

Backup di Emergenza

Progetto di Programmazione di Sistema

Gruppo 32:
Gianluca Maida - s334263
Matteo Petrera - s331356
Davide Proglio - s324103

Problemi e Soluzioni

Il backup può essere critico in situazioni di emergenza. Senza uno schermo funzionante, i metodi tradizionali falliscono. Si ha la necessità di un sistema alternativo affidabile.

Un'applicazione che esegue il backup tramite input con il mouse. Funzionamento in background con minimo impatto sulle prestazioni. Conferma del backup con un secondo input.

Backup di Emergenza

Obiettivi Del Progetto

01

Realizzare un sistema di backup di emergenza con interfaccia grafica.

02

Consentire all'utente di avviare il backup tracciando un segno sul monitor.

03

Registrare metriche di utilizzo della CPU e statistiche di backup.

Architettura dell' Applicazione

Moduli Principali

01 main.rs

Gestione principale

Coordina l'esecuzione dell'applicazione, inizializza i moduli e gestisce l'avvio in background.

02 mouse_input.rs

Riconoscimento input

Rileva i movimenti del mouse e verifica la correttezza del segnale per avviare il backup.

03 backup.rs

Esecuzione del backup

Copia i file dalla sorgente alla destinazione, scrivendo un log dettagliato.

04 logging.rs

Monitoraggio CPU e backup

Registra il consumo di CPU ogni 2 minuti e salva statistiche dettagliate sul backup completato.

Flusso di Esecuzione

L'applicazione segue un flusso strutturato per garantire un backup sicuro ed efficiente. Ogni fase è progettata per minimizzare l'interazione dell'utente, consentendo un utilizzo semplice anche in situazioni critiche.

Fase di configurazione: Se non sono stati definiti i path sorgente e destinazione del backup, viene mostrata una finestra di setup.

Attivazione del backup: L'utente esegue il movimento specifico del mouse per iniziare il processo.

Conferma backup: Viene mostrata una finestra di conferma oppure, in assenza di schermo, si può confermare tracciando un segno meno.

Esecuzione del backup: I file vengono copiati e viene emesso un segnale acustico al completamento.

Registrazione dati: Viene aggiornato il file di log con dettagli sul backup e il consumo della CPU.

Configurazione

main.rs

main_configuration.rs

configuration.csv

L'applicazione utilizza un file di configurazione per permettere all'utente di personalizzare le impostazioni del backup senza bisogno di modificare manualmente il codice.

File di configurazione

Contiene i percorsi di origine e destinazione del backup.

Prima esecuzione

Se non sono stati definiti i path sorgente e destinazione del backup, una finestra grafica permette di selezionare i percorsi.

Modifica della configurazione

L'utente può cambiare le impostazioni, eseguendo setup.exe.

Eventi Mouse

`mouse_input.rs`

L'applicazione utilizza gli eventi del mouse per attivare e confermare il backup senza bisogno di un'interfaccia grafica. Il riconoscimento dei movimenti permette un utilizzo rapido ed efficace anche in situazioni di emergenza.

Inizializzazione

Inizializzazione di variabili globali per gestire la posizione del mouse e i punti di riferimento.

Attivazione Backup

La sequenza di movimenti del mouse per attivare il backup segue i quattro angoli dello schermo.

Conferma Backup

La conferma avviene tracciando un segno meno con il mouse, in alternativa attraverso la finestra di conferma.

Segnale Sonoro

Viene emesso un segnale sonoro dopo la fase di conferma e dopo il termine del backup.

Backup

backup.rs

backup_log.txt

Copiare i file dalla cartella sorgente alla destinazione registrando alcuni dati di rilevanza

Inizio del processo

Inizio del processo, si avvia un timer per calcolare il tempo totale e si inizializzano delle variabili per raccogliere dati relativi al backup.

Creazione della cartella di destinazione

Se la cartella di destinazione non esiste, la crea.

Backup ricorsivo dei file e cartelle

Scansiona il contenuto della cartella sorgente. Fa una copia di ogni file e ogni cartella nella destinazione di backup.

Scrittura del log di backup

Registra il tempo totale impiegato, dimensione complessiva dei file copiati ed elenco dettagliato dei file trasferiti in un file di log nella cartella di destinazione.

Logging Cpu

`cpu_logger.rs`

`cpu_usage_log.txt`

Il sistema di logging monitora il consumo di CPU. Questo aiuta a verificare le prestazioni dell'applicazione e ottimizzarne il comportamento nel tempo.

Inizializzazione del sistema e del file di log

Crea un'istanza di System per monitorare i processi, definisce il percorso del file di log e lo apre.

Loop di monitoraggio CPU

Il programma esegue un ciclo infinito per aggiornare i dati della CPU.

Scrittura del log CPU

Nel file di testo si registra data, ora e consumo di CPU ogni due minuti.

Pausa tra un aggiornamento e l'altro

Il monitoraggio attende 120 secondi, il ciclo si ripete per registrare nuovi dati.

Tecnologie Utilizzate

Librerie

- | | |
|--|--|
| <p>01 sysinfo - Monitoraggio CPU</p> <hr/> | <p>06 scrap - Rilevazione dimensioni schermo</p> |
| <p>02 chrono - Timestamp nei log</p> <hr/> | <p>07 rodio - Riproduzione suoni di conferma</p> <hr/> |
| <p>03 lazy_static - Inizializzazione di variabili globali</p> <hr/> | <p>08 egui ed eframe - Finestra di configurazione e conferma</p> <hr/> |
| <p>04 rdev - Ascolto eventi mouse</p> <hr/> | <p>09 rfd - Selezione delle cartelle tramite finestre di dialogo native</p> <hr/> |
| <p>05 device_query - Tracciamento movimento cursore</p> | <p>10 auto-launch - Avvio automatico dell'app all'accensione del sistema</p> |

Conclusioni e Sviluppi Futuri

01

L'applicazione permette il backup in situazioni critiche senza interfaccia grafica

02

Il consumo della CPU è monitorato e ottimizzato per garantire efficienza.

03

Il sistema di logging permette di tracciare backup e consumi.

Sviluppi futuri

Possibilità di riconoscere più gesti per attivare il backup.

Ottimizzazione ulteriore del consumo CPU con tecniche avanzate di scheduling.

Implementazione di un backup incrementale per ridurre il tempo di esecuzione.