



ELABORAZIONE – ITERAZIONE 1

Introduzione

Conclusa la fase di ideazione, si passa alla fase di elaborazione. La realizzazione del sistema software seguirà uno sviluppo iterativo, con raffinazioni successive dei requisiti del sistema individuati durante la prima fase di ideazione, ed utilizzerà l'approccio di Unified Process e gli strumenti di modellazione di UML.

Scopo delle iterazioni seguenti sarà quello di:

- raffinare la visione del dominio del problema;
- raffinare la conoscenza dei requisiti precedentemente individuati e la loro portata e/o individuarne di nuovi;
- implementare in maniera iterativa il nucleo dell'architettura software;
- risolvere le problematiche legate ai rischi maggiori e ai centri di interesse per la committenza;
- fornire delle stime realistiche del piano di lavoro e delle risorse complessive.

Durante questa prima iterazione l'attenzione ed il lavoro saranno incentrati su:

- Implementare lo scenario principale di successo (happy path) del caso d'uso UC1: Inserimento di un Nuovo Cliente

tralasciando per il momento l'implementazione degli scenari alternativi individuati.

Chiariti gli obiettivi perseguiti dal committente, in termini di funzionalità da implementare e requisiti da rispettare, si procede quindi con la fase di Analisi Orientata agli Oggetti.

Analisi Orientata agli Oggetti

Obiettivo dell'analisi orientata agli oggetti è quello di ricavare una descrizione ad oggetti del dominio del sistema in discussione, in modo da individuare e definire al meglio i confini del sistema e dei problemi da affrontare.

Per fornire tale descrizione vengono utilizzati seguenti strumenti:

- Modello di Dominio
- Diagrammi di sequenza (System Sequence Diagram o SSD)
- Contratti delle Operazioni

MODELLO DI DOMINIO

Il Modello di Dominio è un elaborato grafico della disciplina della Modellazione di Business di Unified Process nel quale vengono identificati i concetti, gli attributi e le associazioni significativi per la descrizione del dominio del sistema in discussione.

Dopo un'attenta analisi dello scenario principale di successo del caso d'uso UC1, è stato possibile identificare le seguenti classi concettuali:

- **S&PMotorBet**: classe concettuale che rappresenta il sistema software in esame;
- **Cliente**: cliente del centro scommesse, il quale richiede la registrazione per poter disporre del servizio dell'attività;
- **Conto**: conto di deposito di denaro associato univocamente al Cliente, dal quale verranno detratti gli importi puntati dal Cliente sulle proprie scommesse e sul quale verranno accreditate le eventuali vincite.
- **Movimento**: rappresenta un movimento di denaro dal Conto del Cliente.

Individuati gli attributi caratterizzanti ciascuna classe e le associazioni tra queste, è

stato ricavato il seguente Modello di Dominio.

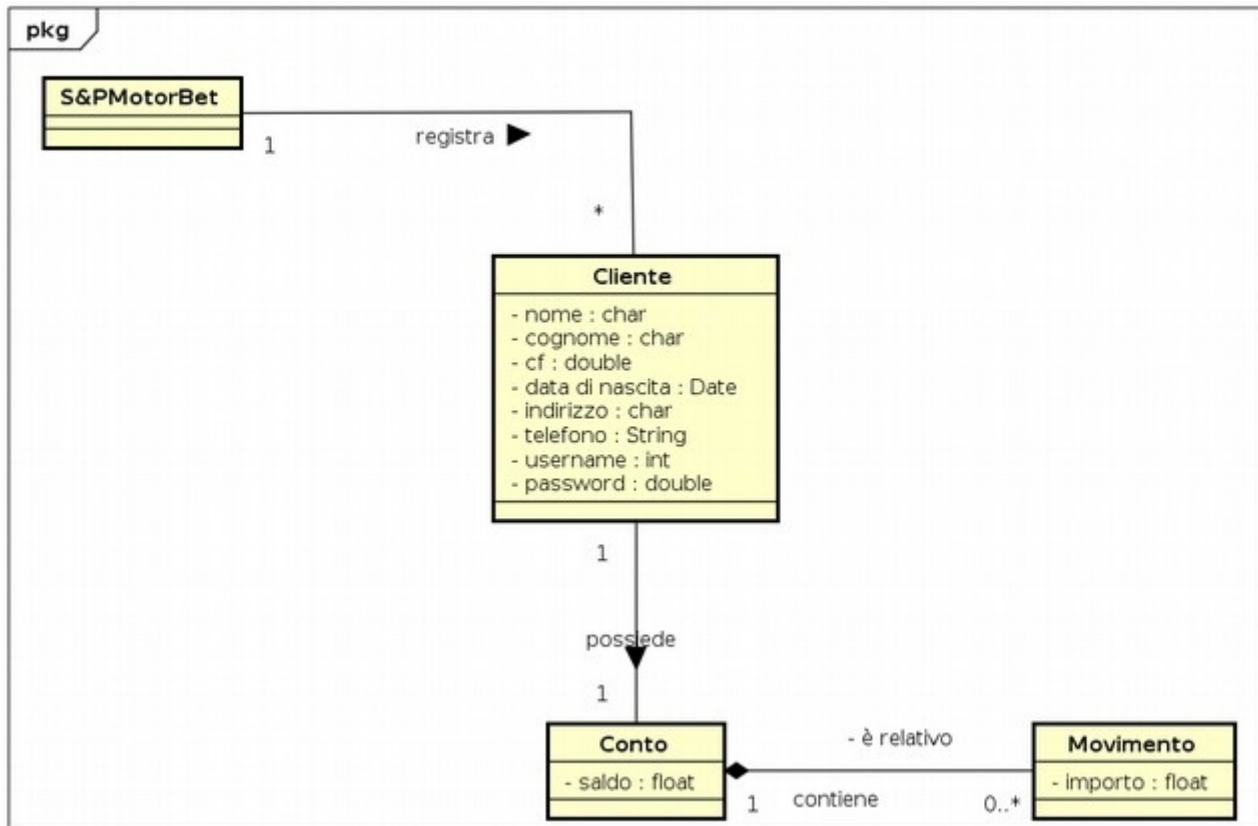
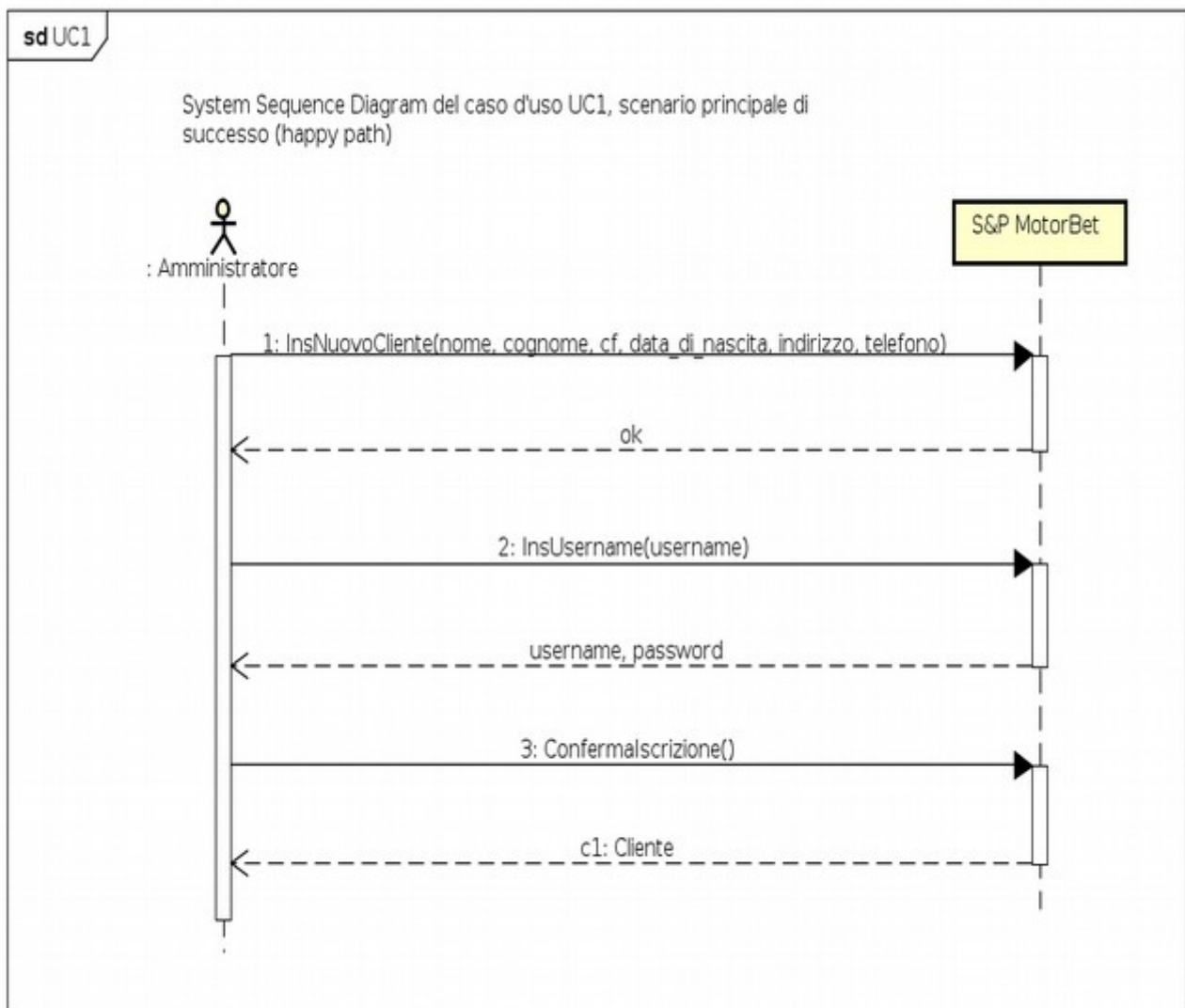


DIAGRAMMA DI SEQUENZA DI SISTEMA

Il Diagramma di Sequenza di Sistema (o SSD) è un elaborato grafico che illustra il corso delle interazioni tra l'attore primario del caso d'uso e il sistema in discussione, in termine di sequenza di input e output prodotti.

Dall'analisi dello scenario principale di successo di UC1 è stato possibile ricavare il seguente diagramma.



Contratti delle Operazioni

I contratti delle operazioni descrivono le principali operazioni di sistema che si occupano di gestire gli eventi di sistema individuati nel Diagramma di Sequenza.

Contratto CO1: InsNuovoCliente

Operazione:

InsNuovoCliente (nome: String, cognome: String, cf: String, data_di_nascita: Date, indirizzo : String, telefono : String)

Riferimenti: Caso d'uso: Inserimento Nuovo Cliente

Pre-condizioni: nessuna

Post-condizioni:

- è stata creata una nuova istanza C1 di Cliente;
- Gli attributi di C1 sono stati inizializzati ai dati anagrafici del cliente;
- è stata creata una nuova istanza A1 di Conto, relativa al Cliente C1;
- Gli attributi di A1 sono stati inizializzati;
- è stata creata una nuova istanza M1 di Movimento, relativa al Conto A1;
- Gli attributi di M1 sono stati inizializzati;

Contratto CO2: InsUsername

Operazione: *InsUsername* (username: String)

Riferimenti: Caso d'uso: Inserimento Nuovo Cliente

Pre-condizioni: è in corso la procedura di registrazione del cliente C1

Post-condizioni:

- l'attributo username dell'istanza C1 è stato inizializzato al valore

inserito;

- l'attributo password dell'istanza C1 è stato inizializzato al valore generato dal sistema.

Contratto CO3: ConfermaIscrizione

Operazione: *ConfermaIscrizione()*

Riferimenti: Caso d'uso: Inserimento Nuovo Cliente.

Pre-condizioni: è in corso la procedura di registrazione del cliente C1.

Post-condizioni:

- l'istanza C1 di Cliente è stata aggiunta alla lista dei Clienti;
- C1 è stata aggiunta all'archivio dei Clienti;
- l'istanza A1 di Conto è stata aggiunta all'archivio dei Conti;
- M1 è stata inserita all'interno della lista di Movimenti di A1;
- l'istanza M1 di Movimento è stata aggiunta all'archivio dei Movimenti.

Progettazione Orientata agli Oggetti

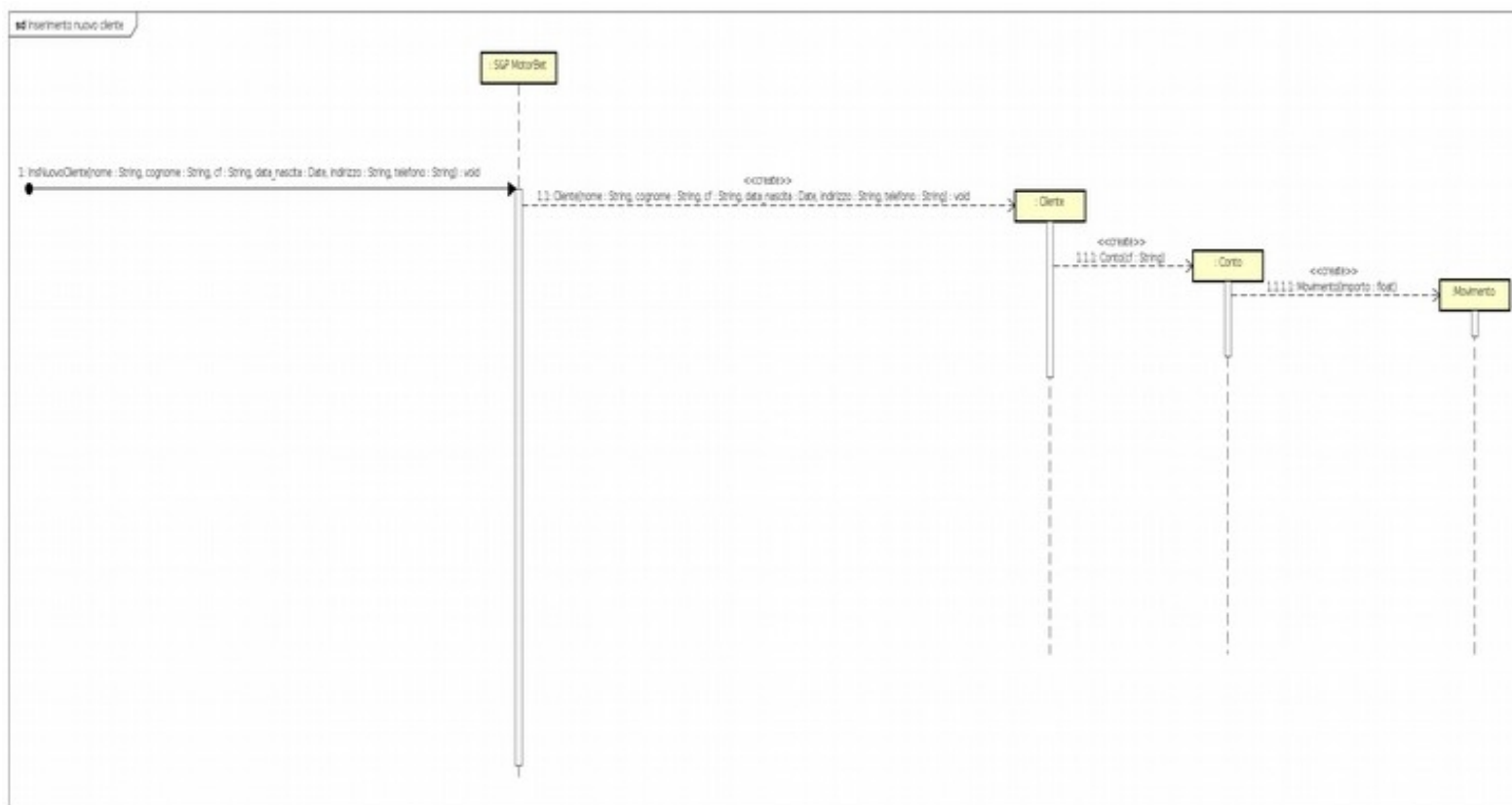
La progettazione orientata agli oggetti è la disciplina di Unified Process che si interessa della definizione degli oggetti software, delle loro responsabilità e di come questi collaborano per soddisfare i requisiti individuati precedentemente.

L'elaborato principale di questa fase è il Modello di Progetto, ovvero l'insieme degli elaborati grafici che descrivono la progettazione logica sia da un punto di vista dinamico, con i Diagrammi delle Interazioni, che da un punto di vista statico, con il Diagramma delle Classi di Progetto.

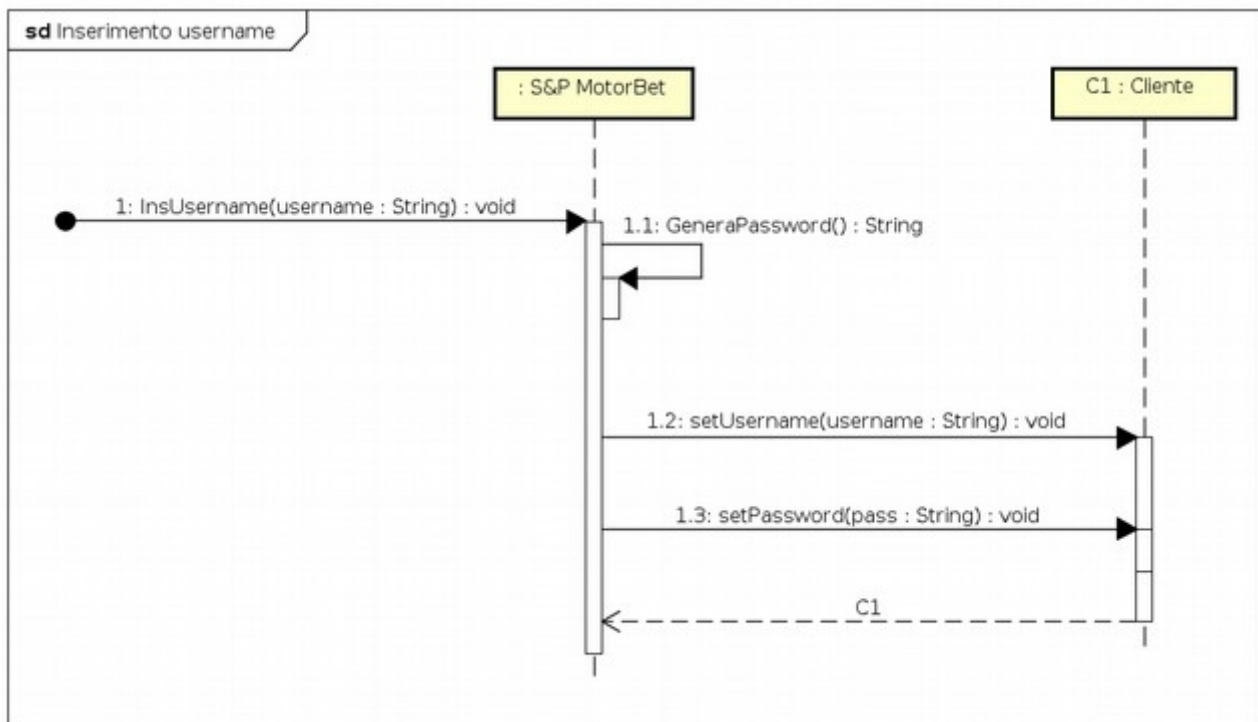
Diagrammi di interazione

A seguito di un attento esame degli elaborati prodotti e mostrati precedentemente, è stato possibile individuare e descrivere i seguenti Diagrammi di Interazione.

- **Inserimento di un nuovo cliente**



- **Inserimento Username**



- Conferma Iscrizione

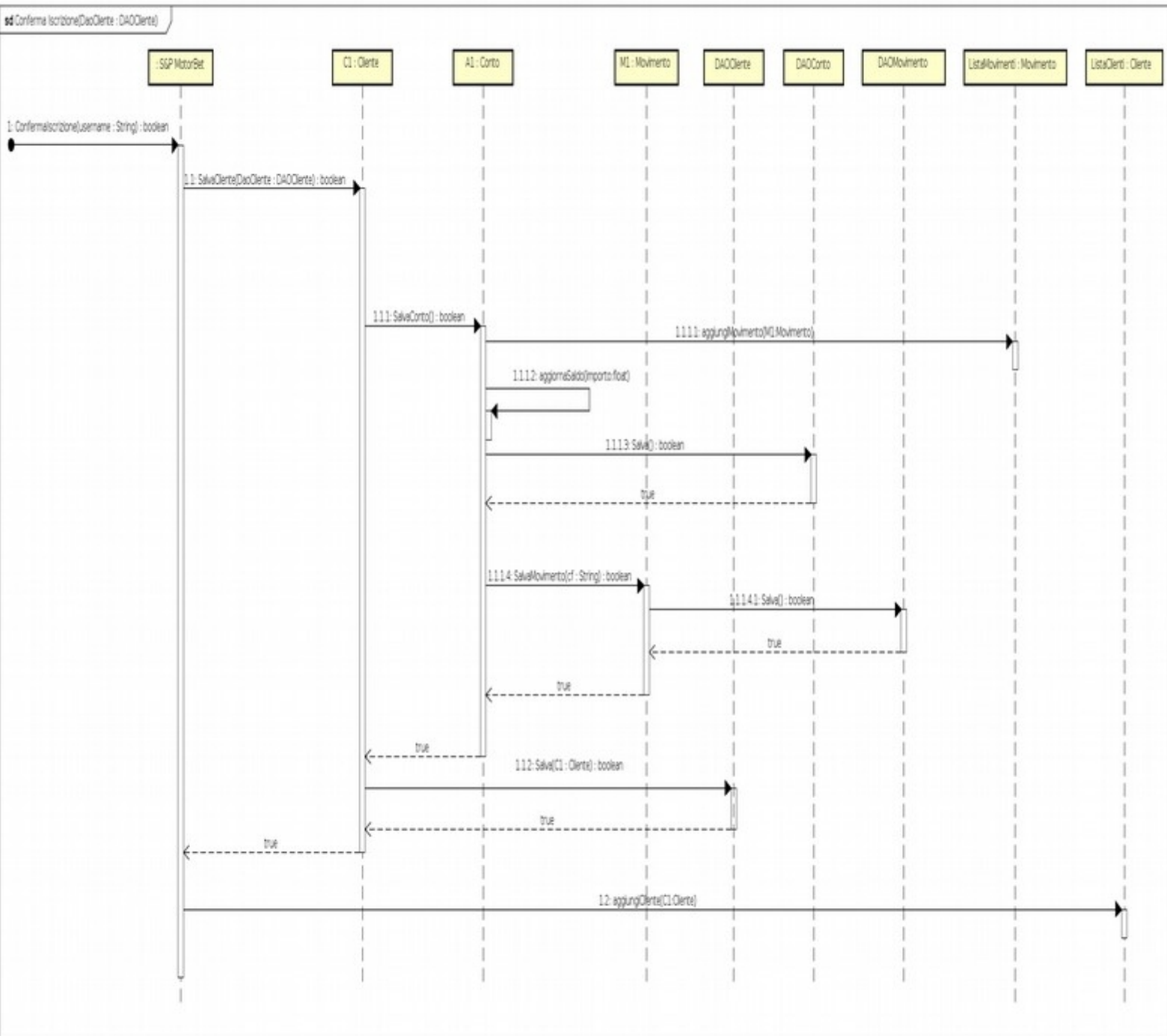
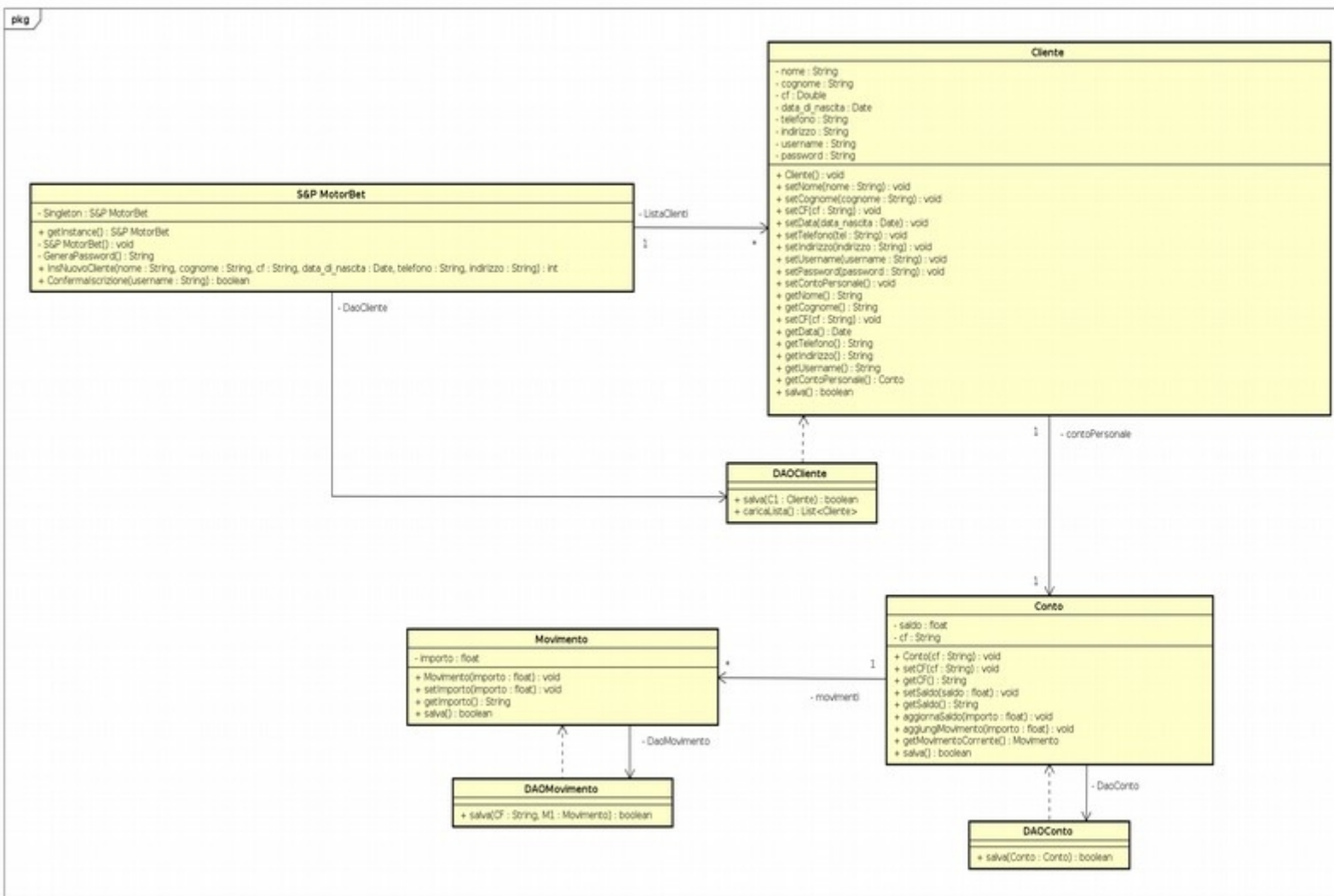


Diagramma delle Classi di Progetto

A seguito di un attento esame degli elaborati prodotti e mostrati precedentemente, è stato possibile individuare e descrivere le seguenti classi di oggetti e le loro associazioni, utilizzate nell'ambito della definizione delle specifiche software.



Scelte implementative della fase di progettazione

Nel presente paragrafo vengono illustrate le scelte implementative operate per la realizzazione degli elaborati della Progettazione Orientata agli Oggetti e del modulo software che risolvono lo scenario principale di successo del caso d'uso UC1.

Per la determinazione delle responsabilità delle classi individuate e dei messaggi tra queste scambiati sono stati applicati diversi pattern GRASP, adottando quindi un approccio Responsibility-Driven. In particolare:

- La classe **S&PMotorBet** è il **Facade Controller** del sistema software in esame. S&PMotorBet ha le seguenti responsabilità:
 - è **Creator** per la classe Cliente, in quanto possiede i dati per inizializzare le istanze di tale classe;
 - è **Information Expert** della classe Cliente, in quanto contiene al suo interno una ArrayList di istanze di Cliente;
- La classe **Cliente** è il **Creator e l'Information Expert** per la classe Conto.
- La classe **Conto** è **Creator e Information Expert** per la classe Movimento, in quanto contiene al suo interno una ArrayList di istanze di Movimento .
- Le classi così individuate e progettate presentano un basso accoppiamento (**Low Coupling**) e risultano coese(**High Cohesion**).

Dato che la classe **S&P MotorBet** rappresenta il Controller del sistema software complessivo, risulta essere conveniente fare in modo che l'applicazione crei solo un'istanza di questa classe. Per ottenere ciò si è scelto di progettare la classe come un **Singleton**, sfruttando l'omonimo pattern GoF.

Per gestire la comunicazione tra l'applicazione ed il RDBMS, demandato alla persistenza dei dati generati ed utilizzati dal sistema nel suo esercizio, sono inoltre state costruite delle classi **DAO (Data Access Object)**, una per ciascuna classe di business del sistema.

Le istanze delle classi di business utilizzano lo strato applicativo dei DAO per gestire gli accessi e le operazioni di creazione, aggiornamento, modifica e cancellazione di entry nelle corrispondenti relazioni presenti nel Database.