

Corso di Laurea Magistrale in Ingegneria Informatica – A.A. 2022/2023

INGEGNERIA DEL SOFTWARE



FRANCESCO DENARO

Prefazione

La seguente documentazione presenta una descrizione relativa allo sviluppo del sistema SportPlanner.

Il software è stato implementato nel linguaggio Java sfruttando l'ambiente di sviluppo NetBeans, mentre la progettazione in UML è stata realizzata con il software Astah.

In questo documento verranno presentate solamente le versioni finali di ciascun elaborato ottenute al termine di tutte le fasi di progettazione (per visionare le versioni intermedie si vedano i documenti nelle corrispondenti sottocartelle della documentazione).

Infine vengono descritte le fasi conclusive di testing e di refactoring del programma

Indice

IDEAZIONE E ANALISI DEI REQUISITI

1	Introduzione.....	5
2	Requisiti	5
2.1	Obiettivi e casi d'uso.....	6
2.2	Casi d'uso.....	7
2.2.1	Caso d'uso UC1: Inserimento nuovo torneo	7
2.2.2	Caso d'uso UC2: Gestione prenotazione.....	8
2.2.3	Caso d'uso UC3: Pagamento prenotazione campo	10
2.2.4	Caso d'uso UC4: Iscrizione torneo	11
2.2.5	Caso d'uso UC5: Pagamento iscrizione torneo	11
2.2.6	Caso d'uso UC6: Registrazione cliente	11
2.2.7	Caso d'uso UC7: Visualizzare dettagli prenotazione	11
2.2.8	Caso d'uso UC8: Inserimento nuovo campo	11
2.2.9	Caso d'uso UC9: Visualizzare prenotazioni partite	12
2.2.10	Caso d'uso UC10: Visualizzare iscrizioni tornei.....	12
3	Documento di visione.....	12
3.1	Posizionamento	12
3.2	Descrizione delle parti interessate	12
3.3	Descrizione generale del progetto	13
4	Regole di business	13
5	Glossario.....	14

ANALISI ORIENTATA AGLI OGGETTI

1	Introduzione.....	15
1.1	Iterazione 1	15
1.2	Iterazione 2	15
1.3	Iterazione 3	15
1.4	Iterazione 4	15
1.5	Iterazione 5	16
2	Modello di dominio	16
3	SSD e contratti.....	17

PROGETTAZIONE

1	Diagramma delle classi	25
2	Diagrammi di sequenza	25

TESTING

1. Test di sistema	30
--------------------------	----

REFACTORING E CONCLUSIONI

1. Refactoring.....	31
2. Test di accettazione.....	31
3. Pattern GoF	31
3.1. Singleton.....	31
3.2. Strategy	31

IDEAZIONE E ANALISI DEI REQUISITI

1 Introduzione

La fase di ideazione è la fase iniziale che permette di stabilire una prima visione del progetto. Infatti in questa fase verrà fatta solamente un'analisi preliminare, mentre la maggior parte dei requisiti verranno definiti nelle fasi successive.

Per capire al meglio come sviluppare e comprendere i requisiti del progetto, in questa fase sono stati presi in considerazione i seguenti documenti:

- Modello dei Casi d'Uso;
- Documento di Visione;
- Specifiche Supplementari;
- Regole di Business;
- Glossario

2 Requisiti

SportPlanner è un software utilizzato da un centro sportivo che disponi di diversi campi per praticare vari sport quali calcio, basket, tennis etc....

Quest'applicazione permette di semplificare la gestione del centro sportivo, in quanto permette ai clienti di effettuare prenotazioni direttamente da remoto, evitando anche eventuali sovrapposizioni tra le varie occupazioni.

Per l'amministratore c'è inoltre la possibilità di inserire tornei ai quali i clienti potranno iscriversi semplicemente effettuando l'accesso con le proprie credenziali.

Quindi il sistema deve gestire il tutto, verificandone la disponibilità dei vari campi, attraverso un'interfaccia.

Il sistema software deve occuparsi della gestione delle seguenti attività:

- Inserimento, da parte dell'Amministratore, di un nuovo Torneo, con fasce orarie, date e campi prestabiliti. Il torneo avrà una quota di iscrizione e numero massimo di partecipanti
- Visualizzazione, da parte dell'Amministratore, di tutti i Tornei presenti nel Sistema, con le relative iscrizioni e con tutti i dettagli necessari.
- Prenotazione, da parte del Cliente, di un campo, in una data e in un orario disponibile, per potere giocare una partita singola

- Iscrizione, da parte del Cliente, ad un Torneo che è stato inserito dall'Amministratore nel Sistema, purché non sia stato raggiunto il numero massimo di iscritti.
- Visualizzazione, da parte del Cliente, di tutti i dettagli della propria prenotazione o iscrizione ad un Torneo, inserendo il codice che è stato rilasciato
- Inserimento di nuovi campi, da parte dell'Amministratore, nel Sistema
- Pagamento, da parte del Cliente, della prenotazione o della quota d'iscrizione

2.1 Obiettivi e casi d'uso

Analizzando i requisiti, sono stati individuati gli attori principali a cui è destinato il Sistema e gli obiettivi che si intende portare a termine.

Attore	Obiettivo	Caso d'uso
Amministratore	Gestire l'inserimento di un nuovo Torneo nel Sistema, specificando le informazioni utili ai Clienti per l'Iscrizione.	UC1: Inserimento torneo
Cliente	Effettuare una prenotazione di un campo se disponibile, in una fascia oraria disponibile	UC2: Gestione prenotazione
Amministratore	Controllo sul corretto svolgimento della Prenotazione, al termine del quale dovrà emettere il codice identificativo di Prenotazione che servirà per il pagamento.	UC2: Gestione prenotazione
Cliente	Pagare la corretta somma in base al prezzo del Campo selezionato	UC3: Pagamento prenotazione campo
Addetto	Controllo sul corretto pagamento	UC3: Pagamento prenotazione campo
Cliente	Iscrizione, da parte del cliente, della propria squadra al Torneo, indicano il numero di partecipanti e il nome della squadra	UC4: Iscrizione torneo
Cliente	Pagamento quota d'Iscrizione al Torneo	UC5: Pagamento iscrizione torneo
Addetto	Controllo sul corretto pagamento	UC5: Pagamento iscrizione

Cliente	Effettuare la Registrazione nel Sistema, specificando nome, cognome, numero di cellulare, indirizzo e-mail e delle Credenziali.	UC6: Registrazione cliente
Cliente	Visualizzare tutte le informazioni della Prenotazione o dell'Iscrizione	UC7: Visualizzare dettagli prenotazione
Amministratore	Inserimento di un nuovo campo, con descrizione dello sport praticabile	UC8: Inserimento nuovo campo
Amministratore	Visualizzare le prenotazioni	UC9: Visualizzare prenotazioni partite
Amministratore	Visualizzare i tornei e le relative iscrizioni	UC10: Visualizzare iscrizioni tornei

2.2 Casi d'uso

Viene qui presentata una descrizione dei casi d'uso;.

In particolare, partendo dalla fase di ideazione in cui la maggior parte dei casi d'uso erano descritti in formato breve, seguendo il processo iterativo si è passati gradualmente ad una descrizione via via più dettagliata per la maggior parte dei casi d'uso.

2.2.1 Caso d'uso UC1: Inserimento nuovo torneo

Nome del caso d'uso	UC1: Inserimento nuovo torneo
Portata	Applicazione SportPlanner
Livello	Obiettivo utente
Attore primario	Amministratore
Parti interessate e interessi	Amministratore → vuole inserire un nuovo torneo specificando le varie informazioni che saranno utili al Cliente
Pre-condizioni	I campi associati al torneo devono essere disponibili nei giorni e nelle fasce orarie specificate dal torneo
Garanzia di successo	Le informazioni relative al torneo sono inserite con successo nel Sistema.

Scenario principale di successo	<ol style="list-style-type: none"> 1. L'Amministratore vuole inserire un nuovo torneo nel Sistema; 2. L'Amministratore sceglie l'attività "Crea nuovo torneo"; 3. L'Amministratore inserisce l'identificativo dei campi da occupare, con le relative fasce orarie, il numero di squadre che parteciperanno, le date di inizio e fine e la quota d'iscrizione al torneo. Al termine dell'operazione il Sistema ritorna un codice univoco; 4. L'Amministratore indica di aver concluso l'operazione
Estensioni	<p>*a. In un qualsiasi momento il Sistema fallisce e ha un arresto improvviso:</p> <ol style="list-style-type: none"> 1. L'Addetto riavvia il software e ripristina lo stato precedente del Sistema; 2. Il Sistema ripristina lo stato;
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizioni	Legata a scelte aziendali.
Varie	

2.2.2 Caso d'uso UC2: Gestione prenotazione

Nome del caso d'uso	UC2: Gestione prenotazione
Portata	Applicazione SportPlanner
Livello	Obiettivo utente
Attore primario	Cliente
Parti interessate e interessi	Cliente → vuole prenotare un campo, in una data e in una fascia oraria disponibile

	Amministratore → vuole controllare che la prenotazione vada a buon fine correttamente
Pre-condizioni	Il cliente deve essere in grado di visualizzare gli eventi già presenti, per vedere fasce orarie e date disponibili
Garanzia di successo	Le informazioni della prenotazione sono state inserite con successo nel Sistema. Il Cliente potrà visualizzare i dettagli della prenotazione
Scenario principale di successo	<ol style="list-style-type: none"> 1. Il Cliente vuole prenotare uno dei campi del centro sportivo; 2. Il Cliente deve prima autenticarsi con le proprie credenziali; 3. Il Cliente sceglie il campo, attraverso il suo identificativo, seleziona la data e preme sul pulsante "Prenota campo". Il Sistema mostra le fasce orarie disponibili per quella data, se presenti e i dettagli relativi al campo; 4. Il Sistema ritorna la quota totale del campo; 5. Il Cliente conferma la prenotazione. 6. Il Sistema notifica l'avvenuta prenotazione al cliente, emettendo una ricevuta con il codice di prenotazione
Estensioni	<p>*a. In un qualsiasi momento il Sistema fallisce e ha un arresto improvviso:</p> <ol style="list-style-type: none"> 1. Il Cliente apre nuovamente l'interfaccia; 2. Il Sistema ripristina lo stato <p>2a. Il Cliente inserisce le credenziali errate.</p> <ol style="list-style-type: none"> 1. Il Sistema genera un messaggio di errore; 2. Il Cliente ripete il passo 2 inserendo nuovamente le credenziali. <p>3a. Il Cliente inserisce l'identificativo di un campo che non è presente nel Sistema.</p> <ol style="list-style-type: none"> 1. Il Sistema genera un messaggio di errore;

	2. Il Cliente ripete il passaggio 3 inserendo un nuovo identificativo. 3b. Il Cliente inserisce una data per cui non ci sono fasce orarie disponibili. 1. Il Sistema genera un messaggio di errore; 2. Il Cliente ripete il passaggio 3 inserendo una nuova data, un nuovo identificativo per un campo o uscire;
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizioni	Legata all'affluenza di Clienti.
Varie	

2.2.3 Caso d'uso UC3: Pagamento prenotazione campo

Il centro sportivo non accetta pagamenti attraverso la piattaforma. Il Cliente che ha prenotato pagherà la somma dovuta prima della partita, insieme agli altri partecipanti.

Nome del caso d'uso	UC3: Pagamento prenotazione campo
Portata	Applicazione SportPlanner
Livello	Obiettivo utente
Attore primario	Addetto
Parti interessate e interessi	Addetto → vuole assicurarsi del corretto pagamento della prenotazione. Cliente → vuole pagare la quota totale corretta
Pre-condizioni	Il Cliente deve aver prenotato correttamente la partita tramite l'interfaccia.
Garanzia di successo	L'Addetto riscuote la corretta somma rispetto alla prenotazione del cliente. Il pagamento verrà registrato nel sistema.
Scenario principale di successo	1. Il Cliente si reca dall'Addetto per pagare la quota corretta;

	<ol style="list-style-type: none"> 2. Il Cliente mostra all'Addetto il codice di prenotazione; 3. L'Addetto sceglie l'attività "Paga prenotazione" passando il codice; 4. Il Sistema ritorna la quota da pagare; 5. Il Cliente paga la somma; 6. L'Addetto registra il pagamento nel sistema;
Estensioni	<p>*a. In un qualsiasi momento il Sistema fallisce e ha un arresto improvviso:</p> <ol style="list-style-type: none"> 3. L'Addetto riavvia il software e ripristina lo stato precedente del Sistema; 4. Il Sistema ripristina lo stato;
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizioni	Legata all'affluenza di Clienti.
Varie	

2.2.4 Caso d'uso UC4: Iscrizione torneo

Un Cliente usa il Sistema per iscriversi al torneo tramite il suo identificativo e il nome scelto per la sua squadra. Il Sistema restituisce un identificativo per il pagamento che verrà gestito dall'Addetto, come per il pagamento delle prenotazioni

2.2.5 Caso d'uso UC5: Pagamento iscrizione torneo

Un Cliente mostra il codice identificativo dell'iscrizione all'Addetto che lo inserisce nel Sistema e mostra la quota corretta da pagare, supervisionando il corretto pagamento.

2.2.6 Caso d'uso UC6: Registrazione cliente

Un Cliente usa il Sistema per registrarsi, inserendo nome, cognome, numero di cellulare e indirizzo e-mail. Dopo sceglierà username e password per gli accessi. Il Sistema salverà tutti i dati e li assocerà al Cliente.

2.2.7 Caso d'uso UC7: Visualizzare dettagli prenotazione

Il Cliente usa il Sistema per visualizzare il riepilogo della prenotazione o dell'iscrizione

2.2.8 Caso d'uso UC8: Inserimento nuovo campo

L'Amministratore usa il Sistema per aggiungere un nuovo campo nel Sistema

2.2.9 Caso d'uso UC9: Visualizzare prenotazioni partite

L'Amministratore usa il Sistema per visualizzare tutte le prenotazioni successive alla data odierna

2.2.10 Caso d'uso UC10: Visualizzare iscrizioni tornei

L'Amministratore usa il Sistema per visualizzare tutti i tornei e le relative iscrizioni che devono ancora iniziare.

3 Documento di visione

L'obiettivo del progetto è la realizzazione di un sistema software che permetta la gestione di Eventi di un Centro Sportivo. Si vuole infatti migliorare l'efficienza della gestione e della programmazione degli Eventi Sportivi sia da un punto di vista economico sia di gestione di disponibilità dei Campi per gli interessati.

3.1 Posizionamento

3.1.1 Opportunità di business

Il software SportPlanner avrà come obiettivo la gestione automatica di una serie di eventi riguardanti diversi sport disponibili in un Centro Sportivo, concentrandosi, nello specifico, nella creazione e gestione di questi., dando la possibilità anche di inserire Tornei per i vari sport.

3.1.2 Formulazione del problema

Un Centro Sportivo offre diversi sport e mette a disposizione del cliente molti Campi. Questo con una gestione non automatizzata rallenta e rende più difficile il compito dell'organizzatore. L'ausilio di un sistema che gestisca il tutto in maniera automatica aiuterà nella coordinazione degli eventi, venendo incontro anche ai Clienti, permettendogli ad esempio la Prenotazione direttamente da casa, ma anche nella gestione dei pagamenti.

3.1.3 Formulazione della posizione del prodotto

La realizzazione del software è rivolta a tutti i Centri Sportivi che mettano a disposizione del Cliente un'ampia scelta di eventi e che vogliano automatizzare la gestione di questi nei diversi ambienti sportivi.

3.2 Descrizione delle parti interessate

3.2.1 Obiettivi a livello dell'utente

Gli utenti del sistema necessitano di:

- Organizzatore: gestione delle attività sportive;
- Cliente: prenotazione dei campi;
- Addetto: gestione dei pagamenti;

3.3 Descrizione generale del progetto

3.3.1 Punto di vista del prodotto

Il sistema SportPlanner sarà utilizzato nei Centri Sportivi e fornirà un'interfaccia ai Clienti per effettuare la propria Prenotazione del campo, in base al tipo di sport che vorrà effettuare e alla disponibilità o di iscriversi ad un Torneo organizzato appositamente dall'Amministratore. Inoltre darà la possibilità agli addetti di gestire i pagamenti delle prenotazioni.

3.3.2 Riepilogo delle caratteristiche del sistema

Il sistema deve avere le caratteristiche riportate di seguito:

- Aggiornamento disponibilità campo per orario
- Gestione prenotazione
- Pagamento prenotazione
- Visualizzazione dettagli prenotazione
- Inserimento nuovo campo
- Inserimento nuovo torneo sportivo
- Gestione iscrizione al torneo
- Pagamento iscrizione

4 Regole di business

Per l'utilizzo del software vanno rispettate le seguenti regole di dominio:

ID	Regola	Modificabilità	Sorgente
R1	Se la fascia oraria del campo prenotato per una partita singola è precedente alle 18, il prezzo della prenotazione sarà scontato del 15%, eccetto nei giorni festivi.	Bassa	Politica interna dell'attività
R2	Se la partita è in un giorno festivo o prefestivo il prezzo della prenotazione aumenta del 10%.	Media	Politica interna dell'attività
R3	Ogni 10 prenotazioni per una Partita o iscrizioni ad un Torneo, il Cliente avrà diritto ad uno sconto del 45% sulla prossima Partita singola.	Media	Politica interna dell'attività

5 Glossario

Termine	Definizione
Amministratore	Gestore dell'attività il quale si occupa della gestione delle prenotazioni e anche dell'inserimento di tornei
Addetto	Persona che svolge la gestione dell'attività, consentendo i pagamenti ai clienti.
Campo	Luogo in cui si svolge l'evento sportivo
Centro Sportivo	Organizzazione a cui è destinato il Software, che deve gestire eventi sportivi con relative prenotazioni e pagamenti
Cliente	Utente che si registra tramite l'interfaccia web per effettuare iscrizioni o prenotazioni
Credenziali	Coppia di valori che saranno username e password relativi ad un cliente.
Evento Sportivo	Rappresenta un evento sportivo che si svolgerà in un campo, permettendo di praticare uno sport specifico
Iscrizione	Possibilità di occupare un posto tra le squadre partecipanti ad un torneo inserito dall'amministratore
Prenotazione	Possibilità di occupare un campo in una particolare fascia oraria
Ricevuta	Strumento che conterrà il codice identificativo di prenotazione utile per il pagamento
Torneo	Evento che ha una data di inizio e di fine, che occuperà diversi campi in diversi giorni, a cui i clienti potranno iscriversi

ANALISI ORIENTATA AGLI OGGETTI

1 Introduzione

Seguendo l'approccio iterativo evolutivo suggerito da UP, la realizzazione del software è stata distribuita in cinque diverse iterazioni. In questo modo è stato possibile implementare in maniera iterativa il nucleo dell'architettura del software, sono state risolte le problematiche relative ai rischi maggiori ed è stata affrontata un'analisi dei requisiti graduale in modo da limitare al minimo il danno causato da eventuali errori di progettazione e implementazione.

Per ciascuna iterazione in particolare ci si è occupati di gestire le seguenti problematiche:

1.1 Iterazione 1

- Implementazione dello scenario principale di successo del caso d'uso *UC1: Inserimento nuovo torneo*.
- Implementazione di un caso d'uso di *SportPlanner* necessario per gestire le esigenze di inizializzazione per questa iterazione.

1.2 Iterazione 2

- Implementare lo scenario principale di successo del caso d'uso *UC6: Registrazione cliente*. L'implementazione del caso d'uso citato è stata scelta in quanto lo scenario di registrazione del cliente è necessario per il proseguimento con il caso d'uso *UC2: Gestione prenotazione*.
- Aggiornamento del caso d'uso di *SportPlanner* necessario per gestire le esigenze di inizializzazione per questa iterazione.

1.3 Iterazione 3

- Implementare lo scenario principale di successo del caso d'uso *UC2: Gestione prenotazione*, considerando che la prenotazione creerà una nuova istanza di una partita che occupi un particolare campo.
- Aggiornamento del caso d'uso di *SportPlanner* necessario per gestire le esigenze di inizializzazione per questa iterazione.

1.4 Iterazione 4

- Implementazione dello scenario principale di successo e delle relative estensioni del caso d'uso *UC3: Pagamento prenotazione campo*.
- Implementazione degli scenari secondari, e quindi delle estensioni, per quanto riguarda i casi d'uso *UC1: Inserimento nuova torneo*, *UC2: Gestione prenotazione* e *UC6: Registrazione cliente*.

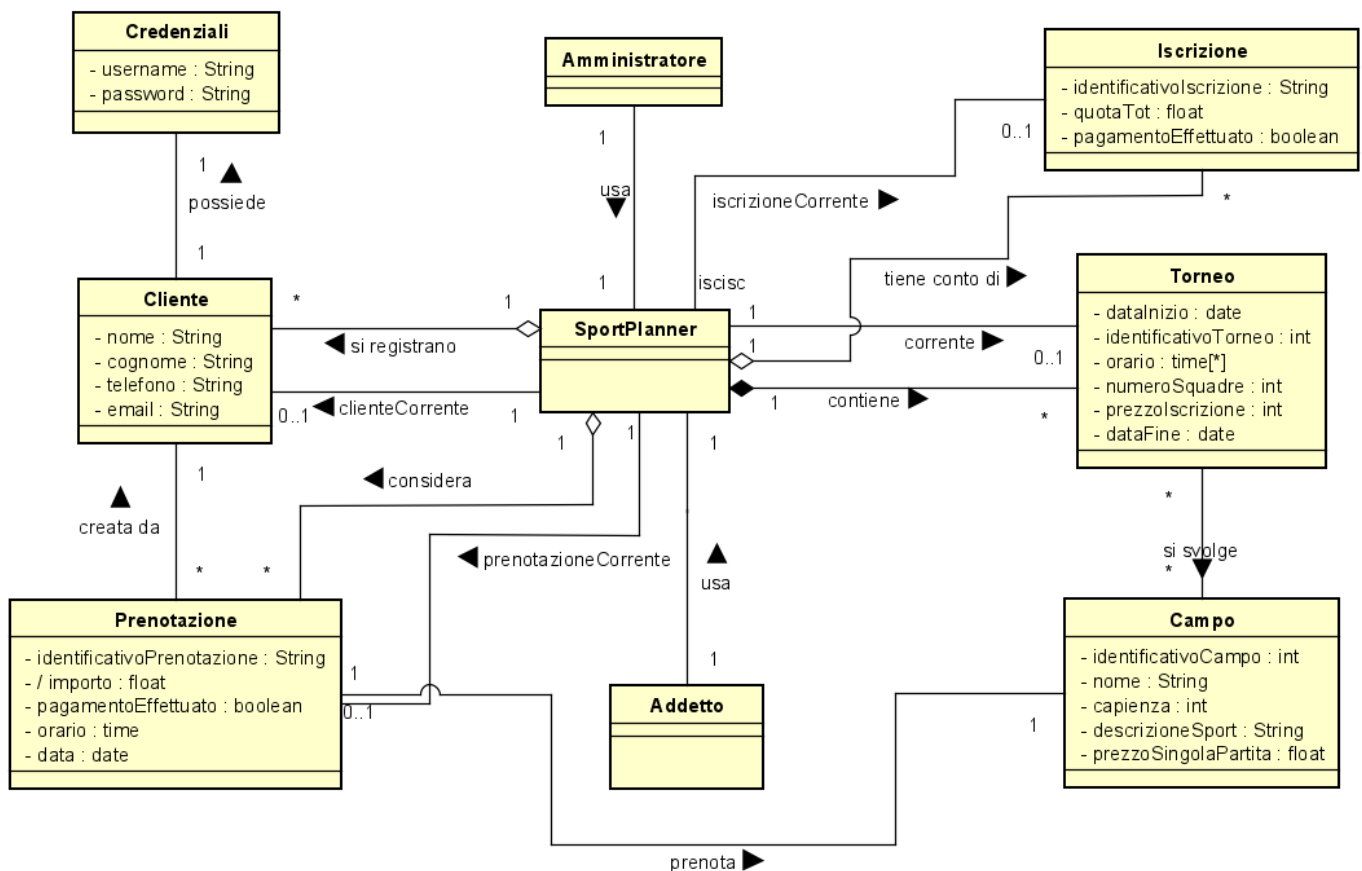
- Implementazione delle regole di dominio;
- Aggiornamento classi Prenotazione e Partita che vengono racchiusi in un'unica entità e con un'aggiunta in SportPlanner.
- Applicazione del pattern *Strategy* per la politica di gestione del prezzo di una Prenotazione.

1.5 Iterazione 5

- Progettazione dei rimanenti Casi d'uso: *UC4: Iscrizione al torneo*, *UC5: Pagamento iscrizione torneo* (che sarà un'evoluzione del caso d'uso UC3), *UC7: Visualizzare dettagli prenotazione*, *UC8: Inserimento nuovo campo*, *UC9: Visualizzare prenotazioni partite*, *UC10: Visualizzare iscrizioni tornei*;
- Miglioramento delle funzionalità offerte dalle interfacce grafiche

2 Modello di dominio

La disciplina che in termini di UP si occupa di fornire dettagli sul dominio è la Modellazione del Business, in particolare essa comprende la stesura del Modello di Dominio, elaborato grafico in cui vengono identificati i concetti, gli attributi e le associazioni che sono stati considerati più importanti. Tenendo conto del contributo dato da ciascuna iterazione, il modello di dominio è dato da:



(Allegato A1-Modello di Dominio)

Sono state individuate le seguenti classi concettuali:

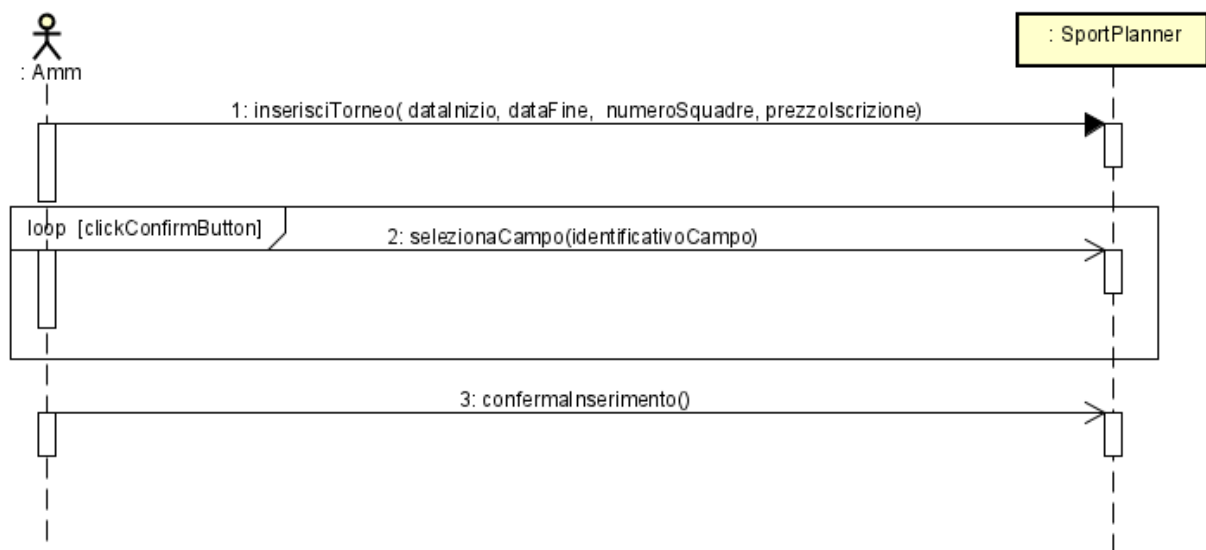
- **Amministratore**: rappresenta l'attore primario, che interagisce con il sistema per eseguire le operazioni;
- **SportPlanner**: rappresenta il sistema per la gestione dell'attività;
- **Torneo**: entità che rappresenta l'arco temporale in cui si potrà usufruire dei servizi dell'attività;
- **Campo**: rappresenta la l'ambientazione in cui si svolge un torneo;
- **Cliente**: rappresenta l'attore primario, che interagisce con il sistema per eseguire la registrazione;
- **Credenziali**: rappresentano la coppia di username e password che il Cliente utilizzerà per accedere al Sistema;
- **Prenotazione**: rappresenta il diritto del Cliente ad avere un campo riservato in una determinata data ad un determinato orario;
- **Addetto**: Persona fisica che si occupa della attività di gestione dei pagamenti, consentendoli ai Clienti che hanno effettuato una prenotazione di un campo o si sono iscritti ad un Torneo.
- **Iscrizione**: rappresenta il diritto del Cliente ad avere un posto riservato all'interno di un torneo;

3 SSD e contratti

Procedendo con l'analisi Orientata agli Oggetti, il passo successivo è la creazione dei Diagramma di Sequenza di Sistema (SSD) al fine di illustrare il corso degli eventi di input e di output per i vari casi d'uso esaminati in ciascuna iterazione. Inoltre, le principali operazioni di sistema individuate negli SSD verranno descritte attraverso i Contratti, quindi avremo:

- **Iterazione 1**

Scenario principale di successo del caso d'uso UC1:



Contratti delle operazioni del caso d'uso UC1

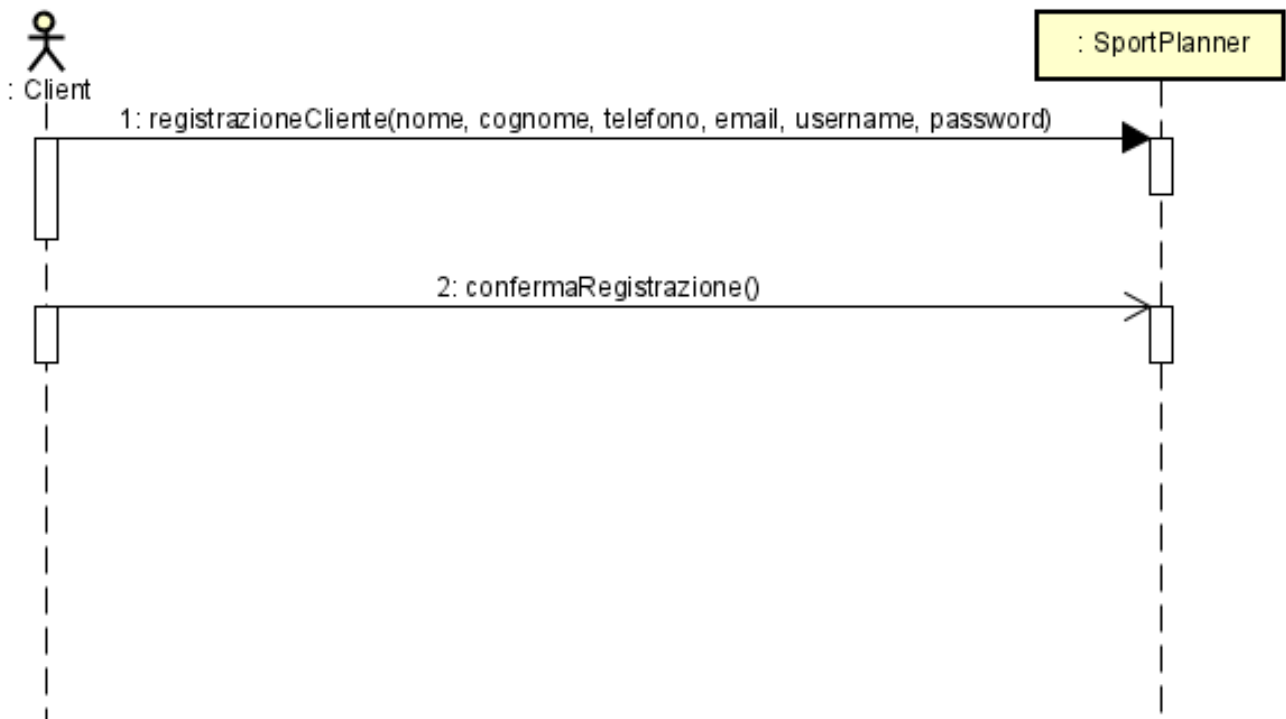
CONTRATTO CO1 inserisciTorneo	
Operazione	inserisciTorneo (dataInizio, dataFine, numeroSquadre, prezzoIscrizione)
Riferimenti	Caso d'uso UC1: Inserimento nuovo torneo
Pre-condizioni	-
Post-condizioni	<ul style="list-style-type: none">– è stata creata una nuova istanza trn di tipo Torneo;– gli attributi di trn sono stati inizializzati;– trn è stato associato a SportPlanner tramite l'associazione <i>corrente</i>

CONTRATTO CO2 selezionaCampo	
Operazione	selezionaCampo (identificativoCampo)
Riferimenti	Caso d'uso UC1: Inserimento nuovo torneo
Pre-condizioni	Sta avvenendo l'inserimento del Torneo trn
Post-condizioni	<ul style="list-style-type: none">– trn è stato associato a Campo tramite l'associazione <i>si svolge</i>

CONTRATTO CO3 confermaInserimento	
Operazione	confermaInserimento ()
Riferimenti	Caso d'uso UC1: Inserimento nuovo torneo
Pre-condizioni	Sta avvenendo l'inserimento del Torneo trn
Post-condizioni	<ul style="list-style-type: none">– trn è stato associato a <i>SportPlanner</i> tramite l'associazione <i>contiene</i>

- **Iterazione 2**

Scenario principale di successo del caso d'uso UC6:



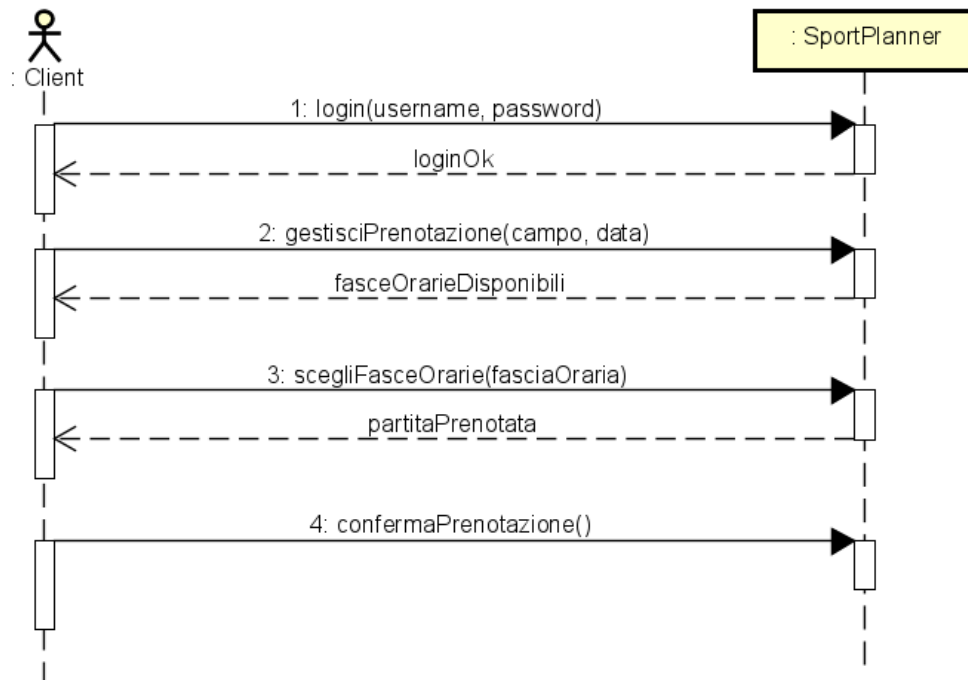
Contratti delle operazioni del caso d'uso UC6

CONTRATTO CO1	registrazioneCliente
Operazione	registrazioneCliente (nome, cognome, telefono, email, username, password)
Riferimenti	Caso d'uso UC6: Registrazione cliente
Pre-condizioni	-
Post-condizioni	<ul style="list-style-type: none"> – è stata creata una nuova istanza <i>cln</i> di tipo Cliente; – gli attributi di <i>cln</i> sono stati inizializzati;

CONTRATTO CO2	confermaRegistrazione
Operazione	confermaRegistrazione ()
Riferimenti	Caso d'uso UC6: Registrazione cliente
Pre-condizioni	Sta avvenendo l'inserimento del Cliente <i>cln</i> tramite l'associazione si registrano
Post-condizioni	– <i>cln</i> è stato associato a SportPlanner tramite l'associazione <i>si svolge</i>

- **Iterazione 3**

Scenario principale di successo del caso d'uso UC2:



Contratti delle operazioni del caso d'uso UC2

CONTRATTO CO1 login	
Operazione	login (username, password)
Riferimenti	Caso d'uso UC2: Gestisci prenotazione
Pre-condizioni	– Il Cliente <i>cln</i> si è precedentemente registrato
Post-condizioni	– Il Cliente <i>cln</i> è autenticato nel Sistema

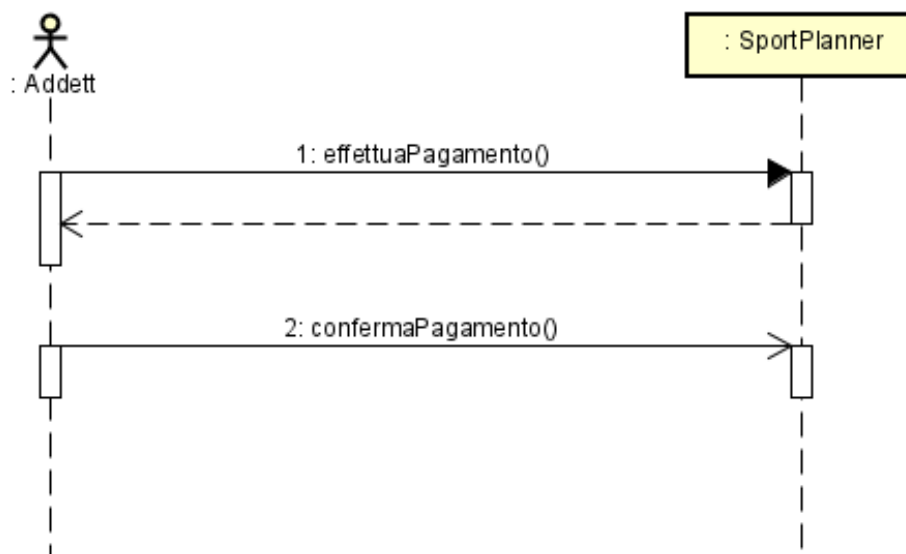
CONTRATTO CO2 gestisciPrenotazione	
Operazione	gesticiPrenotazione (campo, data)
Riferimenti	Caso d'uso UC2: Gestione prenotazione
Pre-condizioni	- Il Cliente <i>cln</i> è autenticato nel Sistema
Post-condizioni	– è stata creata una nuova istanza <i>partita</i> di tipo Partita – gli attributi di <i>partita</i> sono stati inizializzati – è stata creata una nuova istanza <i>prnt</i> di tipo Prenotazione – gli attributi di <i>prnt</i> sono stati inizializzati

CONTRATTO CO3 scegliFasciaOraria	
Operazione	scegliFasciaOraria(orario)
Riferimenti	Caso d'uso UC2: Gestione prenotazione
Pre-condizioni	– il Cliente <i>cln</i> ha scelto l'orario della partita
Post-condizioni	– viene inizializzato l'attributo <i>orario</i> dell'istanza <i>partita</i> .

CONTRATTO CO4 confermaPrenotazione	
Operazione	confermaPrenotazione()
Riferimenti	Caso d'uso UC2: Gestione prenotazione
Pre-condizioni	– è in corso la definizione di una Partita <i>partita</i> . è in corso la definizione di una Prenotazione <i>prnt</i>
Post-condizioni	– la Partita <i>partita</i> è stata associata a <i>SportPlanner</i> tramite l'associazione <i>contiene</i> . – la Prenotazione <i>prnt</i> è stata associata a <i>SportPlanner</i> tramite l'associazione <i>considera</i> . – la Prenotazione è associata a <i>Credenziali</i> tramite l'associazione <i>creata da</i> .

- **Iterazione 4**

Scenario principale di successo del caso d'uso UC3:



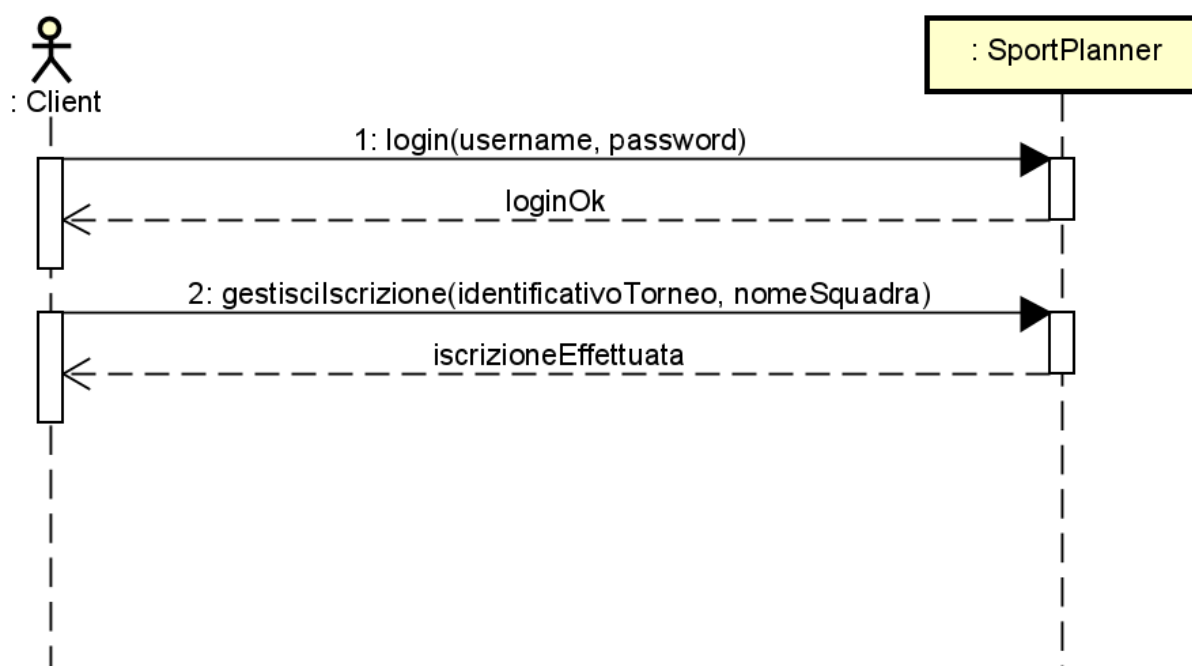
Contratti delle operazioni del caso d'uso UC3

CONTRATTO CO1	effettuaPagamento
Operazione	effettuaPagamento (identificativoPrenotazione)
Riferimenti	Caso d'uso UC3: Pagamento prenotazione partita
Pre-condizioni	–
Post-condizioni	–

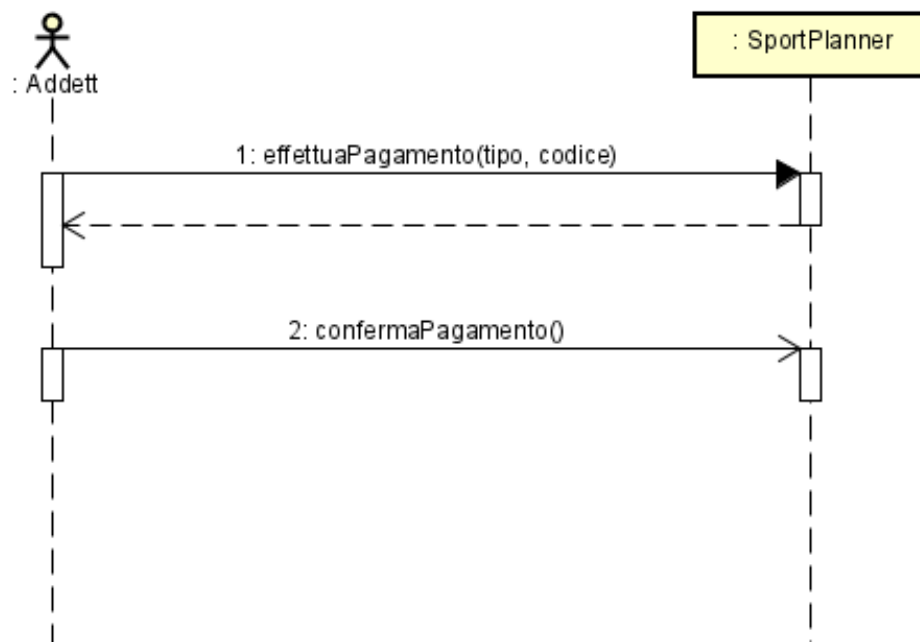
CONTRATTO CO2	confermaPagamento
Operazione	confermaPagamento ()
Riferimenti	Caso d'uso UC3: Pagamento prenotazione partita
Pre-condizioni	– <i>identificativoPrenotazione</i> è stato accettato dal Sistema
Post-condizioni	– gli attributi di Prenotazione sono stati aggiornati

- Iterazione 5**

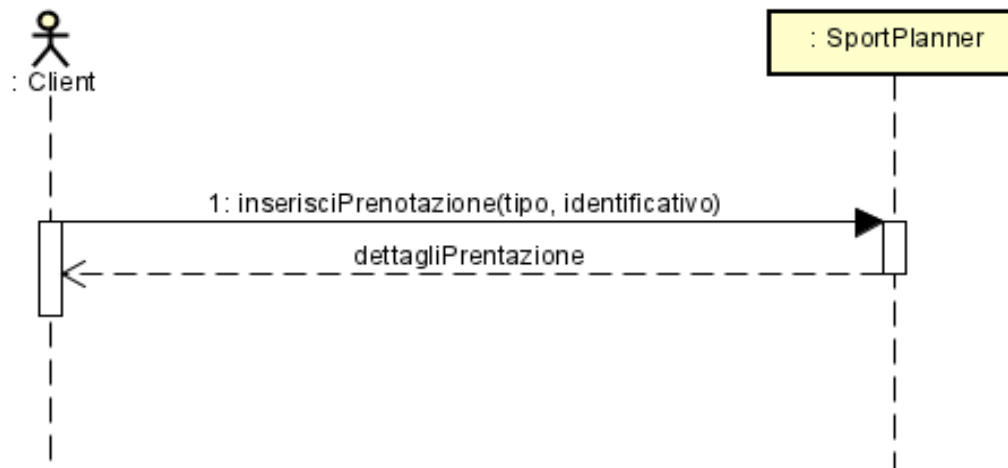
Scenario principale di successo del caso d'uso UC4:



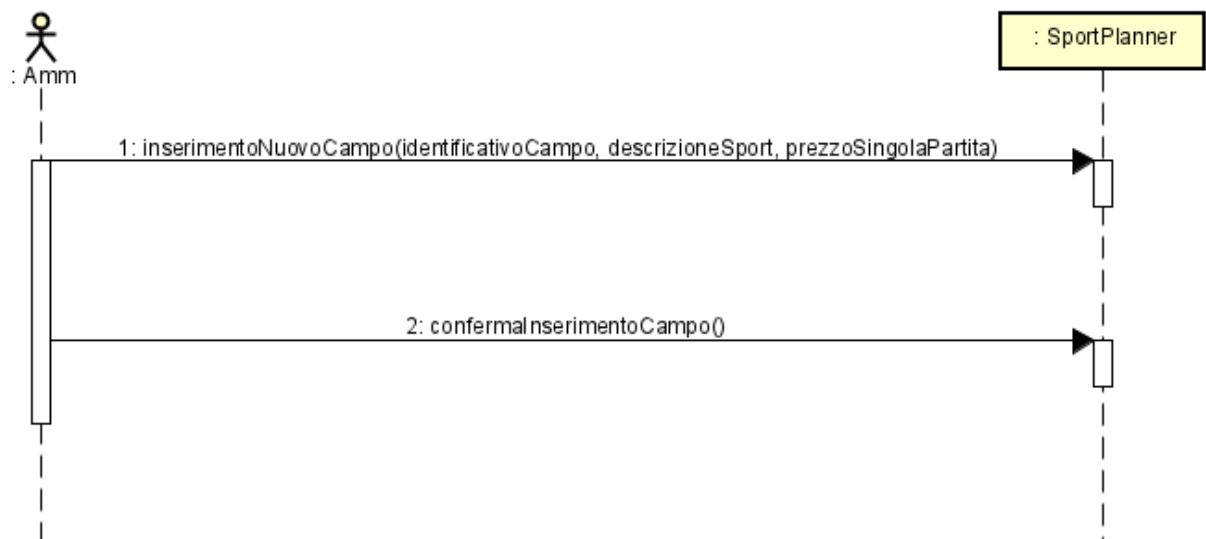
Scenario principale di successo del caso d'uso UC5:



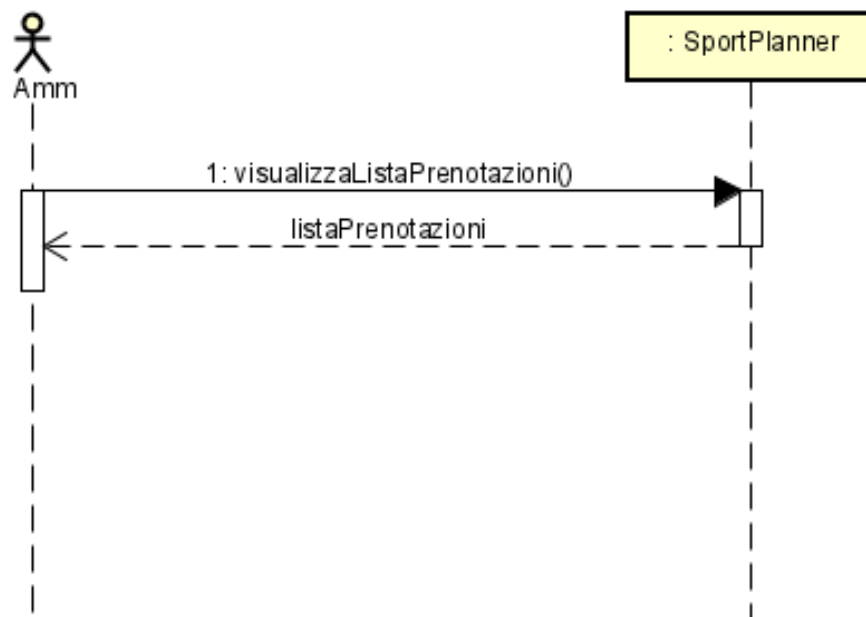
Scenario principale di successo del caso d'uso UC7:



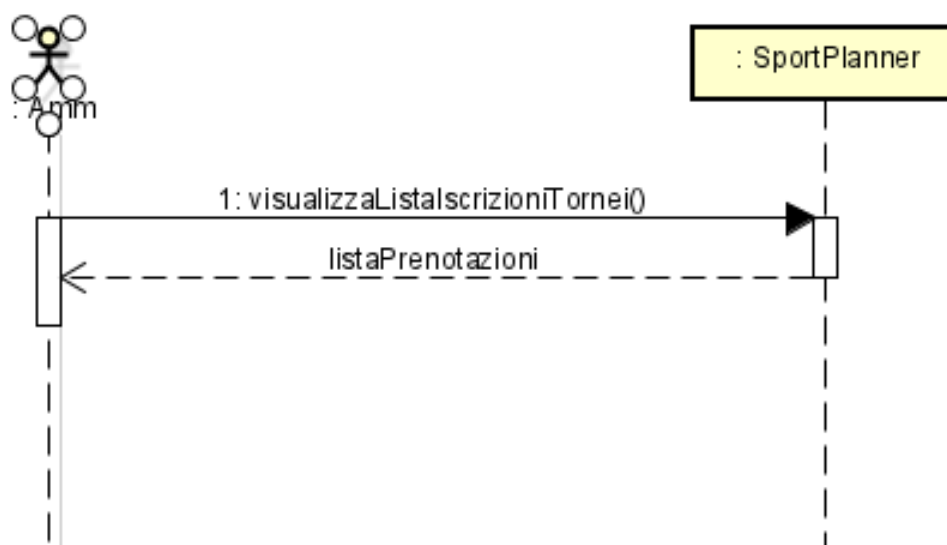
Scenario principale di successo del caso d'uso UC8:



Scenario principale di successo del caso d'uso UC9:



Scenario principale di successo del caso d'uso UC10:

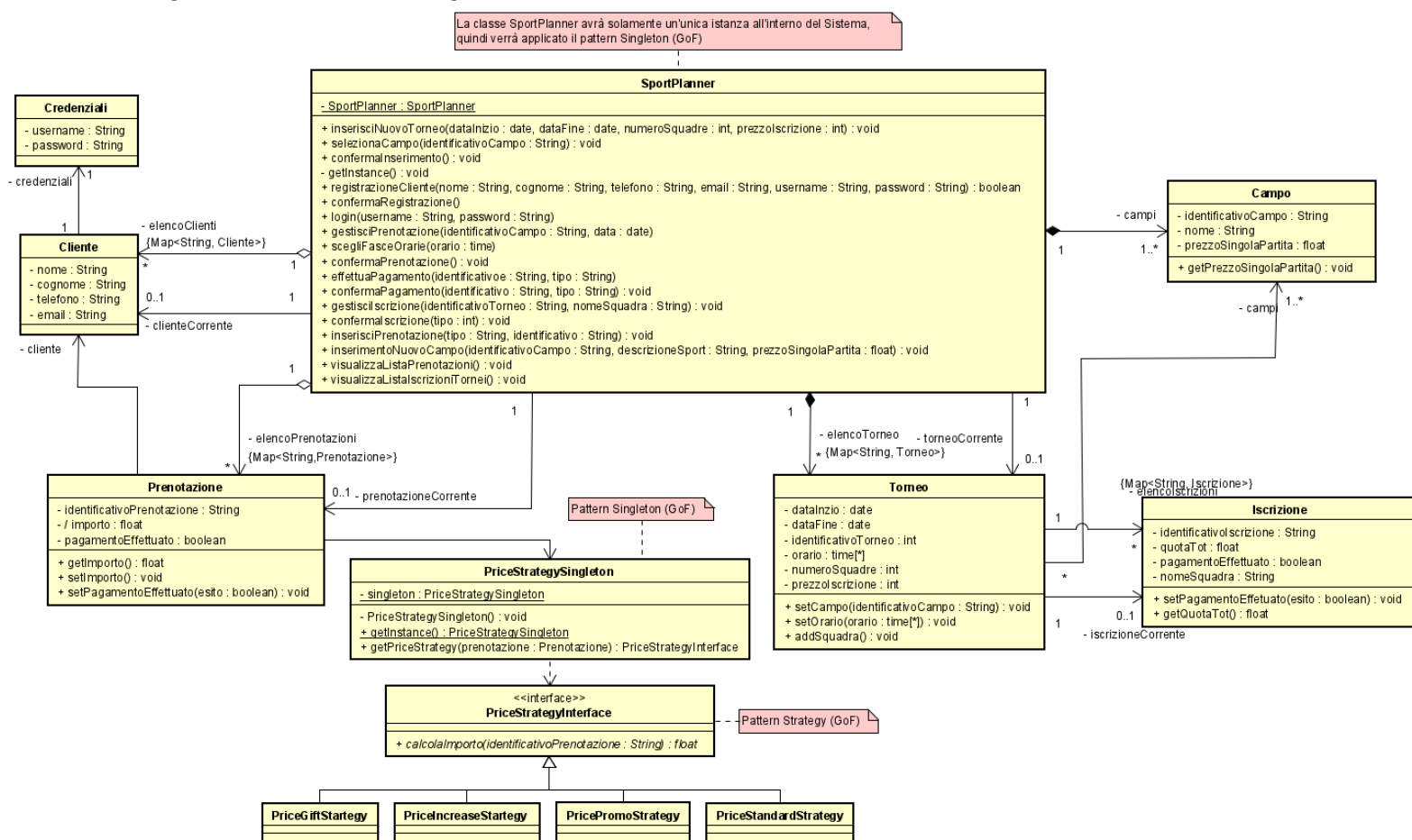


PROGETTAZIONE

1 Diagramma delle classi

La progettazione orientata agli oggetti è la disciplina di UP che si riferisce alla definizione degli oggetti software, delle loro responsabilità e a come questi collaborano per soddisfare i requisiti individuati precedentemente. L'elaborato principale di questa fase che è stato preso in considerazione è il **Modello di Progetto**, ovvero l'insieme dei diagrammi che descrivono la progettazione logica sia da un punto di vista dinamico con l'utilizzo dei Diagrammi di Interazione che da un punto di vista statico attraverso i Diagramma delle Classi.

Il diagramma delle Classi è il seguente:

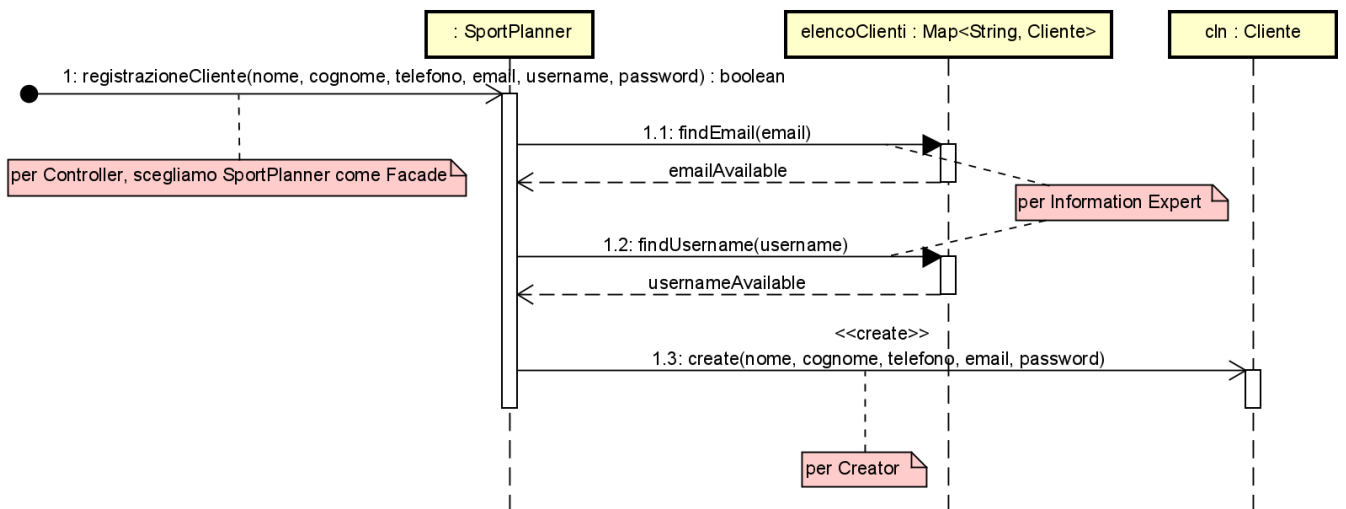


(AllegatoB1-Diagramma delle Classi)

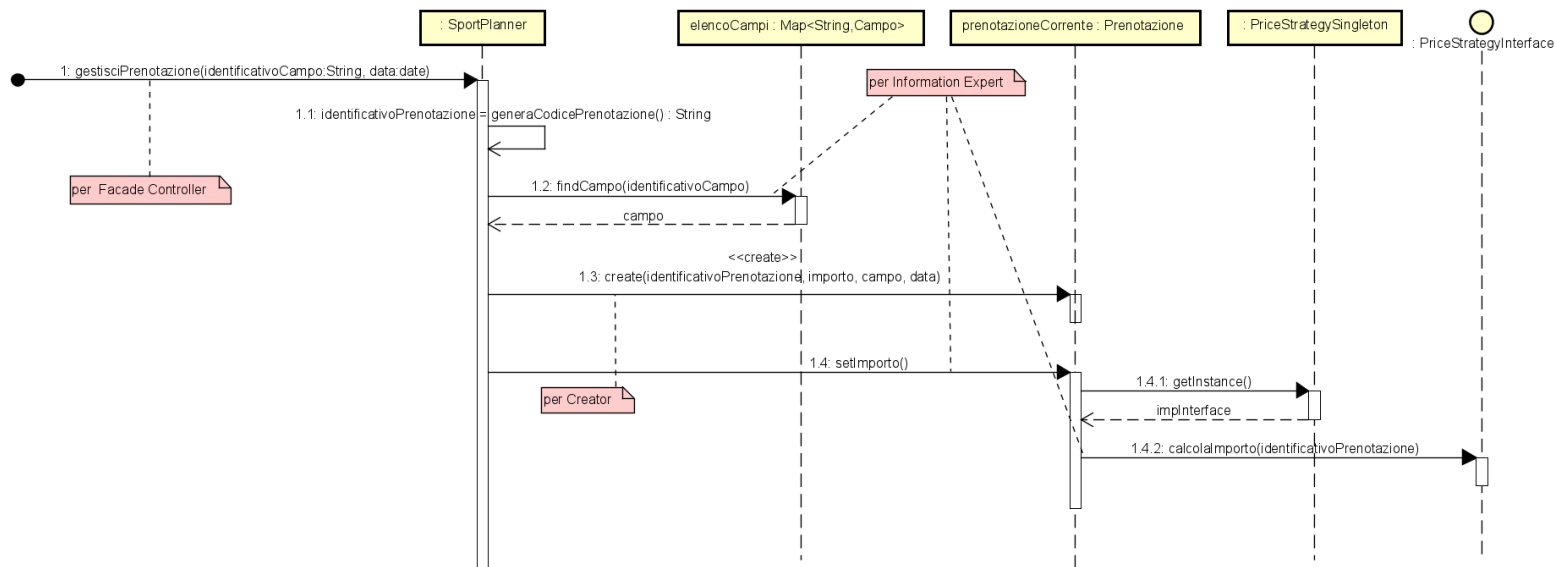
2 Diagrammi di sequenza

Di seguito sono riportati i diagrammi di sequenza delle operazioni considerate più importanti. Per visualizzare tutti i diagrammi si rimanda al realizzato con il software *Astah* (nome file *progetto-finale.asta*) o ai documenti sulle varie fasi di iterazioni nelle relative sottocartelle.

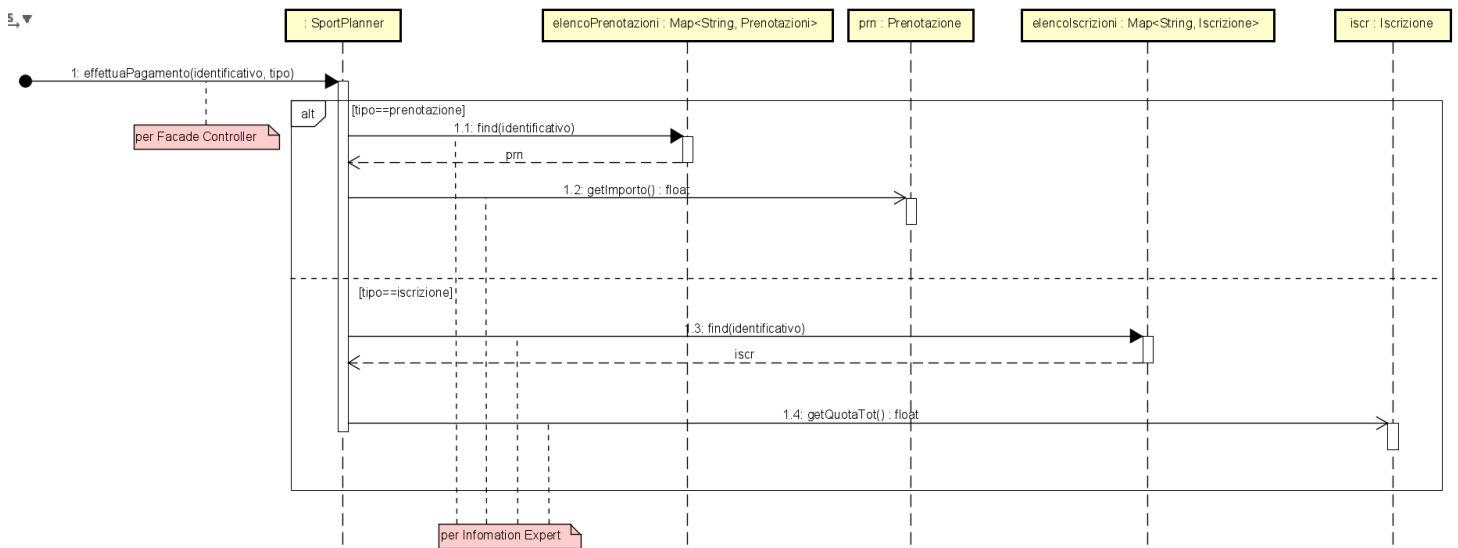
Registrazione cliente (UC6)



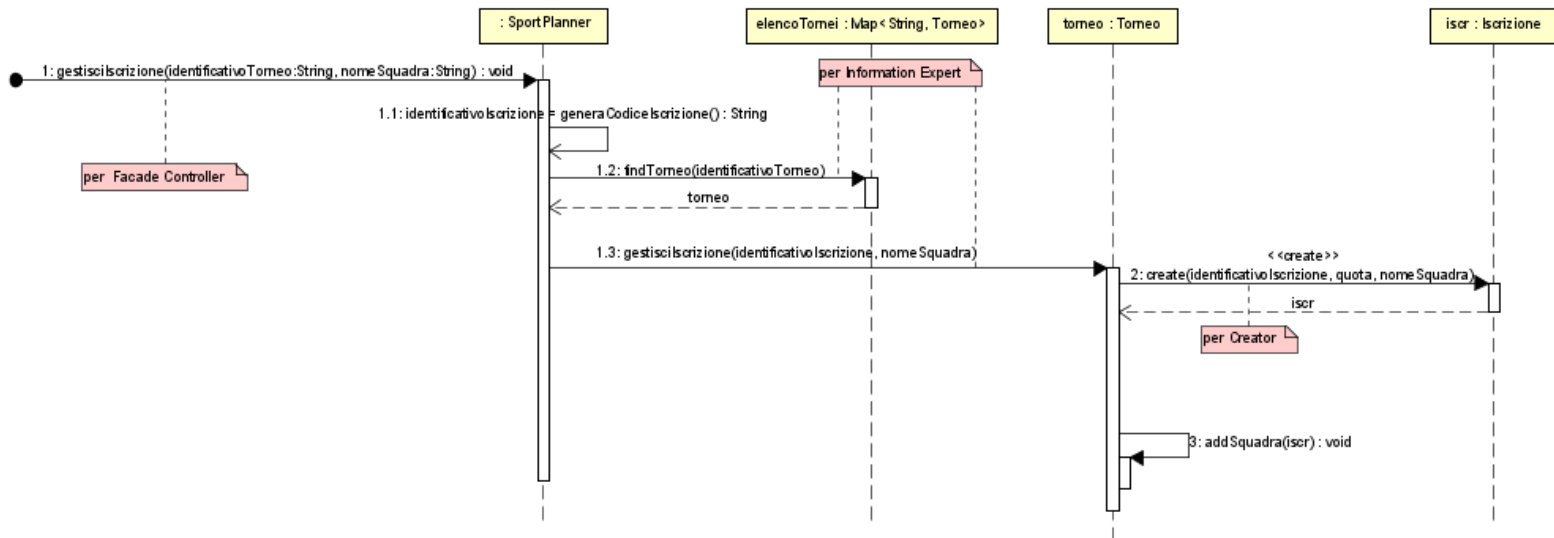
Gestione prenotazione (UC2)



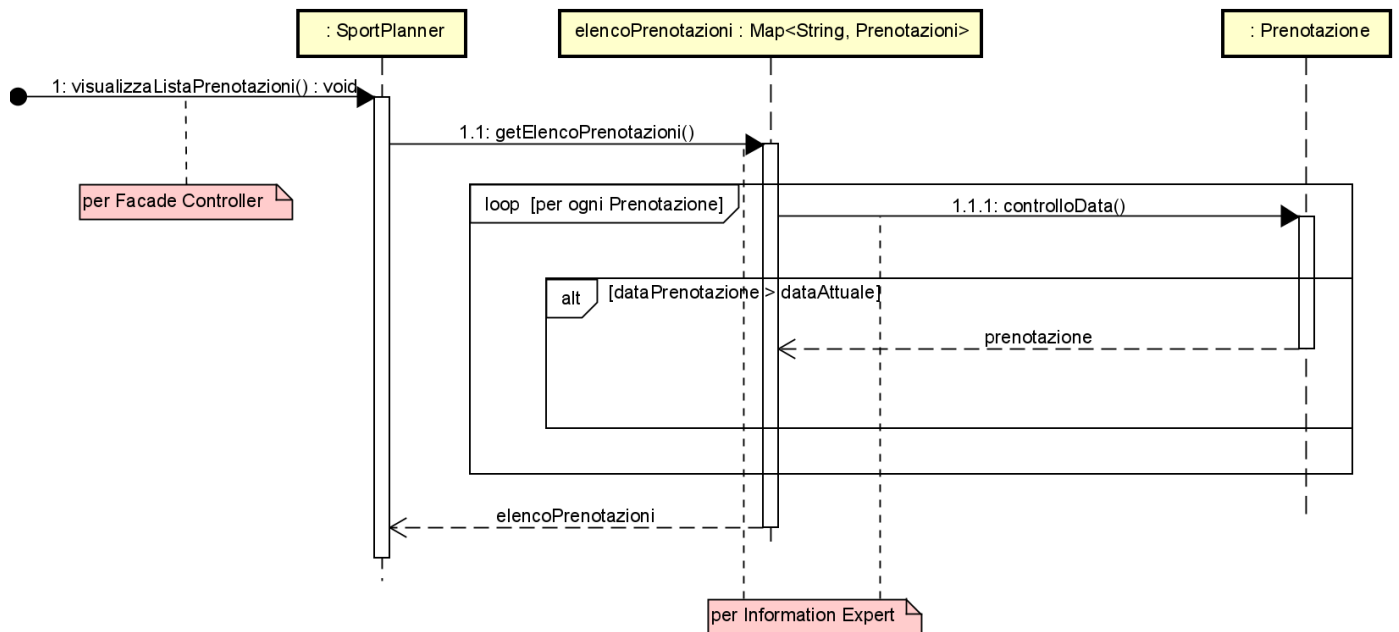
Effettua pagamento (UC3/UC5)



Iscrizione torneo (UC4)



Visualizza lista prenotazioni (UC9)



TESTING

Il testing è una fase essenziale del processo di sviluppo di un programma che deve essere robusto ed efficiente e permette inoltre di ridurre in modo sostanziale i costi di manutenzione di un'applicazione.

La probabilità di rilevare malfunzionamenti è proporzionale al numero di test eseguiti; tuttavia, è difficile testare un programma per intero. Infatti, le possibili combinazioni di valori di input da prendere in considerazione sarebbero enormi e non sempre possono essere riprodotte in un tempo ragionevole. Però un testing progettato accuratamente può rivelare comportamenti anomali ed indesiderati al fine di rendere il software realizzato funzionante e funzionale secondo le specifiche richieste.

I test si dividono principalmente in due famiglie:

- **I test unitari** (o Unit Test) → Sono semplici test/prove che vanno a verificare la correttezza direttamente del codice, in ogni sua piccola parte. L'idea dello Unit Test in Java è quella di valutare ogni singolo metodo in funzione dei valori attesi. Esistono diversi tool per Unit Testing per i diversi linguaggi di programmazione, in particolare JUnit è il framework più diffuso attualmente per l'automazione del testing di unità di programmi Java.
- **I test funzionali** → Sono dei test che vanno a verificare che il sistema software nella sua completezza funzioni correttamente. Questi test trattano il sistema come se fosse una scatola nera alla quale danno degli input e verificano la correttezza degli output.

Per eseguire il testing dell'applicazione realizzata, si è scelto di concentrarsi principalmente su test unitari eseguiti tramite l'ausilio di JUnit e quindi di seguire un approccio detto **BottomUp** collaudando prima le piccole unità che comporranno poi il programma completo.

Si è scelto di effettuare i seguenti test nelle rispettive classi:

- **SportPlanner**
 - *inserisciNuovoTorneo*
 - Il torneo che si sta inserendo diventa corrente per SportPlanner.
 - *selezionaCampo*
 - Si settano i campi per il torneo che si sta inserendo.
 - *confermaInserimento*
 - Un torneo corrente per SportPlanner viene correttamente aggiunto all'elenco di tornei di SportPlanner
 - *registrazioneCliente*
 - Il Cliente che sta effettuando la registrazione diventa corrente per SportPlanner.
 - *confermaRegistrazione*
 - Il Cliente corrente per SportPlanner viene inserito in modo corretto all'interno dell'elenco dei clienti di SportPlanner
 - *login*
 - Il Cliente inserisce le proprie credenziali e diventa *clienteCorrente*.

- *gestisciPrenotazione*
 - Viene creata l'istanza di Prenotazione
- *scegliFasciaOraria*
 - viene settato l'attributo *orario* nell'istanza di Partita
- *confermaPrenotazione*
 - Una prenotazione corrente viene inserita nell'elenco delle prenotazioni del Sistema
- *gestisciscrizione*
 - Viene creata l'istanza di Iscrizione
- *validaCampiTorneo*
 - Verifica se l'identificativo del torneo inserito è vuoto
 - Verifica se la descrizione del torneo inserito è vuota
 - Controlla che la data di inizio del torneo sia successiva alla data odierna
 - Controlla che la data di fine sia successiva a quella odierna
 - Controlla che la data di inizio sia precedente a quella di fine del torneo
 - Controlla che sia stato inserito un valore di quota di iscrizione valido
- *validaCampiCliente*
 - Verifica se tutti i campi sono stati compilati
 - Consente di inserire un nome non troppo breve
 - Consente di inserire un cognome non troppo breve
 - Controlla che il numero di telefono inserito sia valido
 - Controlla che la mail abbia un formato valido ('mail@prova.it')
 - Controlla che l'username e la password non siano troppo brevi
- *validaInserimentoCredenziali*
 - Verifica se tutti i campi sono stati compilati
 - Controlla che l'username e la password inseriti non siano troppo brevi
- *validaCampoData*
 - Verifica che il codice del campo non sia vuoto
 - Controlla che il codice del campo scelto sia valido
 - Controlla che la data scelta sia successiva a quella odierna
- *validaTorneo*
 - Verifica che il codice del torneo non sia vuoto
 - Verifica che il codice del torneo sia valido
 - Controlla che il nome della squadra inserito non sia nullo
- **Torneo**
 - *gestisciscrizione*
 - Viene creata una nuova iscrizione ad un Torneo e viene aggiunta la squadra nell'elenco di partecipanti al Torneo, incrementando la dimensione di uno.
- **Credenziali**
 - *verificaPassword*
 - Controlla che la password non sia nulla e che sia valida
 - Valida l'accesso, controllando se la password inserita corrisponde a quella inserita dall'utente in fase di registrazione

A seguito dell'esecuzione di questi test unitari con JUnit, si è passati a correggere le relative porzioni di codice che portavano al fallimento di alcuni test.

1 Test di sistema

A seguito di tali test non sono emersi grandi errori nel funzionamento e le prestazioni del sistema si sono mantenute piuttosto elevate.

REFACTORING E CONCLUSIONI

1. Refactoring

In questa fase conclusiva principalmente è stata prevista l'aggiunta di alcune funzionalità secondarie dell'applicazione, quali la visualizzazione dei dettagli di una prenotazione o di un'iscrizione, ma anche il recupero di tutta la lista di prenotazioni e iscrizioni da parte dell'amministratore.

2. Test di accettazione

A conclusione delle attività necessarie per portare a completamento i passi precedenti, e dopo un'attenta revisione di tutto il codice sorgente, è stato eseguito il test di accettazione conclusivo.

Sono state fatte delle simulazioni sia nei panni del Cliente che dell'Amministratore e sono state testate tutte le principali attività del programma, per verificare che tutto funziona e che i casi d'uso implementati vengano portati a termine con successo.

3. Pattern GoF

3.1. Singleton

Questo pattern, che fa parte dei pattern *creazionali*, ci assicura che una classe abbia un'unica istanza e punto di accesso globale.

La classe SportPlanner e la classe PriceStrategySingleton sono state rese Singleton, in modo tale da impedire la creazione di più istanze e consentire un accesso unico attraverso un metodo statico getInstance().

3.2. Strategy

Questo pattern comportamentale è stato utilizzato per mettere in pratica le regole di dominio del sistema.

L'obiettivo di questa architettura è isolare un algoritmo all'interno di un oggetto, in maniera tale da risultare utile in quelle situazioni dove sia necessario modificare dinamicamente gli algoritmi utilizzati da un'applicazione.

Così sono state aggiunte le seguenti classi:

- Interfaccia PriceStrategyInterface
- Classe PriceStandardStrategy
- Classe PricePromoStrategy
- Classe PriceIncreaseStrategy
- Classe PriceGiftStrategy
- Classe PriceStrategySingleton

