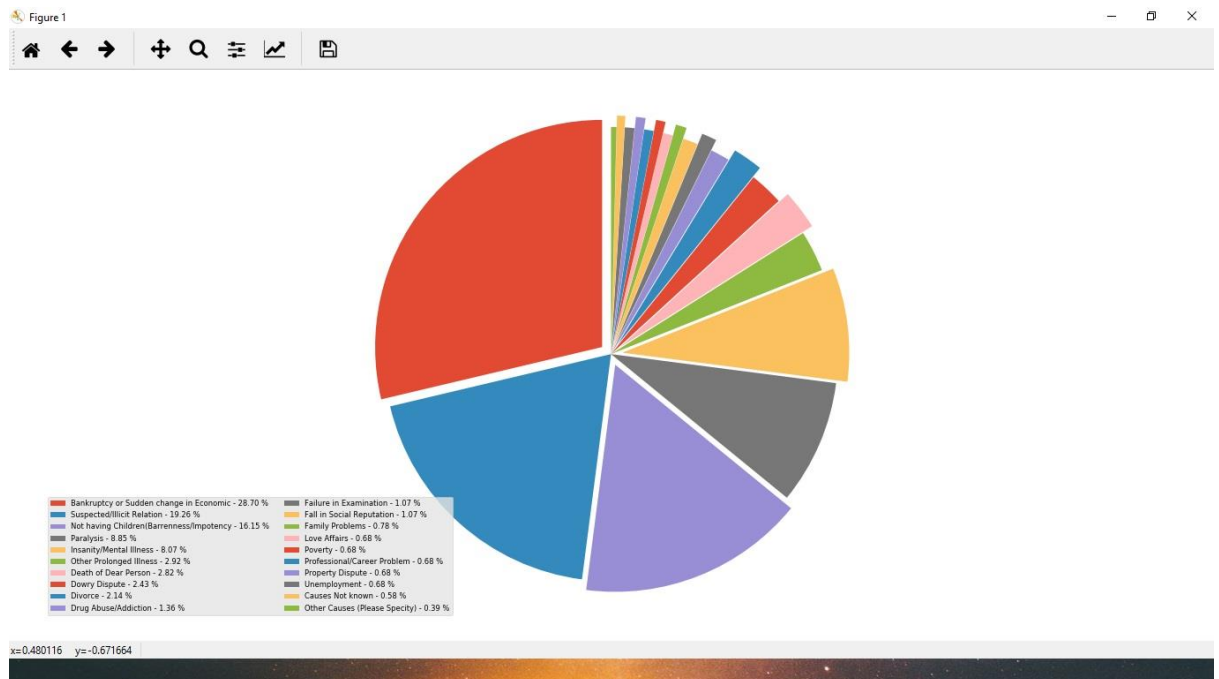
Fig 3. Data for all states for all Causes in Year 2012

This graph indicates that the Punjab have more suicides rate than Goa. It also correlates to the higher population in Punjab and then the total suicide rate has gone up in Punjab.

Fig 4. Bar Graph for all causes for Punjab, 2012

This graph indicates that the Punjab have more suicides rate than Goa. It also correlates to the higher population in Punjab and then the total suicide rate has gone up in Punjab.
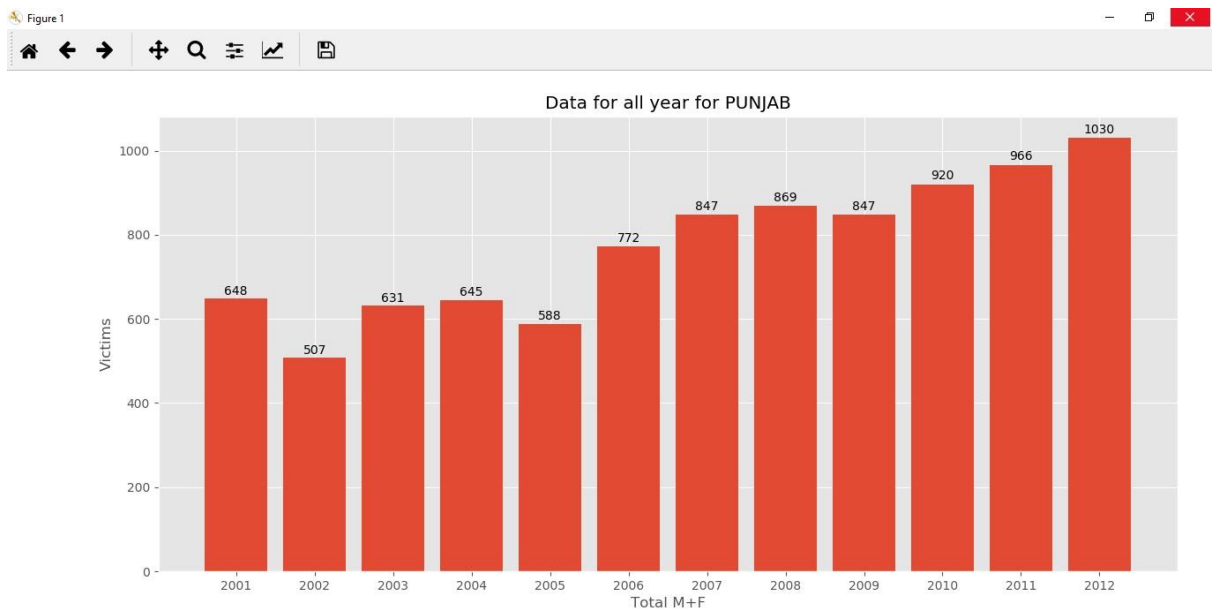
Fig 5. Bar Graph showing no. of suicide Victims from year 2001-2012 in Punjab

This graph indicates that the Punjab have more suicides rate than Goa. It also correlates to the higher population in Punjab and then the total suicide rate has gone up in Punjab.
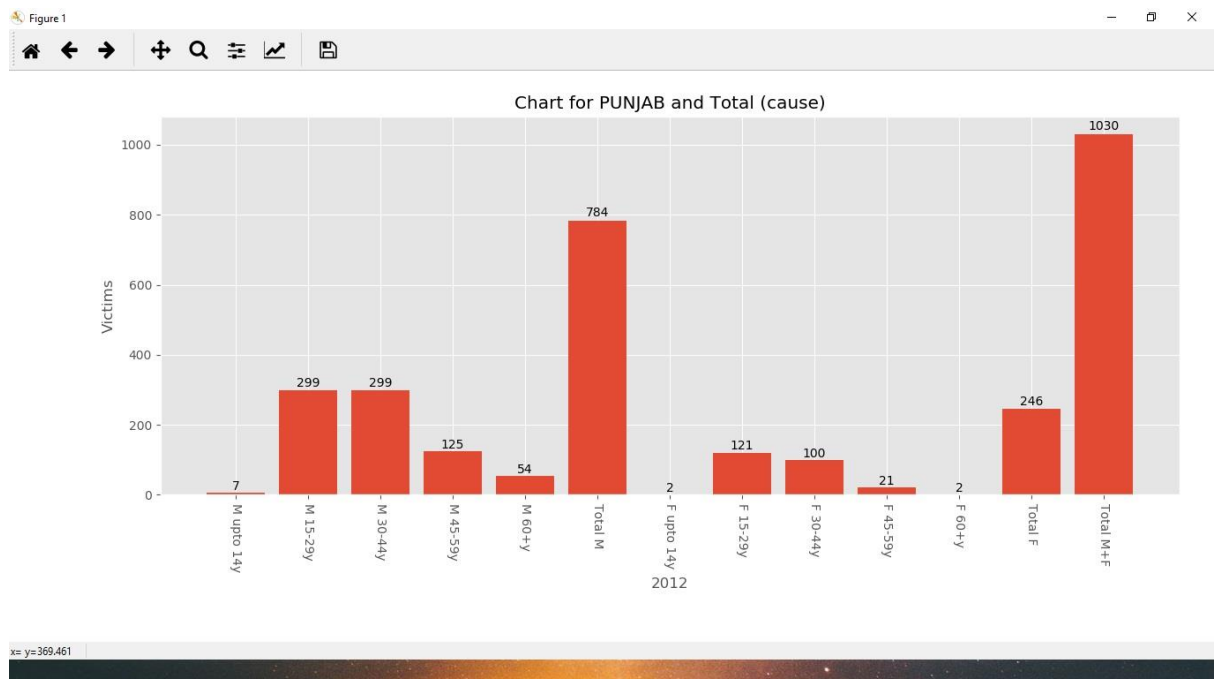
**Fig 6.** Bar Graph showing difference between 2 years for same
cause, Punjab

This graph indicates that the Punjab have more suicides rate than Goa. It
also correlates to the higher population in Punjab and then the total suicide
rate has gone up in Punjab.

Fig 7. A Pie Chart showing which causes contributes to suicide the most for Particular state and Year.

This graph indicates that the Punjab have more suicides rate than Goa. It also correlates to the higher population in Punjab and then the total suicide rate has gone up in Punjab.

Fig 8. A Bar graph divided by age group showing which age group has most victims in particular state, Punjab and for which cause i.e. in this case Total

This graph indicates that the Punjab have more suicides rate than Goa. It also correlates to the higher population in Punjab and then the total suicide rate has gone up in Punjab.
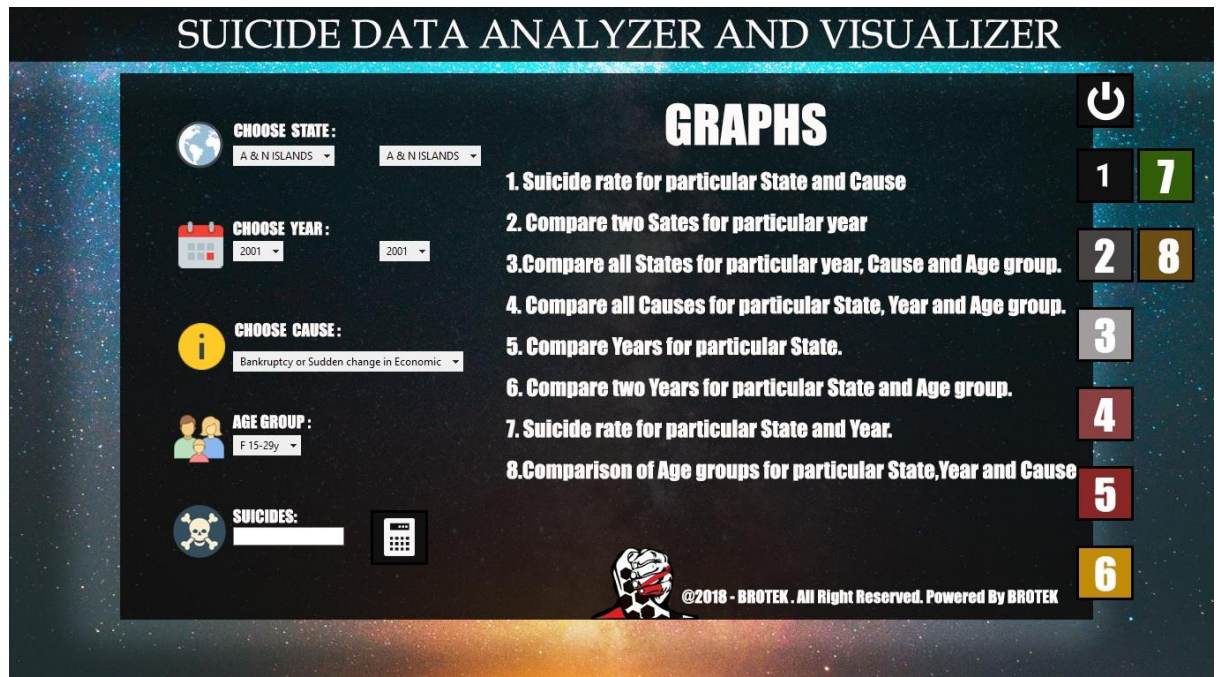
Fig 9. View of our project's Main Screen along with the dropdowns and all the buttons present.

This is the main GUI of the program. This has several dropdowns which gives accurate results of victims of suicides, the user can see which graph will be generated from which button. The first 2 rows have extra dropdowns for comparisons and contrast.
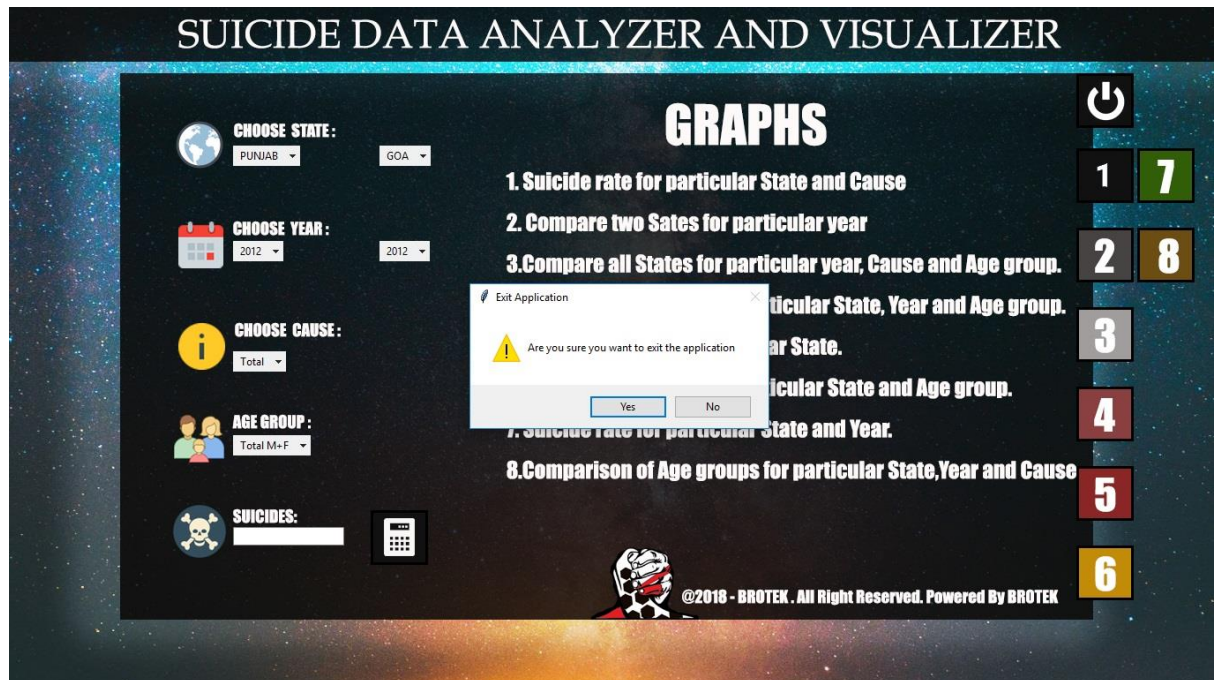
Fig 10. A quit confirmation message box for quitting

For quitting confirmation there is a popup message box which ask for confirmation.

CHAPTER 6
CONCLUSION

By using data visualization techniques and several basic data science techniques we have created this software which enables a user to have in-depth knowledge in graphical format which makes the user to understand the result and effect the data and also determine the causation behind it.

CHAPTER 7

REFERENCES

https://matplotlib.org/gallery/index.html

Used for help in graphs

https://jetbrains.com/pycharm/download/

IDE for using Python

http://effbot.org/tkinterbook/

Used for help in GUI

https://www.anaconda.com/download/

Used for adding library in python

https://pandas.pydata.org/pandas-docs/stable/tutorials.html

Used for data manipulation techniques

https://www.python.org/downloads/

Used to get latest version of Python