

SALA GIOCHI



**IL PIATTINO
D'ORO**

Ideazione e analisi dei requisiti

◆ Requisiti

Il titolare di una sala giochi vuole chiedere la realizzazione di un software per la gestione della sala e dei clienti. Tale software rappresenta uno strumento per gestire tutte le varie funzionalità che la sala giochi offre, dalle macchine da gioco al bar interno, oltre che a sovrintendere alle operazioni svolte con la carta fedeltà del cliente. In particolare si richiedono le seguenti funzionalità:

- Il gestore può tenere traccia dei clienti registrati mediante una carta fedeltà, oltre che verificare le loro preferenze e tener traccia dei punti realizzati nei vari cabinati.
- Il gestore vuole tener traccia, tramite la carta fedeltà, dei gettoni attualmente presenti, spesi e acquisiti dal cliente e poterli comunicare.
- Un cliente può decidere di visualizzare la classifica generale per un dato gioco visionando i primi 10 posti ed il suo eventuale posizionamento, oltre ai vari punteggi accumulati singolarmente per ogni gioco a cui ha partecipato tramite la carta fedeltà.
- Il software monitora lo stato dei vari giochi e macchinari, i quali si suddividono in cabinati (flipper, giochi arcade, freccette), distributori (claw crane, distributori di palline) e tavoli (biliardo, biliardino). Di ognuno ne monitora lo stato di attività, se è attualmente in corso una partita e il loro andamento.
- Il gestore tiene anche conto anche delle scorte di magazzino per il bar interno e per i premi in palio.
- Quando un cliente si presenta, se è la prima volta che accede, si crea la tessera fedeltà e si caricano i primi gettoni; in alternativa se è già registrato, il sistema verifica lo stato della carta e l'ammontare dei gettoni e dei punti presenti in essa. A dipendenza di quanti gettoni e punti possiede, il cliente può scegliere tra:
 1. acquistare gettoni per poter giocare se essi risultano insufficienti
 2. spendere i gettoni per poter giocare ai vari giochi.
 3. spendere i punti per poter riscattare premi od ordinare cibi e bevande al bar interno
 4. rifiutarsi di acquistare o spendere gettoni/punti e andarsene

Nel caso di acquisto gettoni e riscossione premi non è disponibile il rimborso, a meno di possibili errori durante la transazione o di guasti al cabinato durante la partita.

- Nel caso in cui serva una nuova scheda, il sistema calcola il prezzo base a meno di promozioni in corso e in aggiunta precarica un ammontare di 100 gettoni come politica base. Se il cliente conferma si inoltrerà alla cassa tutte le relative informazioni per poi concludere l'acquisto da parte del cliente. La carta verrà erogata da una colonnina apposita, che terrà traccia anche delle carte in bianco rimanenti.
- Un cliente può decidere di fare una carta fedeltà VIP ad un prezzo maggiorato, che gli consente di ottenere premi extra e una consumazione al bar gratuita a giornata.
- Un cliente può anche decidere di prenotare un gioco per una quantità fissa di 30 min,

fornendo le varie informazioni quali il tipo di gioco, la data, l'orario, il numero di partecipanti e un recapito telefonico per eventuali contatti.

- Nello stipulare la prenotazione è possibile scegliere se usufruire o no della consumazione al bar, influenzando il prezzo finale che il cliente dovrà pagare.
- La prenotazione può essere annullata e verrà garantito il rimborso dei soldi spesi.
- Il software permette anche di gestire i dirigenti e i dipendenti al momento in servizio nella sala giochi, gestendo sia il check in che il check out e la disponibilità. Entrambe le figure sono caratterizzate da un ID di riconoscimento ed un codice da utilizzare per accedere alla macchina.
- Tramite il sistema, un cliente può visionare l'andamento di una partita, visualizzando quanti giocatori sono presenti, il tempo di gioco e il punteggio attuale

◆ Obiettivi e casi d'uso

Analizzando i requisiti sono stati individuati i vari attori principali e gli obiettivi che esso intende portare a termine usando l'applicazione; da queste informazioni, e dopo un'ulteriore analisi rispetto alla prima stesura, sono stati ricavati i seguenti casi d'uso principali.

Attore	Obiettivo	Caso d'uso
Amministratore / Dipendente	Creare un nuovo tipo di premio ed inserirne copie, oppure inserire copie di premi già esistenti.	UC1: Gestisci Premi
Dipendente	Prenotare un gioco per una determinata data per un cliente, riservando i posti ed eventuali consumazioni.	UC2: Gestisci Prenotazione
Cliente	Permettere al cliente di riscattare i premi in base ai punti accumulati.	UC3: Riscatta Premio
Dipendente	Su richiesta del cliente, monitorare l'andamento di una partita in corso	UC4: Monitora Partita
Cliente	Creare una carta associata al cliente (1 per cliente).	UC5: Crea Carta
Amministratore / Dipendente	Creare un oggetto riscattabile al bar ed inserirne copie, oppure inserire copie di oggetti già esistenti.	UC6: Gestisci Bar
Cliente	Ricaricare i gettoni presenti sulla carta per poter giocare.	UC7: Ricarica Carta

Cliente	Permettere al cliente di acquistare cibarie al bar.	UC8: Compra Bar
Cliente	Recuperare il codice univoco della carta.	UC9: Recupera Codice
Dipendente	Registrare un dipendente come presente quel giorno a lavoro.	UC10: Check-in Dipendente
Dipendente	Registrare che un dipendente ha finito il turno lavorativo.	UC11: Check-out Dipendente
Amministratore	Controllo dei dipendenti al momento a lavoro.	UC12: Monitoraggio dipendenti
Cliente	Controllare i record personali nei vari giochi.	UC13: Record Personali
Cliente	Controllare la top 10 di un gioco.	UC14: Visualizza Classifica
Amministratore	Aggiungere un nuovo gioco presente in sala.	UC15: Aggiungi Gioco
Amministratore	Controllare i giochi attualmente attivi e se è presente una partita in corso.	UC16: Monitora Macchina
Cliente	Avviare una partita per un gioco scelto dal cliente.	UC17: Avvia Partita

◆ Modello dei casi d'uso

Di seguito saranno mostrati tutti i modelli di caso d'uso nel dettaglio e nello scenario di successo, ad eccezione del caso d'uso UC8: Compra Bar, ritenuto meno rilevante ai fini dell'applicazione generale e per questo esposto solo sinteticamente.

UC1: Gestisci premi

Nome del caso d'uso	UC1: Gestisci premi
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Amministratore / Dipendente
Parti interessate e Interessi	<ul style="list-style-type: none"> Amministratore / Dipendente: vuole organizzare l'inventario dei premi, aggiungendone tipi nuovi o rifornendo i tipi già presenti.
Pre-Condizioni	La sala giochi ha acquistato i premi che possono essere riscattati.
Garanzia di successo	Le informazioni relative al premio sono inserite con successo nel Sistema.
Scenario principale di	1. L'amministratore (Dipendente) vuole inserire un nuovo premio

successo	<p>con le relative copie.</p> <ol style="list-style-type: none"> 2. L'amministratore (Dipendente) sceglie l'attività "Inserisci nuovo premio". 3. L'amministratore (Dipendente) inserisce il nome, il codice e il prezzo standard del premio. 4. L'amministratore (Dipendente) inserisce per il premio appena inserito le informazioni della copia. <i>Il passaggio 4 viene ripetuto finché serve.</i> 5. L'amministratore (Dipendente) indica di aver finito.
Estensioni	<p>*a. In un qualsiasi momento il Sistema fallisce e si arresta improvvisamente.</p> <ol style="list-style-type: none"> 1. L'amministratore (Dipendente) riavvia il software e ripristina lo stato precedente del Sistema. 2. Il Sistema ripristina lo stato. <p>3a. L'amministratore (Dipendente) inserisce un codice univoco già presente nel sistema.</p> <ol style="list-style-type: none"> 1. Il Sistema genera un messaggio per chiedere se si vuole inserire un altro codice o se si vuole inserire delle copie per il premio con il codice inserito. <p>2a. L'amministratore (Dipendente) vuole un nuovo premio e ripete il passaggio 3 inserendo un codice diverso.</p> <p>2b. L'amministratore (Dipendente) sceglie l'opzione "Inserisci copie del premio", si passa al passaggio 4 (non viene creato un nuovo premio).</p>
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizioni	Ogni qual volta la sala giochi intende mettere disponibile un nuovo premio o si vogliano aggiornare le scorte.
Varie	

UC 2: Gestisci Prenotazione

Nome del caso d'uso	UC2: Gestisci prenotazione
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Dipendente

Parti interessate e Interessi	<ul style="list-style-type: none"> • Dipendente: vuole poter gestire la prenotazione di un gioco da parte di un cliente. • Cliente: vuole poter prenotare un gioco in una data specifica per sé ed altri.
Pre-Condizioni	La sala giochi ha i giochi richiesti, deve essere aperta nella data indicata dal cliente e non avere una prenotazione già attiva per quel gioco nella stessa data
Garanzia di successo	La prenotazione è avvenuta senza problemi e comunicata al cliente.
Scenario principale di successo	<ol style="list-style-type: none"> 1. Un cliente arriva e intende prenotare un gioco per sé ed un eventuale comitiva. 2. Il dipendente controlla la carta fedeltà del cliente per verificarne la registrazione. 3. Il cliente comunica il giorno e l'ora della prenotazione. 4. Il dipendente utilizza il software per verificare quali giochi sono disponibili per quella data e li comunica al cliente. 5. Il cliente comunica il numero degli altri partecipanti. 6. Il cliente comunica il gioco che vuole prenotare. 7. Il dipendente registra quel gioco come prenotato. (i passi 6 e 7 si ripetono per ogni gioco che vuole prenotare) 8. il dipendente registra le informazioni. 9. Il software elabora il prezzo totale della prenotazione. 10. Il cliente comunica il proprio recapito telefonico. 11. Il dipendente registra il numero e lo associa al cliente. 12. Il cliente conferma la prenotazione, paga e se ne va. 13. Il sistema aggiorna i giochi disponibili per un'eventuale nuova prenotazione.
Estensioni	<p>* In qualsiasi momento, il sistema ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il dipendente riavvia il software e richiede il ripristino. 2. Il sistema restituisce lo stato precedente. <p>2a. Il cliente non possiede la carta fedeltà.</p> <ol style="list-style-type: none"> 1. Il cliente se già registrato comunica il proprio numero tessera e il nominativo. 2. Il dipendente controlla se la tessera risulta registrata.[<i>vedi Recupero Codice</i>]. 3. Se il cliente non è registrato si registra. [<i>vedi Crea Carta</i>]. <p>4a. Il software non trova giochi disponibili per quella data.</p> <ol style="list-style-type: none"> 1. Il cliente comunica un'altra data oppure se ne va. <p>8a. Il cliente comunica anche la volontà di usufruire della consumazione al bar.</p> <ol style="list-style-type: none"> 1. il cliente comunica la bevanda.

	<p>2. il dipendente associa il cibo alla prenotazione. (i passi 2 e 3 si ripetono per ogni cibo)</p> <p>3. Il software elabora il prezzo totale.</p> <p>12a. Il cliente sceglie se pagare in contanti o usando i gettoni presenti nella carta.</p> <p>1. Se il cliente sceglie di pagare in gettoni si verifica prima se i gettoni sono in numero sufficiente.</p> <p>2. Se i gettoni non sono sufficienti viene mostrato un segnale di errore.</p>
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	<p>Per ogni prenotazione si devono registrare il nome del cliente, il tipo di gioco (1 o più), la data, l'orario, il numero di partecipanti, il numero telefonico e l'eventuale consumazione.</p> <p>Il costo è di 10€ per partecipante, con ulteriori 3€ per la consumazione. Se il cliente è VIP il prezzo relativo alla sua partecipazione è di 7€</p> <p>Se si sceglie di pagare con i gettoni la conversione è di 1 gettone = 0,50€</p>
Frequenza di ripetizioni	Ogni qual volta che un cliente decide di prenotare un gioco.
Varie	La prenotazione vale 30 minuti per gioco per ogni prenotazione

UC3: Riscatta Premi

Nome del caso d'uso	UC3: Riscatta premi
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Dipendente
Parti interessate e Interessi	<ul style="list-style-type: none"> Dipendente: vuole poter gestire la riscossione di un premio da parte di un cliente. Cliente: vuole riscattare un premio in base ai punti ottenuti.
Pre-Condizioni	Il cliente possiede abbastanza punti per poter riscattare un premio.
Garanzia di successo	Il cliente usa i suoi punti e riceve un premio in omaggio.
Scenario principale di successo	<ol style="list-style-type: none"> Un cliente arriva e decide di riscattare un premio. Il cliente inserisce nel sistema la propria carta fedeltà. Il dipendente controlla l'ammontare dei punti del cliente. Il dipendente ricerca tutti i premi disponibili che abbiano un costo pari o inferiore ai punti del cliente. Il sistema recupera i premi e la loro disponibilità. Il cliente decide quale premio ritirare.

	<p>7. Il dipendente consegna il premio al cliente e diminuisce il totale di punti cumulati dalla carta fedeltà</p> <p>8. Il sistema aggiorna l'ammontare dei premi disponibili. (i passi dal 3 al 8 si ripetono fino a quanto necessario)</p> <p>9. Il cliente è soddisfatto e se ne va.</p>
Estensioni	<p>* In qualsiasi momento, il sistema ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il dipendente riavvia il software e richiede il ripristino. 2. Il sistema restituisce lo stato precedente. 3. Se i punti risultano spesi vengono restituiti <p>2a. Il cliente non possiede la tessera.</p> <ol style="list-style-type: none"> 1. Il cliente se già registrato comunica il proprio numero tessera e il nominativo. 2. Il dipendente controlla se la tessera risulta registrata.[vedi UC9]. <p>4a. Il cliente possiede una tessera VIP.</p> <ol style="list-style-type: none"> 1. Il sistema recupera i premi disponibili contrassegnati come VIP che sono di costo pari o inferiore al punteggio del cliente <p>5a. Il software non trova premi disponibili in base ai punti comunicati.</p> <ol style="list-style-type: none"> 1. Il cliente se ne va.
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	Se il cliente ha una carta fedeltà VIP i premi disponibili cambiano.
Frequenza di ripetizioni	Ogni qual volta che un cliente decide di riscattare un premio.
Varie	

UC4: Monitora Partita

Nome del caso d'uso	UC4: Monitora Partita
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Dipendente
Parti interessate e Interessi	<ul style="list-style-type: none"> • Dipendente: vuole poter verificare lo stato delle partite in corso. • Cliente: vuole poter vedere l'andamento di una partita in corso.
Pre-Condizioni	Deve essere attualmente attiva una partita
Garanzia di successo	Il cliente visualizza lo stato della partita per il gioco richiesto.

Scenario principale di successo	<ol style="list-style-type: none"> 1. Il cliente arriva e comunica il gioco di cui vuole vedere l'andamento. 2. Il dipendente inserisce il nome del gioco. 3. Il sistema recupera il codice del gioco. 4. Il sistema ricerca per quel codice la presenza di una partita. 5. Il sistema recupera il numero di giocatori, l'orario di inizio e il punteggio attuale. 6. Il dipendente mostra al cliente i dati richiesti.
Estensioni	<p>* In qualsiasi momento, il sistema ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il dipendente riavvia il software e richiede il ripristino. 2. Il sistema restituisce lo stato precedente. <p>3a. Il gioco non risulta presente nel sistema.</p> <ol style="list-style-type: none"> 1. Viene visualizzato un segnale di errore. 2. Il cliente comunica un altro gioco. <p>4a. Non risulta alcuna partita in corso per quel gioco.</p> <ol style="list-style-type: none"> 1. Viene visualizzato un segnale di errore.
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizioni	Ogni qual volta un cliente vuole monitorare lo stato di una partita.

UC5: Crea carta

Nome del caso d'uso	UC5: Crea Carta
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Cliente
Parti interessate e Interessi	<ul style="list-style-type: none"> • Cliente: Vuole creare una carta per poter usufruire dei servizi della sala giochi.
Precondizioni	Devono essere presenti delle carte rilasciabili all'interno del macchinario.
Garanzia di successo	Il Cliente paga, la carta viene generata dal Sistema e consegnata al Cliente, che la prende e va via. Il Sistema salva le informazioni relative alla carta erogata.
Scenario principale di successo	<ol style="list-style-type: none"> 1. Un Cliente vuole creare una carta. 2. Seleziona l'opzione "Crea nuova carta". 3. Il Sistema richiede l'inserimento di un documento d'identità. 4. Il Cliente inserisce il documento d'identità. 5. Il Sistema legge le informazioni del cliente.

	6. Il Sistema chiede al cliente di pagare per la creazione della carta. 7. Il Cliente sceglie il metodo di pagamento. 8. Il Sistema chiede se si vuole la carta VIP o normale. 9. Il Cliente seleziona il tipo di carta desiderata e paga. 10. Il Sistema rilascia da una colonnina al cliente la carta appena creata ed il documento d'identità. 11. Il Cliente prende entrambe e se ne va.
Estensioni	*a. Il Cliente può decidere in qualsiasi momento, prima dell'avvenimento del pagamento, di annullare l'operazione. <ol style="list-style-type: none"> 1. Il Sistema annulla l'operazione e ritorna il documento d'identità (o altre cose che sono state inserite). 2. Il Sistema torna allo stato iniziale. *b. In un qualsiasi momento il Sistema fallisce e si arresta improvvisamente. <ol style="list-style-type: none"> 1. Il dipendente riavvia il software e richiede il ripristino. 2. Il sistema restituisce lo stato precedente. 2a. Non sono presenti carte rilasciabili all'interno del macchinario. <ol style="list-style-type: none"> 1. Il Sistema genera un messaggio per indicare l'impossibilità di generare una nuova carta. 5a. Esiste già una carta associata a quel Cliente. <ol style="list-style-type: none"> 1. Il Sistema torna il documento d'identità al Cliente. 2. Il Sistema torna allo stato iniziale. 10a. Il Sistema controlla il numero di carte rimanenti. <ol style="list-style-type: none"> 1a. Ci sono altre carte, il Sistema non fa nulla. 1b. Non ci sono altre carte, il Sistema segnala che non ci sono più carte e che deve essere riempito.
Requisiti speciali	Il Sistema alla creazione della carta precarica un ammontare di 100 gettoni.
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizione	Ogni qual volta arriva un nuovo Cliente.
Varie	

UC6: Gestisci Bar

Nome del caso d'uso	UC6: Gestisci Bar
----------------------------	--------------------------

Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Dipendente / Amministratore
Parti interessate e Interessi	<ul style="list-style-type: none"> • Dipendente / Amministratore: Vuole aggiungere un nuovo cibo consumabile al bar
Precondizioni	Il cibo deve essere presente in magazzino con adeguate scorte.
Garanzia di successo	Il nuovo cibo è stato inserito all'interno del menù del Bar
Scenario principale di successo	<ol style="list-style-type: none"> 1. L'amministratore (Dipendente) vuole inserire un nuovo cibo consumabile. 2. L'amministratore (Dipendente) sceglie l'attività "Inserisci nuovo cibo". 3. L'amministratore (Dipendente) inserisce il nome, il codice, la quantità disponibile e una breve descrizione. 4. L'amministratore (Dipendente) inserisce il prezzo standard per quella consumazione. <p><i>Il passaggio 3 e 4 viene ripetuto finché serve.</i></p> <ol style="list-style-type: none"> 5. L'amministratore (Dipendente) indica di aver finito. 6. Il sistema restituisce l'elenco dei cibi disponibili.
Estensioni	<p>*a. In un qualsiasi momento il Sistema fallisce e si arresta improvvisamente.</p> <ol style="list-style-type: none"> 1. Il dipendente riavvia il software e richiede il ripristino. 2. Il sistema restituisce lo stato precedente. <p>3a. L'amministratore (Dipendente) inserisce un codice univoco già presente nel sistema.</p> <ol style="list-style-type: none"> 1. Il Sistema genera un messaggio per chiedere se si vuole inserire un altro codice o se si vuole modificare le quantità per il cibo con il codice inserito. <p>2a. L'amministratore (Dipendente) sceglie l'opzione "Inserisci nuovo codice", ripete il passaggio 3 inserendo un codice diverso.</p> <p>2b. L'amministratore (Dipendente) sceglie l'opzione "modifica quantità", si passa al passaggio 4 (non viene creato un nuovo cibo).</p>
Requisiti speciali	I cibi devono avere una quantità disponibile ed un prezzo in gettoni ≥ 1
Elenco delle varianti tecnologiche e dei dati	Nel caso in cui il cibo inserito sia "Acqua naturale", il prezzo in gettoni deve essere 0.
Frequenza di ripetizione	Ogni qual volta il Dipendente vuole aggiornare il catalogo delle consumazioni al bar.

Varie	
-------	--

UC7: Ricarica Carta

Nome del caso d'uso	UC7: Ricarica Carta
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Cliente
Parti interessate e Interessi	<ul style="list-style-type: none"> • Cliente: Vuole ricaricare i gettoni presenti nella carta per poter giocare.
Precondizioni	UC7 deve essere stato eseguito
Garanzia di successo	Il parametro numGettoni della carta viene aggiornato con successo.
Scenario principale di successo	<ol style="list-style-type: none"> 1. Il Cliente comunica i dati della carta al dipendente 2. Il Dipendente recupera la carta del cliente 3. Il Cliente seleziona la funzione di ricarica della carta. 4. Il Cliente comunica l'ammontare dei gettoni che vuole ricaricare. 5. Il Sistema comunica il totale del pagamento. 6. Il Cliente seleziona il tipo di pagamento e paga. 7. Il Sistema aggiorna il parametro numGettoni della carta. 8. Il Sistema rilascia la carta. 9. Il Cliente ritira la carta e se ne va.
Estensioni	<p>*a. Il Cliente può decidere in qualsiasi momento, prima dell'avvenimento del pagamento, di annullare l'operazione.</p> <ol style="list-style-type: none"> 1. Il Sistema annulla l'operazione e ritorna il documento d'identità (o altre cose che sono state inserite). 2. Il Sistema torna allo stato iniziale. <p>*b. In un qualsiasi momento il Sistema fallisce e si arresta improvvisamente.</p> <ol style="list-style-type: none"> 1. Il dipendente riavvia il software e richiede il ripristino. 2. Il sistema restituisce lo stato precedente. <p>2a Il cliente non risulta registrato nel Sistema.</p> <ol style="list-style-type: none"> 1. Vedere caso d'uso UC5.
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	Il massimo numero di gettoni acquistabili giornalmente è 1000.
Frequenza di ripetizione	Ogni qualvolta il cliente vuole ricaricare di gettoni la sua carta per poter giocare

Varie	
-------	--

UC8: Compra Bar

1. Il Cliente crea un ordine in base alla disponibilità del bar.
2. Il Sistema comunica l'importo totale.
3. Il Cliente sceglie il metodo di pagamento e paga.
4. Il Sistema rilascia lo scontrino da mostrare al bar.
5. Il Cliente prende lo scontrino e se ne va.

UC 9: Recupero Carta

Nome del caso d'uso	UC9: Recupero Carta
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Cliente
Parti interessate e Interessi	<ul style="list-style-type: none"> • Cliente: Vuole recuperare il codice della sua carta fedeltà e stamparla.
Precondizioni	Il cliente deve essere già registrato nel sistema
Garanzia di successo	Il cliente recupera la sua carta
Scenario principale di successo	<ol style="list-style-type: none"> 1. Il cliente arriva e chiede il recupero del codice della tessera fedeltà. 2. Il dipendente chiede al cliente il documento di identità 3. Il sistema controlla che il documento inserito risulti registrato. 4. Il dipendente comunica al cliente cosa fare. 5. Il sistema rilascia il documento d'identità. 6. il sistema comunica alla colonnina di stampare una nuova carta con il codice recuperato. 7. La colonnina fornisce la tessera. 8. Il cliente se ne va.
Estensioni	<p>*a. Il Cliente può decidere in qualsiasi momento, prima dell'avvenimento del pagamento, di annullare l'operazione.</p> <ol style="list-style-type: none"> 1. Il Sistema annulla l'operazione e ritorna il documento d'identità (o altre cose che sono state inserite). 2. Il Sistema torna allo stato iniziale. <p>*b. In un qualsiasi momento il Sistema fallisce e si arresta improvvisamente.</p> <ol style="list-style-type: none"> 1. Il dipendente riavvia il software e richiede il ripristino. 2. Il sistema restituisce lo stato precedente. <p>1a. Non sono presenti carte rilasciabili all'interno del macchinario.</p>

	<p>1. Il Sistema genera un messaggio per indicare l'impossibilità di generare una nuova carta.</p> <p>3a. Non è presente il codice fiscale che il cliente ha fornito.</p> <ol style="list-style-type: none"> 1. Il Sistema genera un messaggio di errore. 2. Il dipendente comunica al cliente se vuole registrarsi (<i>vedi UC 5</i>) <p>4a. Il cliente decide di cambiare il proprio codice della tessera.</p> <ol style="list-style-type: none"> 1. Il Sistema genera un nuovo codice 2. Il Sistema aggiorna i dati relativi a quel cliente (si riprende dal passo 5) <p>8a. Il Sistema controlla il numero di carte rimanenti.</p> <ol style="list-style-type: none"> 1a. Ci sono altre carte, il Sistema non fa nulla. 1b. Non ci sono altre carte, il Sistema segnala che non ci sono più carte e che deve essere riempito.
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizione	Ogni qual volta il cliente smarrisce la tessera fedeltà
Varie	

UC 10: Check-In Dipendente

Nome del caso d'uso	UC10: Check-In Dipendente
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Dipendente
Parti interessate e Interessi	<ul style="list-style-type: none"> • Dipendente: Vuole eseguire il chek-in e iniziare il turno di lavoro
Pre-Condizioni	Il dipendente deve essere già registrato nel sistema
Garanzia di successo	Il Dipendente inizia il suo turno lavorativo.
Scenario principale di successo	<ol style="list-style-type: none"> 1. Il Dipendente inserisce le proprie credenziali. 2. Il sistema verifica le credenziali e lo stato. 3. il sistema verifica che il turno del dipendente non sia ancora stato svolto. 4. Il sistema esegue il check-in.

	5. Il dipendente inizia il turno di lavoro.
Estensioni	<p>* In qualsiasi momento, il sistema ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il dipendente riavvia il software e richiede il ripristino. 2. Il sistema restituisce lo stato precedente. <p>2a. Il Sistema non trova le credenziali inserite dal dipendente.</p> <ol style="list-style-type: none"> 1. Il Sistema comunica l'errore. <p>3a. Il Sistema trova il turno del dipendente già svolto o lo stato risulta non idoneo</p> <ol style="list-style-type: none"> 1. Il Sistema comunica un errore.
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	Uno stato del dipendente è "non idoneo" se esso si trova in malattia, ferie o congedo
Frequenza di ripetizioni	Ogni qual volta che un dipendente inizia il proprio turno
Varie	

UC 11: Check-Out Dipendente

Nome del caso d'uso	UC11: Check-Out Dipendente
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Dipendente
Parti interessate e Interessi	<ul style="list-style-type: none"> • Dipendente: Vuole eseguire il check-out e finire il turno di lavoro
Pre-Condizioni	Il dipendente deve essere già registrato nel sistema e avere un turno di lavoro attivo
Garanzia di successo	Il Dipendente finisce il suo turno lavorativo.
Scenario principale di successo	<ol style="list-style-type: none"> 1. Il Dipendente inserisce le proprie credenziali. 2. Il sistema verifica le credenziali e lo stato. 3. il sistema verifica che il turno del dipendente sia attivo e che l'orario sia corretto. 4. Il sistema esegue il check-out. 5. Il dipendente ritorna a casa.
Estensioni	<p>* In qualsiasi momento, il sistema ha un arresto improvviso.</p> <ol style="list-style-type: none"> 3. Il dipendente riavvia il software e richiede il ripristino. 4. Il sistema restituisce lo stato precedente. <p>2a. Il Sistema non trova le credenziali inserite dal dipendente.</p> <ol style="list-style-type: none"> 2. Il Sistema comunica l'errore.

	3a. Il Sistema trova il turno del dipendente ancora in svolgimento. <ol style="list-style-type: none"> Il Sistema segnala che il turno non è ancora finito. Se il dipendente sceglie di finirlo viene effettuato il check-out Il turno viene segnato come “non terminato”
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizioni	Ogni qual volta che un dipendente finisce il proprio turno
Varie	

UC 12: Monitoraggio Dipendenti

Nome del caso d'uso	UC12: Monitoraggio Dipendenti
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Amministratore
Parti interessate e Interessi	<ul style="list-style-type: none"> Amministratore: Vuole visualizzare e monitorare i dipendenti attualmente al lavoro
Pre-Condizioni	La sala giochi deve essere aperta
Garanzia di successo	L'amministratore visualizza i dipendenti a lavoro.
Scenario principale di successo	<ol style="list-style-type: none"> L'amministratore inserisce le proprie credenziali. Il sistema verifica le credenziali. l'amministratore inserisce la data attuale il sistema cerca i dipendenti che hanno un turno attivo. il sistema visualizza le informazioni dei dipendenti assegnati al turno corrente
Estensioni	<p>* In qualsiasi momento, il sistema ha un arresto improvviso.</p> <ol style="list-style-type: none"> si riavvia il software e richiede il ripristino. Il sistema restituisce lo stato precedente. <p>2a. Il Sistema non trova le credenziali inserite dall'amministratore.</p> <ol style="list-style-type: none"> Il Sistema comunica l'errore.
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	

Frequenza di ripetizioni	Ogni qual volta si vuole verificare che un dipendente sia svolgendo il turno
Varie	

UC 13: Record Personale

Nome del caso d'uso	UC13: Record Personale
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Dipendente
Parti interessate e Interessi	<ul style="list-style-type: none"> Dipendente: Vuole visualizzare i record personali richiesti dal cliente per ogni gioco
Pre-Condizioni	Il cliente deve aver giocato ad almeno 1 gioco
Garanzia di successo	Vengono visualizzati i record.
Scenario principale di successo	<ol style="list-style-type: none"> Il cliente inserisce la propria carta. Il sistema verifica le credenziali e recupera i giochi. il sistema ordina i giochi in base al punteggio. il sistema cerca i punteggi relativi ai giochi. il sistema visualizza le informazioni
Estensioni	<p>* In qualsiasi momento, il sistema ha un arresto improvviso.</p> <ol style="list-style-type: none"> si riavvia il software e richiede il ripristino. Il sistema restituisce lo stato precedente.
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizioni	Ogni qual volta si vuole verificare il punteggio di un cliente
Varie	

UC 14: Visualizza Classifica Generale

Nome del caso d'uso	UC14: Classifica Generale
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Dipendente
Parti interessate e Interessi	<ul style="list-style-type: none"> Dipendente: Vuole visualizzare i primi 10 record per ogni gioco

Pre-Condizioni	Il gioco deve essere o essere stato in servizio
Garanzia di successo	Vengono visualizzati i record.
Scenario principale di successo	<ol style="list-style-type: none"> 1. il dipendente inserisce il codice del gioco 2. il sistema recupera le informazioni 3. il sistema ordina in base al punteggio 4. il sistema visualizza le informazioni
Estensioni	<p>* In qualsiasi momento, il sistema ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. si riavvia il software e richiede il ripristino. 2. Il sistema restituisce lo stato precedente.
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizioni	Ogni qual volta si vuole verificare il punteggio di un gioco
Varie	

UC15: Aggiungi Gioco

Nome del caso d'uso	UC15: Aggiungi Gioco
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Amministratore
Parti interessate e Interessi	<ul style="list-style-type: none"> • Amministratore: vuole aggiornare il catalogo dei giochi disponibili in sala.
Pre-Condizioni	La sala giochi ha acquistato un nuovo macchinario\tavolo.
Garanzia di successo	Le informazioni relative al premio sono inserite con successo nel Sistema.
Scenario principale di successo	<ol style="list-style-type: none"> 1. L'amministratore vuole aggiungere un gioco disponibile all'interno della sala giochi. 2. L'amministratore sceglie l'attività "Inserisci nuovo gioco". 3. L'amministratore inserisce il nome, il codice, la tipologia di gioco e il numero di giocatori. 4. L'amministratore inserisce per quel gioco il costo in gettoni. <i>Il passaggio 3 e 4 viene ripetuto finché serve.</i> 5. L'amministratore indica di aver finito.
Estensioni	<p>*a. In un qualsiasi momento il Sistema fallisce e si arresta improvvisamente.</p> <ol style="list-style-type: none"> 1. L'amministratore riavvia il software e ripristina lo stato

	<p>precedente del Sistema.</p> <p>2. Il Sistema ripristina lo stato.</p> <p>3a. L'amministratore inserisce un codice univoco già presente nel sistema.</p> <p>1. Il Sistema genera un messaggio per chiedere se si vuole inserire un altro codice.</p> <p>2. L'amministratore ripete il passaggio 3 inserendo un codice diverso.</p> <p>4a. L'amministratore inserisce costo non valido.</p> <p>1. Il Sistema genera un messaggio di errore.</p>
Requisiti speciali	Il costo in gettoni va da un minimo di 1 ad un massimo di 30.
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizioni	Ogni qual volta la sala giochi intende mettere disponibile un nuovo gioco.
Varie	

UC16: Gestione Macchina

Nome del caso d'uso	UC16: Gestione Macchina
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Dipendente
Parti interessate e Interessi	<ul style="list-style-type: none"> Dipendente: vuole verificare lo stato delle macchine/giochi.
Pre-Condizioni	I giochi sono inseriti nel sistema.
Garanzia di successo	Il dipendente visualizza lo stato del gioco richiesto.
Scenario principale di successo	<ol style="list-style-type: none"> Il Dipendente inserisce il nome del gioco della quale vuole controllare lo stato. Il Sistema recupera il codice del gioco. Il Sistema recupera lo stato del gioco. Il Sistema comunica lo stato del gioco desiderato.
Estensioni	<p>* In qualsiasi momento, il sistema ha un arresto improvviso.</p> <ol style="list-style-type: none"> Il dipendente riavvia il software e richiede il ripristino. Il sistema restituisce lo stato precedente. <p>3a. Lo stato del gioco è on.</p>

	1. Il Sistema controlla se per il gioco è presente una partita in corso.
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizioni	Ogni qual volta si vuole controllare lo stato delle macchine/giochi.

UC 17: Avvia Partita

Nome del caso d'uso	UC17: Avvia Partita
Portata	Software Il Piattino d'Oro
Livello	Obiettivo utente
Attore primario	Cliente
Parti interessate e Interessi	<ul style="list-style-type: none"> • Cliente: vuole iniziare una partita ad un gioco.
Pre-Condizioni	Il gioco è libero e funzionante.
Garanzia di successo	Il cliente può effettuare una partita al gioco da lui selezionato.
Scenario principale di successo	<ol style="list-style-type: none"> 1. Il Cliente arriva al gioco e vuole avviare una partita. 2. Il Cliente inserisce la carta per avviare la partita. 3. Il Sistema acquisisce il codice della carta e del gioco. 4. Il Sistema controlla se sono presenti abbastanza gettoni per avviare la partita del gioco. 5. Il Sistema scala i gettoni dalla carta. 6. Il Sistema associa la carta, il gioco e l'orario, creando una partita. 7. Il Sistema, alla fine della partita, carica il punteggio fatto durante la partita. 8. Il Sistema chiede se si desidera continuare a giocare. (i passi 4, 5, 6, 7 e 8 si ripetono se necessari) 9. Il Cliente segnala di aver finito di giocare. 10. Il Sistema restituisce la carta. 11. Il Cliente prende la carta e se ne va.
Estensioni	<p>* In qualsiasi momento, il sistema ha un arresto improvviso.</p> <ol style="list-style-type: none"> 1. Il dipendente riavvia il software e richiede il ripristino. 2. Il sistema restituisce lo stato precedente. <p>4a. Nella carta non sono presenti abbastanza gettoni per</p>

	<p>giocare.</p> <p>1. Il Sistema comunica l'insufficienza dei gettoni e va al passo 10.</p> <p>8a. Il Sistema fa il passo 4 prima di chiedere se si vuole continuare e non sono presenti abbastanza gettoni.</p> <p>1. Il Sistema non dà la possibilità di continuare.</p> <p>8b. Il Sistema fa il passo 4 prima di chiedere se si vuole continuare e sono presenti abbastanza gettoni.</p> <p>1. Il Sistema dà la possibilità di continuare.</p>
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizioni	Ogni qual volta che un cliente decide di prenotare un gioco.
Varie	La partita ha una durata (media) fissa di 30 minuti per gioco.

◆ Regole di business

ID	Regola	Modificabilità	Sorgente
R1	Se il cliente rimane nelle prime 5 posizioni nella classifica di un gioco per più di 7 giorni lavorativi, riceve uno sconto del 15% sulle ordinazioni al bar.	Media	Politica interna
R2	Se un cliente sceglie la carta fedeltà VIP, al prezzo base della carta si aggiungono 15€.	Media	Politica interna
R3	Il giovedì si paga tutto il 10% in meno. Sabato e domenica si paga tutto il 10% in più. Dalle 8 alle 14 si paga il 10% in più. Dalle 18 alle 2 si paga il 10% in meno.	Alta	Politica interna
R4	Il Cliente può decidere di pagare usando i punti accumulati. 1€ = 50 punti.	Bassa	Politica interna
R5	I punti accumulati dal cliente dipendono dal punteggio totale ottenuto e dal gioco stesso.	Media	Politica interna
R6	I giochi hanno un costo di avvio in gettoni che va da un minimo di 1 ad un massimo di 30.	Media	Politica interna

Delle regole di business ivi esposte le regole R2, R5 ed R6 sono state implementate nell'applicazione.

◆ Glossario

Vengono qui riportati i termini più significativi e le relative definizioni.

- **Carta Fedeltà:** termine riferito ad una carta utilizzata nella sala giochi per tenere traccia dei punti accumulati e dei gettoni.
- **Punto:** valuta ottenuta da un cliente dopo la sua partecipazione ad un gioco, l'ammontare dipende dalla prestazione del cliente al suddetto gioco
- **Gettone:** valuta di scambio valida solo all'interno della sala giochi che permette di accedere ad un gioco, riscattare un premio o ordinare al bar; tutte le attività hanno un costo in termini di gettoni differenti.
- **Tavolo:** Struttura con il quale il cliente può giocare ad un gioco non elettronico, quale biliardo o calcetto.
- **Cabinato:** Struttura con il quale il cliente può interagire per poter partecipare ad un gioco elettronico.
- **Gioco:** attività principale che si svolge all'interno della struttura, spaziando da quelli arcade, elettronici o tradizionali.

Analisi Orientata agli Oggetti

Seguendo l'approccio iterativo, la realizzazione dell'applicazione è stata articolata su 5 iterazioni, nelle quali sono state realizzate iterativamente le funzioni lato software, risolte le eventuali problematiche riscontrate, un'analisi dei requisiti graduale e ulteriori modifiche e migliorie, man mano che si è proseguito nell'implementazione, particolarmente relative al carico di lavoro svolto dalla classe di sistema "Il Piattino d'Oro".

Successivamente, e per ogni iterazione, si è sviluppata una classe *Test* ove verificare, mediante, per l'appunto, dei test, le funzionalità di ogni operazione e di ogni caso d'uso analizzato. Per ciascuna iterazione sono state affrontate, nel particolare, le seguenti problematiche:

1. Iterazione 1

- Implementazione del caso d'uso di Start Up per l'inizializzazione del sistema.
- Implementazione del caso d'uso *UC1: Gestisci Premi* nel suo scenario di successo, rimandando poi l'implementazione delle sue estensioni in un secondo momento.
- Implementazione del caso d'uso *UC16: Aggiungi Gioco* sempre nel suo scenario di successo, per avere disponibili già in prima istanza le funzioni base del programma.

2. Iterazione 2

- Implementazione del caso d'uso *UC5: Crea Carta* in prima istanza nel suo scenario di successo, per poi passare all'implementazione delle sue estensioni

in un secondo momento.

- Implementazione del caso d'uso *UC6: Gestisci Bar* nel suo scenario di successo.
- Implementazione degli scenari alternativi del caso d'uso *UC1: Gestisci Premi*.
- Implementazione degli scenari alternativi del caso d'uso *UC16: Aggiungi Gioco*.

3. Iterazione 3

- Implementazione del caso d'uso *UC2: Gestisci Prenotazione* sia nel suo scenario di successo, che nelle sue ulteriori estensioni.
- Implementazione del caso d'uso *UC9: Recupera Carta* sia nel suo scenario di successo, che nella sua estensione.
- Implementazione in sequenza del caso d'uso *UC7: Ricarica Carta* nel suo scenario di successo.

4. Iterazione 4

- Implementazione del caso d'uso *UC18: Avvia Partita* sia nel suo scenario di successo, che nelle sue ulteriori estensioni.
- Implementazione dei casi d'uso *UC4: Monitora Partita* e *UC17: Gestione Macchina* nel suo scenario di successo.
- Implementazione del caso d'uso *UC3: Riscatta Premi* sia nel suo scenario di successo, che nelle sue ulteriori estensioni 4a e 5a.

5. Iterazione 5

- Implementazione dei casi d'uso *UC10: Check-in Dipendenti* e *UC11: Check-out Dipendenti*, entrambi nel loro scenario di successo.
- Implementazione del caso d'uso *UC12: Monitora Dipendenti*, sempre nel suo scenario di successo.
- Implementazione dei casi d'uso *UC13: Record Personali* e *UC14: Visualizza Classifica Generale*.

Nell'effettuare l'analisi dei requisiti Orientata agli Oggetti, sono stati utilizzati e qui successivamente elencati diversi strumenti: Modello di Dominio, SSD (Sequence System Diagram) e Contratti delle operazioni.

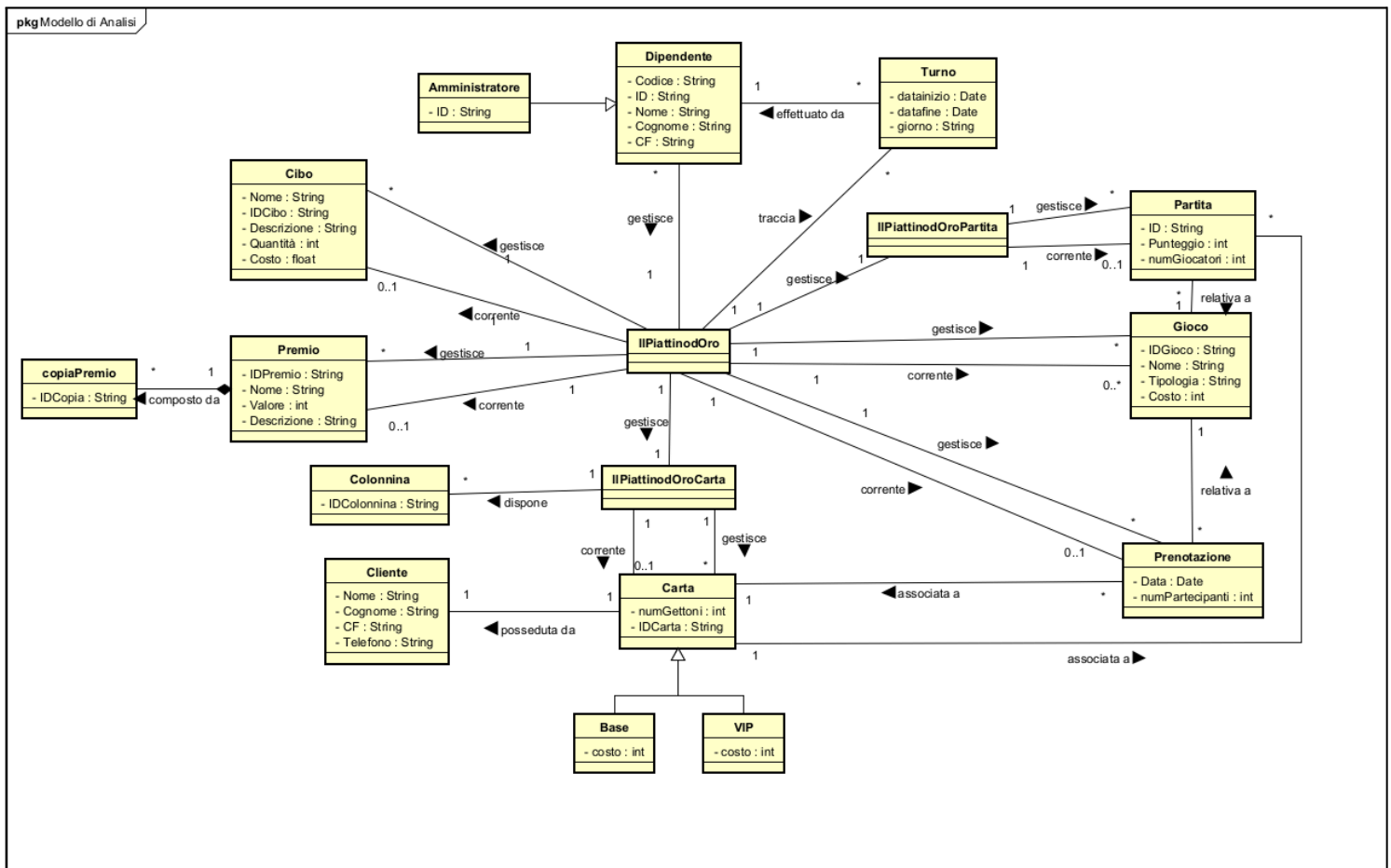
◆ Modello di Dominio

Nel Modello di Dominio vengono identificati i concetti, gli attributi e le associazioni più significativi, in particolare mettendo in risalto le classi concettuali. Tenendo conto di tutte le iterazioni, alla fine il modello ricavato e le relative classi sono le seguenti:

- ***IIPIattinodOro***: Classe che rappresenta il sistema stesso, il quale si occupa della maggior parte delle funzioni; ad eccezione delle funzioni relative alla Carta e alla Partita, gestite, rispettivamente, da ***IIPIattinodOroCarta*** e ***IIPIattinodOroPartita***.
- ***Dipendente***: Attore primario che utilizza il sistema.
- ***Amministratore***: Ulteriore attore primario che utilizza il sistema. Si tratta di un dipendente speciale che ha i compiti amministrativi e che può eseguire, al contrario degli altri, il

monitoraggio.

- **Gioco:** Attività principale della sala giochi. Se sono in stato operativo, giocando ad uno di essi si guadagnano punti per ottenere premi.
- **Prenotazione:** Entità con la quale si identifica un'associazione tra cliente e gioco in una determinata ora.
- **Partita:** Attività principale che avviene con un gioco. è dettata da alcune regole specifiche per ogni gioco e ha una durata limitata. Alla fine di ogni partita vengono assegnati dei punteggi al giocatore/cliente.
- **Premio:** Oggetto che un cliente può riscattare spendendo dei punti, ogni premio ha delle copie di esso disponibili.
- **Cibo:** Alimento o pietanza che viene offerta al bar interno della sala giochi.
- **Turno:** Classe che rappresenta il turno stesso, associando il dipendente ad orari e giorni della settimana specifici. Ad essi si è aggiunto un parametro Stato per indicarne la disponibilità o meno del dipendente ad eseguirlo.
- **Cliente:** Persona che vuole registrarsi all'interno della sala giochi per usufruire dei suoi servizi. Presenta numerosi parametri identificativi, tra cui la Carta.
- **Carta:** Oggetto usato nella sala giochi per tener traccia dei punteggi registrati a fine partita e dei gettoni di un cliente registrato.
- **Colonnina:** Macchinario adibito alla stampa e al monitoraggio delle carte fedeltà in bianco.



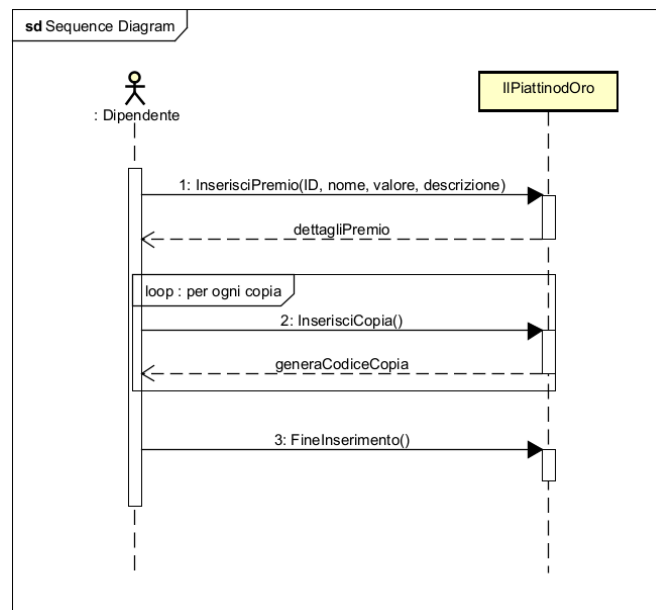
◆ SSD e Contratti

Nell'analisi Orientata agli Oggetti, è presente anche la creazione dei Diagramma di Sequenza di Sistema (SSD) per illustrare il corso degli eventi di input/output per i vari casi d'uso esaminati nelle varie iterazioni. Per le principali operazioni individuate negli SSD verranno descritti anche i relativi Contratti delle operazioni.

Di seguito sono elencati, per ogni iterazione un SSD esemplificativo relativo alle operazioni analizzate; in particolare per l'iterazione 1 l' *UC1: Gestisci Premi*; per l'iterazione 2 l' *UC5: Crea Carta*; per l'iterazione 3 l' *UC2: Gestisci Prenotazione*; per l'iterazione 4 l' *UC18: Avvia Partita* e per l'iterazione 5 l' *UC12: Monitora Dipendenti*.

1. Iterazione 1 *UC1: Gestisci Premi*

SSD



Contratti

Contratto CO 1: InserisciPremio

Operazione: InserisciPremio(ID: String, nome: String, descrizione: String, valore: int)

Riferimenti: caso d'uso UC1

Precondizioni: -

PostCondizioni: è stata creata un'istanza *p* di Premio
gli attributi di *p* sono stati inizializzati
p è stato associato al sistema tramite associazione "corrente"

Contratto CO 2: InserisciCopia

Operazione: InserisciCopia()

Riferimenti: caso d'uso UC1

Precondizioni: è in corso la definizione di *p*

PostCondizioni: è stata creata un'istanza *pc* di PremioCopia
gli attributi di *pc* sono stati inizializzati
pc è stato associato al sistema tramite associazione "prevede"

Contratto CO 3: FineInserimento

Operazione: FineInserimento()

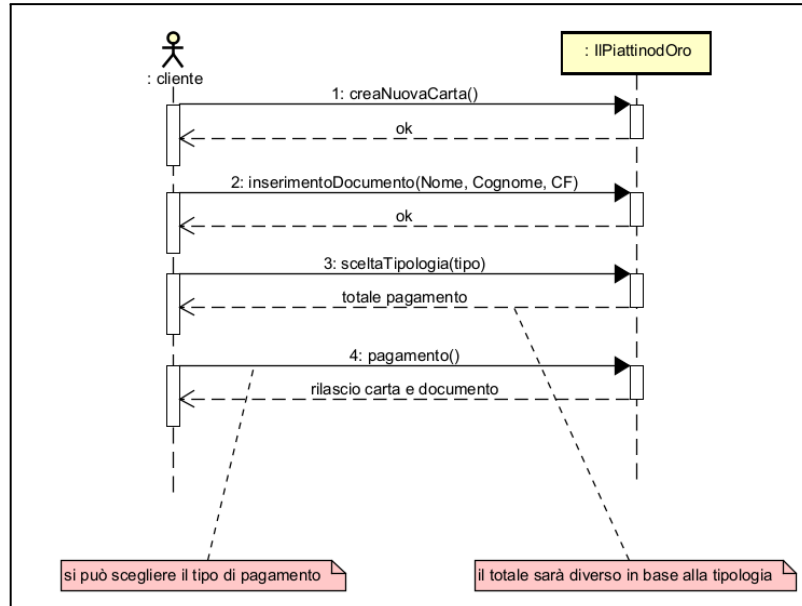
Riferimenti: caso d'uso UC1

Precondizioni: è in corso la definizione di *p*

PostCondizioni: *p* è stato associato al sistema tramite associazione “*gestisce*”

2. Iterazione 2 UC5: Crea Carta

SSD



Contratti

Contratto CO 1: InserisciPremio

Operazione: InserisciPremio(ID: String, nome: String, descrizione: String, valore: int)

Riferimenti: caso d'uso UC1

Precondizioni: -

PostCondizioni: è stata creata un'istanza *p* di Premio

gli attributi di *p* sono stati inizializzati

p è stato associato al sistema tramite associazione “*corrente*”

Contratto CO 2: InserisciCopia

Operazione: InserisciCopia()

Riferimenti: caso d'uso UC1

Precondizioni: è in corso la definizione di *p*

PostCondizioni: è stata creata un'istanza *pc* di PremioCopia

gli attributi di *pc* sono stati inizializzati

pc è stato associato al sistema tramite associazione “*prevede*”

Contratto CO 3: FineInserimento

Operazione: FineInserimento()

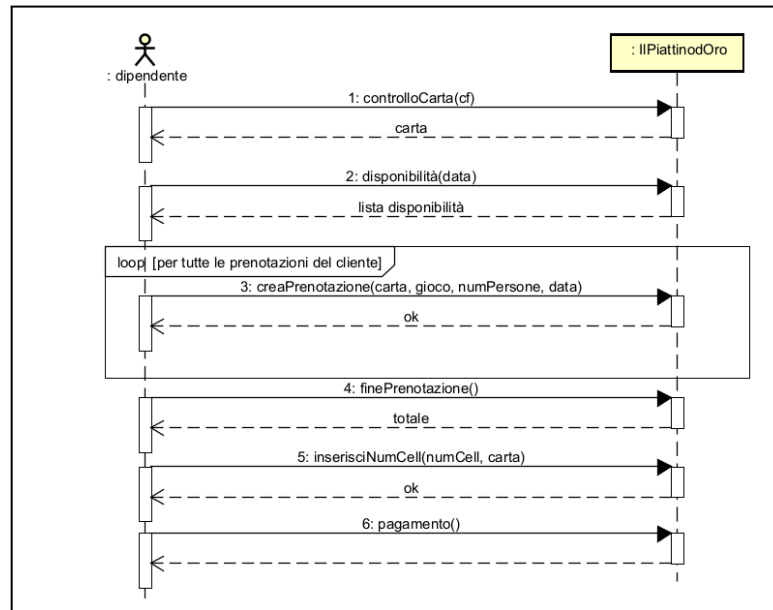
Riferimenti: caso d'uso UC1

Precondizioni: è in corso la definizione di *p*

PostCondizioni: p è stato associato al sistema tramite associazione “*gestisce*”

3. Iterazione 3 UC2: *Gestisci Prenotazione*

SSD



Contratti

Contratto CO 1: InserisciPremio

Operazione: InserisciPremio(ID: String, nome: String, descrizione: String, valore: int)

Riferimenti: caso d'uso UC1

Precondizioni: -

PostCondizioni:   stata creata un'istanza p di Premio
gli attributi di p sono stati inizializzati
 p   stato associato al sistema tramite associazione “*corrente*”

Contratto CO 2: InserisciCopia

Operazione: InserisciCopia()

Riferimenti: caso d'uso UC1

Precondizioni:   in corso la definizione di p

PostCondizioni:   stata creata un'istanza pc di PremioCopia
gli attributi di pc sono stati inizializzati
 pc   stato associato al sistema tramite associazione “*prevede*”

Contratto CO 3: FineInserimento

Operazione: FineInserimento()

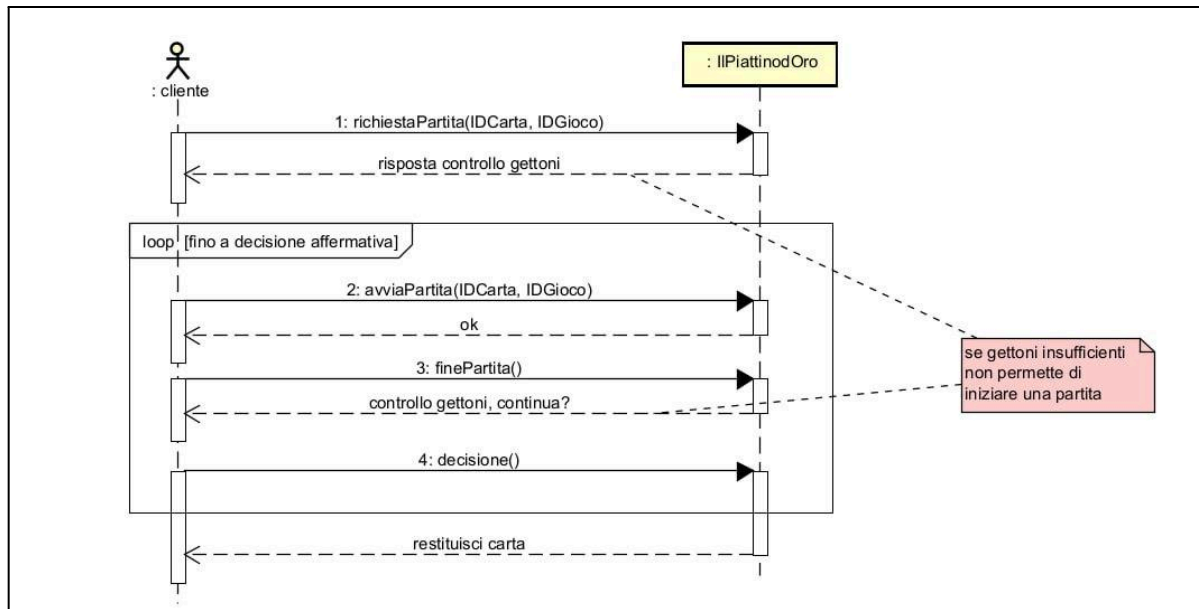
Riferimenti: caso d'uso UC1

Precondizioni:   in corso la definizione di p

PostCondizioni: p   stato associato al sistema tramite associazione “*gestisce*”

4. Iterazione 4 UC18: *Avvia Partita*

SSD



Contratti

Contratto CO 1: InserisciPremio

Operazione: InserisciPremio(ID: String, nome: String, descrizione: String, valore: int)
 Riferimenti: caso d'uso UC1
 Precondizioni: -
 PostCondizioni: è stata creata un'istanza *p* di Premio
 gli attributi di *p* sono stati inizializzati
p è stato associato al sistema tramite associazione "corrente"

Contratto CO 2: InserisciCopia

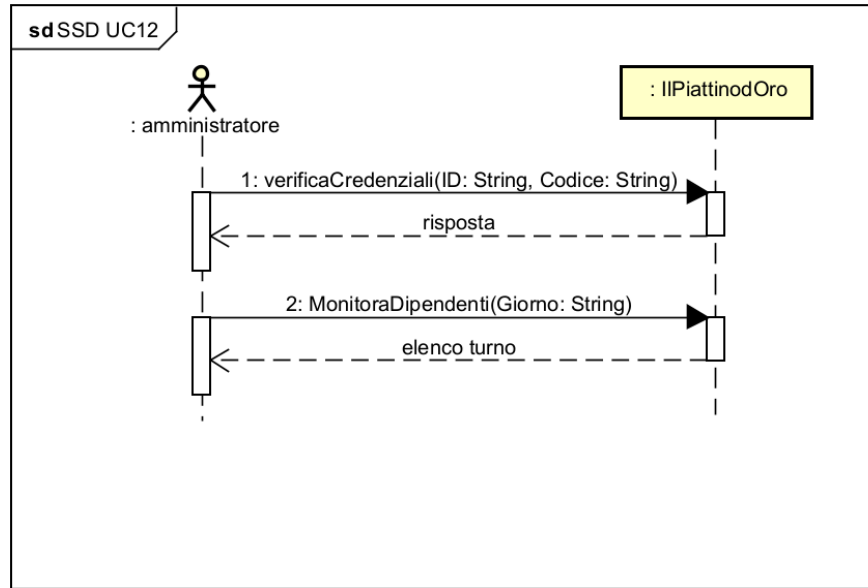
Operazione: InserisciCopia()
 Riferimenti: caso d'uso UC1
 Precondizioni: è in corso la definizione di *p*
 PostCondizioni: è stata creata un'istanza *pc* di PremioCopia
 gli attributi di *pc* sono stati inizializzati
pc è stato associato al sistema tramite associazione "prevede"

Contratto CO 3: FineInserimento

Operazione: FineInserimento()
 Riferimenti: caso d'uso UC1
 Precondizioni: è in corso la definizione di *p*
 PostCondizioni: *p* è stato associato al sistema tramite associazione "gestisce"

5. Iterazione 5 UC12: Monitora Dipendenti

SSD



Contratti

Contratto CO 1: InserisciPremio

Operazione: InserisciPremio(ID: String, nome: String, descrizione: String, valore: int)
 Riferimenti: caso d'uso UC1
 Precondizioni: -
 PostCondizioni: è stata creata un'istanza *p* di Premio
 gli attributi di *p* sono stati inizializzati
p è stato associato al sistema tramite associazione "corrente"

Contratto CO 2: InserisciCopia

Operazione: InserisciCopia()
 Riferimenti: caso d'uso UC1
 Precondizioni: è in corso la definizione di *p*
 PostCondizioni: è stata creata un'istanza *pc* di PremioCopia
 gli attributi di *pc* sono stati inizializzati
pc è stato associato al sistema tramite associazione "prevede"

Contratto CO 3: FineInserimento

Operazione: FineInserimento()
 Riferimenti: caso d'uso UC1
 Precondizioni: è in corso la definizione di *p*
 PostCondizioni: *p* è stato associato al sistema tramite associazione "gestisce"

Progettazione

Nella progettazione sono stati definiti gli oggetti software, le loro responsabilità e come questi interagiscono tra loro e soddisfano i requisiti individuati. A corredo vi è il Modello di Progetto, costituito dai diagrammi che ne descrivono la progettazione logica sia da un punto di vista

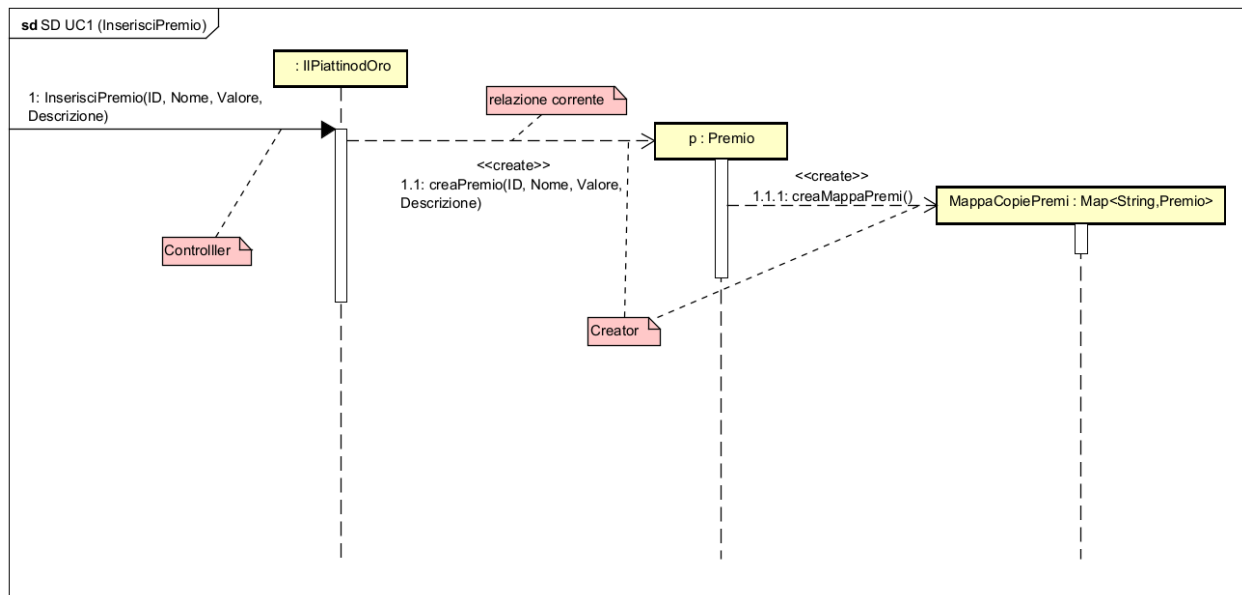
statico (Diagramma delle Classi), che da un punto di vista dinamico (Diagrammi di Interazione).

◆ Diagrammi di Sequenza

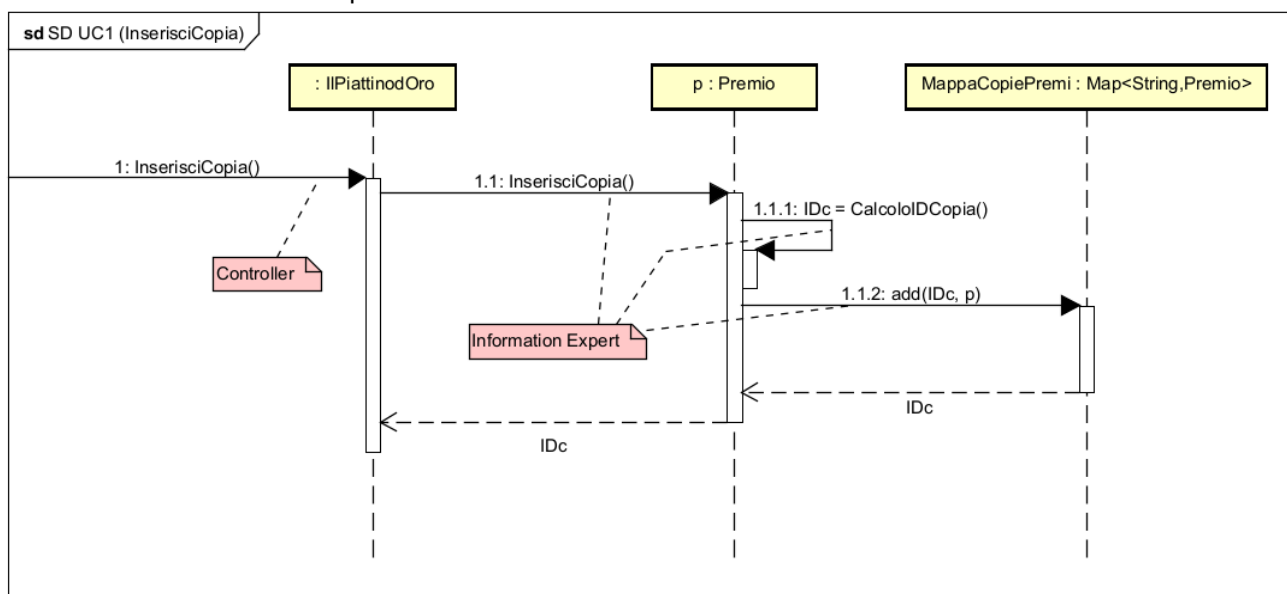
I diagrammi di sequenza esposti in seguito fanno riferimento, per evitare di appesantire il carico informativo, solo ai casi d'uso *UC1: Gestisci Premi* e *UC18: Avvia Partita* come casi emblematici. Per una trattazione completa visionare i documenti relativi alle varie iterazioni.

◆ DS UC1

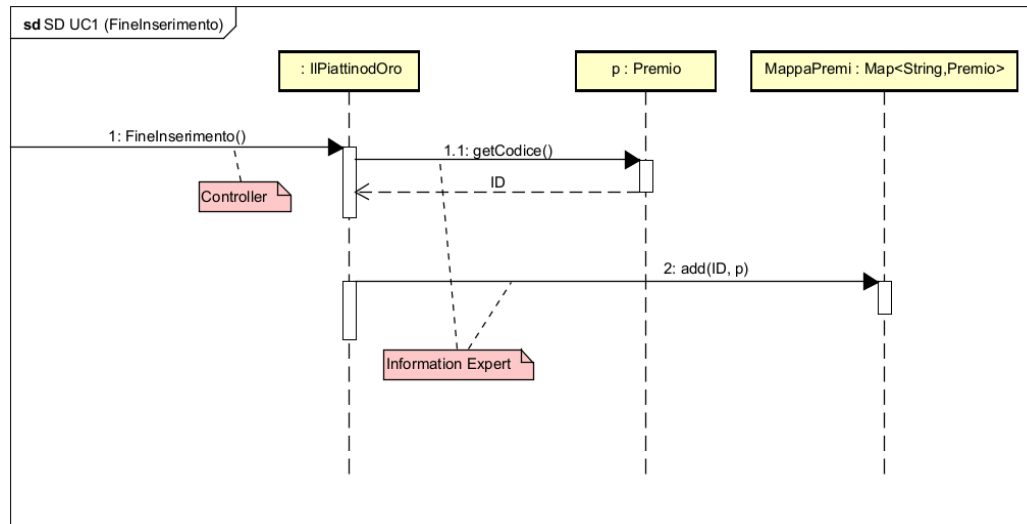
- Inserimento nuovo premio



- Inserimento di una copia

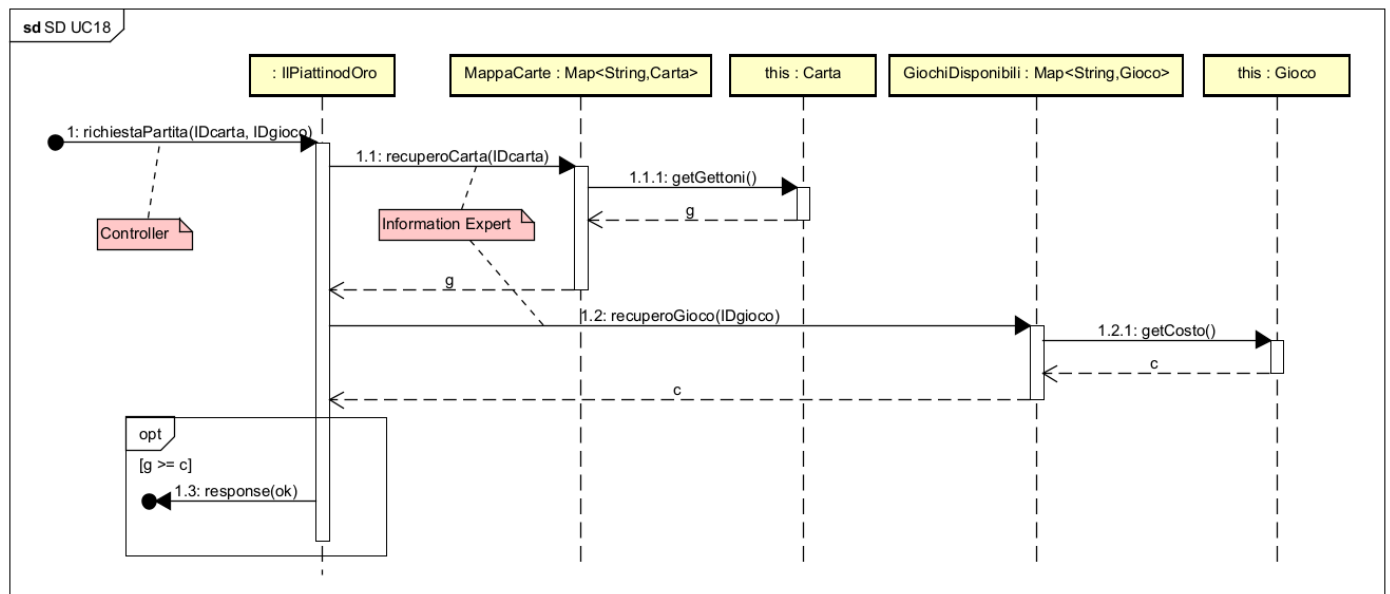


- Fine inserimento

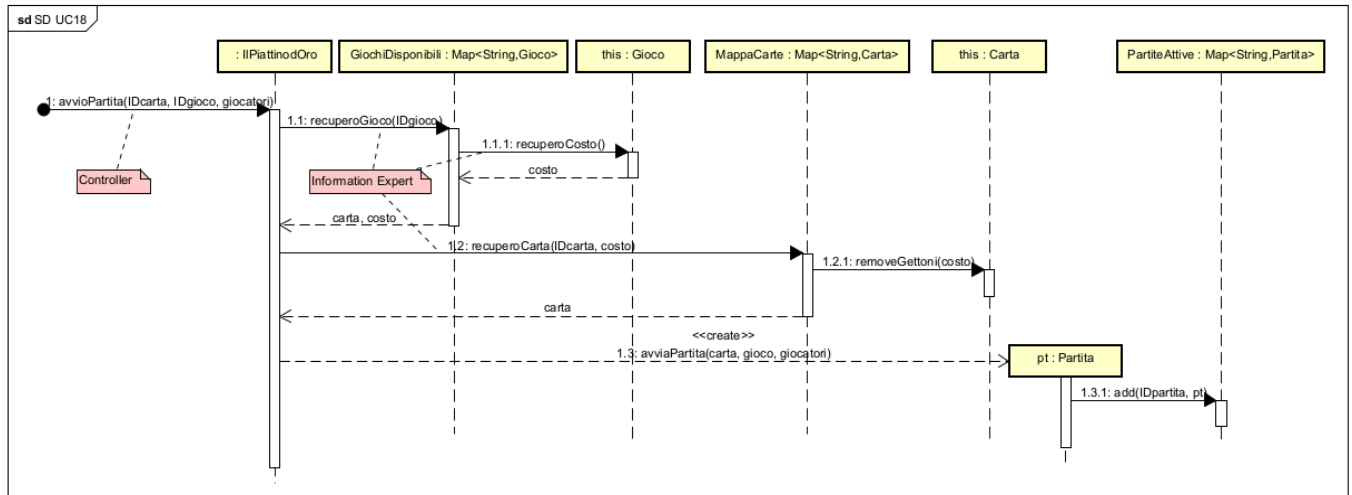


◆ DS UC18

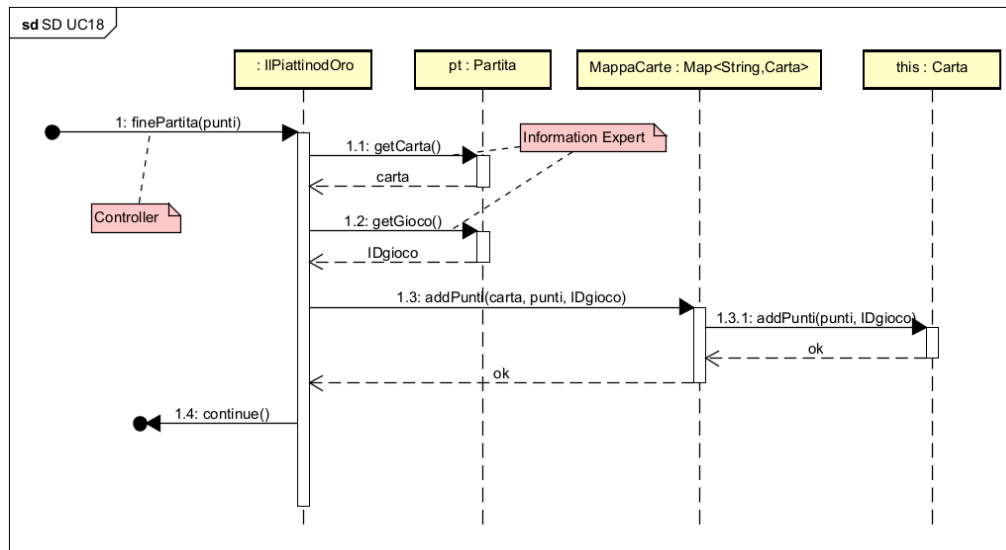
- Controllo della carta e dei gettoni



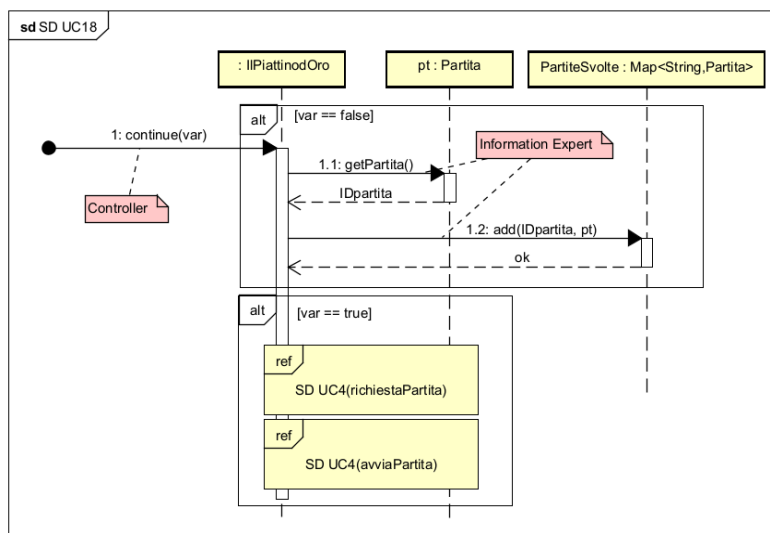
- Avvio della partita e scalaggio gettoni

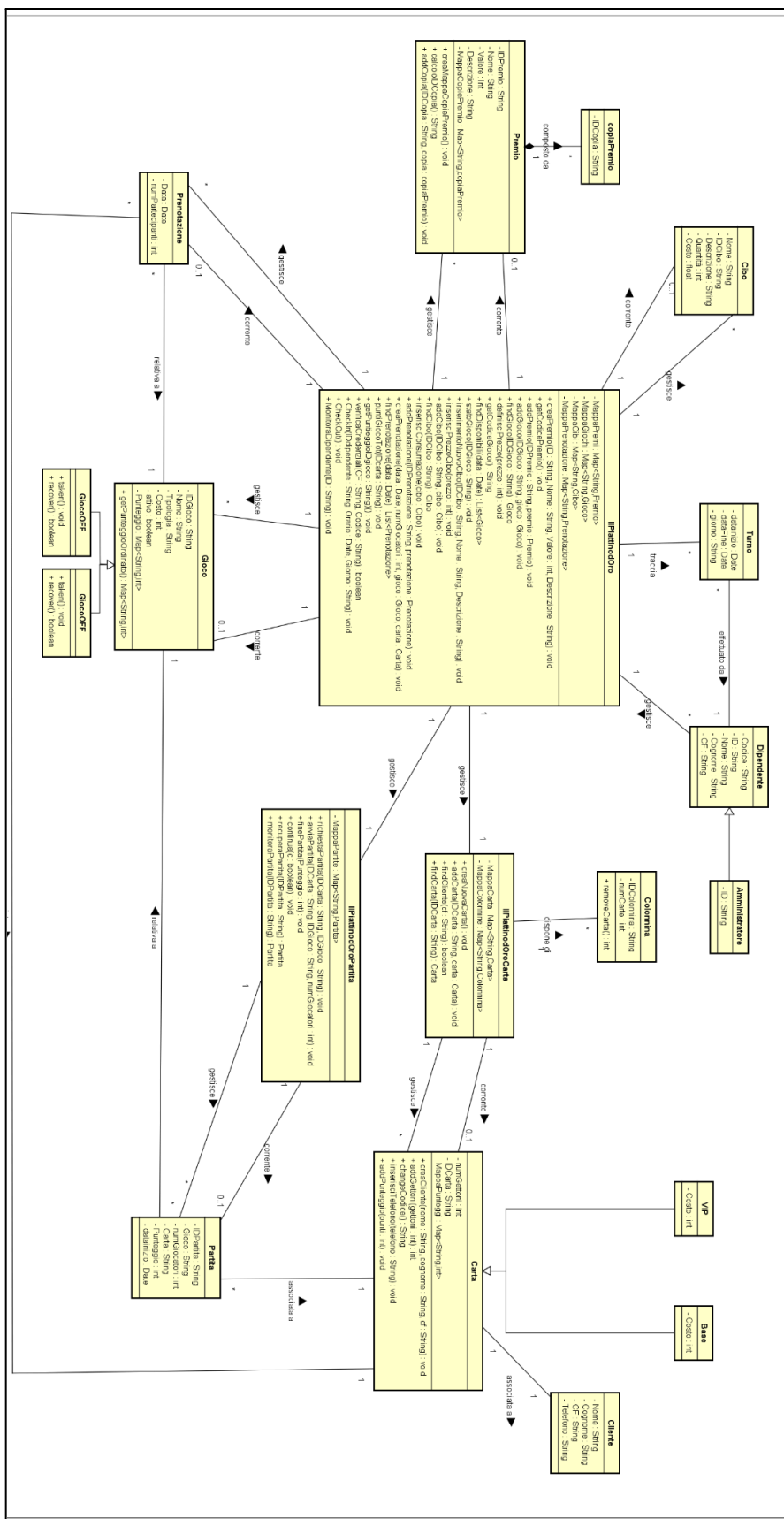


- Fine partita e salvataggio punteggio



- Scelta di continuare a giocare o finire





◆ Testing ed Approfondimento Software

Durante il processo di sviluppo, per poter rilevare dei malfunzionamenti o in generale se il codice che era stato prodotto risultasse valido e verificato, sono stati realizzati dei **test** su di esso. In una fase preliminare, è stata dapprima eseguita una ispezione del codice sorgente al fine di scegliere le classi e i metodi da testare.

Si è data priorità in primo luogo a dei test unitari sui vari metodi relativi alla classe gioco, in particolare al caso d'uso UC51 Inserisci Nuovo Gioco. I test, eseguiti sfruttando JUnit, sono stati condotti sui metodi di inserimento, dettaglio prezzo e conferma, ognuno distinto e con l'accortezza di verificare mediante `assertNotNull()` ed `assertEquals()` se le specifiche venissero rispettate.

In seguito sono stati condotti test analoghi per quanto riguardava altri casi d'uso e le altre classi principali del sistema come la Carta, i Cibi e i Premi.

Per quanto riguardato le funzionalità più complesse come il monitoraggio della partita o l'eseguire una prenotazione sono stati condotti delle prove più complesse facendo eseguire per ogni test una serie di step atte a imitare il funzionamento del caso d'uso e successivamente verificato anche mediante i risultati mostrati in riga di comando.

Tali prove sono state poi eseguite, una volta che veniva realizzata la parte dell'applicazione, lato grafico, ad essa associata e per tutte le funzionalità che essa è in grado di eseguire.

Nella creazione del software, per migliorare la qualità, sono stati inoltre implementati all'interno del codice alcuni design pattern, ascrivibili ai cosiddetti Pattern GOF. Di seguito una loro descrizione e relativo utilizzo.

- **Singleton:** Assicura che una classe abbia solo un'istanza ed un unico punto d'accesso mediante un singolo costruttore privato. Ciò è stato usato nella classe `IlPiatinoDOro`, essendo la classe principale del sistema.
- **Prototype** Creare un'istanza da una classe presa come modello. Viene usato nella classe `Premio` per istanziare numerose copie mediante `java.Clonable`. In tal modo possono esserci più copie dello stesso premio in inventario, con ognuno i propri attributi e senza utilizzare un attributo `num.Copie`, invece si adopera una mappa che ne tiene conto. Analogo ragionamento per il Cibo.
- **State:** Lo stato è implementato nella classe `Gioco`, mediante `GiocoOn` e `GiocoOff` che ne rappresentano i due stati. In base allo stato se si cerca tale gioco come disponibile verrà visualizzata o no la sua effettiva disponibilità, utile soprattutto in caso di prenotazione.
- **Command:** Per incapsulare una richiesta da parte di un invocatore e permettere ai ricevitori di eseguirla. Nella sezione relativa all'applicazione grafica, esso è stato usato per semplicità, data la complessità dell'applicativo, soltanto nei pulsanti relativi all'elencare cibi, giochi, premi e carte.

◆ Refactoring

In ultima istanza si è proceduto ad effettuare, specialmente nelle classi più importanti, un rapido refactoring. La pratica di riscrittura del codice, senza modificarne il comportamento e attuata allo scopo di semplificare e rendere più chiara la sua comprensione, è stata applicata, durante la scrittura del codice, in genere alla fine di un'iterazione, o al suo inizio come segnalato nei vari

documenti ad esse relative.

Tra le migliorie più evidenti vi è stata, come precedentemente documentata, la separazione delle funzioni relative all'uso della Carta e della Partita in due classi distinte, rispettivamente `IIPiattinoDOroCarta` e `IIPiattinoDOroPartita`. Questo è stato fatto per non sovraccaricare la classe principale `IIPiattinoDOro` di troppe operazioni. Queste considerazioni sono state fatte rispettivamente in iterazione 4 e 5.

Un'altra miglioria è stata fatta nella classe `GUI`, la quale si occupa di visualizzare graficamente il programma e permette all'utente, nel caso in questione il Dipendente, di interagire. Ad inizio implementazione le varie funzionalità erano scritte tutte all'interno del `main`, facendo apparire il codice molto lungo e complesso, oltre che difficile da navigare durante la fase di testing. Per questo si è deciso di spostare tutti i comportamenti relativi alle varie "aree" di competenza, quali la visualizzazione e inserimento, in altre funzioni separate e richiamate dal `main`, lasciando in quest'ultimo solo la gestione dei vari pulsanti per l'attivazione delle determinate "aree".

Oltre ad esse è stata migliorata, in vari momenti dello sviluppo, la leggibilità del codice ad esempio per quanto riguarda l'individuazione dei parametri con nomi identificativi adeguati e altro.