

A PROJECT REPORT ON  
“Personal Diary”  
SUBMITTED BY:

---

Mr.Nikam Rohan Atmaram  
2124UCEM1084

SUBJECT:

Programming in Problem Solving Using C++  
Under the guidance of

---

Miss. Ishwari Tirse



Department of Computer Science and Engineering

Sanjivani Rural Education  
Society's

**SANJIVANI UNIVERSITY**

KOPARGAON – 423603, DIST : AHMEDNAGAR

2024-2025

# INDEX

| SR.<br>NO | CONTENT      | PAGE NO. |
|-----------|--------------|----------|
| 1.        | INTRODUCTION | 3        |
| 2.        | CODE         | 4        |
| 3.        | OUTPUT       | 8        |
| 4.        | CONCLUSION   | 11       |

# INTRODUCTION

The Personal Diary project is a console-based application developed in C++ that enables users to maintain and manage their diary entries effectively. In today's fast-paced world, keeping track of thoughts, experiences, and daily activities is essential for personal reflection and mental well-being. This application aims to provide a simple yet functional platform for users to write, read, edit, and delete diary entries, ensuring that their memories are preserved and easily accessible.

**The primary objectives of the Personal Diary project are to:**

1. **Facilitate Writing:** Allow users to effortlessly write and save their thoughts and experiences in a text-based format.
2. **Enable Easy Access:** Provide a straightforward way to view all entries, allowing users to reflect on their past experiences.
3. **Support Editing and Deleting:** Allow users to modify or remove entries, ensuring that their diary remains relevant and accurate.
4. **Utilize File Management:** Implement basic file handling techniques to store diary entries in a text file, ensuring data persistence between sessions.

## **Features :**

**Entry Creation:** Users can create new diary entries with ease.

**Entry Viewing:** The application allows users to read all saved entries, enhancing reflection and mindfulness.

**Entry Editing:** Users can edit existing entries to update information or correct mistakes.

**Entry Deletion:** The application provides functionality to delete specific entries, giving users control over their content.

# CODE

```
#include <iostream>
#include <fstream>
#include <string>
#include <vector>

class Diary { private: std::string
filename;

public: Diary(const std::string& file) :
filename(file) {}

void writeEntry(const std::string&
entry)

{ std::ofstream outFile(filename,
std::ios::app); if (outFile) { outFile <<
entry << std::endl; }
}

void readEntries()
{ std::ifstream inFile(filename);
std::string line; while (getline(inFile,
line)) { std::cout << line << std::endl; }
}
```

```

void deleteEntry(int entryNumber) {

    std::ifstream inFile(filename);
    std::vector<std::string> entries;
    std::string line;

    while (getline(inFile, line)) {
        entries.push_back(line);
    }

    if (entryNumber >= 1 && entryNumber
        <= entries.size()) {
        entries.erase(entries.begin() +
            entryNumber - 1);
    }

    std::ofstream outFile(filename); for
    (const auto& entry : entries) { outFile <<
        entry << std::endl;
    }
}

void editEntry(int entryNumber, const std::string& newEntry) {

    std::ifstream inFile(filename);
    std::vector<std::string> entries;
    std::string line;

    while (getline(inFile, line))
    { entries.push_back(line);
    }

    if (entryNumber >= 1 && entryNumber <=
        entries.size())

        { entries[entryNumber - 1] = newEntry;

        } std::ofstream outFile(filename); for (const auto&
        entry : entries) { outFile << entry << std::endl; }
    }
};

```

```

int main()
{ Diary diary("diary.txt");
int choice;
do {

std::cout << "\n1. Write Entry\n2. Read Entries\n3. Edit
Entry\n4. Delete Entry\n5. Exit\n";

std::cout << "Choose an option: ";

std::cin >> choice; std::cin.ignore();

if (choice == 1) { std::string entry; std::cout
<< "Enter your diary entry: ";

std::getline(std::cin, entry); diary.writeEntry(entry);
}

else if (choice == 2) { diary.readEntries(); } else if (choice ==
3) {

if (choice == 1)
{ std::string entry;
std::cout << "Enter your diary entry: ";
std::getline(std::cin, entry);
diary.writeEntry(entry);
}
else if (choice == 2) { diary.readEntries();
}
else if (choice == 3) {

```

```
    int entryNumber;
    std::string newEntry;
    std::cout << "Enter the entry number to edit: ";
    std::cin >> entryNumber;
    std::cin.ignore();
    std::cout << "Enter the new entry: ";
    std::getline(std::cin, newEntry);
    diary.editEntry(entryNumber, newEntry);

} else if (choice == 4) {
    int entryNumber;

    std::cout << "Enter the entry number to delete: ";
    std::cin >> entryNumber;
    diary.deleteEntry(entryNumber);
}

} while (choice != 5);

return 0;
}
```

# OUTPUT

## 1. Launching the Program:

markdown

Personal Diary Menu:

1. Write Entry
2. Read Entries
3. Edit Entry
4. Delete Entry
5. Exit

Choose an option:

## 2. Writing an Entry:

mathematica

Choose an option: 1

Enter your diary entry: Today was a great day!

Entry added successfully.

## 3. Reading Entries:

yaml

Choose an option: 2

Diary Entries:

Today was a great day!

## 4. Writing Another Entry:

mathematica

Choose an option: 1

Enter your diary entry: I learned something new in C++.

Entry added successfully.



# OUTPUT

## 5. Reading Entries Again:

mathematica

```
Choose an option: 2
Diary Entries:
Today was a great day!
I learned something new in C++.
```

## 6. Editing an Entry:

mathematica

```
Choose an option: 3
Enter the entry number to edit: 1
Enter the new entry: Today was an amazing day!
```

## 7. Reading Entries After Edit:

mathematica

```
Choose an option: 2
Diary Entries:
Today was an amazing day!
I learned something new in C++.
```

## 8. Deleting an Entry:

vbnet

```
Choose an option: 4
Enter the entry number to delete: 2
```

## 9. Reading Entries After Deletion:

yaml

```
Choose an option: 2
Diary Entries:
Today was an amazing day!
```



# OUTPUT

## 10. Exiting the Program:

```
vbnet
```

```
Choose an option: 5
```

```
Exiting the diary application.
```

## CONCLUSION

The Personal Diary project serves as a practical application of fundamental programming concepts in C++. Through the development of this console-based diary application, key skills such as file handling, dynamic data management, and user interaction have been effectively demonstrated.

This project allows users to create, read, edit, and delete diary entries, providing a straightforward way to record personal thoughts and experiences. The use of a text file for storage ensures that entries are persistent, allowing users to access their diary over multiple sessions.

Moreover, the implementation highlights important programming practices such as modular design, data validation, and error handling. It also lays a solid foundation for future enhancements, such as integrating a graphical user interface, adding search functionality, or implementing features like password protection.