

Progetto Advanced Programming Languages

Fernando Riccioli, Daniele Lucifora

March 3, 2025

Abstract

SiteManager è un'applicazione multi-piattaforma sviluppata utilizzando tre linguaggi di programmazione: C#, C++ e Python. L'obiettivo dell'applicazione è fornire una gestione efficiente dei cantieri edili attraverso un'interfaccia utente intuitiva e funzionalità avanzate di analisi dei dati come:

- Gestione dei Materiali: Tracciamento e gestione dei materiali utilizzati nei cantieri.
- Monitoraggio dei Progressi: Monitoraggio in tempo reale dello stato di avanzamento dei lavori attraverso la gestione delle tasks di cantiere.
- Analisi dei Dati: Analisi dei dati raccolti per ottimizzare le operazioni e migliorare la produttività dei cantieri.
- Generazione di Report: Creazione di report dettagliati per la revisione e la presentazione dei dati.

Architettura del Sistema

L'architettura del sistema è progettata per essere modulare e scalabile, consentendo l'integrazione di nuovi componenti e funzionalità senza compromettere la stabilità del sistema. L'applicazione è suddivisa in diversi moduli, ciascuno responsabile di una specifica parte del sistema:

- Modulo di Interfaccia Utente: Sviluppato in C#, questo modulo gestisce tutte le interazioni con l'utente e presenta i dati in modo intuitivo e accessibile.
- Modulo di Logica di Business: Anch'esso sviluppato in C#, questo modulo contiene la logica principale dell'applicazione, inclusa la gestione dei dati e l'esecuzione delle operazioni principali.
- Modulo di Generazione Statistiche: Sviluppato in C++, questo modulo gestisce le operazioni ad alte prestazioni e l'integrazione con librerie di terze parti per la generazione delle statistiche del cantiere.
- Modulo Analisi dei dati: Sviluppato in Python, questo modulo è responsabile della raccolta, elaborazione e analisi dei dati, nonché della generazione di report dettagliati.

Python

Il codice Python fornito è responsabile della generazione di un report per un cantiere e utilizza la libreria reportlab per creare e formattare il pdf. Il report include informazioni sui task completati, i materiali utilizzati e i costi sostenuti. Il codice inizia con l'importazione delle librerie necessarie: reportlab.pdfgen.canvas per la generazione del pdf. reportlab.lib.pagesizes.A4 per definire le dimensioni della pagina. datetime per ottenere la data e l'ora correnti.

Costanti

Sono definite alcune costanti per la formattazione del pdf: `COLONNA_1` e `COLONNA_2` per le posizioni delle colonne. `ALTEZZA_INIZIO` e `ALTEZZA_FINE` per le posizioni verticali della pagina. `INTERLINEA` e `INTERLINEA_TITOLO` per la spaziatura tra le righe.

Classi

La classe `Cursore` tiene traccia della posizione corrente sulla pagina tramite `x` e `y`, le coordinate correnti. Mentre la classe `Cantiere` rappresenta un cantiere e contiene informazioni sui task, i materiali e i costi

Metodi

`crea_report(cantiere)` crea un nuovo PDF con il nome del cantiere e la data corrente e chiama le funzioni per stampare il titolo e le sezioni del report. `stampa_titolo(nome, data)` stampa il titolo del report e la data nella parte superiore della pagina. `stampa_sezione(titolo, collezione, unità)` stampa una sezione del report con un titolo e una collezione di dati (lista o dizionario). `stampa_intestazione(titolo)` stampa l'intestazione di una sezione. `stampa_lista(lista)` stampa una lista di elementi. `stampa_dizionario(dizionario, unità)`: stampa un dizionario di elementi con unità di misura. `aggiorna_cursore()` aggiorna la posizione del cursore sulla pagina e gestisce il cambio di colonna o di pagina.

C#

Service

La classe `TasksService` gestisce le operazioni CRUD (Create, Read, Update, Delete) per i task associati ai cantieri. Utilizza una connessione MySQL per interagire con il database. I metodi principali includono:

- `OttieniTasks(Cantiere cantiere)`: Restituisce una lista di task per un determinato cantiere.
- `AggiungiTask(Tasks task)`: Aggiunge un nuovo task al database.
- `AggiornaTask(Tasks task)`: Aggiorna un task esistente nel database.
- `EliminaTask(int IdTasks)`: Elimina un task dal database utilizzando il suo ID.

La classe `MaterialeService` gestisce le operazioni CRUD per i materiali utilizzati nei cantieri. Include metodi per ottenere, aggiungere, aggiornare ed eliminare materiali, nonché per assegnare materiali ai cantieri. La logica generale è simile a quella di `TasksService`, con metodi che interagiscono con il database MySQL per eseguire le operazioni richieste. La classe `MaterialeCantiereService` gestisce le operazioni per ottenere i materiali utilizzati in un cantiere specifico. La classe `OperaioService` gestisce le operazioni CRUD per gli operai. Include metodi simili a quella di `TasksService` e `MaterialeService`

Model

La classe `Materiale` rappresenta un materiale utilizzato nei cantieri. Include proprietà per l'identificativo del materiale, il nome, la quantità, l'unità di misura e il costo unitario. La classe `MaterialeCantiere` rappresenta l'assegnazione di un materiale a un cantiere. Include proprietà per l'identificativo dell'assegnazione, l'identificativo del cantiere, l'identificativo del materiale, la quantità utilizzata, e gli oggetti `Cantiere` e `Materiale` associati. Gli altri file presenti nel model sono le classi `Operaio` e `Tasks` con i relativi membri.

C++

Il linguaggio C++ è stato utilizzato principalmente per la gestione delle statistiche. Viene creato un server web che fornisce un endpoint per ottenere le statistiche dal database.

Classi

La classe Database gestisce la connessione al database MySQL e l'esecuzione delle query. Il costruttore della classe inizializza la connessione al database. La funzione OttieniMateriali esegue una query per ottenere i materiali dal database e li restituisce come un vettore di strutture Materiale. Il main() configura il server web utilizzando la libreria Crow. Definisce un endpoint /calcolaCostoMateriali che, quando viene chiamato, riceve un JSON contenente i materiali, calcola il costo totale e restituisce il risultato. Restituisce il costo totale come double.

Metodi

CalcolaCostoTotale(Materiale) è un metodo che calcola il costo totale dei materiali moltiplicando la quantità per il costo unitario di ciascun materiale.