



VITYARTHI PROJECT

Title: Personal Notes Saver – Mini Project

Name: Sakshi Dhananjay Bhosale

Registration No.: 25BHI10052

Branch: CSE (Health informatics)

Course: Python essentials/ Intro to problem solving and programming (CSE1021)

Date of submission: 23/11/2025

sakshi.25bhi10052@vitbhopal.ac.in

[Email address]

Problem Statement

In day-to-day college life, we often need a quick place to store small notes — like class reminders, to-do items, assignment points, etc.

Most first-year students switch between mobile, WhatsApp notes, paper, and laptop files, which becomes confusing.

So, I decided to create a simple console-based Personal Notes Saver using Python.

It lets a user:

- Add a note
- View saved notes
- Delete a note

This solves a small real-time problem: keeping all small study or task notes in one place in an organised way.

Objectives

The main objectives of this mini project are:

1. To understand how Python interacts with files.
2. To perform basic operations like read, write, and update.
3. To practise simple working of functions and menus.
4. To build a very small useful program for daily note-keeping.

Proposed Solution

The program follows a menu-driven approach.

The user chooses what action to perform:

- 1 → Add a new note (text is appended to a file)
- 2 → View notes (program reads notes from file)
- 3 → Delete notes (file is cleared)
- 4 → Exit

All notes are stored in a file called notes.txt.

This makes the program simple and suitable for a beginner-level console project.

Tools & Technologies Used

Language: Python

Editor: VS Code

Version Control: Git & GitHub

System Requirements

Any system with Python installed

Any text editor like VS Code or Notepad

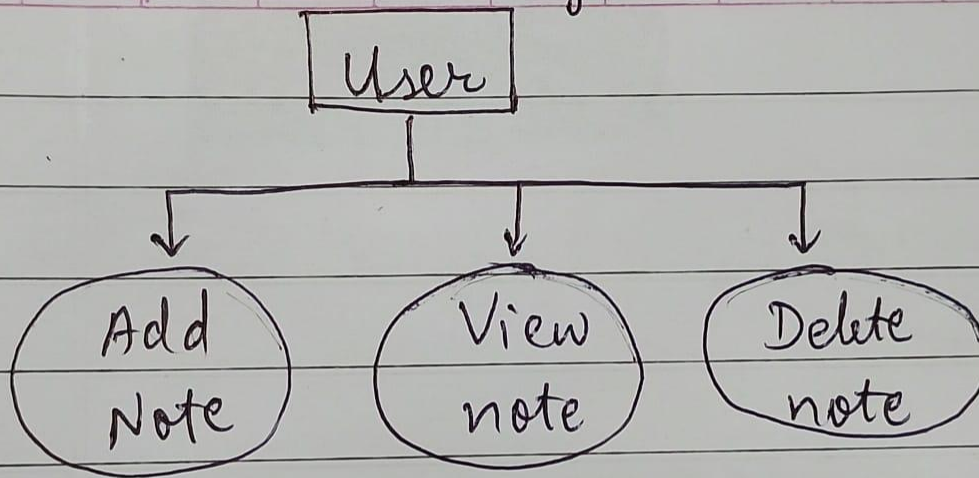
Basic storage for saving .txt file

Operating System: Windows

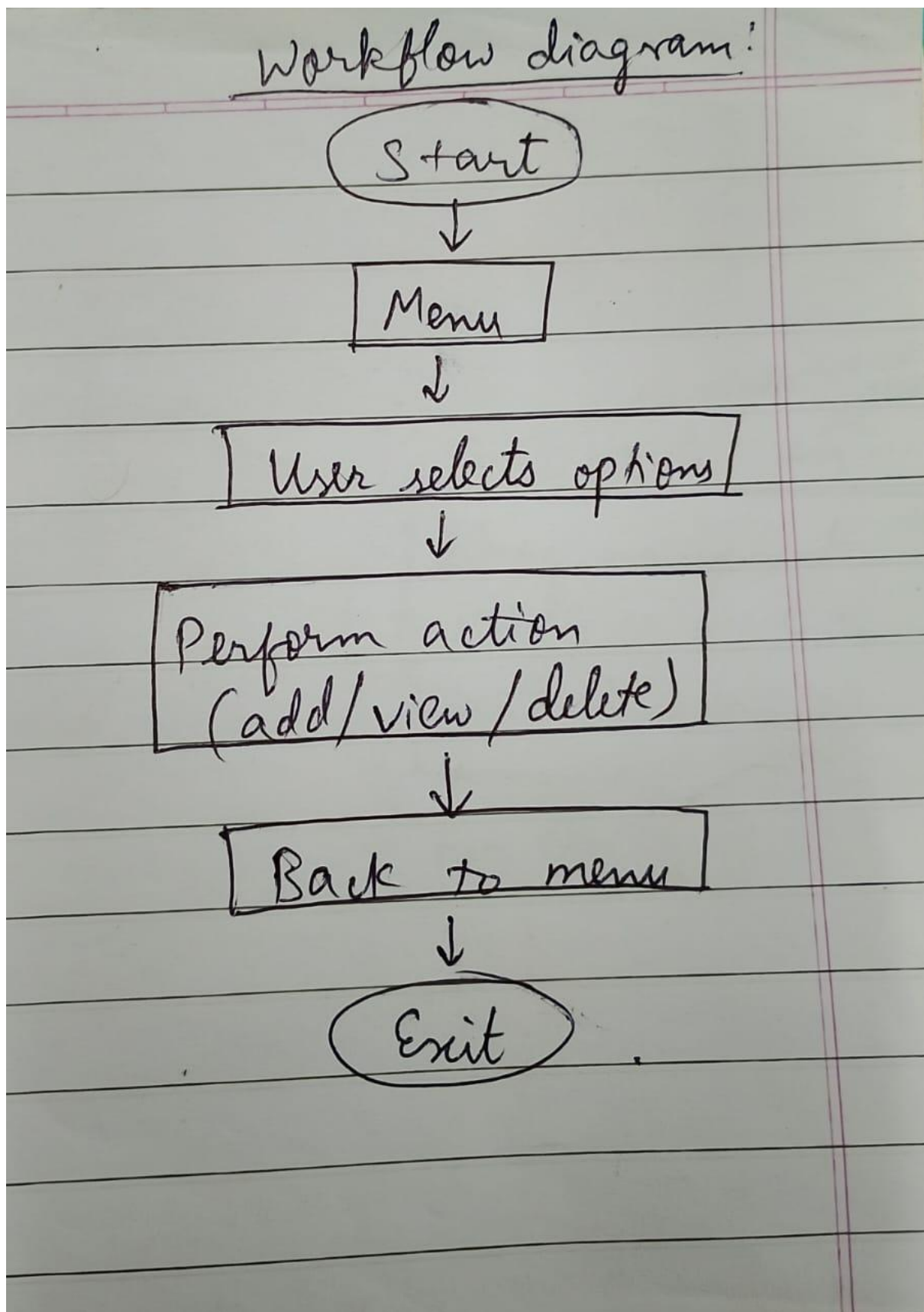
Project Design (Diagrams)

1. Use Case Diagram

Use Case Diagram:

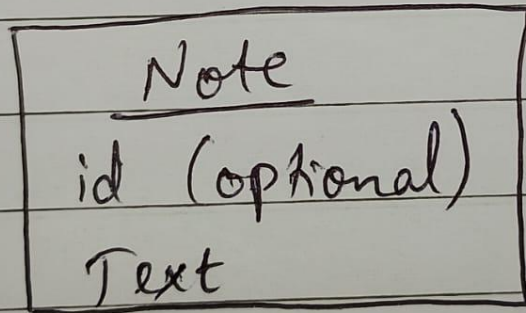


2. Workflow Diagram



3. Class/ER Diagram

Class/ER Diagram:



Implementation

Below is the final Python code used:

```
def add_note():
    note = input("Enter your note: ")
    with open("notes.txt", "a") as file:
        file.write(note + "\n")
    print("Note added successfully.\n")

def view_notes():
    print("\nYour Notes:")
    with open("notes.txt", "r") as file:
        notes = file.readlines()
        if not notes:
            print("No notes found.\n")
        else:
            for idx, note in enumerate(notes, start=1):
                print(f'{idx}. {note.strip()}')
    print()

def delete_notes():
    with open("notes.txt", "w") as file:
        pass
    print("All notes deleted successfully.\n")

def main():
    while True:
```



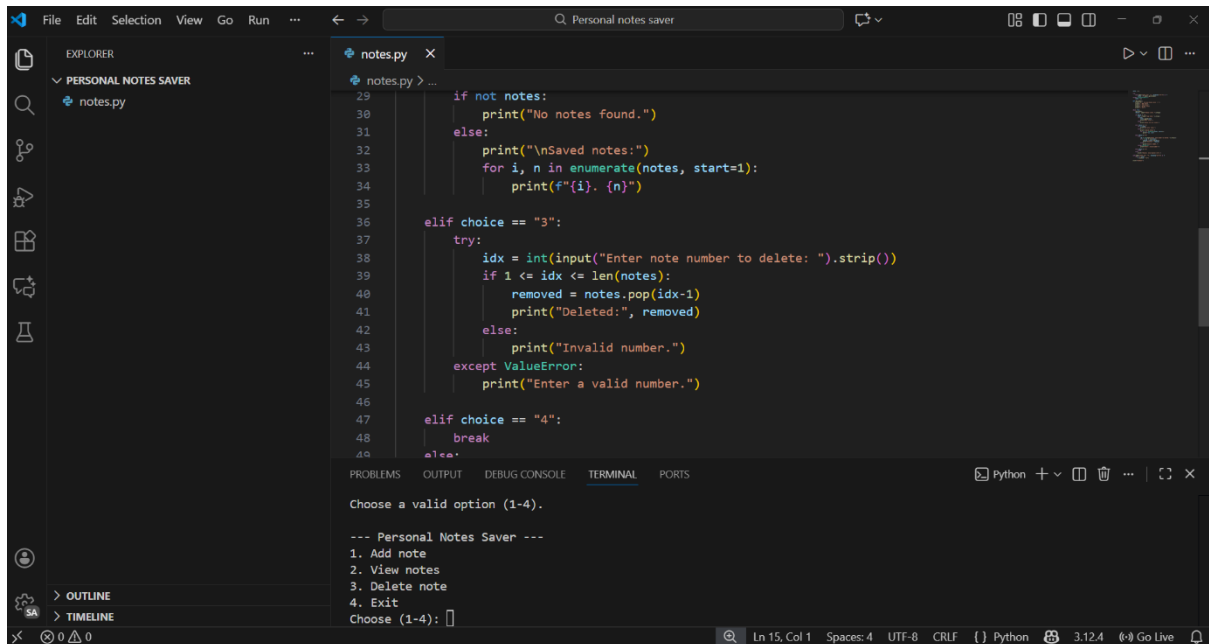
```
print("1. Add Note")
print("2. View Notes")
print("3. Delete Notes")
print("4. Exit")
```

```
choice = input("Enter your choice: ")
```

```
if choice == "1":
    add_note()
elif choice == "2":
    view_notes()
elif choice == "3":
    delete_notes()
elif choice == "4":
    print("Exiting program...")
    break
else:
    print("Invalid choice. Try again.\n")
```

```
if __name__ == "__main__":
    main()
```

Output Screenshots



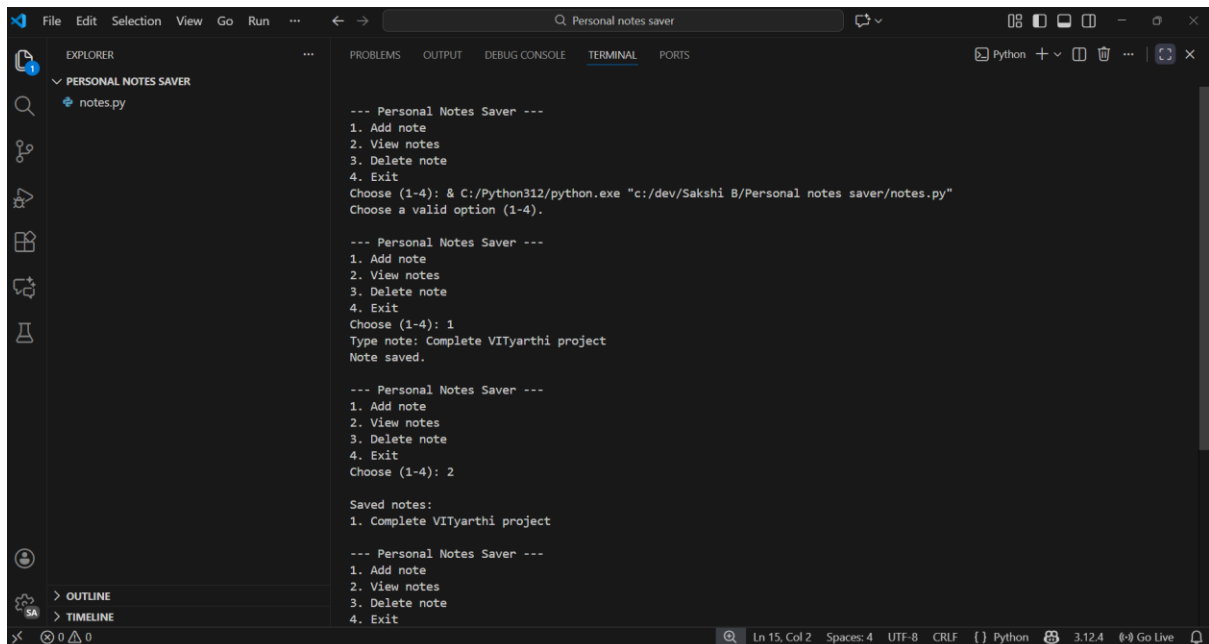
```
29         if not notes:
30             print("No notes found.")
31         else:
32             print("\nSaved notes:")
33             for i, n in enumerate(notes, start=1):
34                 print(f"{i}. {n}")
35
36         elif choice == "3":
37             try:
38                 idx = int(input("Enter note number to delete: ").strip())
39                 if 1 <= idx <= len(notes):
40                     removed = notes.pop(idx-1)
41                     print("Deleted:", removed)
42                 else:
43                     print("Invalid number.")
44             except ValueError:
45                 print("Enter a valid number.")
46
47         elif choice == "4":
48             break
49     else:
```

Choose a valid option (1-4).

--- Personal Notes Saver ---

1. Add note
2. View notes
3. Delete note
4. Exit

Choose (1-4):



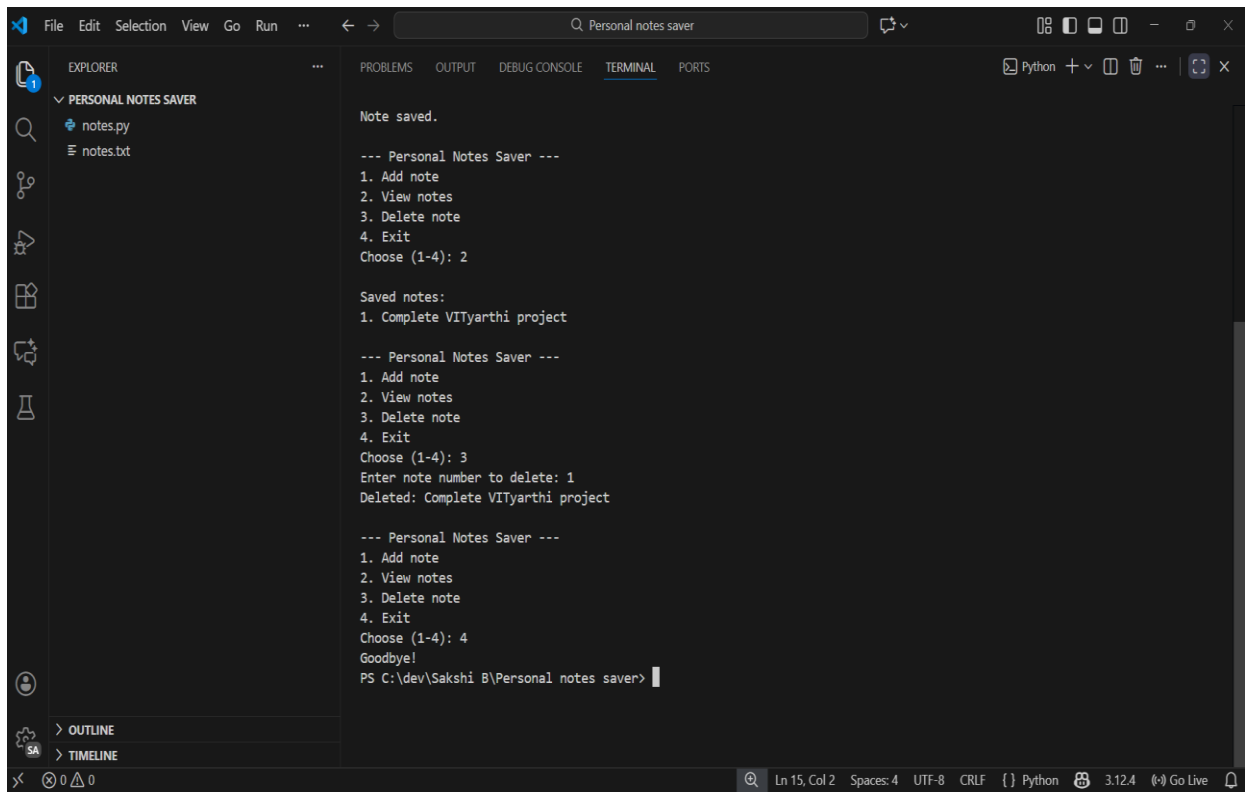
```
--- Personal Notes Saver ---
1. Add note
2. View notes
3. Delete note
4. Exit
Choose (1-4): & C:/Python312/python.exe "c:/dev/Sakshi B/Personal notes saver/notes.py"
Choose a valid option (1-4).

--- Personal Notes Saver ---
1. Add note
2. View notes
3. Delete note
4. Exit
Choose (1-4): 1
Type note: Complete VITYarthi project
Note saved.

--- Personal Notes Saver ---
1. Add note
2. View notes
3. Delete note
4. Exit
Choose (1-4): 2

Saved notes:
1. Complete VITYarthi project

--- Personal Notes Saver ---
1. Add note
2. View notes
3. Delete note
4. Exit
```



The screenshot shows a Visual Studio Code editor with a terminal window open. The terminal displays the output of a Python script named 'Personal Notes Saver'. The script prompts the user to choose an option (1-4) and then performs actions based on the choice. The output shows the script running successfully and displaying the saved notes.

```
Personal notes saver

Note saved.

--- Personal Notes Saver ---
1. Add note
2. View notes
3. Delete note
4. Exit
Choose (1-4): 2

Saved notes:
1. Complete VITyarthi project

--- Personal Notes Saver ---
1. Add note
2. View notes
3. Delete note
4. Exit
Choose (1-4): 3
Enter note number to delete: 1
Deleted: Complete VITyarthi project

--- Personal Notes Saver ---
1. Add note
2. View notes
3. Delete note
4. Exit
Choose (1-4): 4
Goodbye!
PS C:\dev\Sakshi B\Personal notes saver>
```

GitHub Repository Link

<https://github.com/sakshibhosale2107-bit/Personal-notes-saver.git>

Conclusion

This mini project helped me understand the basics of Python programming, file handling, and how simple menu-driven applications work. I also learned how to use GitHub for uploading a project and managing files properly.

The program solves a small practical problem of saving quick notes in one place, and it can be improved later by adding timestamps, categories, or even a GUI.

References

Python official documentation

Class notes

Basic examples from online tutorials

END OF PROJECT

THANK YOU!