Università degli studi di Napoli
Parthenope

Book Shop Parthenope

Relazione Progetto di Programmazione 3
A.A. 2022/2023

Data consegna: 20/02/2023
Codice gruppo: pd2laxji49

Candidati:
Enrico Madonna 0124002279
Lorenzo Mazza 0124002003
Beniamino Nardone 0124002440

Prof. Angelo Ciaramella
Prof. Emanuel Di Nardo

Book Shop Parthenope

Indice

Requisiti del progetto

Il progetto consiste nella creazione di un'applicazione Java che simuli la vendita di prodotti su internet. Abbiamo usato SceneBuilder come tool grafico compatibile con JavaFX per creare un'applicazione standalone e Gradle per automatizzare il processo di compilazione e gestire le librerie di terze parti (dipendenze) necessarie al progetto.
In aggiunta, abbiamo utilizzato XAMPP per creare un database utilizzando MySql. Abbiamo configurato il database attraverso l'interfaccia web di XAMPP e abbiamo creato le tabelle necessarie per il nostro progetto.
Dopo aver creato il database, abbiamo esportato la sua struttura in un file SQL e lo abbiamo inserito nel file .zip che contiene tutti i file necessari per avviare l'applicazione.

Dato che la traccia* ci dava la libertà di scegliere il tipo di prodotti da vendere, abbiamo deciso di creare un negozio di libri, ispirandoci a siti come Amazon e Feltrinelli, rispettando i requisiti richiesti. Ogni libro infatti è identificato da un codice ISBN, nome, descrizione, quantità di scorta, costo, categoria e sottocategoria.
Come richiesto, l'accesso può essere effettuato come amministratore o come cliente.

L'amministratore può effettuare le seguenti operazioni:

- inserire un nuovo prodotto con le sue informazioni
- eliminare un prodotto
- visualizzare periodicamente tutti gli acquisti effettuati da un dato utente

Il cliente può effettuare le seguenti operazioni:

- inserire i prodotti nel carrello della spesa
- eliminare un prodotto precedentemente inserito nel carrello
- effettuare il pagamento. Il pagamento può avvenire secondo le modalità: contanti, carta di credito o bancomat.

Abbiamo riservato la possibilità di registrazione solo ai clienti. Mentre sarà possibile loggarsi come amministratore usando le credenziali:

➔ Username: admin
➔ Password: admin

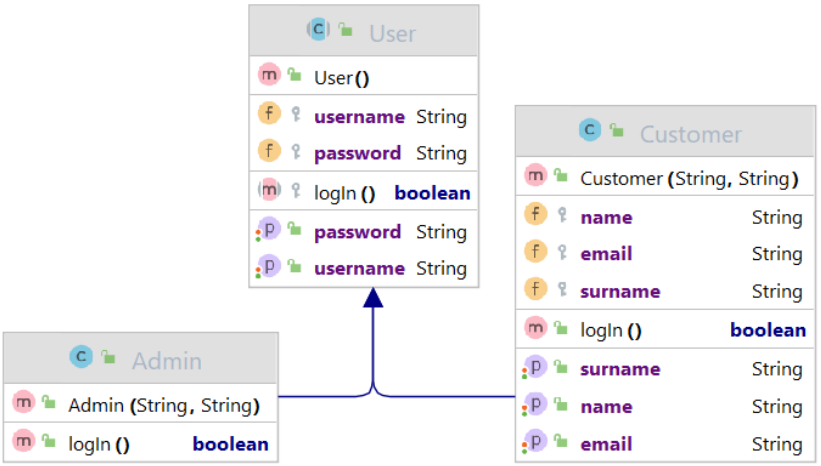
*Viene allegata alla relazione anche il file PDF contenente la traccia originale del progetto.

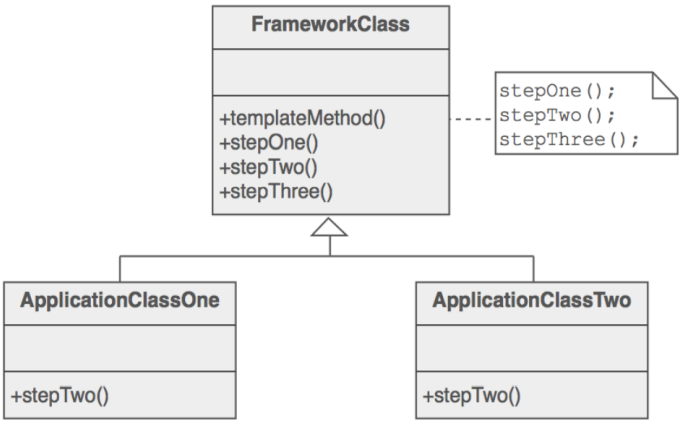
Diagrammi UML delle classi

Abbiamo estrapolato il diagramma UML delle classi usando IntelliJ, si allega il file del diagramma completo, mentre qui evidenzieremo solo alcune classi inerenti all'uso dei Design Pattern.

Template Method

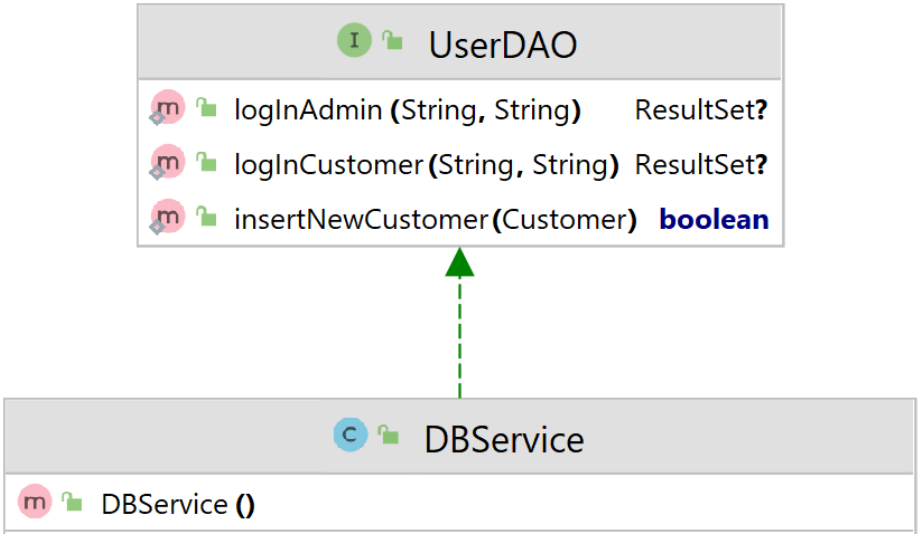
Per gestire l'accesso abbiamo usato il Template Method andando a dividere la logica di accesso per Amministratori e Clienti.
Nel nostro caso, abbiamo applicato questo pattern creando una classe astratta chiamata "User" in cui è creato il metodo "login".
Le classi Customer e Admin estendono User implementando in maniera specifica il metodo login().

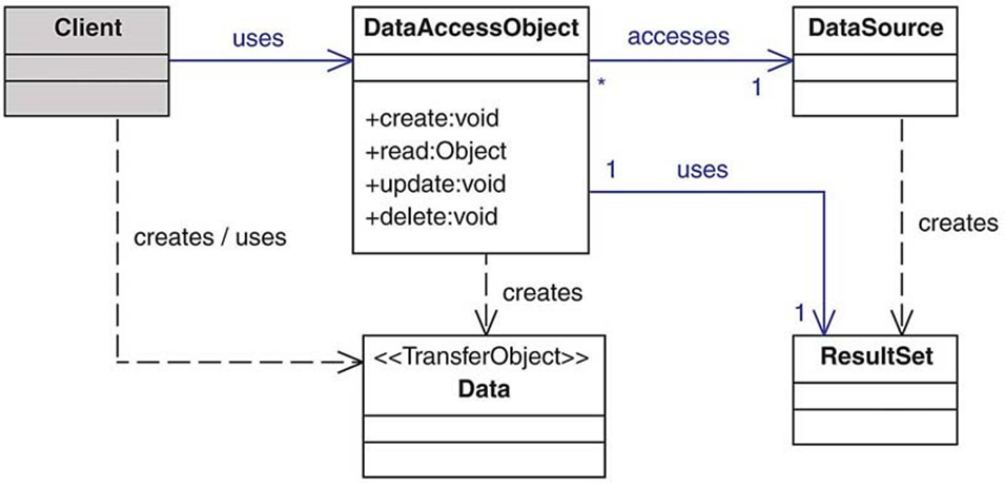




Data Access Object

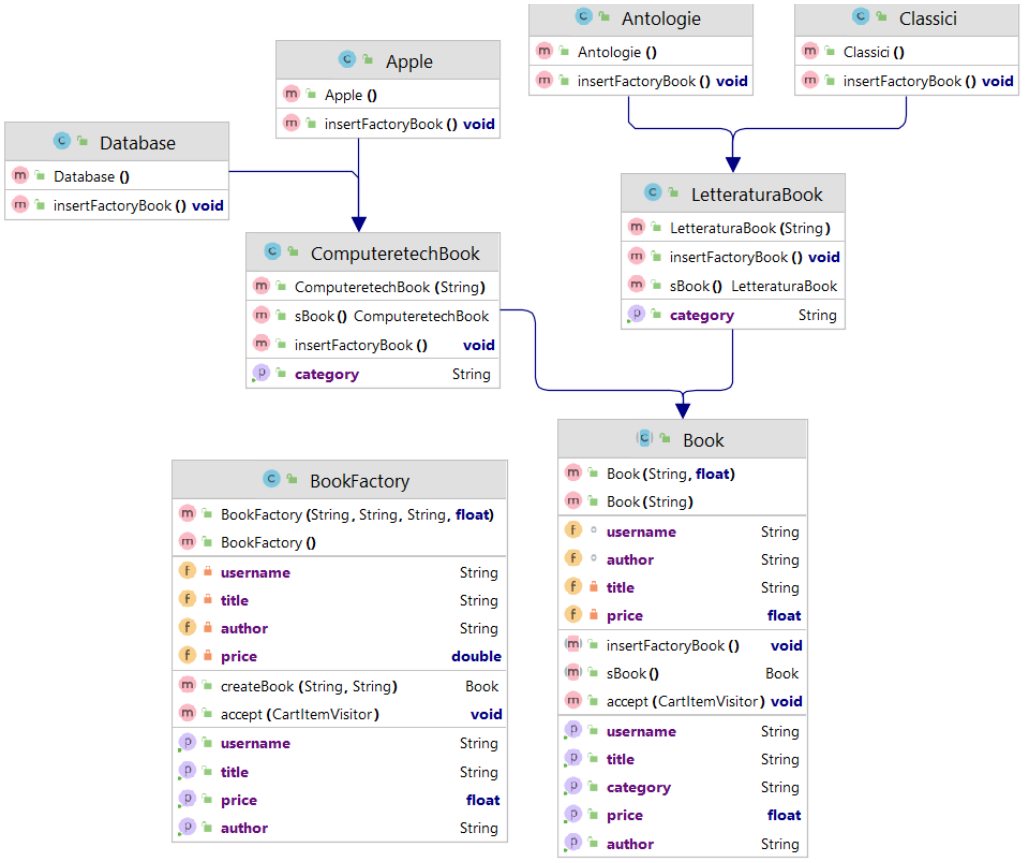
Abbiamo usato il pattern DAO per separare la logica di accesso ai dati per quanto riguarda il login degli utenti; in modo che per poter accedere al DataBase dovrà essere usata l'interfaccia DAO.

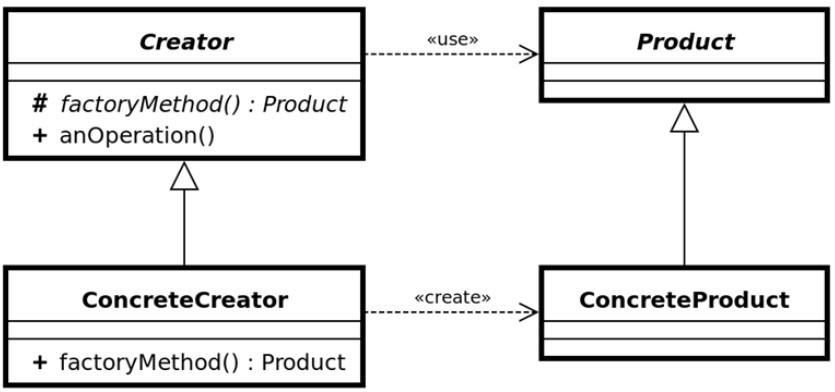




Factory Method

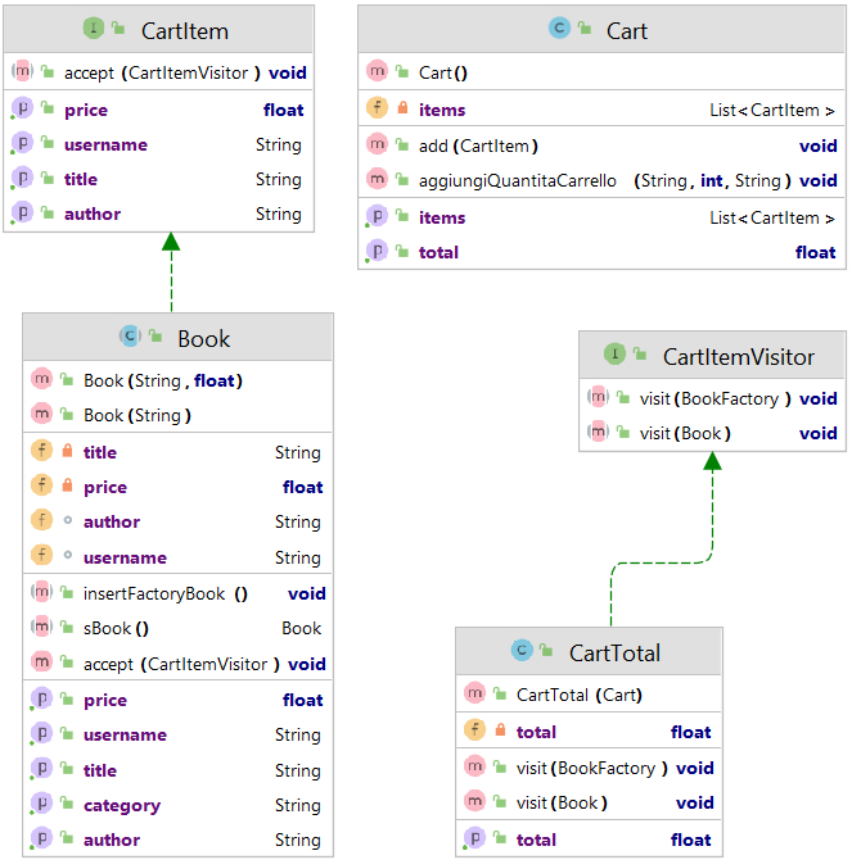
Per occuparci della logica della creazione dei libri (i nostri prodotti), abbiamo usato il Factory Method creando una classe per ogni tipo di libro (categoria del prodotto) che dev'essere creato. Qui sono presenti solo alcune categorie di esempio.

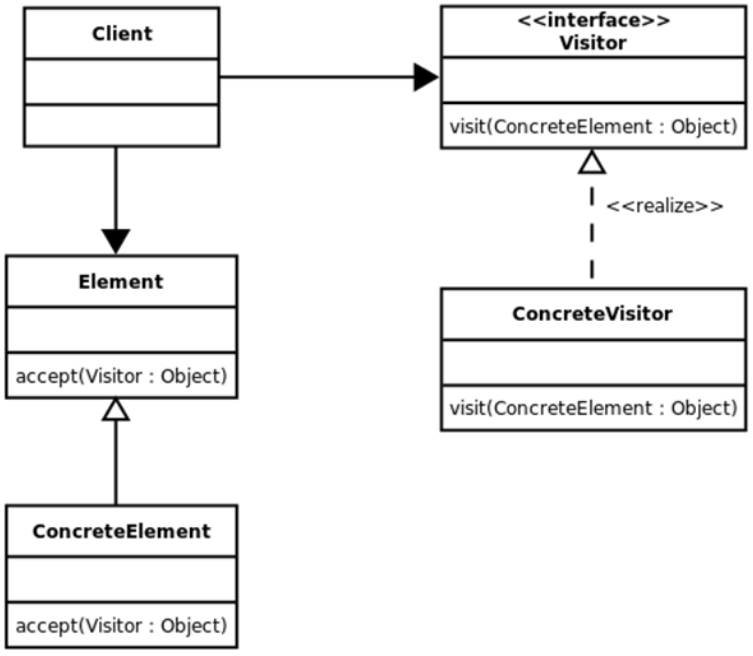




Visitor Pattern

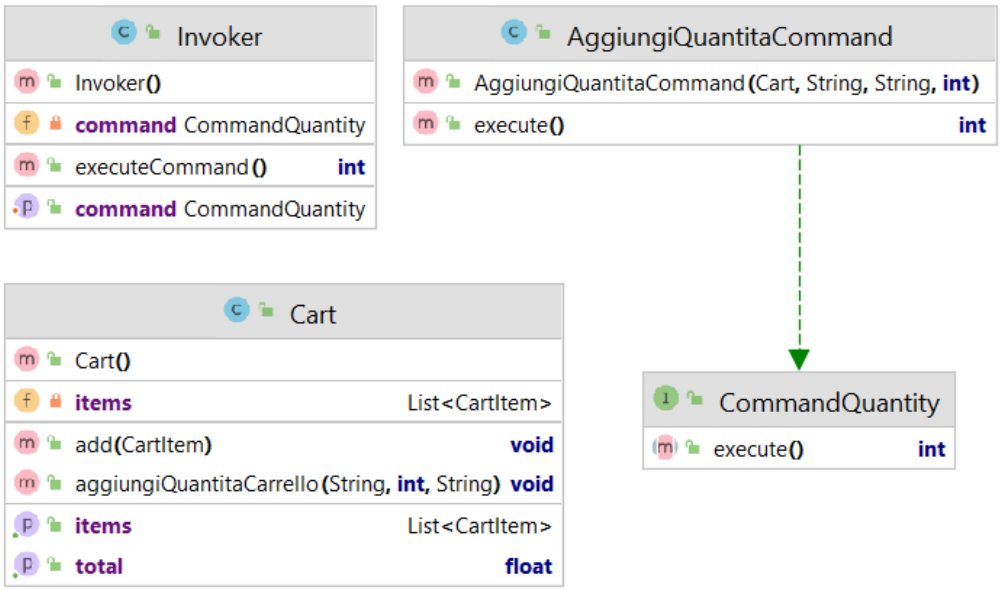
Per la visualizzazione e il calcolo del totale dei libri nel carrello di ogni utente abbiamo optato per l'utilizzo del Visitor Pattern.

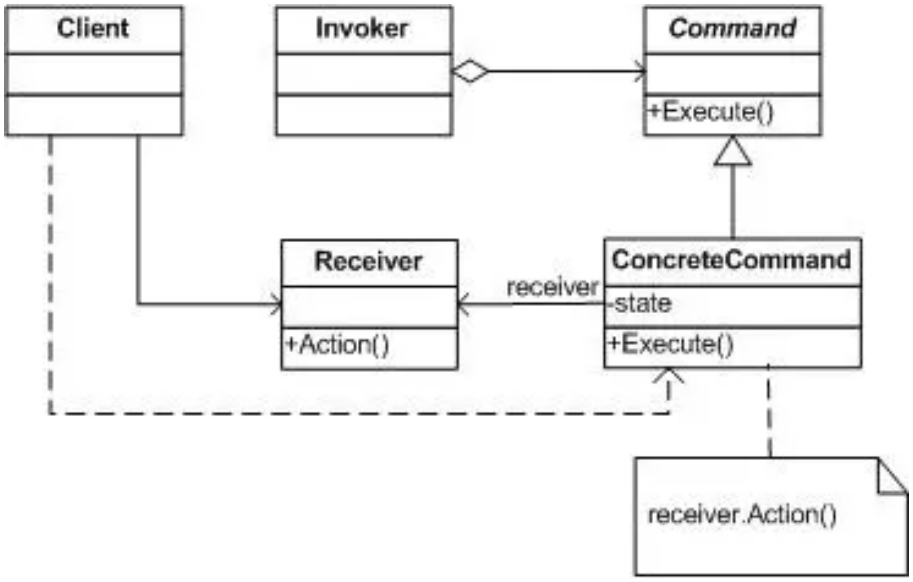




Command Pattern

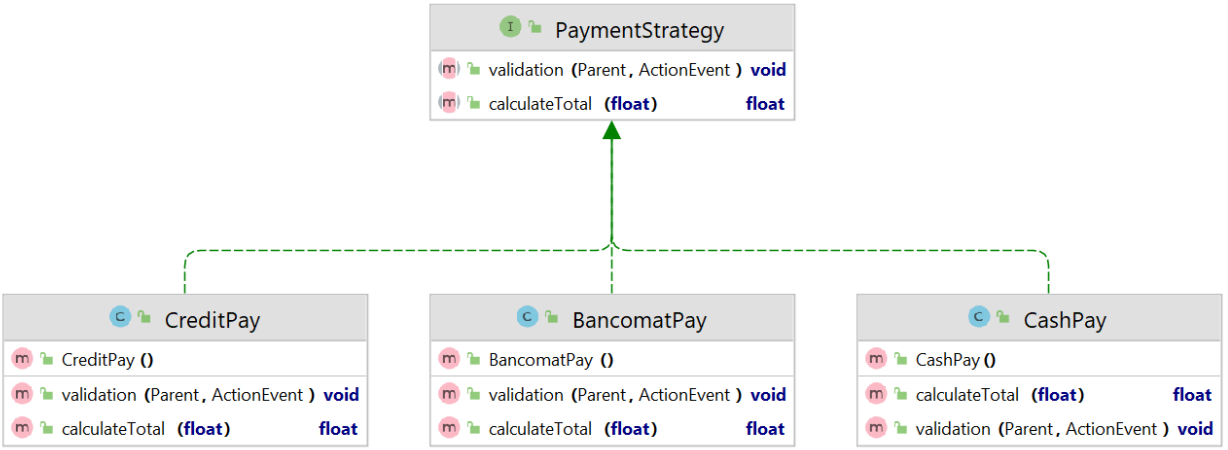
Abbiamo utilizzato il Command Pattern per implementare l'aggiunta di un numero di copie di un libro all'interno del carrello di un utente.

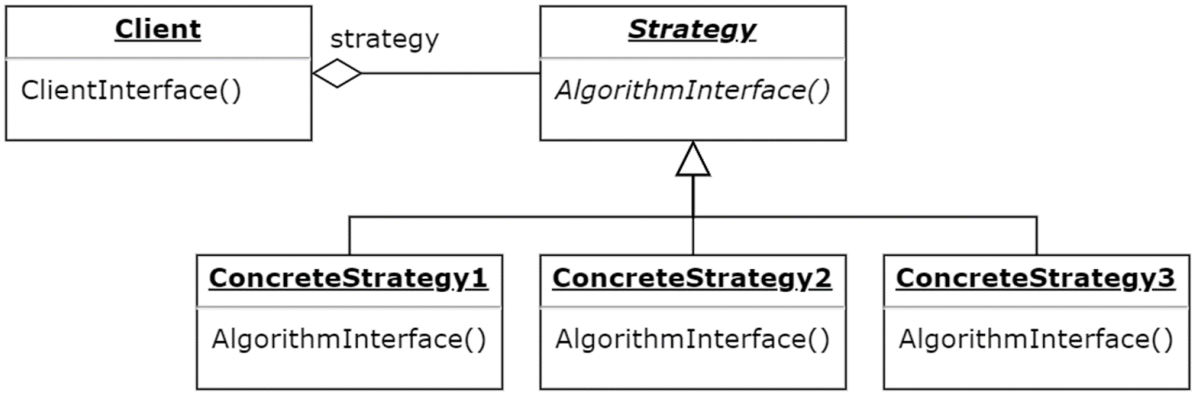




Strategy Pattern

Lo Strategy Pattern è stato utile per implementare in modo diverso l'algoritmo di pagamento, a seconda del metodo scelto (contanti, carta o bancomat).





Avvio del Programma

Come già riportato nel READme, mostriamo i passi necessari all'avvio del programma:

- Estrarre il file .zip che contiene il programma
- Aprire XAMPP e creare un database chiamato bsv2 importando il file SQL
- Aprire il progetto in un ambiente di sviluppo e avviare il programma da App.java

Accesso in modalità utente:

Per accedere come cliente al programma bisogna registrarsi come nuovo utente ed effettuare l'accesso. In alternativa è possibile accedere con uno dei profili già creati e presenti sul db.

Accesso in modalità admin:

Accedere con le credenziali Username: admin Password: admin