**CHAPTER 1**

**INTRODUCTION**

**1.1 PROBLEM DEFINITION**

Nowadays, we often face difficulties in understanding some difficult words in English. Normally we depend on online. It is difficult to find the meaning of these words without an internet connection. Many people cannot find the right dictionary when needed. Also, in this online world, there are some places where internet is unavailable. So, the people who are living there will face difficulty to understand the meaning of the word, opposites and synonyms.

**1.2 OBJECTIVES**

Through this application,

* Using GUI, the user can enter the word for which the meaning is needed.
* The user can also hear an audio output of the entered word.
* The meaning is displayed along with an audio output.
* The antonym of the word is displayed along with an audio output.
* The synonym of the word is displayed along with an audio output.

**1.3 METHODOLOGY TO BE FOLLOWED**

* In this project we are using GUI with the help of Tkinter module for an interactive User Interface.
* This uses available libraries in python.
* This uses text-to-speech attribute.
* It uses the nltk package.
* It uses json file.

**1.4 EXPECTED OUTCOMES**

* Through this application, user can get the meaning, antonym and synonym at the same time when the user enters the word.
* The user can also hear the audio output of the word to understand the correct pronunciation of the word.
* The user can also hear the audio output of the meaning, antonym and synonym of the word.
* Since this application works on offline, the user can easily get the required data in a short time.

**1.5 HARDWARE AND SOFTWARE REQUIREMENTS**

**Hardware**:

o Operating System: Either Windows 10 or higher

o Processor: x86 64-bit CPU (Intel / AMD architecture)

o 4 GB RAM

o 5 GB free disk space

**Software**:

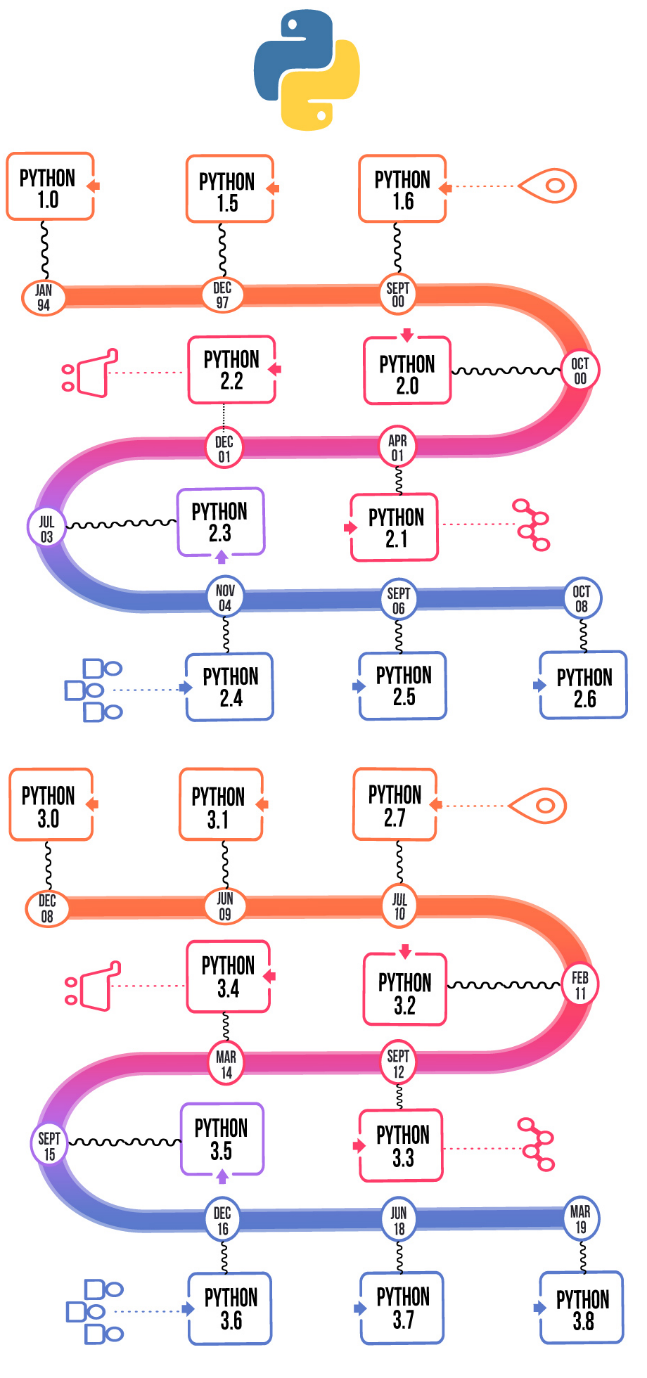
Either Python IDLE, PyCharm, Anaconda or VS Code

**CHAPTER 2**

**FUNDAMENTALS OF PYTHON**

**2.1 INTRODUCTION TO PYTHON**

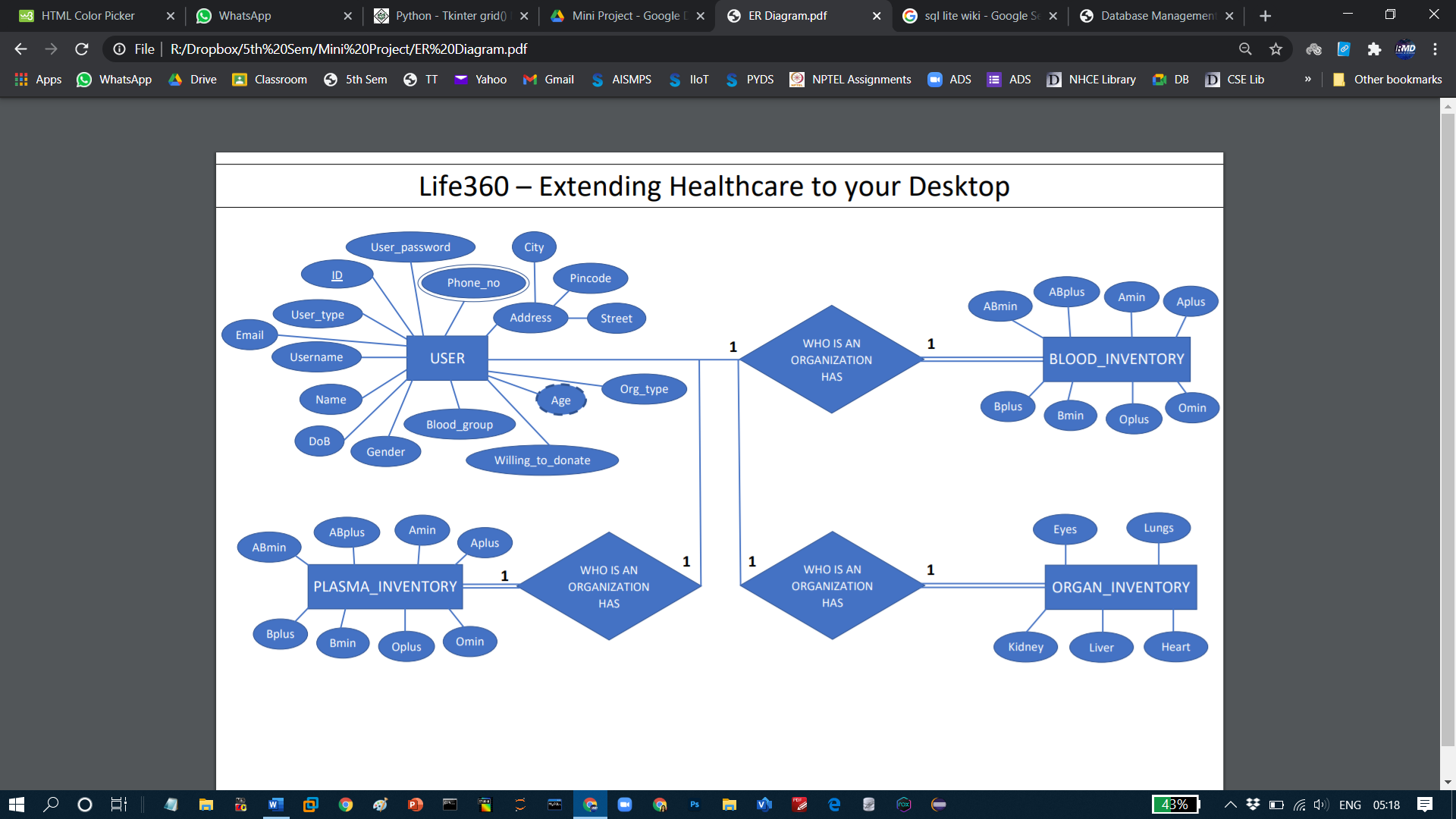
[Python](https://www.geeksforgeeks.org/python-programming-language/) is a commonly and extensively used general-purpose, high-level programming language. Guido van Rossum in 1991 was the founder of Python and was later developed by Python Software Foundation. It was primarily designed to emphasize on code readability, and its syntax allows programmers to express ideas in few lines of code. Python can be used for things like:



**Figure 2.1:** **Different versions of Python over the years**

**2.1.1 ENTITY-RELATIONSHIP (ER) MODEL**

The Entity-Relationship (ER) Model is an attractive high level conceptual data model. It has an entity which may be an object with a physical existence like a particular car, house, person or employee or it may be an object with a conceptual existence like an organization, a profession, or a university course. Each entity has attributes—the definite properties that characterize it. For example, a student entity may be described by the student’s name, age, address, USN etc.



**Figure 2.2:** **DBMS Component Modules**

**2.2 ADVANTAGES OF PYTHON**

* Presence of third-party modules
* Extensive support libraries (NumPy for numerical calculations, Pandas for data analytics, etc.)
* Open source and large active community base
* Versatile, Easy to read, learn and write
* User-friendly data structures
* High-level language
* Dynamically typed language (No need to mention data type based on the value assigned, it takes data type)
* Object-Oriented and Procedural Programming language
* Portable and Interactive
* Ideal for prototypes – provide more functionality with less coding
* Highly Efficient (Python’s clean object-oriented design provides enhanced process control, and the language is equipped with excellent text processing and integration capabilities, as well as its own unit testing framework, which makes it more efficient.)
* Internet of Things (IoT) Opportunities
* Interpreted Language
* Portable across Operating systems

**2.3 PYTHON LISTS**

Lists are used to store multiple items in a single variable.

Lists are one of 4 built-in data types in Python used to store collections of data, the other 3 are [Tuple](https://www.w3schools.com/python/python_tuples.asp), [Set](https://www.w3schools.com/python/python_sets.asp), and [Dictionary](https://www.w3schools.com/python/python_dictionaries.asp) all with different qualities and usage.

Lists are created using square brackets.

List items are ordered, changeable, and allow duplicate values.

List items are indexed, the first item has index [0], the second item has index [1] etc.

**2.4 PYTHON TUPLES**

A tuple in Python is similar to a [list](https://www.programiz.com/python-programming/list). The difference between the two is that we cannot change the elements of a tuple once it is assigned whereas we can change the elements of a list.

## Creating a Tuple

A tuple is created by placing all the items (elements) inside parentheses (), separated by commas. The parentheses are optional however, it is a good practice to use them.

A tuple can have any number of items and they may be of different types (integer, float, list, [string](https://www.programiz.com/python-programming/string), etc.).

**2.5 PYTHON SETS**

A set is an unordered collection of items. Every set element is unique (no duplicates) and must be immutable (cannot be changed).

However, a set itself is mutable. We can add or remove items from it.

Sets can also be used to perform mathematical set operations like union, intersection, symmetric difference, etc.

## Creating Python Sets

A set is created by placing all the items (elements) inside curly braces {}, separated by comma, or by using the built-in set() function.

It can have any number of items and they may be of different types (integer, float, tuple, string etc.). But a set cannot have mutable elements like [lists](https://www.programiz.com/python-programming/list), sets or [dictionaries](https://www.programiz.com/python-programming/dictionary) as its elements.

**2.6 PYTHON DICTIONARIES**

Dictionaries are used to store data values in key: value pairs.

A dictionary is a collection which is ordered\*, changeable and do not allow duplicates.

As of Python version 3.7, dictionaries are ordered. In Python 3.6 and earlier, dictionaries are unordered.

Dictionaries are written with curly brackets, and have keys and values:

## Dictionary Items:

Dictionary items are ordered, changeable, and does not allow duplicates.

Dictionary items are presented in key: value pairs, and can be referred to by using the key name.

**2.7 PYTHON FUNCTIONS**

A function is a block of code that performs a specific task.

Suppose, you need to create a program to create a circle and colour it. You can create two functions to solve this problem:

* create a circle function
* create a color function

Dividing a complex problem into smaller chunks makes our program easy to understand and reuse.

## Types of function

There are two types of function in Python programming:

* **Standard library functions** - These are built-in functions in Python that are available to use.
* **User-defined functions** - We can create our own functions based on our requirements.

**CHAPTER 3**

**FUNDAMENTALS OF TKINTER**

**3.1 INTRODUCTION**

Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUI methods, tkinter is the most commonly used method. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with tkinter is the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an easy task.

To create a tkinter app:

Importing the module – tkinter

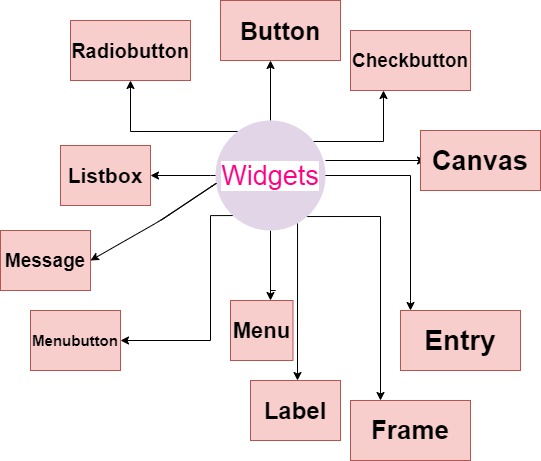
Create the main window (container)

Add any number of widgets to the main window

Apply the event Trigger on the widgets.

**3.2 WIDGETS**

There are various controls, such as **buttons**, **labels**, **scrollbars**, **radio buttons**, and **text boxes** used in a GUI application. These**little components** or controls of **Graphical User Interface (GUI)** are known as **widgets** in Tkinter.



**Fig 3.2 Widgets**

These are**19 widgets** available in Python Tkinter module.

**3.3 LABELS**

The Label is used to specify the container box where we can place the text or images. This widget is used to provide the message to the user about other widgets used in the python application.

There are the various options which can be specified to configure the text or the part of the text shown in the Label.

Example: anchor, bg, bd, fg, image, cursor, justify, wrap

**3.4 FRAME**

The Frame widget is very important for the process of grouping and organizing other widgets in a somehow friendly way. It works like a container, which is responsible for arranging the position of other widgets.

It uses rectangular areas in the screen to organize the layout and to provide padding of these widgets. A frame can also be used as a foundation class to implement complex widgets.

**3.5 BUTTONS**

The Button widget is used to add buttons in a Python application. These buttons can display text or images that convey the purpose of the buttons. You can attach a function or a method to a button which is called automatically when you click the button.

## Syntax

Here is the simple syntax to create this widget:

w= Button (master, option=value, …)

**CHAPTER 4**

**DESIGN**

**4.1 DESIGN GOALS**

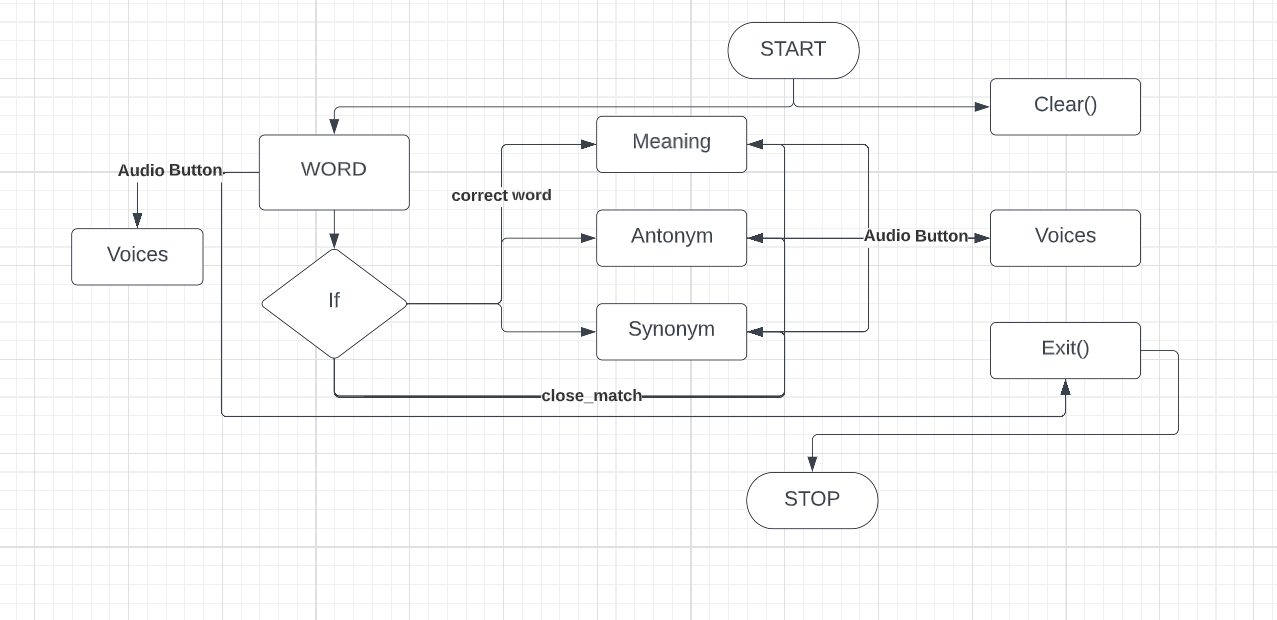
* This mini project has ensured that the user has an interactive and explorable environment.
* The interface is user friendly, simple to understand and has tried to ensure that there are no bugs.
* This project is fast and secure.
* Easy to access.

**4.2 ALGORITHM**

MAIN PAGE

* Enter the word
* If search button is pressed meaning, antonyms and synonyms is displayed.
* If mic button is pressed the word is said aloud.
* If the entered word is typed wrong, it asks the user whether the required word is the correct word instead of wrong word and displays the meaning, antonym and synonym of the correct word.
* If mic button of meaning is pressed, the meaning is said aloud.
* If mic button of antonym is pressed, the antonym is said aloud.
* If mic button of synonym is pressed, the synonym is said aloud.
* If clear button is pressed, it clears all the fields.
* If exit button is pressed, it exits the window.

**4.3 FLOWCHART**

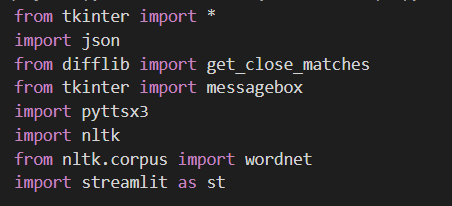
****

**Fig 4.3 FLOWCHART**

**CHAPTER 5**

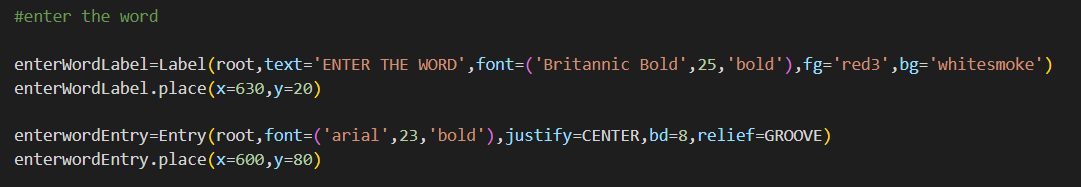
**IMPLEMENTATION**

**MODULE 5.1 LIBRARIES**

****

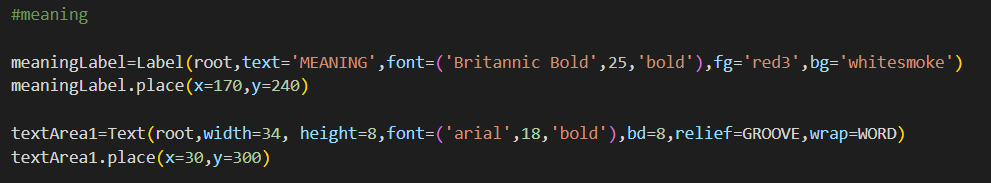
These are the libraries used in this project.

**MODULE 5.2 WORD ENTRY**



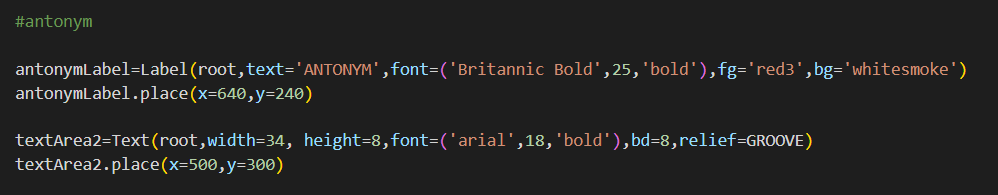
This code is used to create the field to enter the word.

**MODULE 5.3 MEANING**

****

This code is used to create the field to display meaning.

**MODULE 5.4 ANTONYM**

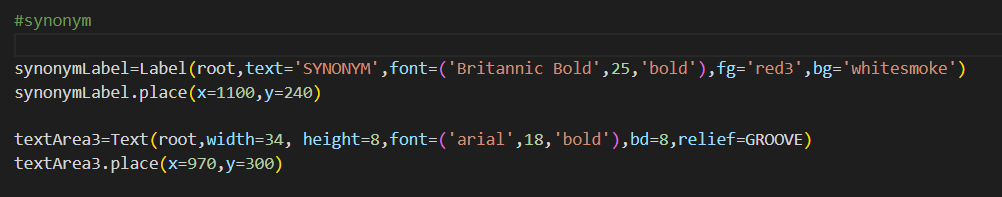
****

This code is used to create the field to display antonym.

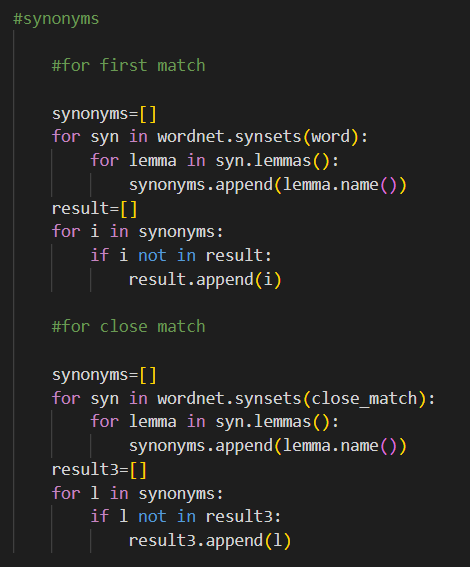
****

This code is used to access the antonym of the required word from the nltk library.

**MODULE 5.5 SYNONYM**

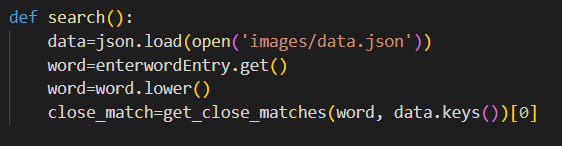
****

This code is used to create the field to display synonym.

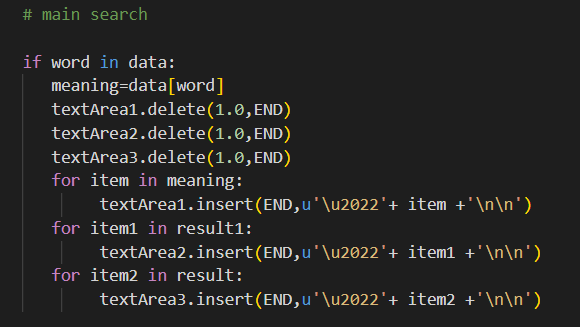
****

This code is used to access the synonym of the required word from the nltk library.

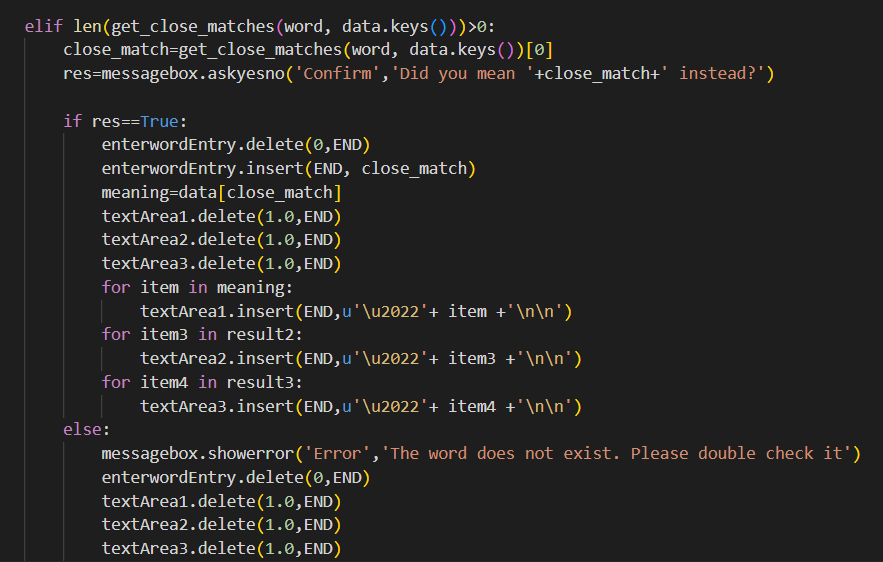
**MODULE 5.6 SEARCH**

****

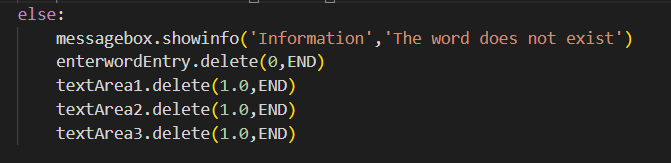
This code is used to access the required word for meaning from the json file.

****

This code is used to get the meaning, antonym and synonym of the correct word.

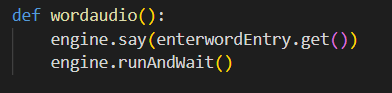
****

This code is used to get the meaning, antonym and synonym of the close match word.

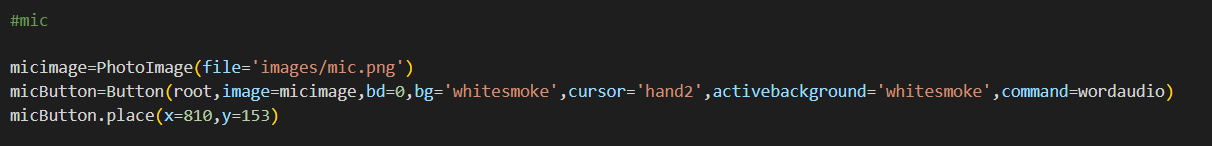
****

This code is used to display the error message when wrong word is entered.

**MODULE 5.7 WORD AUDIO**

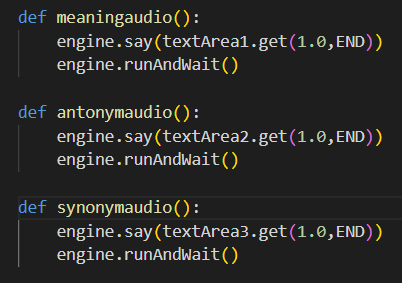
****

This function is used to give the audio output of the word.

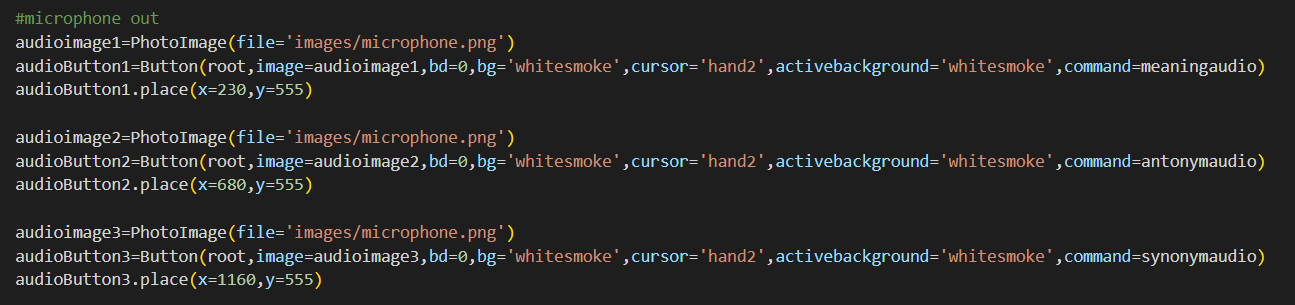
****

This code is used to create the mic button for the word.

**MODULE 5.8 MEANING, ANTONYM AND SYNONYM AUDIO**

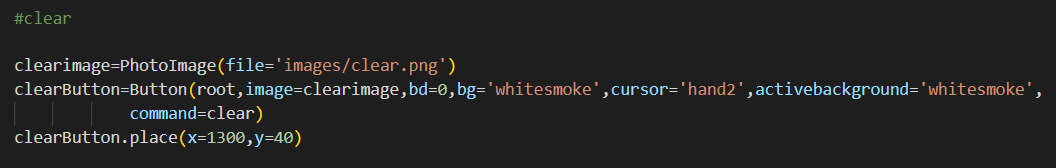
****

These functions are used to give the audio output of meaning, antonym and synonym.

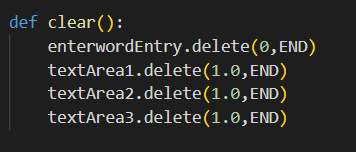
****

This code is used to create the mic button for meaning, antonym and synonym.

**MODULE 5.9 CLEAR**

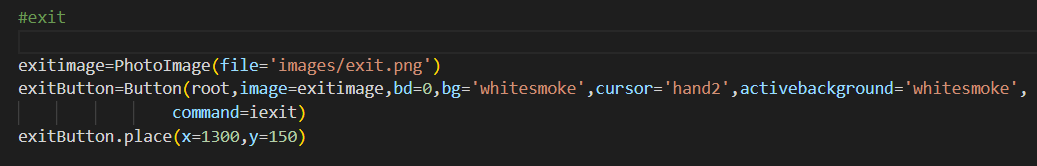
****

This code is used to create the clear button.

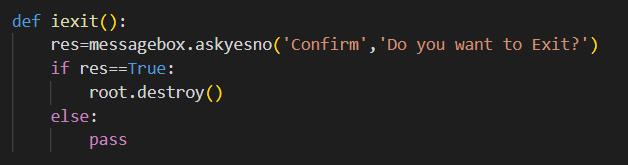
****

This code is used to clear the page.

**MODULE 5.10 EXIT**

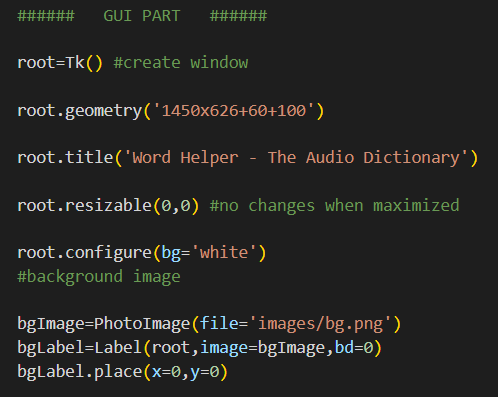
****

This code is used to create the exit button.

****

This code is used to exit the application.

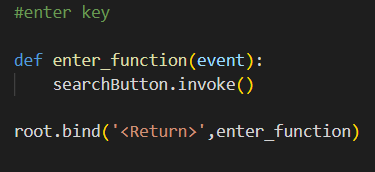
**MODULE 5.11 TKINTER**

****

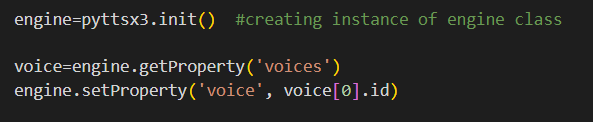
****

This code is used to create the GUI using tkinter library.

**MODULE 5.12 OTHER FUNCTIONS**

****

This code is used to make the enter key to submit the word.

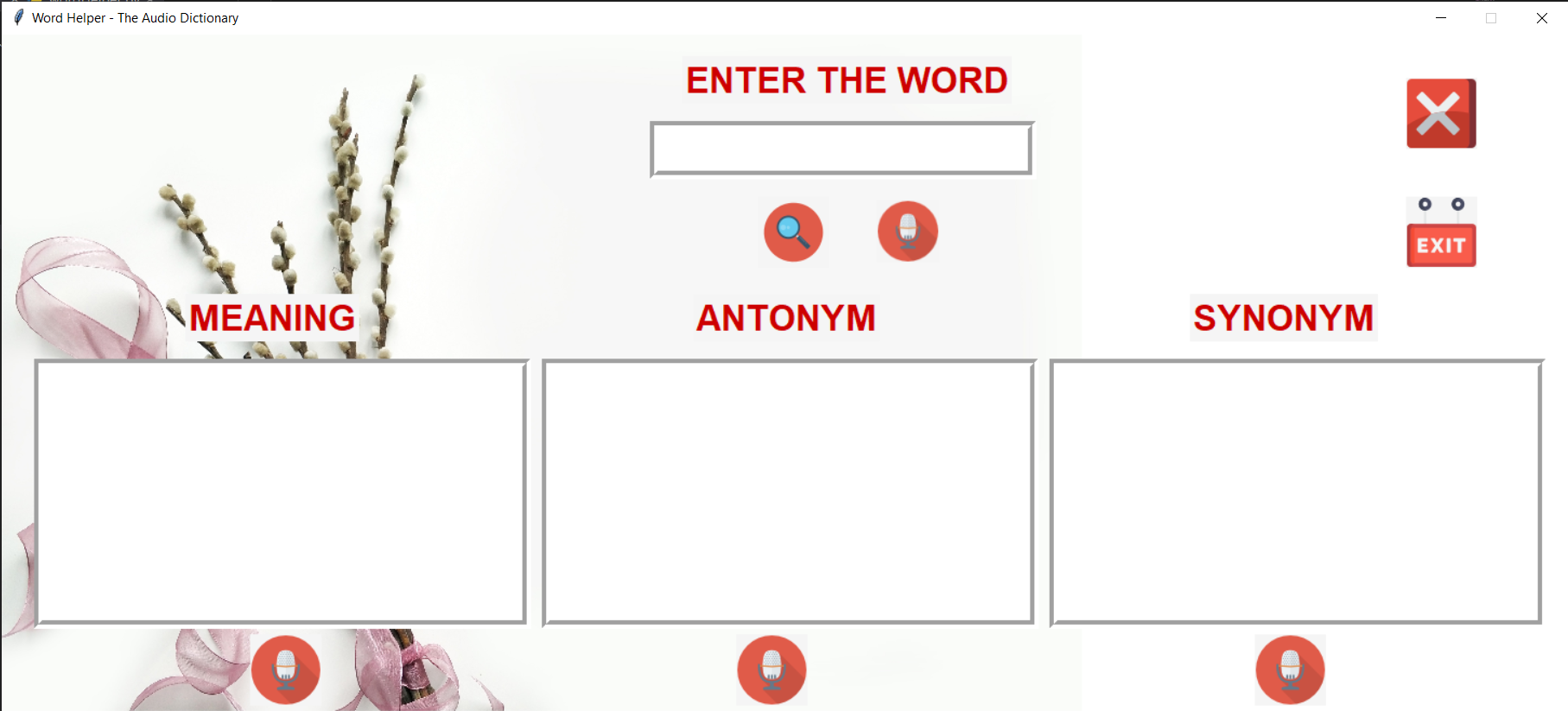
****

This function is used to access the voices.

**CHAPTER 6**

**RESULTS**

**6.1 MAIN SCREEN**

****

**Fig 6.1** **Main Screen**

This is the UI’s main screen.

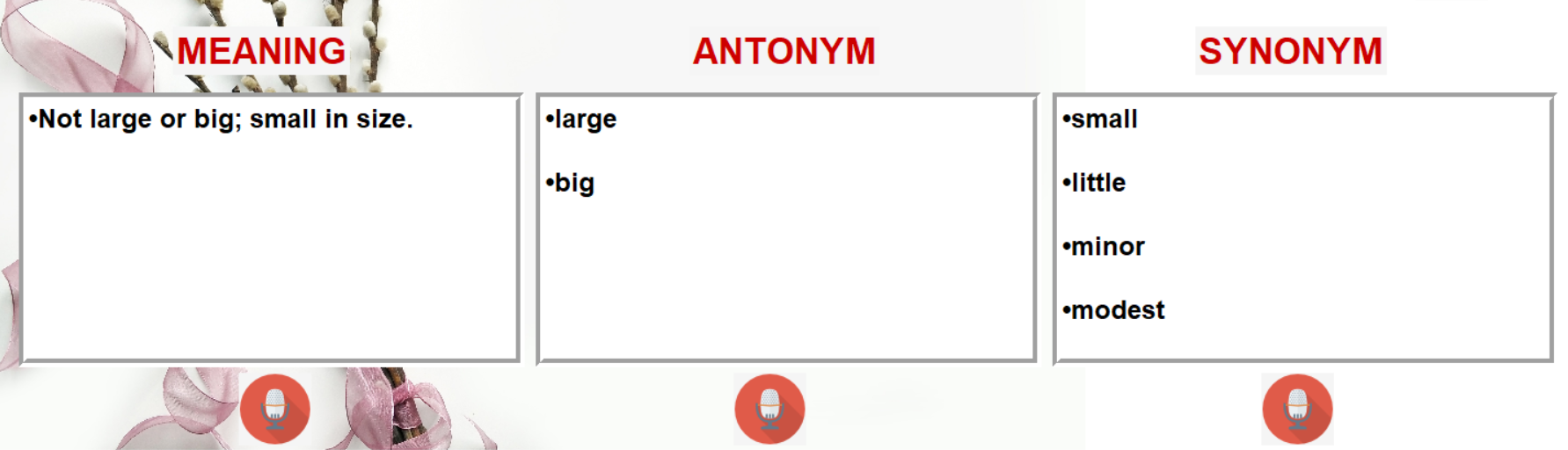
**6.2 WORD ENTRY**

****

**Fig** **6.2 Word Entry Field**

Here, we enter the word for which we need to find the meaning, antonym and synonym.

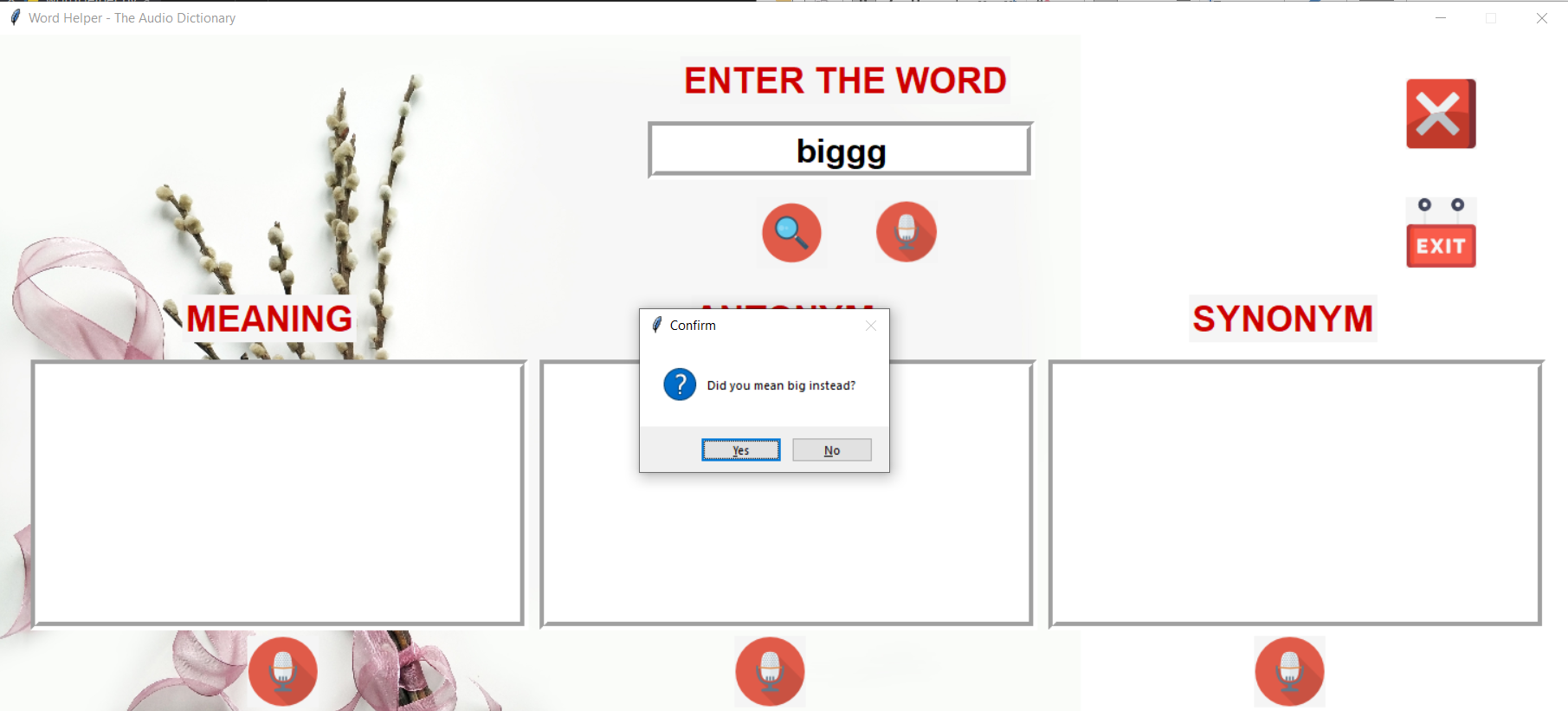
**6.3 MEANING, ANTONYM AND SYNONYM**

****

**Fig** **6.3 Meaning, Antonym and Synonym Field**

Here, meaning, antonym and synonym of the word is displayed.

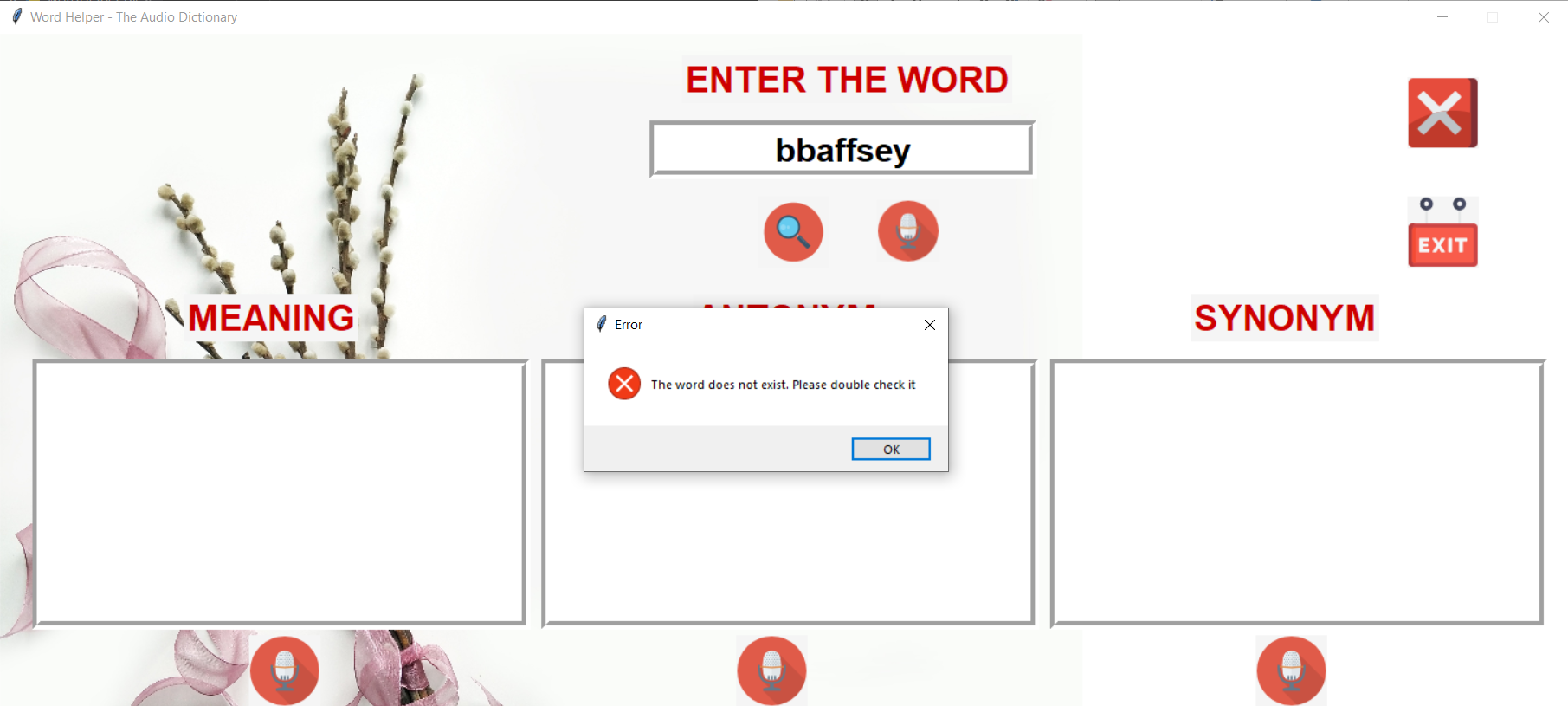
**6.4 WRONG WORD BUT NEARER TO CLOSER WORD**

****

**Fig** **6.4 Wrong Entry but Closer Word**

When a wrong word but a closer match word is entered, a messagebox is displayed.

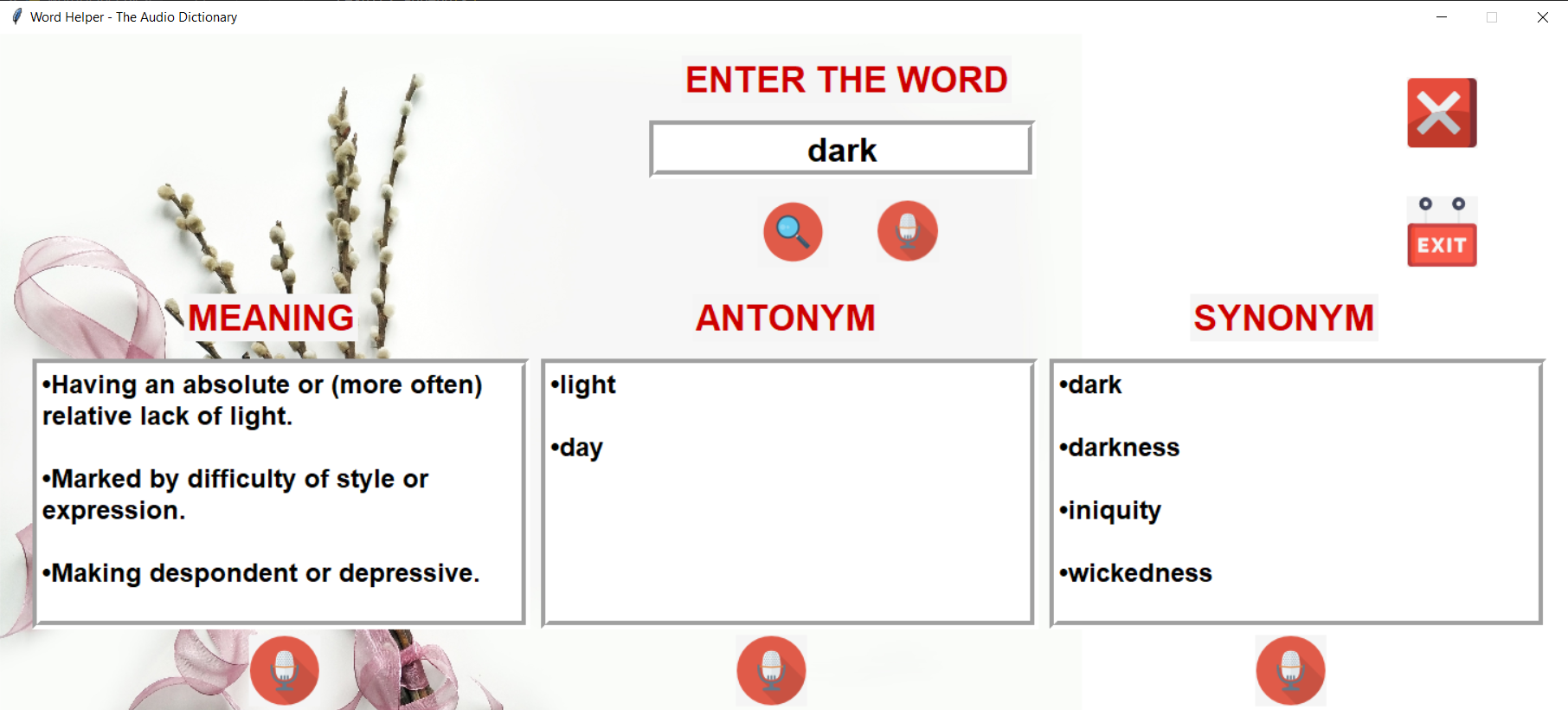
**6.5 WRONG WORD**

****

**Fig** **6.5 Wrong Word Entry**

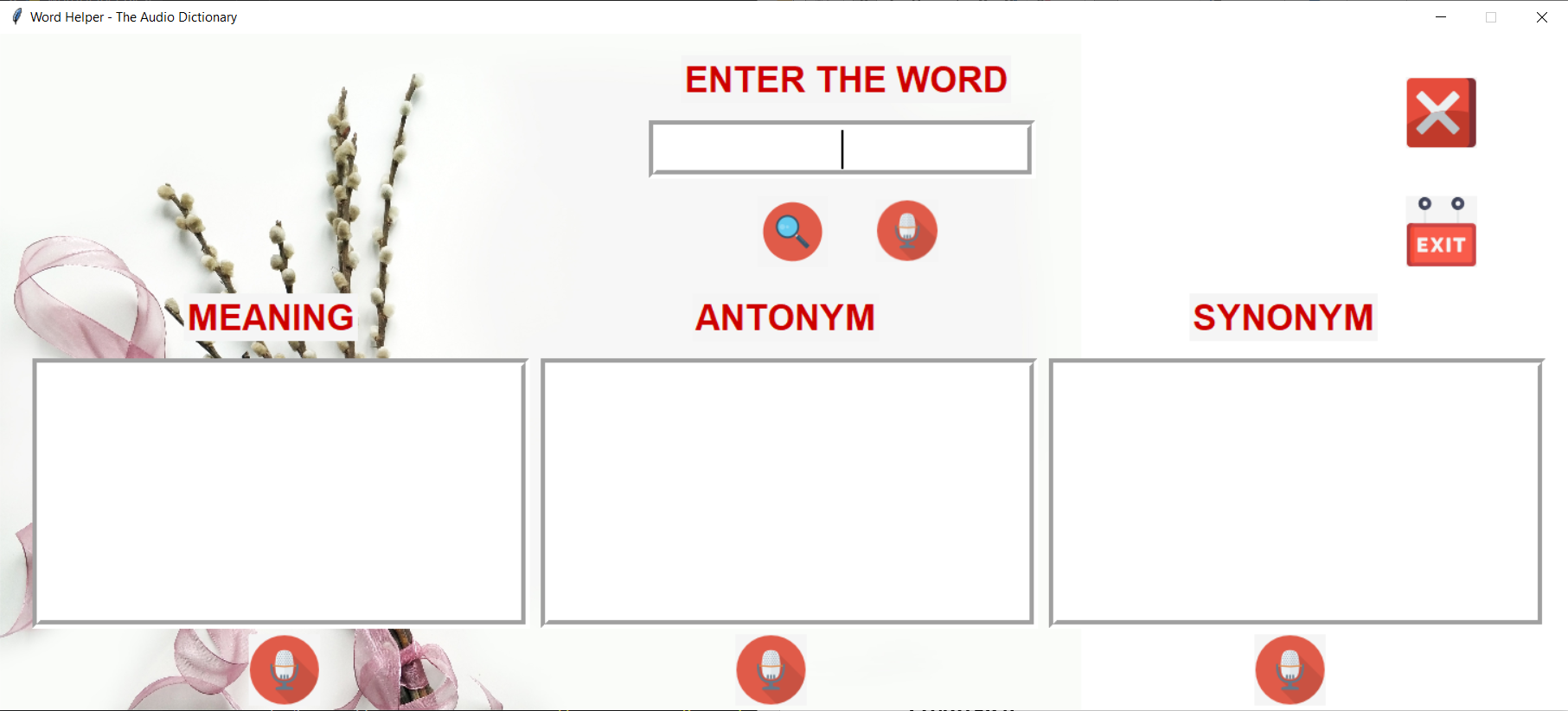
When wrong word is entered, it displays a messagebox.

**6.6 CLEAR**

****

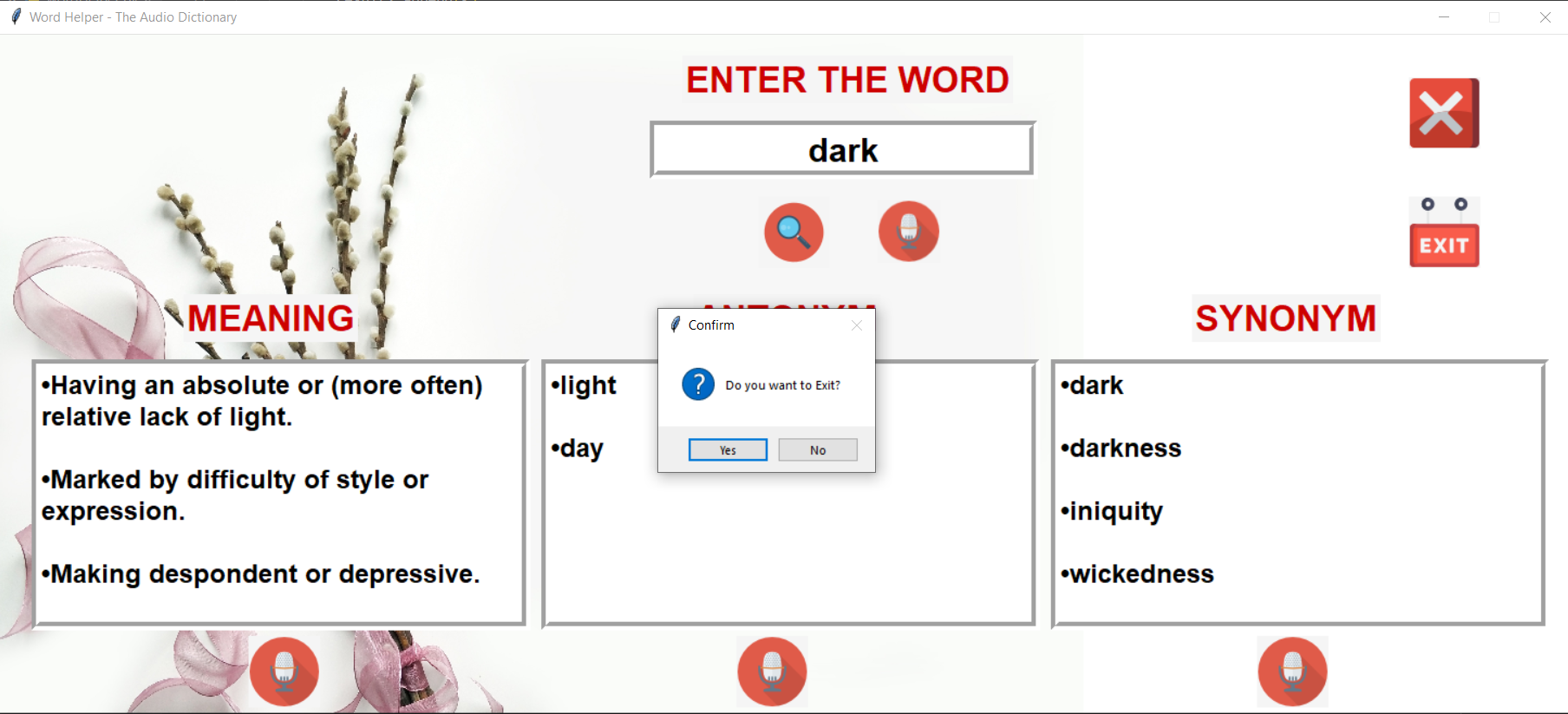
**Fig** **6.6.1 Before Clear**

When we want to clear the screen, the screen is changed from Fig 6.6.1 to Fig 6.6.2.

****

**Fig** **6.6.2 After Clear**

**6.7 EXIT**

****

**Fig** **6.7 Exit**

When exit button is pressed, it displays a messagebox to exit the application.

**CHAPTER 7**

**CONCLUSION**

The mini project has successfully accomplished the goals it had set out in the objectives and design sections of this report.

This UI implements all the modules successfully that are mentioned in this report.

Hence, using this project people will be able to find the meaning, antonym and synonym of the required word successfully in offline mode in fast and efficient way.

**REFERENCES**

[1] [Pinterest](https://www.javatpoint.com/) (example for background image referred)

{2] YouTube (For understanding the concepts used in this project)

[3] <https://www.geeksforgeeks.org/> (for examples)

[4] <https://www.javatpoint.com/> (for referring libraries)

[5] <https://stackoverflow.com/> (for clarifying doubts related to project)