**Proiect Programare si limbaj de programare 2 C++/C**

“Sistem de gestiune a fisierelor”:

Un program care sa creeze,sa citeasca,sa modifice si sa stearga un fisier

**Studenți: Profesori coordonatori:**

Boza Silviu Gabriel Olteanu Gabriela

Neculoiu Georgian

Cuprins

1. Introducere…………………………………………………….....1
2. Analiza detaliata a codul………………………………………………..3
3. Structura principala………………………………………………………3
4. Functii implementate: Creerea de fisier…………………………4
5. Functii implementate: Citirea fisier………………………………….5
6. Functii implementate: Modificarea fisierului……………………..6
7. Functii implementate: Stergerea Fisierului………………………..7
8. Utilizarea codului in alte aplicatii……………………………………..9
9. Analiza rezultatelor……………….………………………………………10
10. Posibile imbunatatiri………………………………….………………….10
11. Codul integral………………….……………………………….…………..11
12. Concluzie………………………………………………..…………………..13
13. Bibliografie………….……………..…………………..……………………14

# 

# I. Introducere

Proiectul „Sistem de gestiune a fișierelor” are ca scop dezvoltarea unei aplicații în limbajul C++ care permite utilizatorului să gestioneze fișiere text în mod interactiv, prin intermediul unei interfețe simple în linia de comandă. Gestiunea fișierelor este o componentă esențială în orice sistem de operare și aplicație. informatică Funcționalitățile principale sunt:

* Proiectul „Sistem de gestiune a fișierelor” are ca scop dezvoltarea unei aplicații în limbajul **Crearea** unui fișier nou
* **Citirea** conținutului unui fișier
* **Modificarea** (suprascrierea) fișierului
* **Ștergerea** fișierului

Acest tip de aplicație este util pentru învățarea lucrului cu fluxuri de fișiere (fstream) și interacțiunea de bază cu sistemul de operare.

# II. Analiza detaliata a codului

## 2.1. Structura peincipala”:

Int main() {

// Afiseaza meniul si citeste optiunea utilizatorului

Se foloseste un meniu interactive pentru a permite utilizatorului sa aleaga actiuni.

Optiunile sunt procesate printr-un switch,in functie de input-ul utilizatorului.

.

## 

## 2.2. Functii implementate: Creerea de fisier

creeazaFisier()

ofstrem fisiser(test.txt);

Creeaza un fisier gold aca nu exista deja

Daca exista,il suprascrie (implicit in ofstream)

A screenshot of a computer

AI-generated content may be incorrect.

## 

## 2.3. Functii implementate: Citirea fisierului

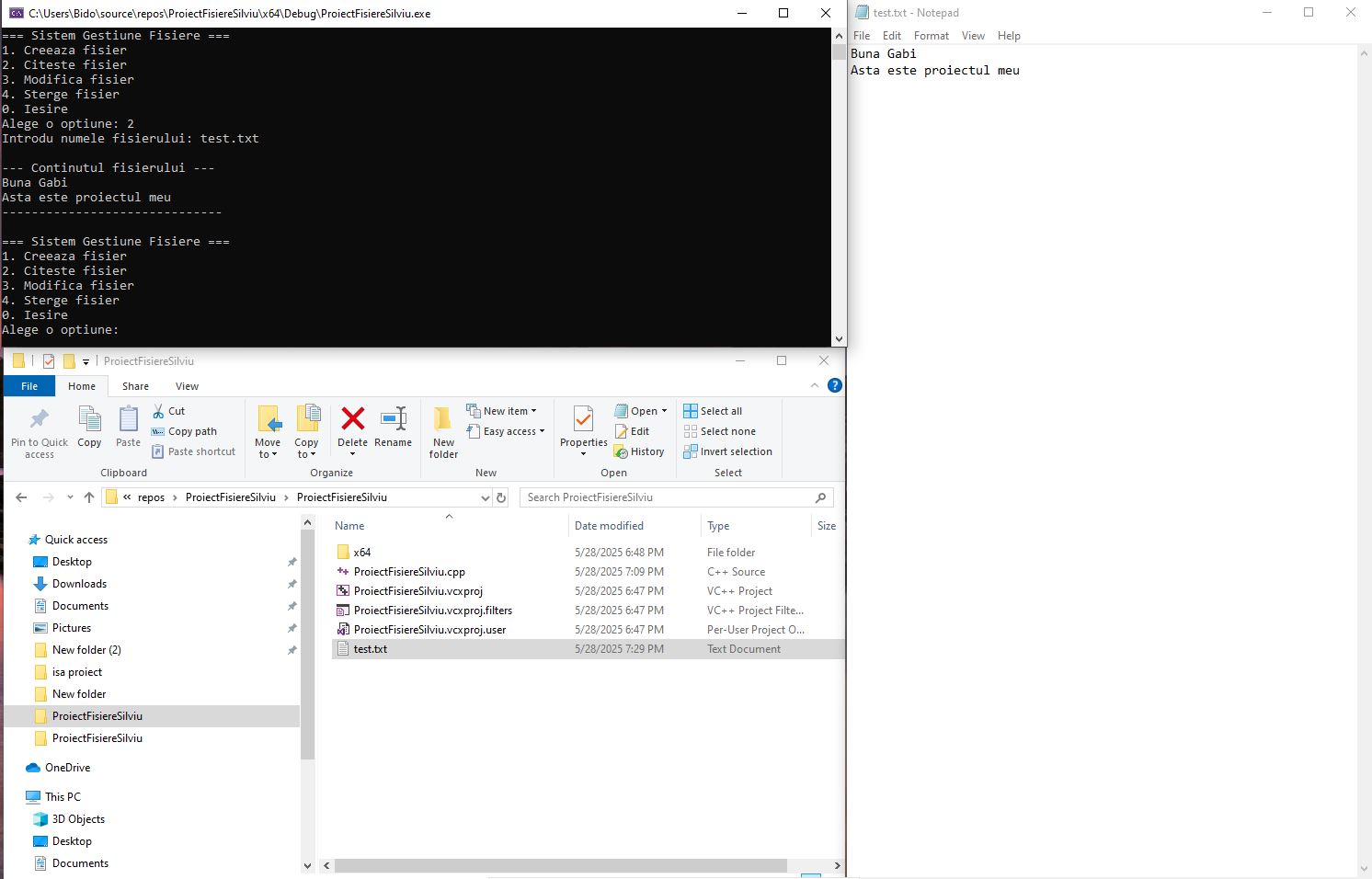
citesteFisier()

ifstream fisier(test.txt);

getline(fisier,linie);

Deschide fisierul in mod “read-only”.

Afiseaza continutul linie cu linie.

****

## 

## 2.4. Functii implementate: Modificarea fisierului

modificaFisier()

ofstream fisier(text.txt);

fisier <<linie <<end1;

Deschide fisierul in mod “write”,suprascriind continutul

Permite introducerea de texp pana la o linie goala

# 

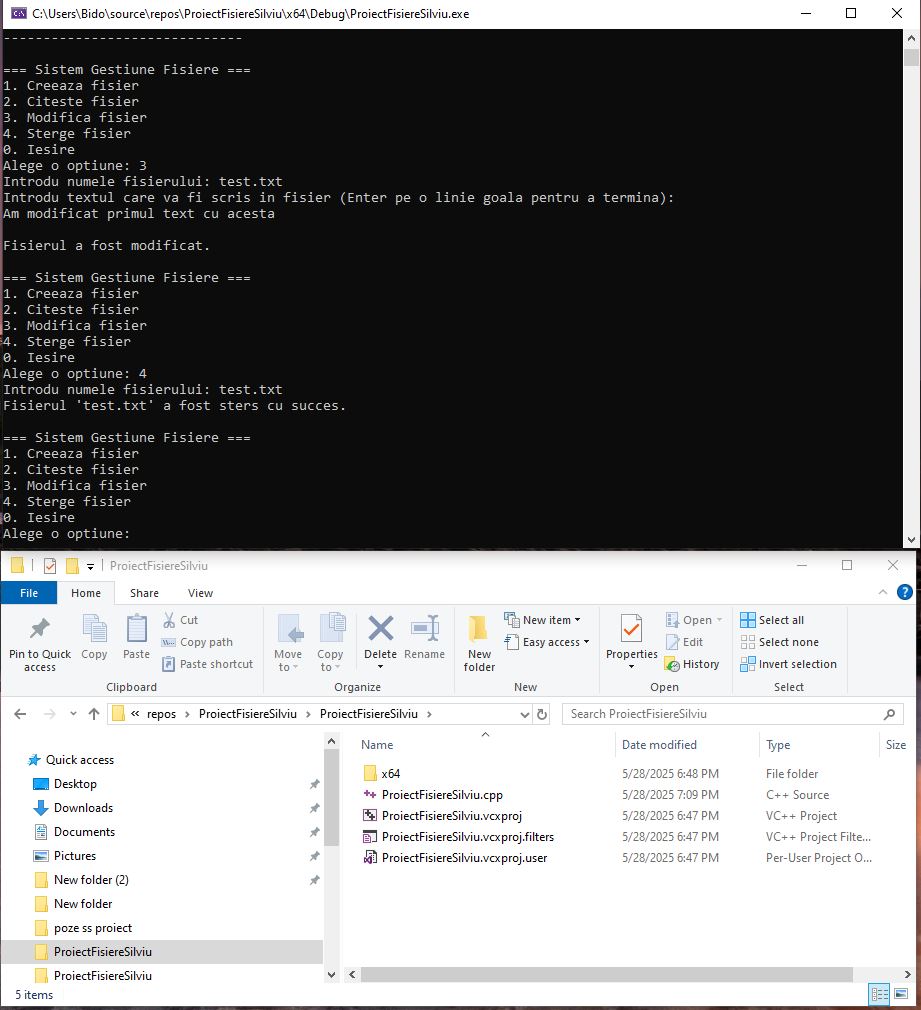
## 2.5. Functii implementate: Stergerea Fisierului

stergeFisier()

remove(test.txt.c\_str());

Foloseste functia C remove() pentru a sterge un fisier

Returneaza 0 la success,astfel afiseaza eroare



 **Fluxuri de fișiere**: C++ utilizează obiectele ifstream, ofstream și fstream pentru citire, scriere și ambele simultan.

 **Tratamentul erorilor**: Este implementat minim (ex: verificarea dacă fișierul a fost deschis cu succes). Codul poate fi extins cu tratamente specifice de excepții (try-catch).

 **Funcționalitate interactivă**: Programul rulează într-o buclă do-while, ceea ce permite utilizatorului să efectueze mai multe acțiuni într-o singură sesiune.

 **Compatibilitate multiplatformă**: Codul este portabil, fiind scris folosind doar biblioteci standard.

## 2.6. Utilizarea codului si in alte aplicatii:

Codul programului poate sa ruleze si in alte aplicatii,nu doar in VS 2022,cum ar fi “GDB Online Debugger”

A screenshot of a computer

AI-generated content may be incorrect.

## 3. Analiza rezultatelor

Aplicația a fost testată în Visual Studio 2022 pe Windows 10. Funcțiile de creare, citire, modificare și ștergere au fost validate cu mai multe tipuri de fișiere și conținut.

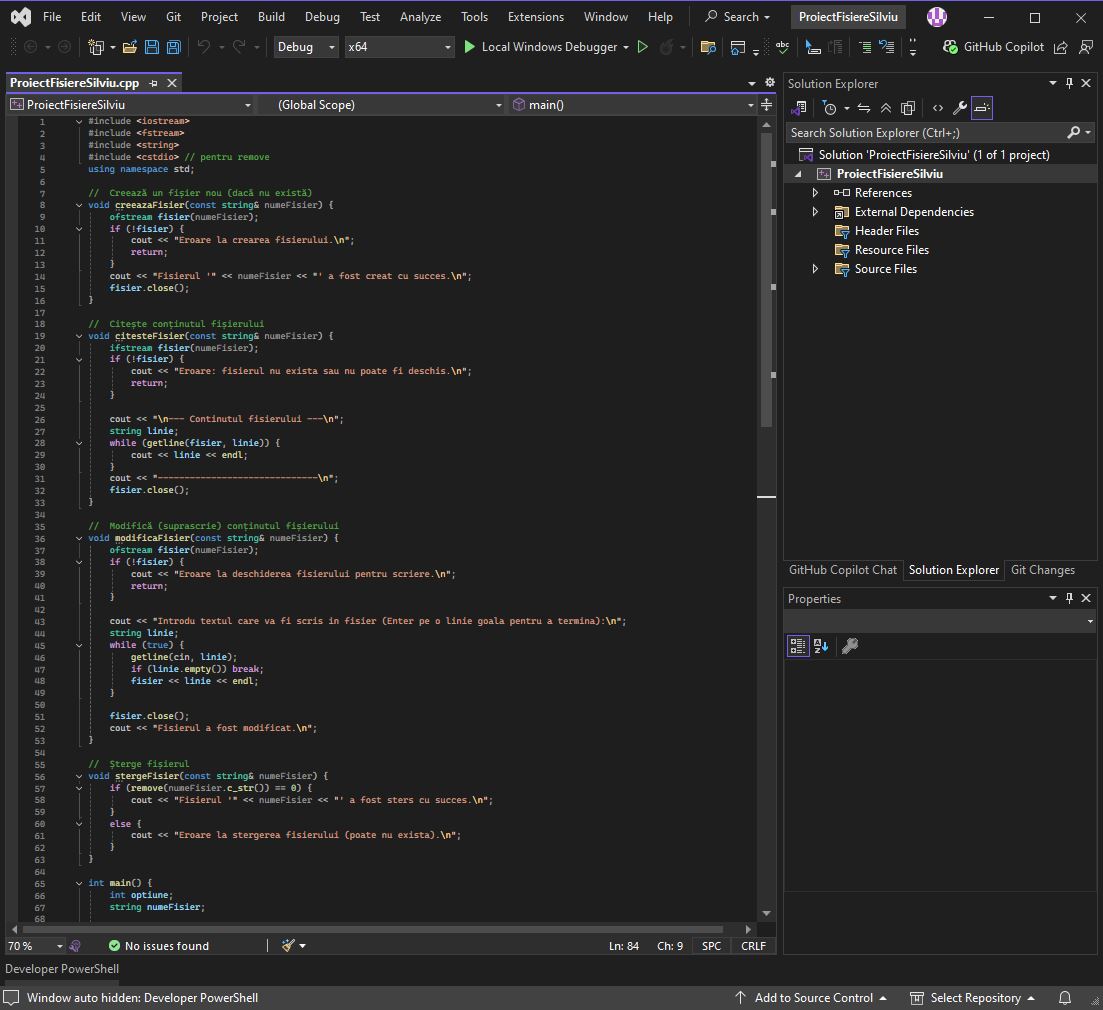
**Observații:**

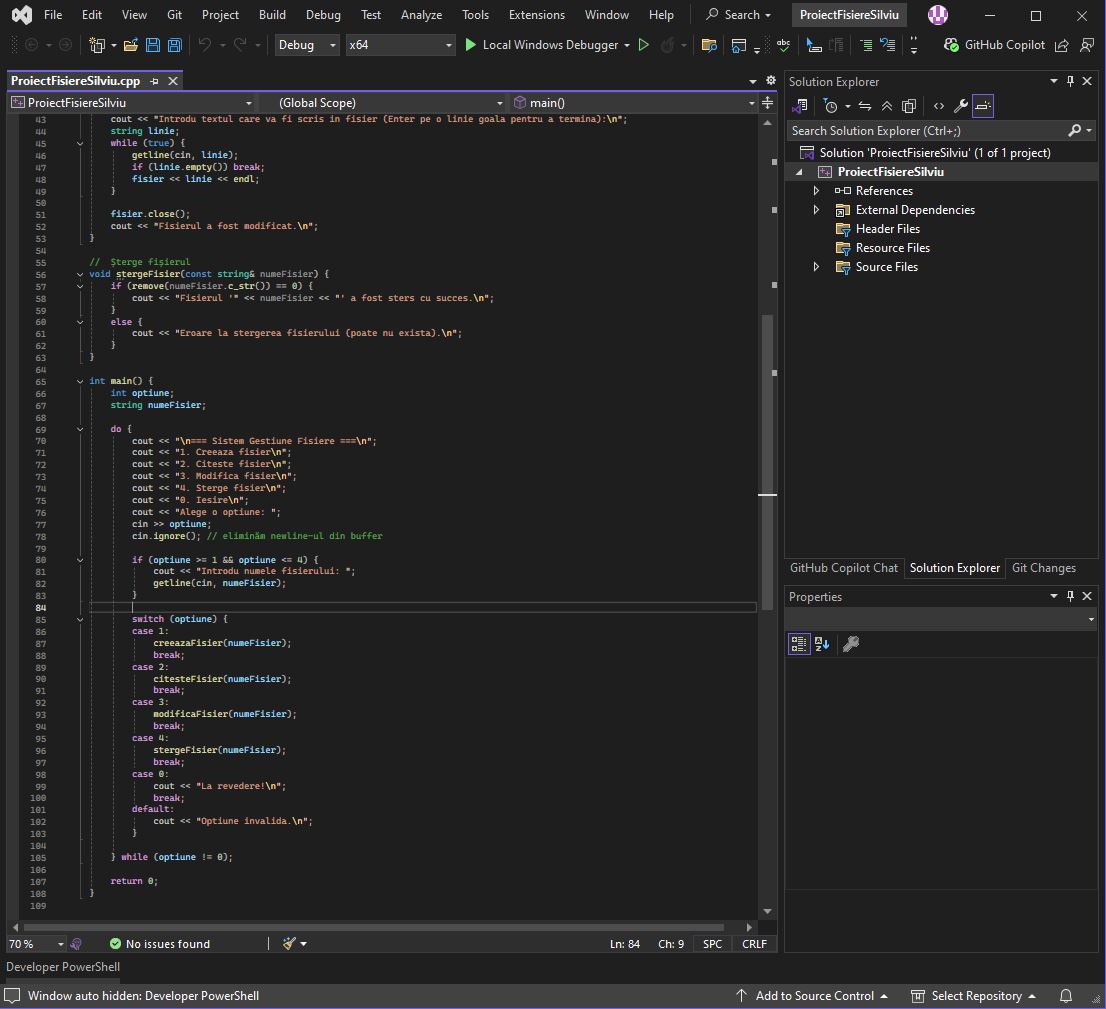
* Utilizatorul poate crea și edita oricâte fișiere dorește, atâta timp cât are permisiuni de scriere.
* Nu apar erori critice dacă fișierul nu există, dar afișarea de mesaje informative poate fi îmbunătățită.
* Utilizatorul trebuie să introducă numele fișierului manual — nu există completare automată sau validare avansată.

## 4. Posibile imbunatatiri:

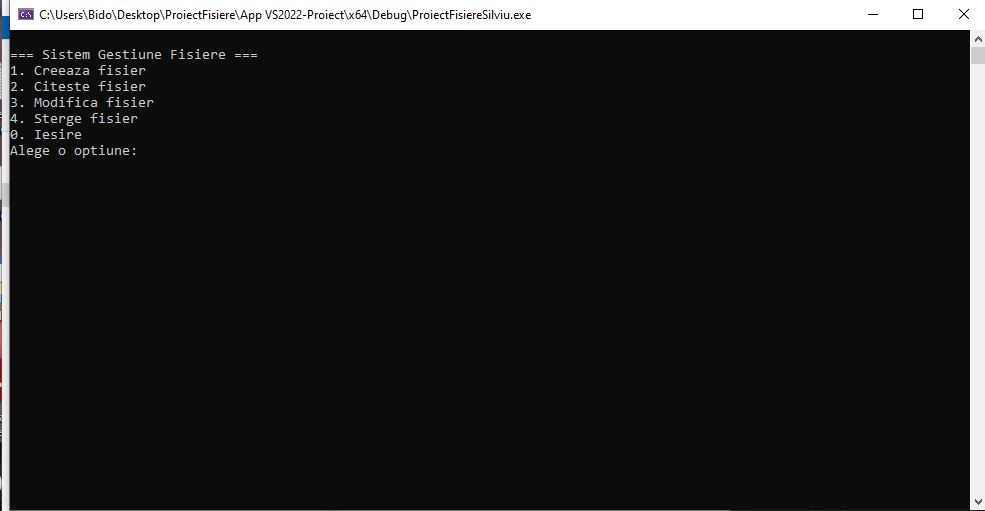
|  |  |
| --- | --- |
| Adăugare în fișier | Deschiderea fișierului în mod ios::app pentru a păstra conținutul existent |
| Listarea fișierelor | Integrarea unei funcții care afișează toate fișierele .txt din director |
| Jurnalizare/logging | Salvarea fiecărei acțiuni într-un fișier „log.txt” |
| Validare input | Verificarea ca numele fișierului să nu conțină caractere nepermise (?, \*, : etc.) |
| Interfață grafică | Folosirea bibliotecii Qt pentru a realiza o versiune cu butoane, meniuri și ferestre |
| Criptare simplă | Aplicarea unui algoritm XOR sau Caesar pentru criptarea textului din fișier |

## 5. Codul integral:





Dupa ce codul a fost scris cu succes se foloseste comanda Ctrl+ F5 sau se apasa pe Local Windows Debugger pentru a rula codul si a deschide aplicatia de unde putem creea,modifica,sterge si citii un fisier aratand in felul urmator:



## 6. Concluzie

Proiectul „Sistem de gestiune a fișierelor” oferă o bază solidă pentru înțelegerea lucrului cu fișiere în C++. Codul este clar, modular și poate fi extins cu ușurință. Fiind scris într-un limbaj compilat precum C++, asigură eficiență și control total asupra operațiilor de I/O.

Această aplicație poate servi ca punct de plecare pentru proiecte mai complexe, cum ar fi:

* un editor de fișiere simplificat
* un program de backup automatizat
* un sistem de organizare a fișierelor personale

Prin extinderea funcționalităților, acest proiect poate evolua într-o aplicație reală, cu impact educațional sau chiar profesional.

# 

# Biblliografie

**cplusplus.com – File streams**

Tutorial oficial despre lucrul cu fișiere în C++, explică utilizarea ifstream, ofstream și fstream.

**GeeksforGeeks – File Handling in C++**

Explicații clare și moderne despre clasele pentru fișiere și modul de citire/scriere în C++.

[**Microsoft Learn – Working with files in C++**](https://learn.microsoft.com/en-us/cpp/standard-library/fstream?view=msvc-170)

Documentația oficială Microsoft pentru lucrul cu fișiere folosind biblioteca standard C++ în Visual Studio.

**Programiz – C++ File Handling**

Un ghid didactic cu exemple simple, orientat spre începători.

[**Stack Overflow – C++ file input/output questions**](https://stackoverflow.com/questions/tagged/c%2b%2b+file-io)

Forum activ cu întrebări și soluții legate de gestiunea fișierelor în C++. Util pentru rezolvarea erorilor.

**TutorialsPoint – C++ Files and Streams**

Prezentare structurată a operațiilor cu fișiere în C++.