

Sfide Affrontate e Soluzioni Adottate

Durante lo sviluppo del progetto, ho incontrato diverse sfide che hanno richiesto soluzioni creative e adattamenti per essere superate. Ecco le principali sfide affrontate e come sono state risolte:

1. Gestione degli Input dell'Utente

Sfida: La gestione degli input dell'utente si è rivelata complessa, soprattutto nel garantire che gli input fossero validi e nel formato corretto.

Soluzione: Ho implementato metodi specifici per controllare la validità degli input (`checkInt` e `checkDouble`), garantendo che il programma richiedesse all'utente di inserire valori validi fino a quando non veniva fornito un input corretto.

Lezione Appresa: L'importanza di validare sempre gli input dell'utente per prevenire errori e migliorare la robustezza del software.

2. Modularità e Manutenibilità del Codice

Sfida: Organizzare il codice in modo che fosse facilmente manutenibile e modulare.

Soluzione: Ho suddiviso il programma in diverse funzioni, ciascuna responsabile di una specifica parte del menu (es. `prodotti`, `gestioneOfferte`, `cercaOfferte`). Questo ha permesso una maggiore chiarezza e facilità di aggiornamento del codice.

Lezione Appresa: La suddivisione del codice in moduli chiari e ben definiti facilita la manutenzione e l'espansione del progetto.

3. Gestione delle Offerte e Prodotti

Sfida: Integrare correttamente la gestione delle offerte e dei prodotti nel sistema, assicurandosi che le offerte fossero associate ai prodotti esistenti e che potessero essere modificate o eliminate.

Soluzione: Ho sviluppato metodi per cercare, aggiungere, modificare e rimuovere offerte, assicurando che ogni operazione fosse intuitiva e sicura.

Lezione Appresa: La gestione di relazioni complesse tra dati richiede una pianificazione attenta e un design accurato delle classi e dei metodi.

4. Interazione con l'Utente tramite la Console

Sfida: Creare un'interfaccia utente efficace utilizzando solo la console. **Soluzione:** Ho progettato un menu strutturato e chiaro, con messaggi espliciti per guidare l'utente attraverso le diverse operazioni.

Lezione Appresa: Anche con risorse limitate, è possibile creare un'interfaccia utente efficace attraverso una comunicazione chiara e strutturata.

Suggerimenti per Miglioramenti Futuri

1. **Interfaccia Grafica:** Considerare l'implementazione di un'interfaccia grafica (GUI) per migliorare l'usabilità e l'esperienza utente.
2. **Persistenza dei Dati:** Integrare un sistema di database per la persistenza dei dati, consentendo il salvataggio e il recupero dei dati anche dopo la chiusura del programma.
3. **Test Automatizzati:** Implementare test automatizzati per assicurare che tutte le funzionalità del sistema funzionino correttamente e per facilitare la manutenzione e l'espansione del progetto.

In sintesi, il progetto ha rappresentato una sfida stimolante che ha contribuito notevolmente alla mia crescita professionale, fornendo competenze pratiche nella gestione dei dati e nell'interazione con l'utente. Le lezioni apprese saranno fondamentali per i progetti futuri, con un'attenzione particolare alla robustezza del codice, all'usabilità e alla manutenibilità.

Esperienze Personali

Durante il lavoro su questo progetto, ho avuto l'opportunità di approfondire la mia conoscenza della programmazione orientata agli oggetti in Java e di migliorare le mie competenze nella gestione dei dati e nelle interazioni con l'utente tramite la console. Ho imparato l'importanza di implementare controlli rigorosi per l'input dell'utente per garantire la robustezza del sistema e ho sviluppato una maggiore attenzione per la strutturazione del codice in modo modulare e riutilizzabile. Questa esperienza mi ha anche permesso di comprendere meglio come progettare e gestire un'applicazione software completa, dalla fase di ideazione alla realizzazione pratica, affrontando e risolvendo problemi reali durante il processo.