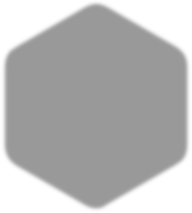
**KI - SAFE**

#### Class: 20ECP-118\_20BCS10\_A Group No.: 6 Group Members Details

|  |  |  |
| --- | --- | --- |
| **S.**  **No.** | **Name** | **UID** |
| **1.** | **DHANESH LAKHANI** | **20BCS1781** |
| **2.** | **PARMAR ASKHAT BHANUPRASAD** | **19BCS1753** |
| **3.** | **ROHAN GODHA** | **20BCS1762** |
| **4.** | **SABAHUL MUNZARIN** | **20BCS1715** |
| **5.** | **SANDEEP SHAW** | **20BCS1676** |
| **6.** | **SIKANDAR SIDDIQUE** | **20BCS1716** |



**INDEX**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **TOPIC** | **PAGE NO.** |
| **1.** | **PROBLEM STATEMENT** | **2** |
| **2.** | **FEATURES** | **4-6** |
| **3.** | **SOFTWARES USED** | **7** |
| **4.** | **DELIVERABLEs** | **7** |
| **5.** | **WORKFLOW** | **8-23** |
| **6.** | **Bibliography** | **24-25** |

***Problem Statement***

*In this HIGH-SPEED DATA world, every day we all have requirements to keep our data safe and have its access fast at the same time. So, why not prepare an App for that.*

*Thus, we as a part of our Foundations of Artificial Intelligence Subject, have come up with an application in which we have added multiple features in which we can keep our passwords safe and some of our day-to-day data in form of short notes.*

# KEY

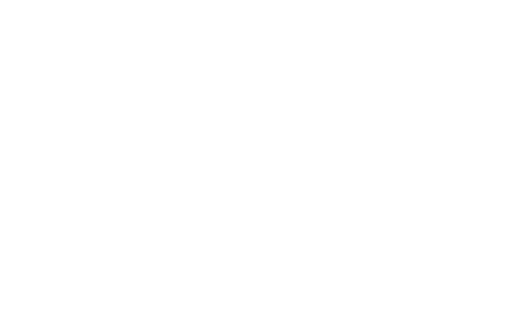
FEATURES

**2nd**

### 3rd

**MAIN FEATURES**

### Insert a picture or enter the key tasks completed

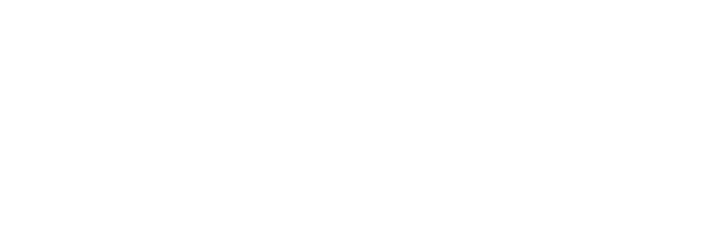


The main features and functionalities of our project are:

PASSWORD GENERATOR PASSWORD MANAGER NOTES MANAGER

3

PASSWORD



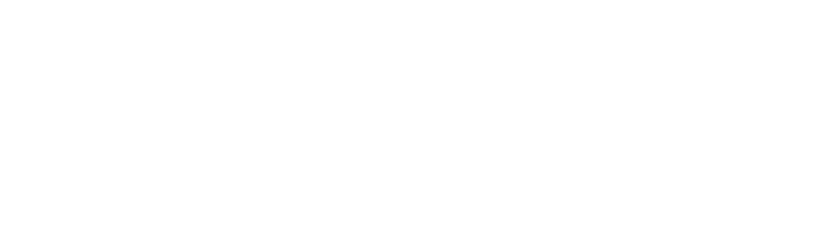
MANAGER

A password manager is a computer program that allows users to store, generate, and manage their personal passwords for online services. A password manager assists in generating and retrieving complex passwords, potentially storing such passwords in an encrypted database or calculating them on demand.

Many password manager applications offer additional capabilities that enhance both convenience and security such as storage of credit card and frequent flyer information and autofill functionality.

Types of password managers include: locally installed software applications online services accessed through website portals locally accessed hardware devices that serve as keys Depending on the type of password manager used and, on the functionality, offered by its developers, the encrypted database is either stored locally on the user's device or stored remotely through an online file-hosting service.

4



PASSWORD

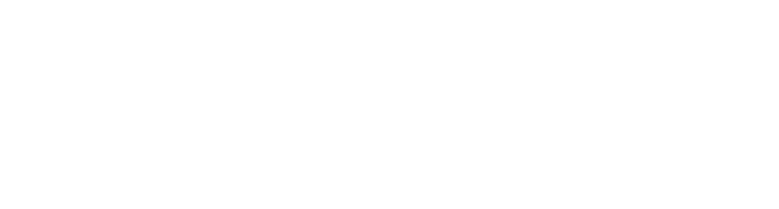
GENERATOR

A password generator is a software tool that creates random or customized passwords for users. It helps users create stronger passwords that provide greater security for a given type of access. Some password generators are simply random password generators.

These programs produce complex/strong passwords with combinations of numbers, uppercase and lowercase letters, and special characters such as braces, asterisks, slashes, etc.

Other types of password generators are made to generate more recognizable passwords rather than a completely random set of characters. There are tools for generating pronounceable passwords, as well as custom tools that allow users to set detailed criteria.

For instance, a user could set a request for a certain number of characters, a certain mix of letters and numbers, a certain number of special characters, or any other criteria for generating a new password. Password generators help those who have to constantly come up with new passwords to ensure authorized access for programs and to manage a large number of passwords for identity and access management. location.



NOTES MANAGER



A note-taking app is an app that lets you keep track of all kinds of notes on your smartphone. You can capture and prioritize ideas on the go, organize your thoughts, and even keep track of images and pictures

Notes Management Software is a notes manager for all your business management and business accounting tasks. A person has to maintain a diary to keep track of all the work that has to be completed effectively and on time. It becomes difficult to maintain the list of tasks manually in an efficient manner. Hence a computerized system needs to be generated that will keep track of the complete information about an individual’s task. This notes manager

software does just that. The Notes Manager Application provides the functionality to maintain the list of tasks.

We have divided Notes Manager in 2 Parts :

1. Speech to Text
2. Direct Text Entry

SOFTWARES USED

1. PyCharm Professional
2. Python IDLE 3.7.9
3. Adobe Photoshop
4. Web Browser
5. Windows PowerShell
6. One Note

**Deliverable(s)**

## [*Python file*](https://drive.google.com/file/d/1WDCg961RHxrGsrT0IoesYEM-UKakHJmB/view?usp=sharing)

* + [*Executable file*](https://drive.google.com/file/d/1E7MCTARdx0kEepVdA9qHJd3v_81_-PdL/view?usp=sharing)
  + [*Screencast*](https://drive.google.com/file/d/10x1qHd0z7L37wGsmmsAjJ09KQNzCQZnq/view?usp=sharing)

7

***WORK FLOW***

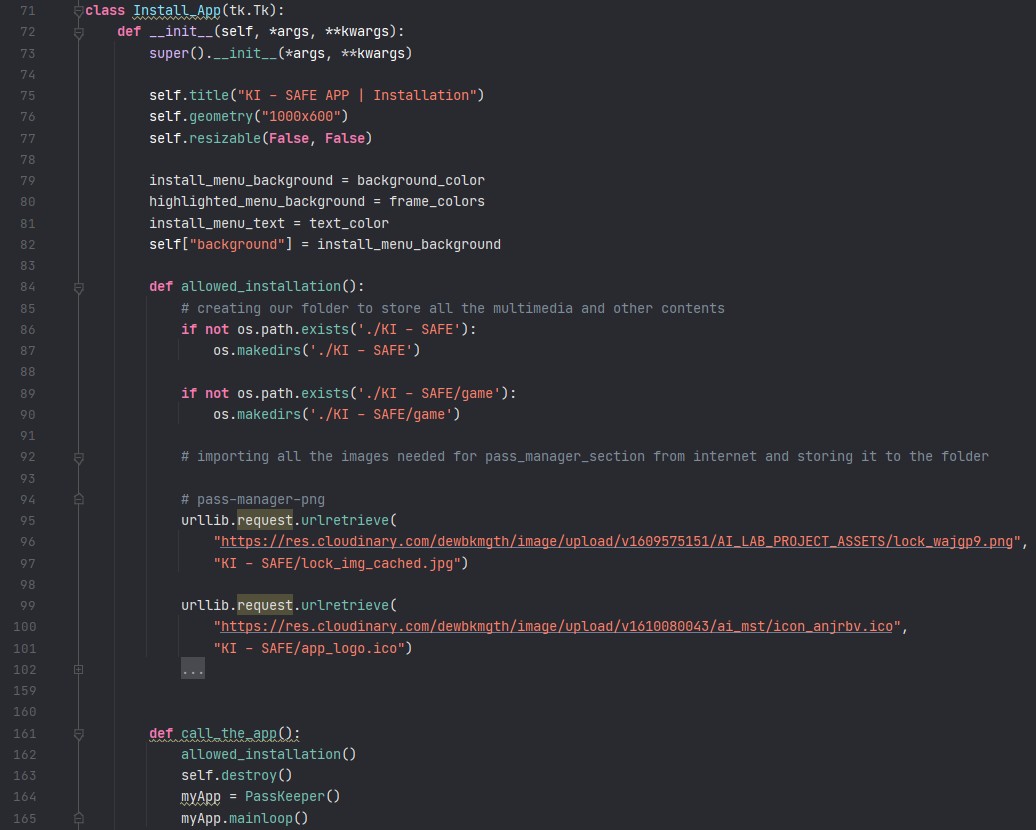
Now let us go through the application so that we get an easy understanding of the app.

**Step – 1: *Installation Process***

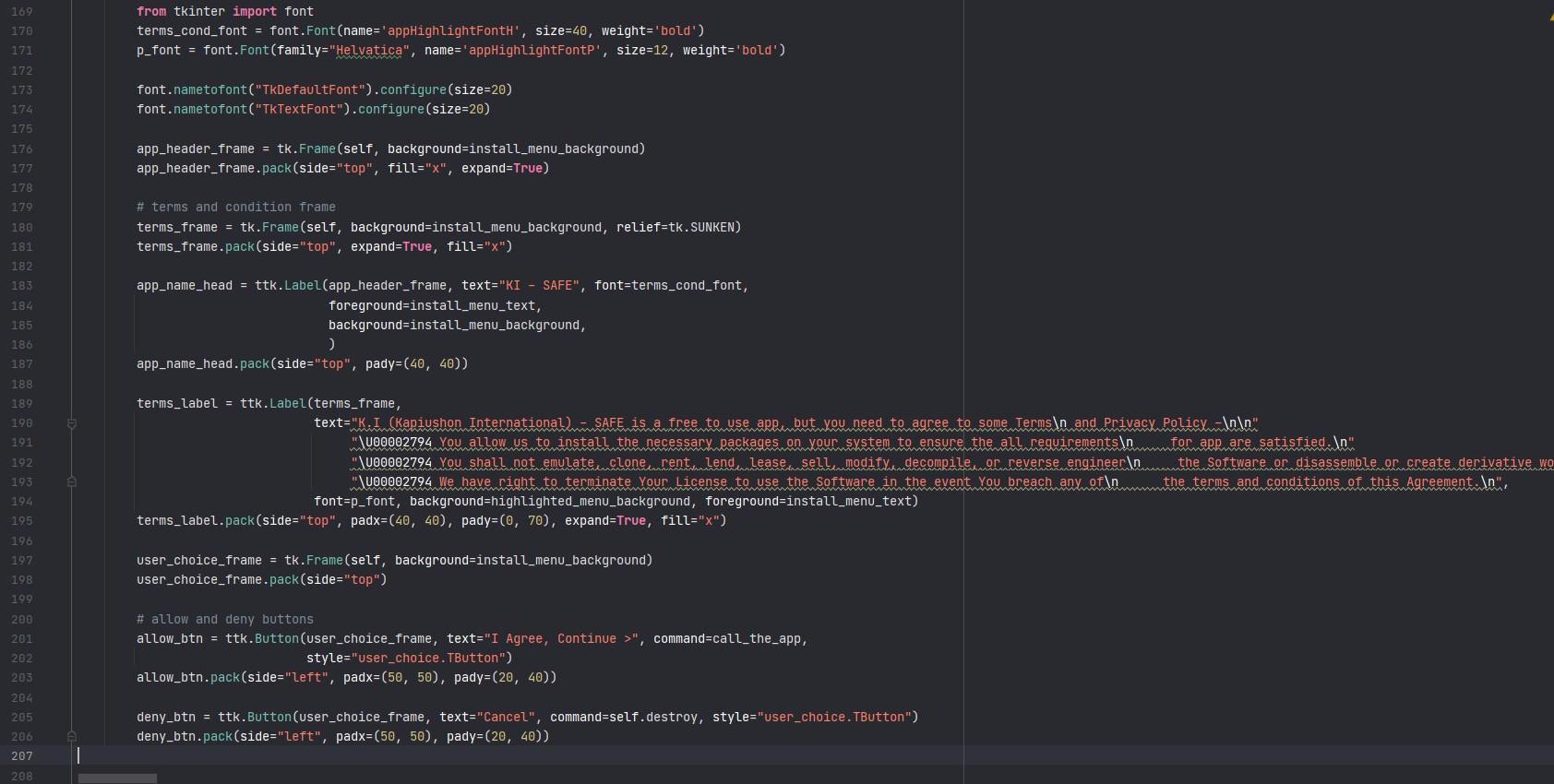


*Before using the app, we’ve installed all the necessary packages required so you only need* ***Python version 3.7.0 – 3.7.9*** *to run*

*the app. We’ve used subprocess to module to automatically install packages if you don’t have them.*



*The above two snapshots display the source code for our app in which basically allows us to download all the images, icons, and* files we need from the internet using ***urlib*** *and* ***request*** *module to locally on our hard drive. We’ve scrapped things for internet to* make one single execution file of the app.

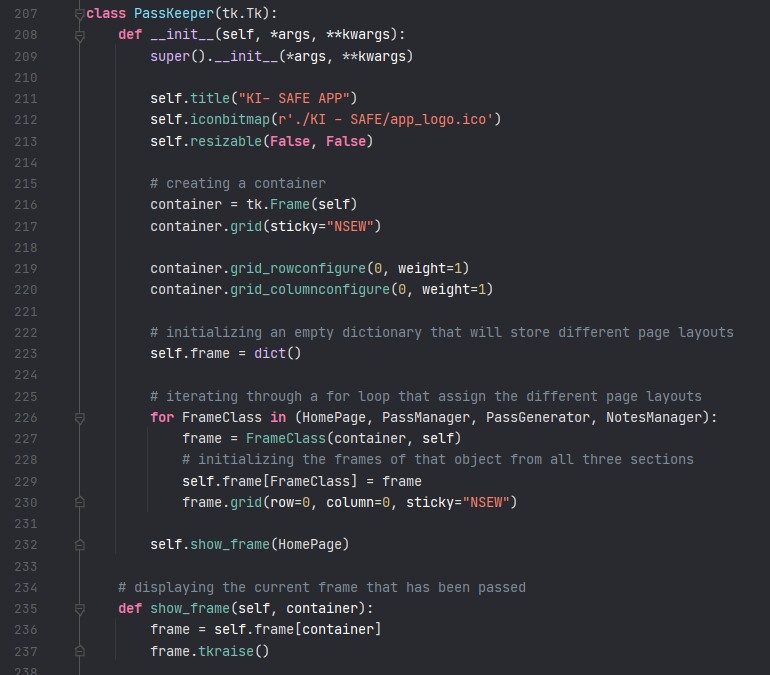


*In this class we’ve inherited everything of* ***tk (tkinter)*** *class into our class so we can use the same functionality along with our* own. We’ve displayed some instructions about the process using labels in ***tkinter*** *and some buttons to continue installation.* Here we’ve used ***pack*** *as geometry manager.*



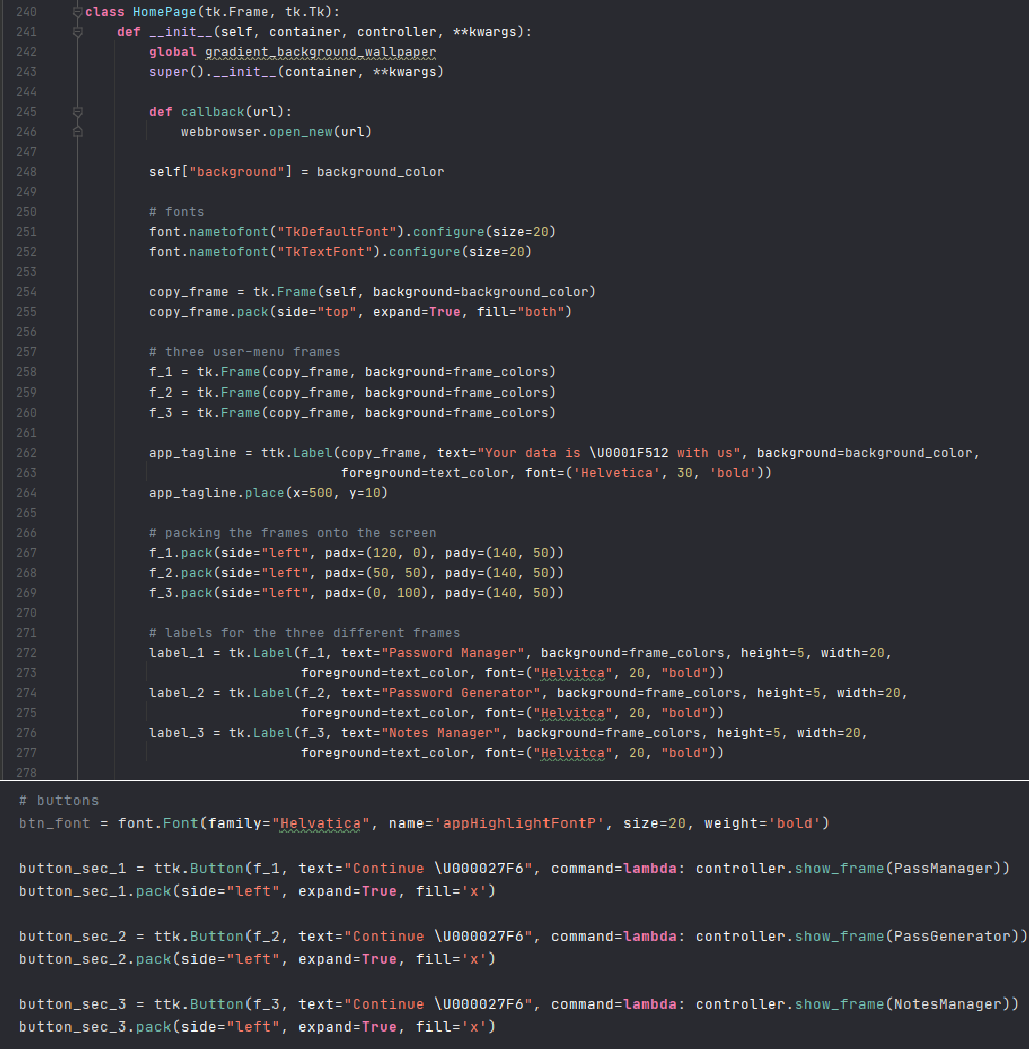
*This is the output of the Install\_App class that we created.*

**Step – 2: *General frame hierarchy***

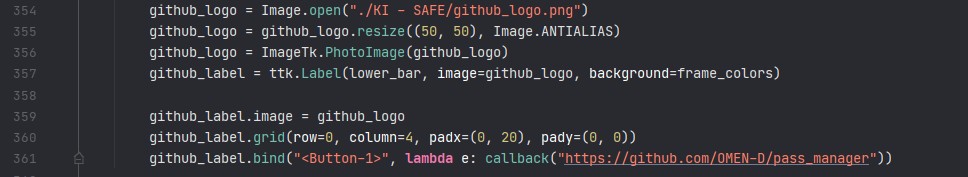


*In tkinter to built an app with multiple screens we use something called controller concept. The above screenshot shows the* controller that stores the specific frame along with its name in a key-value format. It will store all frame layouts and we’ve binded the buttons with controller to switch the frames using show\_frame() function.

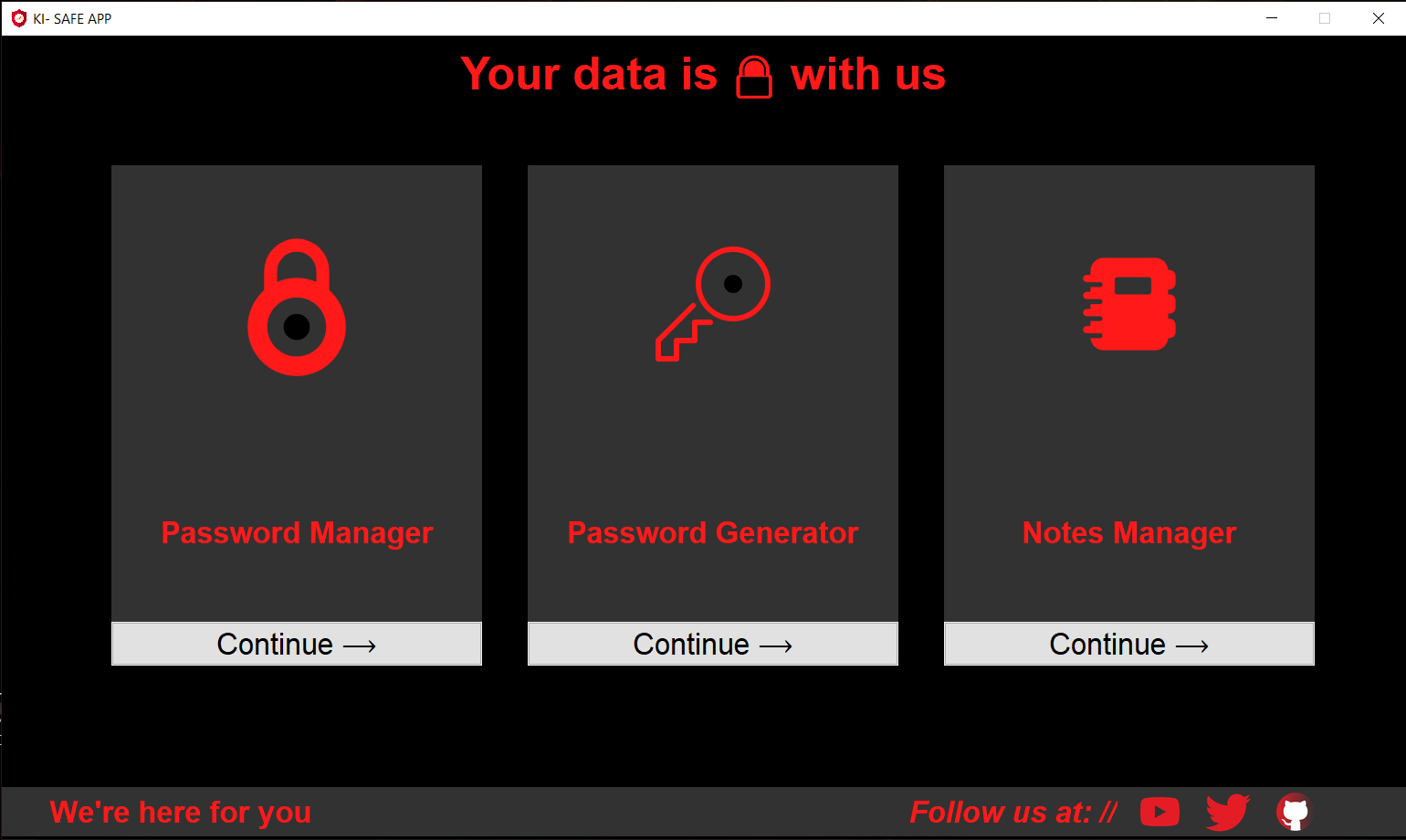
**Step – 3: *Home Page***



*In these above screenshots I have used basically made a Home Page frame which nested frames (header frame > body frame >* footer frame) to display everything in order. In the seconds snapshot you can see that how I have used controller and have bind them with the buttons to show the respective frame.



*We’ve used PIL (pillow) library to display all the images, logo, icons. This is a three-step process first we open images, then use* ImageTk.PhotoImage to convert it in tkitner images format and then we put it into a label. After that we just need to pack it onto the screen.

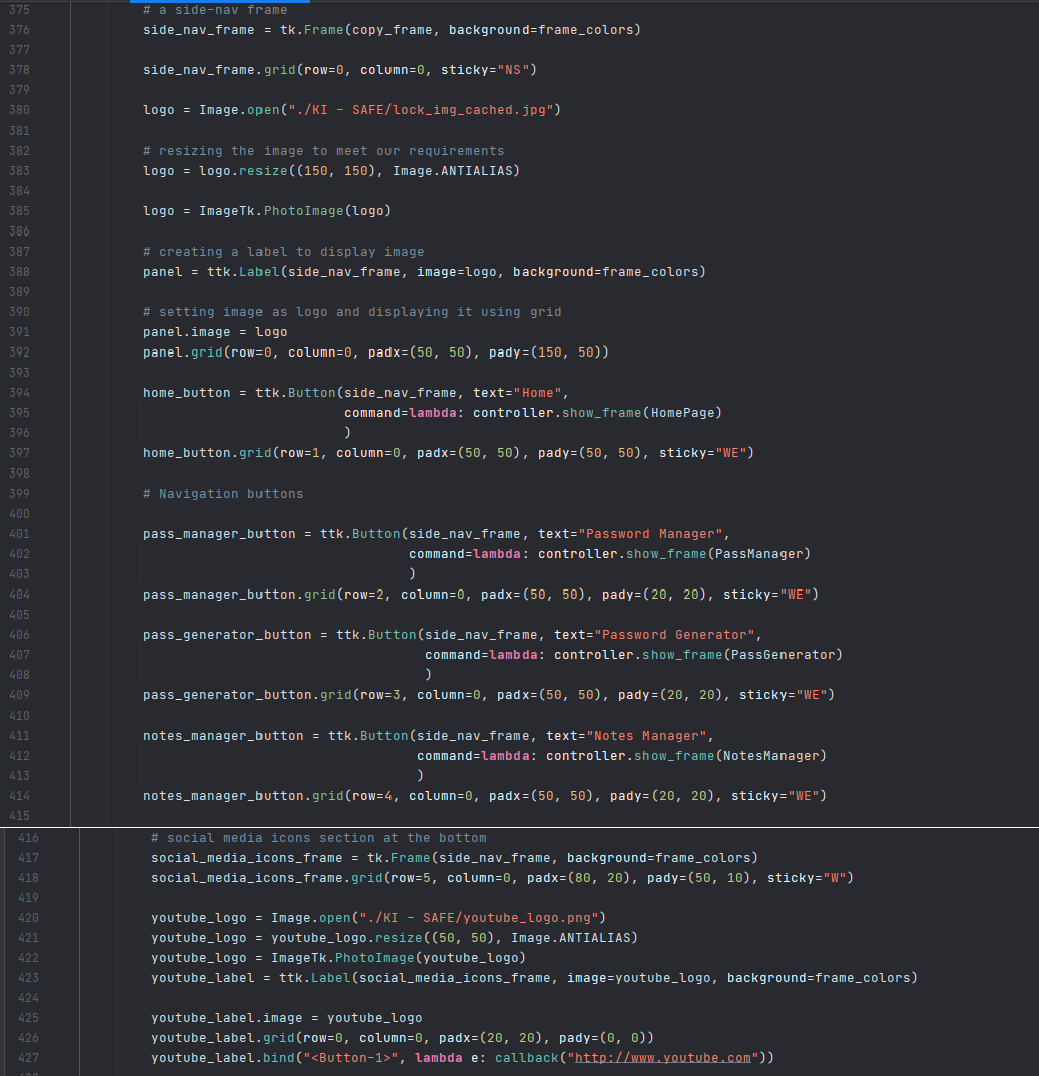


*This screenshot displays output of HomePage class. We can navigate to other three sections of the app using the ‘Continue->’*

*button. In the footer section of the app, we’ve used hyperlinks for social media accounts.*

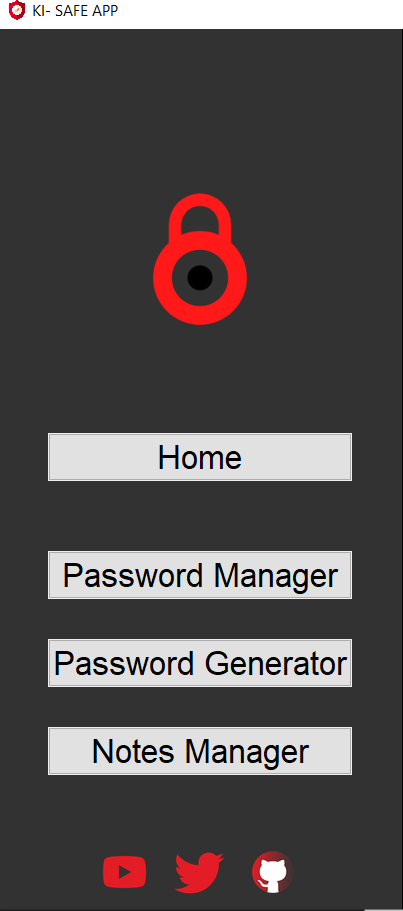
**Step – 4: *Password Manager:***

***For all the other three sections namely Password Manager, Password Generator, Notes Manager we’ve created a list-navigation menu as one section inside every frame and other frame for the respective content which will be explained below.***



*In these snapshots we’ve created a side navigation menu using a Frame property in tkinter. Inside the frame we’ve created the* buttons through which the different sections of app can be explored. And we’ve binded these buttons same as before with the controller to show the respected frame using tk.raise(frame) property. At the bottom we’ve also shown the social media handles for our app. Here we have used grid geometry to give the exact coordinates to labels and frames relative to each other.

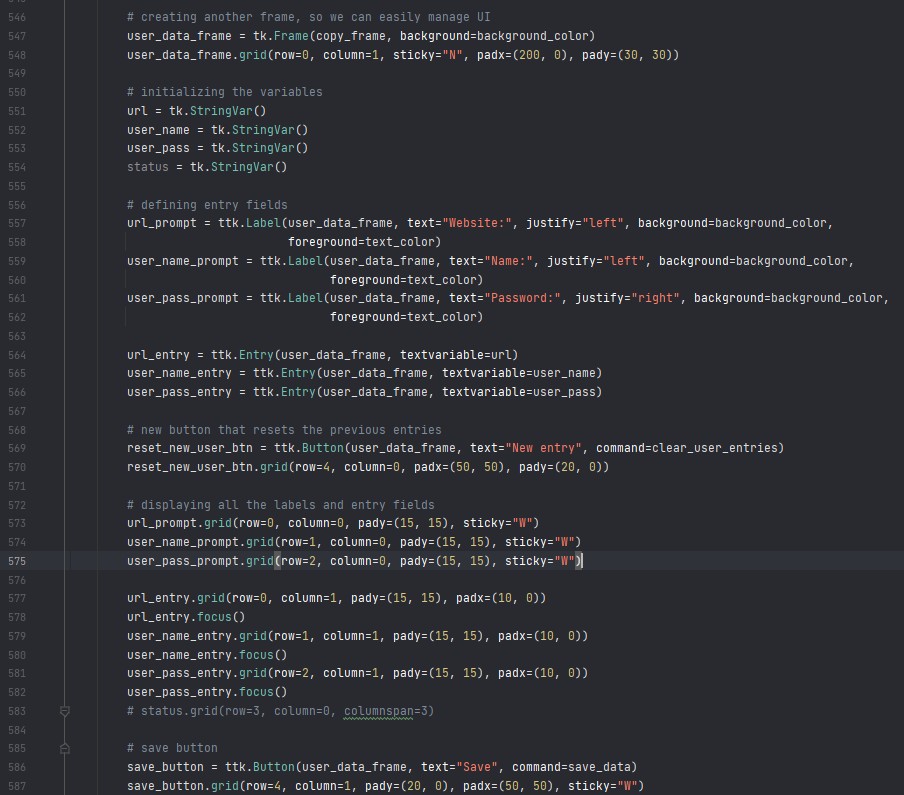
##### *Output:*



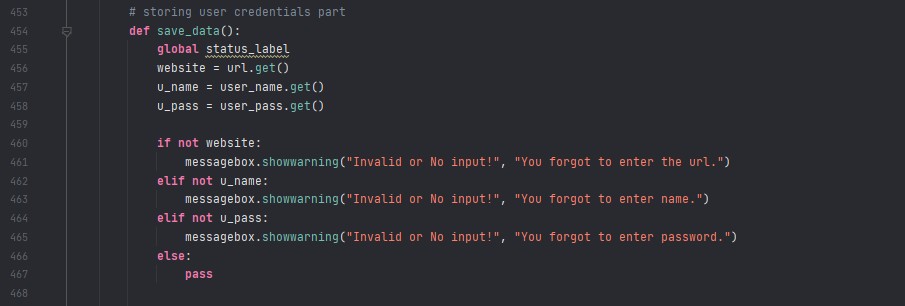
*We’ve also created some visual effects like whenever the button is hovered border color changes to red and text to light-grey*

*color. To achieve this we’ve used style.configure property.*

##### *Password Manager:*

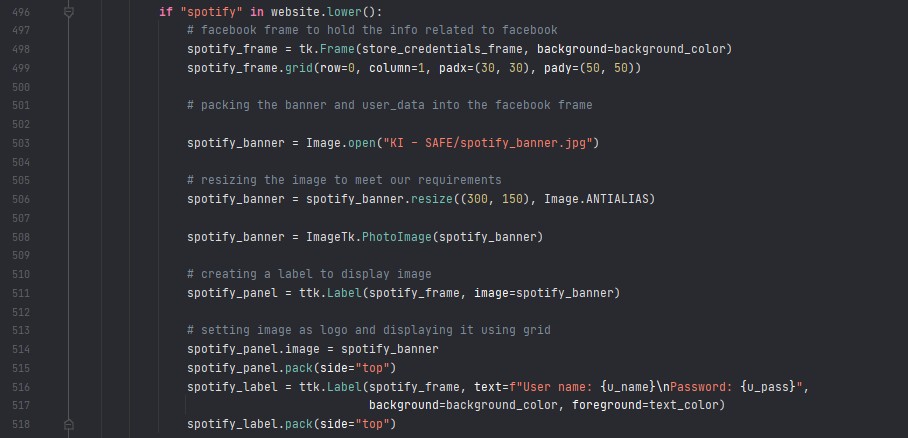


*Here we’ve created a new frame (user\_data) inside the Password manager frame as we mentioned earlier to display the* content apart from navigation menu. We’ve used input field property to take user credentials as input inside a StringVar variable we’ve created. As soon as the we the button is clicked the save\_data function is used to store and display the credentials to user in GUI format. Here .focus() is used to highlight the input field.



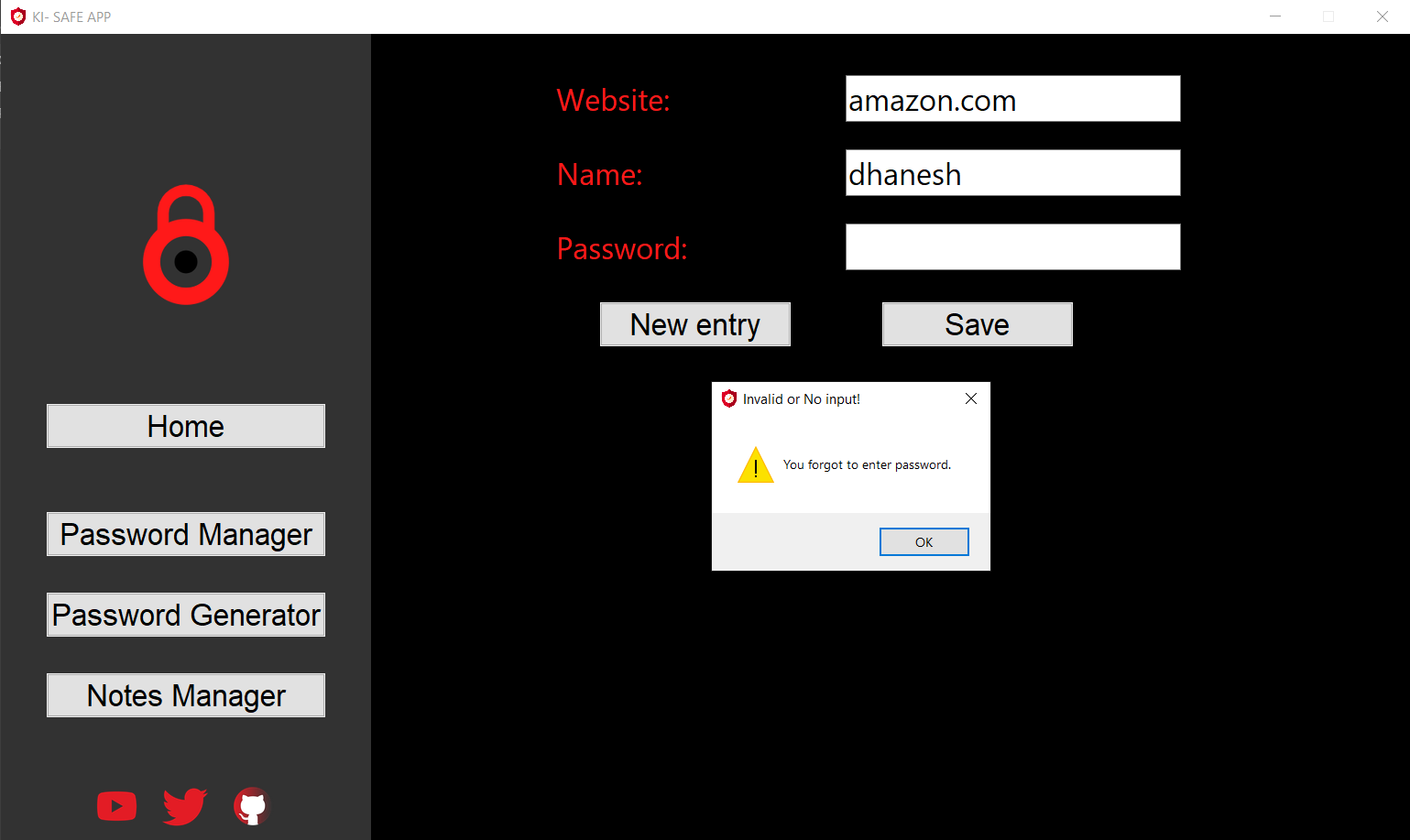
*Here we’ve checked whether the input fields are empty or not, if they are empty and user hasn’t entered anything we’ve raised*

*the messagebox with the warning that he/she forgot to enter particular input field.*



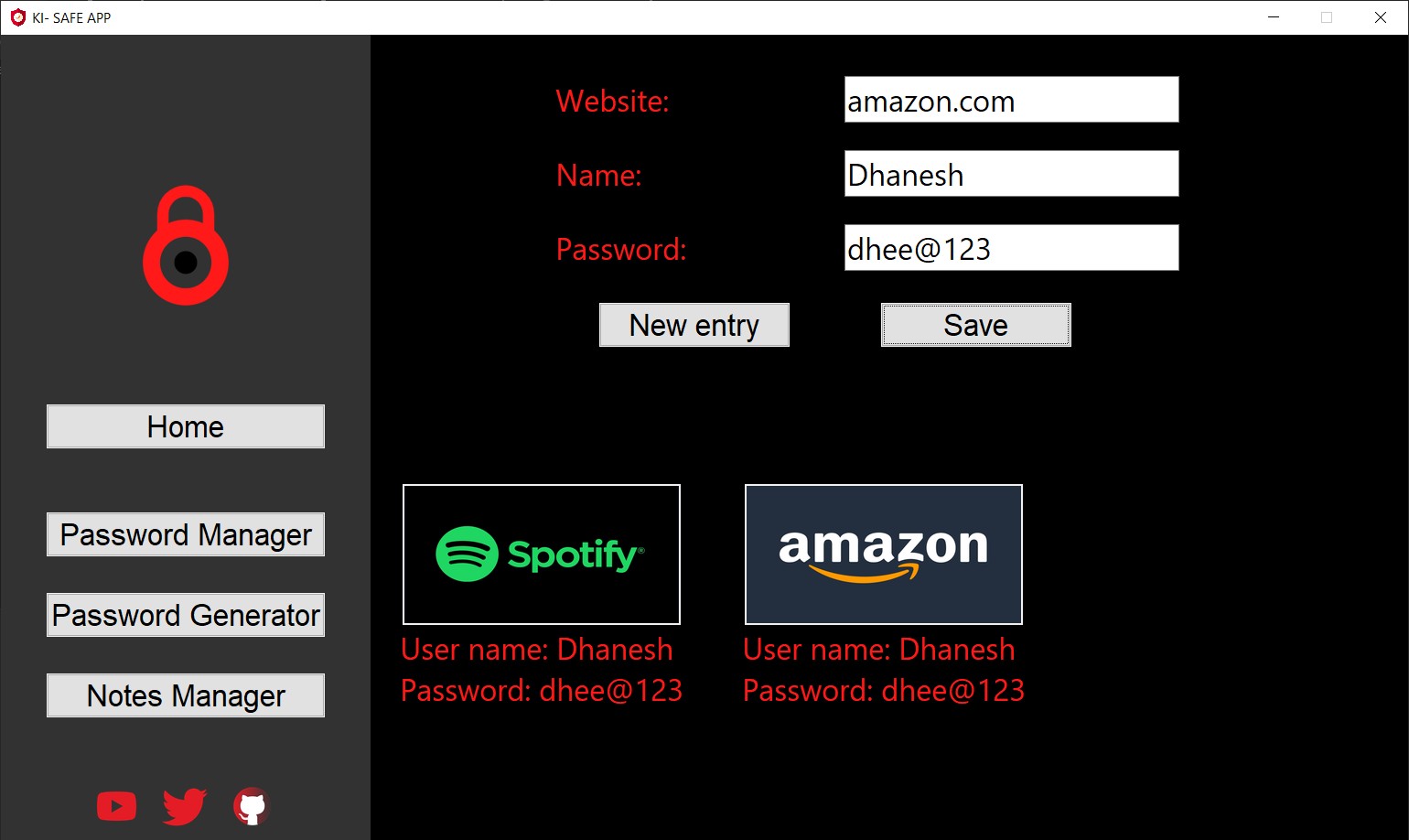
*Here we’ve created a new frame to display to stored credentials for user. Here we’ve used basic ‘if’ statement to check whether* there is anything related to ‘spotify’ inside the user input (url), if condition is true it will display the spotify banner along with the user credentials inside a label. We’ve used .get() method to capture whatever the user has entered.

##### *Outputs:*



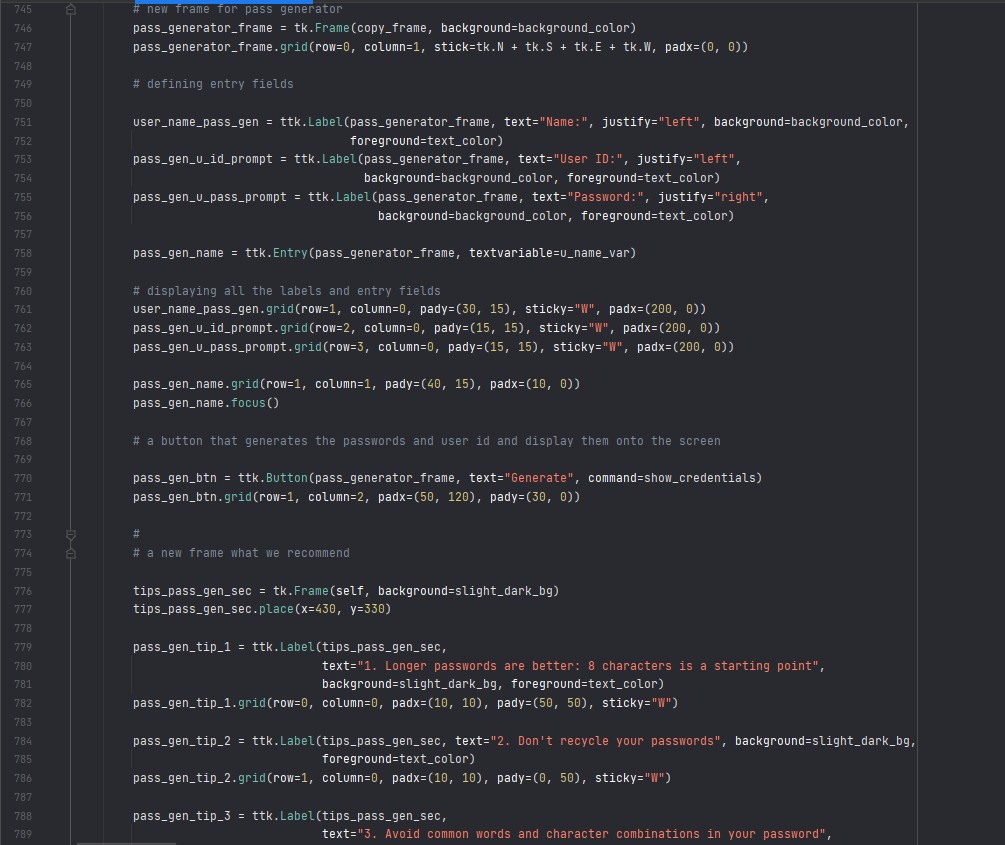
*Here you can see when I entered my credentials and I didn’t entered password, it shows me the pop messagebox to enter the*

*details.*

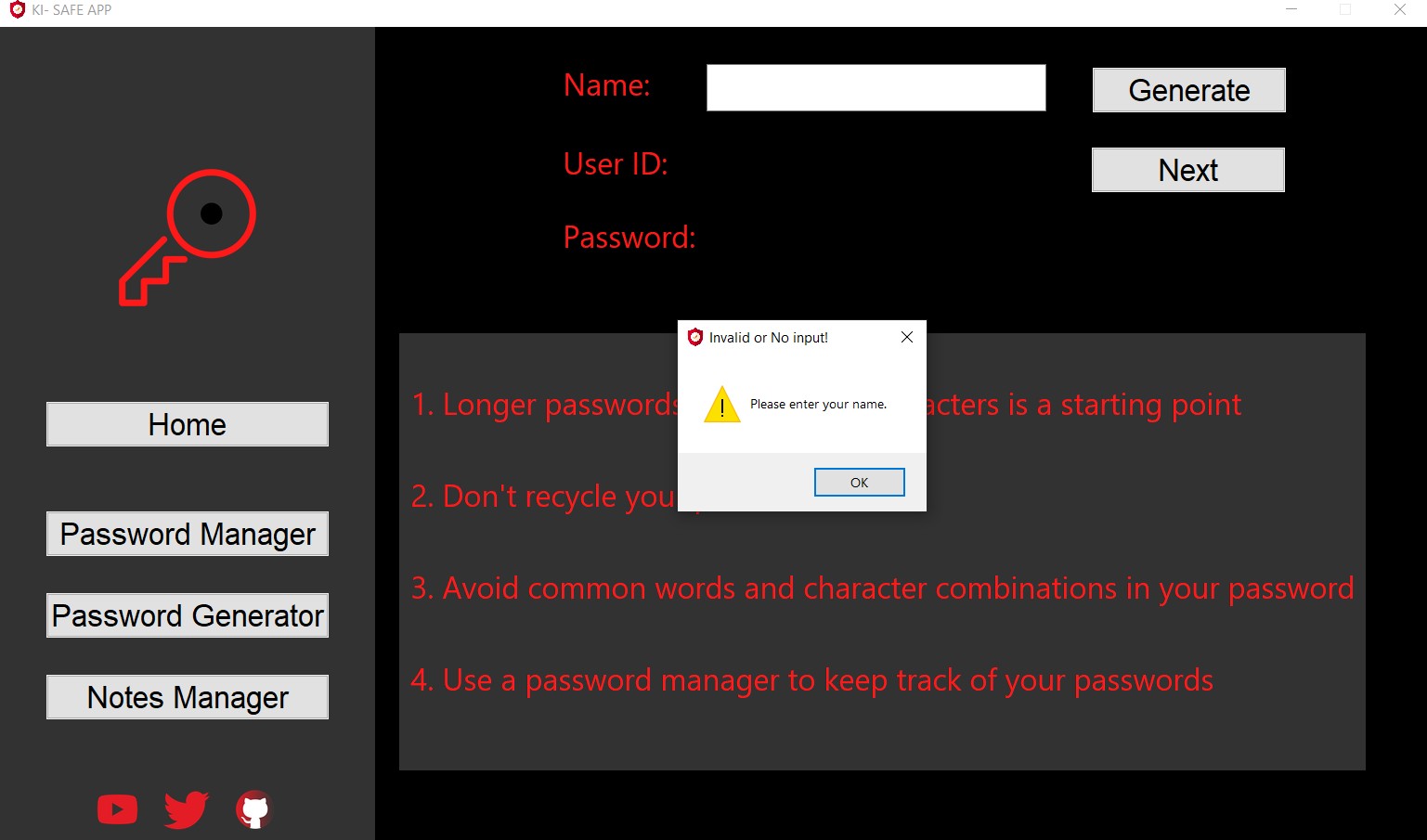


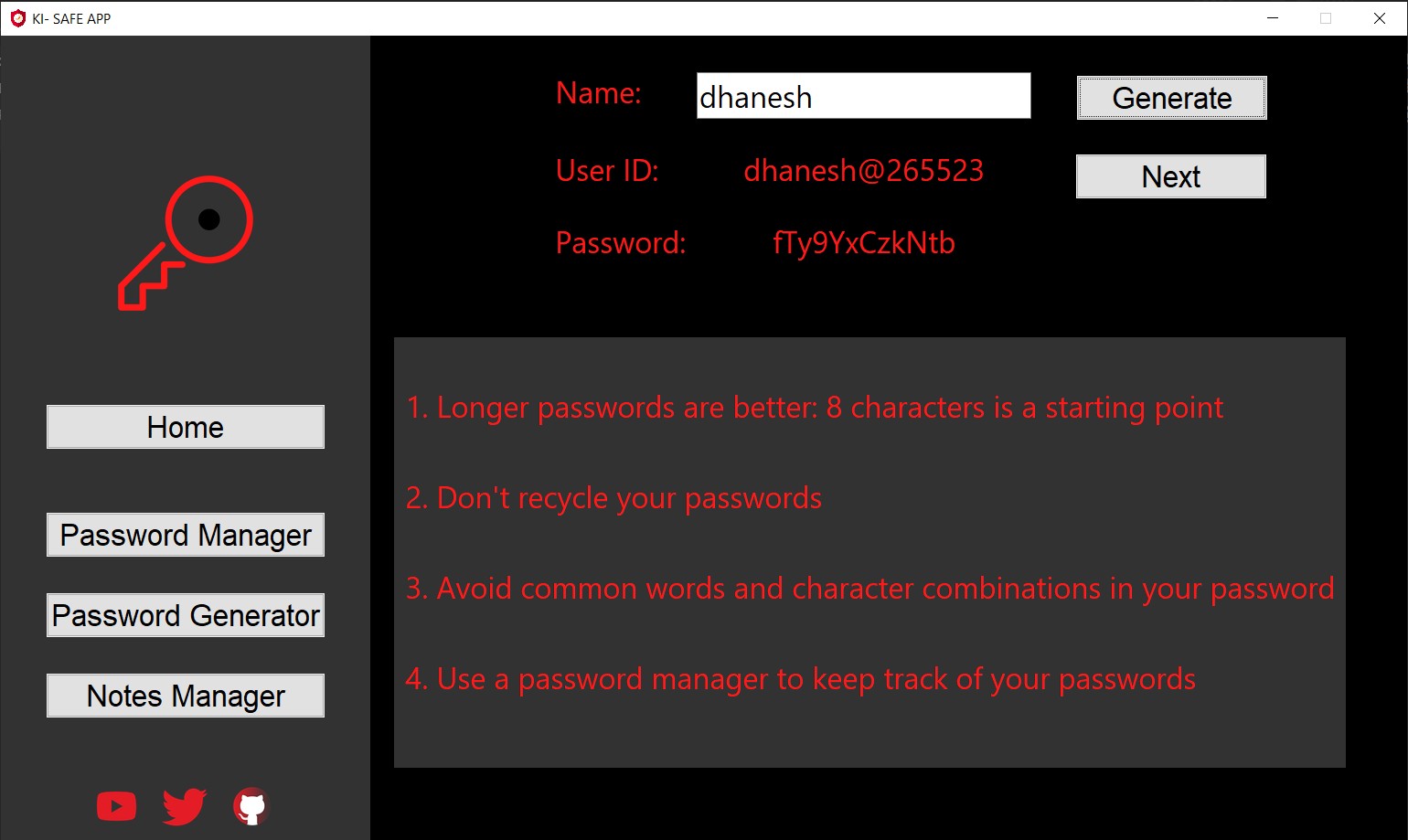
*This is the output it will show if user entered all input fields correctly.*

**Step – 5: *Password Generator***



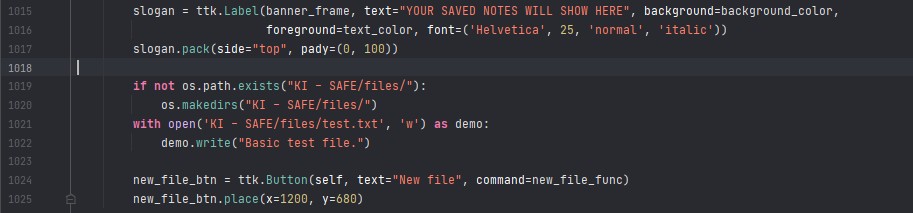
*In this screenshot we’ve made another two frames inside the password generator frame as we’ve done earlier. We’ve used* random module in python to create a secure password generator. There is only one input field, i.e, user name and credentials will be based on the username. We’ve used labels to show the tips we recommend to stay protected.



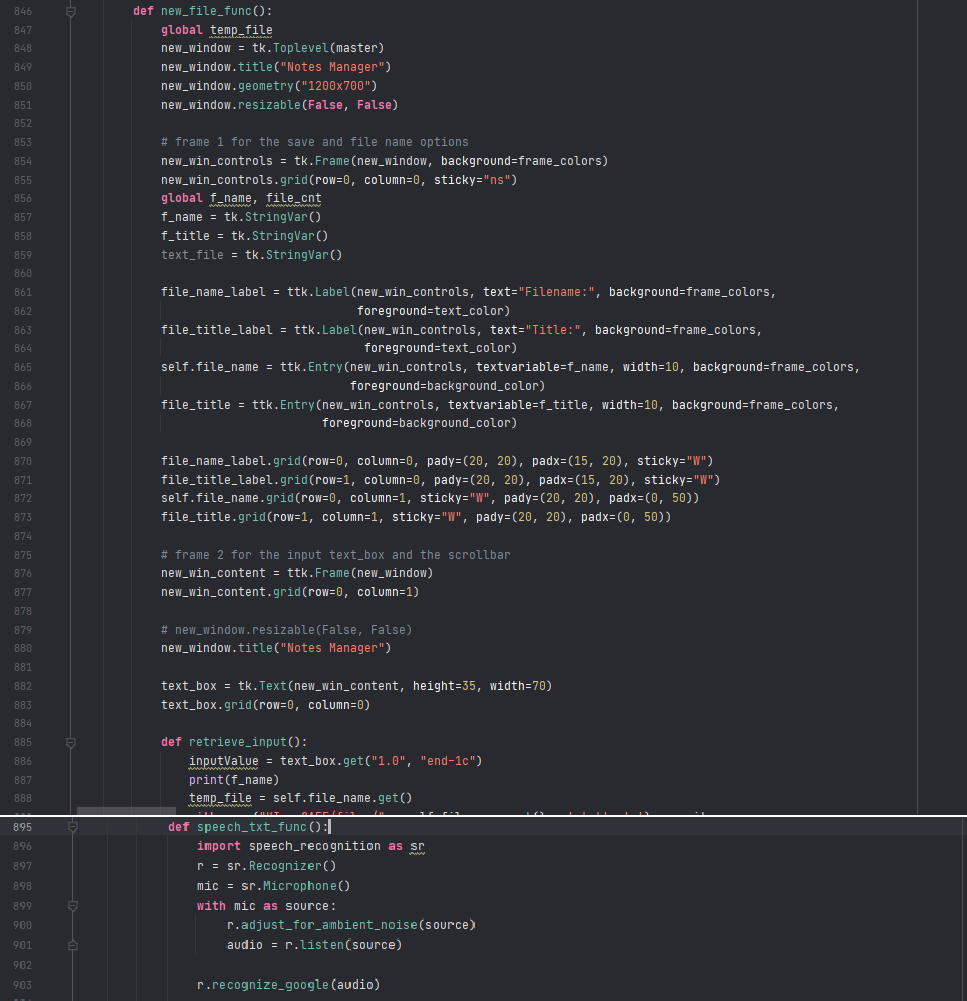


*Here also you can see in above screenshot that if the username is not given it will display the messagebox with the error with the* input fields that are missing.

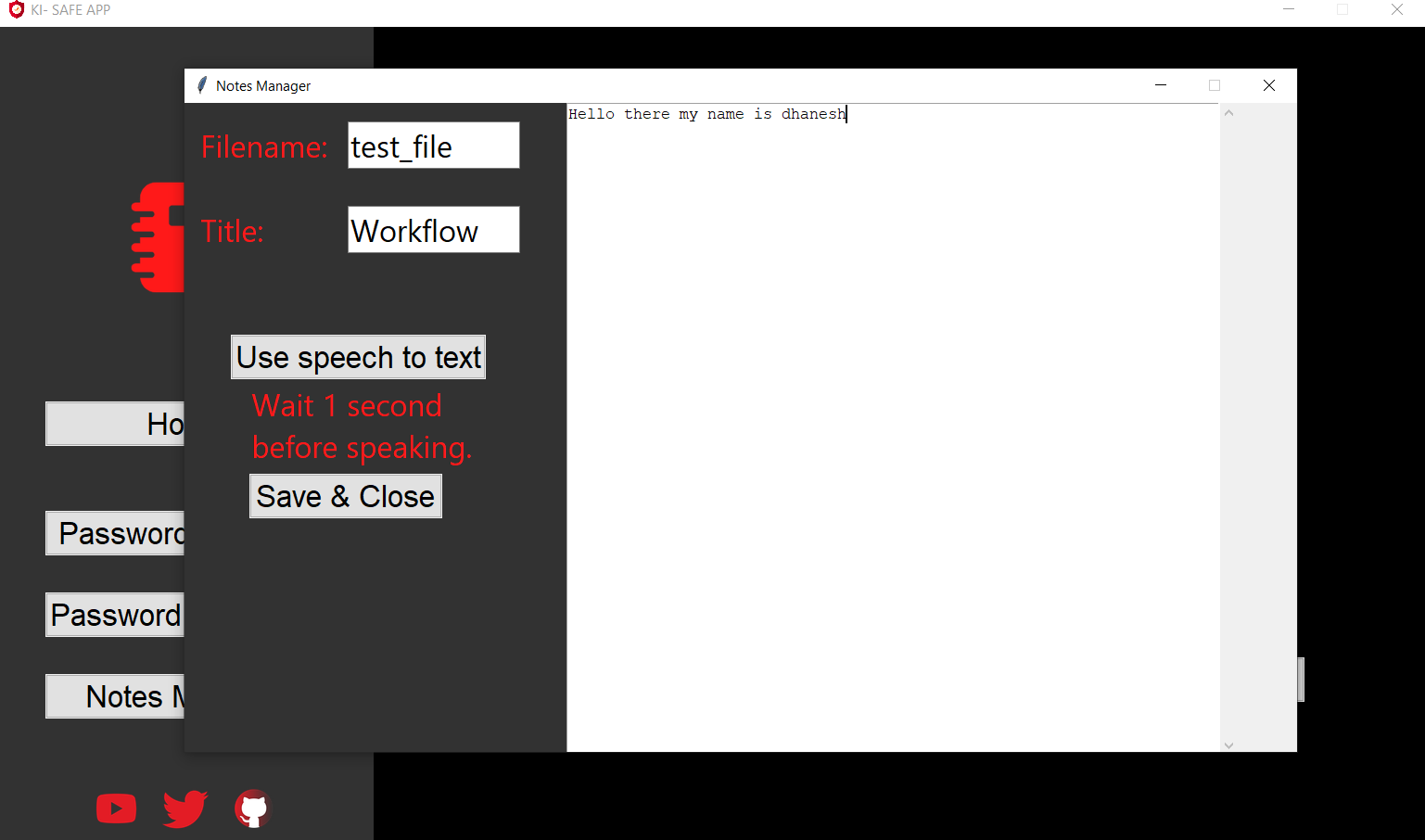
**Step – 6: *Notes Manager***



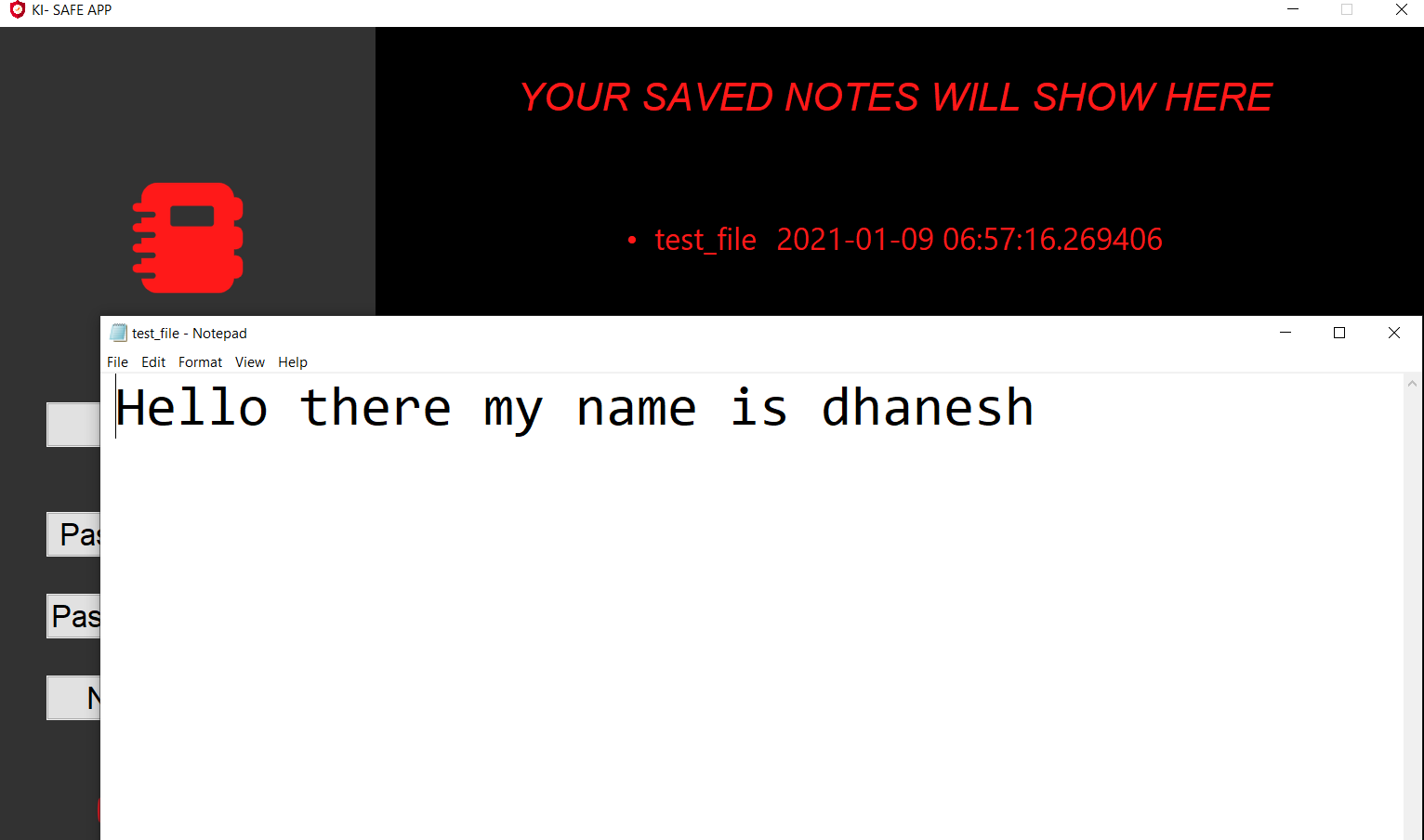
*Here we’ve just initialized the notes manager frame with the side-navigation menu and the basic title. We have also created a* directory using reading and writing files functionality in python with a test file. There is a new file button which will be used to call a new windows to save the notes.



*In the above two screenshots you will see that the new file function creates a new window inside the root one using Toplevel* method in tkinter. Similarly, we’ve created multiple input fields like file title and name along with labels and we’ve used grid geometry. We’ve also used speech\_to\_text using SpeechRecognition module feature which we learned in unit 2.



*In this snapshot you can see whenever the user clicks on the new button new window pops up and here we can write our notes* and save them by clicking on Save & Close button.



*Here you can see that the file name is displayed in the notes manager frame and I’ve binded it to display the content in notepad* whenever its clicked by user.

***Bibliography:***

**Works Cited**

###### “Build A Text Editor Part 7 - Change Text Colors - Python Tkinter GUI Tutorial #110 - YouTube.” *Www.Youtube.com*, [www.youtube.com/watch?v=CtENi3AhuY4.](http://www.youtube.com/watch?v=CtENi3AhuY4) Accessed 9 Jan. 2021.

“Loading Images in Tkinter Using PIL.” *GeeksforGeeks*, 29 Nov. 2019, [www.geeksforgeeks.org/loading-images-in-tkinter-using-pil/.](http://www.geeksforgeeks.org/loading-images-in-tkinter-using-pil/) Accessed 9 Jan. 2021.

“Open a New Window with a Button in Python-Tkinter.” *GeeksforGeeks*, 10 Mar. 2020, [www.geeksforgeeks.org/open-a-new-window-with-a-button-in-python-tkinter/.](http://www.geeksforgeeks.org/open-a-new-window-with-a-button-in-python-tkinter/) Accessed 9 Jan. 2021.

“Python | Add Style to Tkinter Button.” *GeeksforGeeks*, 13 Mar. 2019, [www.geeksforgeeks.org/python-add-style-to-tkinter-button/.](http://www.geeksforgeeks.org/python-add-style-to-tkinter-button/) Accessed 9 Jan. 2021.

“Python | Binding Function in Tkinter.” *GeeksforGeeks*, 3 Apr. 2019, [www.geeksforgeeks.org/python-binding-function-in-tkinter/.](http://www.geeksforgeeks.org/python-binding-function-in-tkinter/) Accessed 9 Jan. 2021.

“Python - How to Create a Hyperlink with a Label in Tkinter?” *Stack Overflow*, stackoverflow.com/questions/23482748/how-to-create-a-hyperlink-with-a-label-in- tkinter.

“Python - Tkinter Background Image in Frame.” *Stack Overflow*, stackoverflow.com/questions/54250448/tkinter-background-image-in- frame#:~:text=1%20Answer&text=You%20could%20put%20an%20image.

“Python 3.x - How to Open Multiple Files in a Directory.” *Stack Overflow*, stackoverflow.com/questions/38991923/how-to-open-multiple-files-in-a- directory/38992988.

Python, Real. “The Ultimate Guide To Speech Recognition With Python – Real

Python.” *Realpython.com*, realpython.com/python-speech-recognition/#working-with- audio-files. Accessed 9 Jan. 2021.

“TkDocs Tutorial - Fonts, Colors, Images.” *Tkdocs.com*, tkdocs.com/tutorial/fonts.html.

Accessed 9 Jan. 2021.

“Tkinter Application to Switch Between Different Page Frames.” *GeeksforGeeks*, 1 May 2020, [www.geeksforgeeks.org/tkinter-application-to-switch-between-different-page-frames/.](http://www.geeksforgeeks.org/tkinter-application-to-switch-between-different-page-frames/) Accessed 9 Jan. 2021.

“Twitter Icons - Free Download, PNG and SVG.” *Icons8.com*, icons8.com/icons/set/twitter.

Accessed 9 Jan. 2021.