**5. Progettazione e Sviluppo Iterazione 4**

**5.1. Implementazione singleton connessione database e login + GUI**

In questa quarta e ultima iterazione, sono state implementate le funzionalità di connessione al database, il sistema di login e l'interfaccia utente grafica (GUI) per l'applicazione. Sono state introdotte due finestre principali: la pagina di login e la finestra di visualizzazione.

1. Singleton connessione al database: È stato implementato un singleton per la connessione al database, garantendo che ci sia una sola istanza attiva della connessione durante l'esecuzione dell'applicazione. Questo permette di gestire in modo efficiente le operazioni di accesso al database e di condividerle tra le diverse componenti dell'applicazione.
2. Pagina di login: La pagina di login è stata implementata per consentire all'utente di accedere all'applicazione. L'utente inserisce le sue credenziali, come nome utente e password, e queste informazioni vengono confrontate con i dati presenti nella tabella "utenti" del database. Utilizzando l'entità UtenteDAO, viene creato un oggetto Utente specificando il ruolo corrispondente.
3. Finestra di visualizzazione: La finestra di visualizzazione è la principale interfaccia utente dell'applicazione. Il suo comportamento varia in base al ruolo dell'utente:
   1. Ruolo "Admin": Se l'utente ha il ruolo di "Admin", la finestra di visualizzazione mostra informazioni complete su tutte le spedizioni, clienti e corrieri presenti nel database. L'utente "Admin" ha la possibilità di modificarli, aggiungerli o eliminarli utilizzando appositi pulsanti o comandi nell'interfaccia.
   2. Ruolo "Corriere": Se l'utente ha il ruolo di "Corriere", la finestra di visualizzazione mostra solo le spedizioni che sono assegnate a quel corriere specifico. Il corriere può visualizzare le spedizioni che gli sono state assegnate e può aggiornare lo stato di una spedizione specificando il suo ID tramite un apposito pulsante o comando nell'interfaccia.
   3. Ruolo "Cliente": Se l'utente ha il ruolo di "Cliente", la finestra di visualizzazione mostra solo le spedizioni di cui è destinatario. Il cliente può visualizzare le spedizioni che sono state inviate a lui, ma non ha la possibilità di interagire con l'interfaccia in alcun modo, oltre alla visualizzazione delle informazioni.

Queste funzionalità permettono una gestione differenziata delle informazioni e delle azioni disponibili in base al ruolo dell'utente, garantendo una corretta separazione dei privilegi e una navigazione personalizzata nell'interfaccia.

**5.2. Conclusioni**

Durante le quattro iterazioni di sviluppo descritte in questa documentazione, sono state implementate diverse funzionalità per un sistema di gestione delle spedizioni. Ogni iterazione ha introdotto nuovi requisiti e miglioramenti, portando l'applicazione a un livello di completezza sempre maggiore. Di seguito, vengono riassunti i punti salienti di ciascuna iterazione e le relative conclusioni:

**Prima Iterazione:**

* Implementazione delle entità principali: Cliente, Corriere e Spedizione.
* Utilizzo del Factory Method per creare le spedizioni in base al tipo specificato.
* Implementazione delle strategie di spedizione per calcolare il prezzo e la data di consegna.
* Realizzazione di diagrammi di sequenza per illustrare il flusso delle operazioni.

**Seconda Iterazione:**

* Aggiunta del metodo "Update Stato" per le spedizioni.
* Utilizzo del pattern Observer per permettere al Corriere di ricevere notifiche sugli aggiornamenti delle spedizioni.
* Descrizione dettagliata del comportamento del Corriere e del Magazzino in relazione all'aggiornamento dello stato delle spedizioni.

**Terza Iterazione:**

* Implementazione dei metodi di assegnazione automatica del Magazzino e del Corriere durante la creazione di una nuova spedizione.
* Descrizione del metodo "getNearestCap" per determinare il Magazzino e il Corriere più vicini al CAP del destinatario.
* Introduzione dei DAO (Data Access Object) per la gestione delle operazioni di CRUD (Create, Read, Update, Delete) delle entità.

**Quarta Iterazione:**

* Implementazione del singleton per la connessione al database, garantendo una singola istanza attiva durante l'esecuzione dell'applicazione.
* Realizzazione del sistema di login per autenticare gli utenti e assegnare loro ruoli specifici.
* Creazione di una GUI con due finestre principali: la pagina di login e la finestra di visualizzazione.
* Descrizione del comportamento della finestra di visualizzazione in base al ruolo dell'utente, consentendo operazioni diverse per gli amministratori, i corrieri e i clienti.

Complessivamente, le quattro iterazioni hanno permesso di sviluppare un sistema di gestione delle spedizioni con funzionalità complete e differenziate in base al ruolo degli utenti. L'implementazione dei design pattern come il Factory Method, l'Observer e il Singleton ha contribuito a una migliore organizzazione del codice e a una maggiore flessibilità nell'estendere le funzionalità dell'applicazione.

In conclusione, il sistema di gestione delle spedizioni sviluppato offre una piattaforma robusta per gestire il processo di spedizione, consentendo agli utenti di visualizzare, aggiornare e interagire con le spedizioni in modo efficiente e personalizzato.