AKADEMIA GÓRNICZO-HUTNICZA

Dokumentacja do projektu

Biblioteka do obsługi systemu plików

z przedmiotu

Języki Programowania Obiektowego

Elektronika i Telekomunikacja, III rok

Oskar Kierys

Piątek, godz. 13:15

prowadzący: mgr. Inż. Jakub Zimnol

09.01.2025

Opis projektu

Projekt to aplikacja konsolowa do zarządzania katalogami i plikami .txt. Umożliwia wykonywanie operacji takich jak przeglądanie katalogów, wyświetlanie drzewa katalogów, tworzenie, modyfikowanie, odczytywanie i usuwanie plików.

Struktura projektu

- **Plik** `DirectoryManager.hpp/cpp`: Obsługuje operacje związane z katalogami.
- **Plik** `FileManager.hpp/cpp`: Obsługuje operacje na plikach.
- **Plik** `main.cpp`: Punkt wejścia aplikacji i implementacja interfejsu użytkownika.

Główne klasy i funkcjonalności

Klasa `DirectoryManager`

Obsługuje operacje na katalogach.

Atrybuty

- `m_currentDirectory`: Ścieżka do bieżącego katalogu.

Metody

1. `DirectoryManager(const std::string& directoryPath)`
   Inicjalizuje bieżący katalog.
2. `void changeDirectory(const std::string& newDirectoryPath)`
   Zmienia bieżący katalog na podany, jeśli istnieje.
3. `std::string getCurrentDirectory() const`
   Zwraca ścieżkę do bieżącego katalogu.
4. `std::vector<std::string> listFilesInDirectory(const std::string& directoryPath)`
   Zwraca listę plików w podanym katalogu.
5. `bool fileExists(const std::string& fileName) const`
   Sprawdza, czy plik o podanej nazwie istnieje w bieżącym katalogu.
6. `void displayTree(const std::string& directoryPath, int indent = 0)`
   Wyświetla strukturę katalogów w formie drzewa z wcięciami.
7. `std::uintmax_t calculateSize(const std::string& directoryPath)`
   Oblicza całkowity rozmiar plików w katalogu.

Klasa File

Obsługuje operacje na plikach .txt.

Atrybuty

- m_filePath: Ścieżka do pliku.

Metody

1. File(const std::string& path)
   Konstruktor ustawiający ścieżkę do pliku.
2. bool createFile()
   Tworzy nowy plik i otwiera go w domyślnym edytorze (tylko Windows).
3. std::string readFile()
   Odczytuje zawartość pliku.
4. bool writeFile(const std::string& content)
   Dopisuje zawartość do pliku.
5. bool deleteFile()
   Usuwa plik po potwierdzeniu użytkownika.
6. void openInEditor()
   Otwiera plik w domyślnym edytorze (Windows).

Funkcja listFilesInDirectory (z main.cpp)

Wyświetla zawartość katalogu, w tym katalogi oraz pliki .txt wraz z ich rozmiarami.

Menu użytkownika

Aplikacja posiada interfejs konsolowy z następującymi opcjami:

- a: Wyświetla bieżący katalog.
- c: Zmienia katalog.
- n: Tworzy nowy plik .txt.
- l: Wyświetla listę plików w bieżącym katalogu.
- t: Wyświetla strukturę drzewa katalogów.
- r: Odczytuje zawartość pliku .txt.
- m: Dopisuje zawartość do pliku .txt.
- d: Usuwa plik .txt.
- q: Zamyka aplikację.

Przykładowe użycie

1. **Tworzenie pliku**: Użytkownik wybiera opcję **n**, wprowadza nazwę pliku z rozszerzeniem .txt, a następnie edytuje go w domyślnym edytorze (na Windows).
2. **Wyświetlanie drzewa katalogów**: Opcja **t** pokazuje strukturę katalogów z wcięciami.
3. **Odczyt pliku**: Opcja **r** odczytuje zawartość wybranego pliku .txt.

Obsługiwane platformy

Obsługa otwierania plików w edytorze jest dostępna tylko na Windows. Na innych systemach funkcja `openInEditor` jest ignorowana.

Kompilacja projektu

Do kompilacji projektu wymagany jest kompilator obsługujący standard C++20. Aby skompilować projekt, użyj następującego polecenia:

```
g++ -std=c++20 main.cpp file.cpp directory.cpp -o fileman
```

Uruchomienie aplikacji odbywa się za pomocą komendy:

```
./fileman
```