

02 – Elaborazione - Iterazione 1

1 Introduzione

Una volta conclusa la fase di ideazione si passa alla fase di elaborazione. Viene quindi iniziata la serie di iterazioni che portano alla definizione finale del progetto.

Lo scopo delle iterazioni sarà quello di programmare e verificare l'architettura software principale, scoprire e stabilizzare la maggior parte dei requisiti, e attenuare o eliminare i rischi maggiori.

Durante la prima iterazione i requisiti su cui mi sono concentrato sono stati:

- Implementazione dello scenario principale di successo del caso d'uso *UC1: Inizio nuova partita solitario*, in modo da permettere al giocatore di poter iniziare una nuova partita del gioco Solitario.

2 Presentazione del caso d'uso UC1

Andiamo nuovamente a presentare lo schema del caso d'uso *UC1: Inizio nuova partita solitario*.

UC1: Inizio nuova partita solitario

Nome del caso d'uso	UC1: Inizio nuova partita solitario
Portata	Applicazione gioco Solitario
Livello	obiettivo utente
Attore primario	Giocatore del Sistema
Parti interessate e Interessi	<ul style="list-style-type: none">▪ Giocatore: vuole giocare al gioco del Solitario, nel modo più fluido possibile. Vuole una visualizzazione chiara degli elementi presenti sul tavolo da gioco.
Precondizioni	Il giocatore ha avviato l'eseguibile del gioco oppure, all'interno della finestra, ha cliccato sul pulsante " <i>Nuova partita</i> ", presente nella barra del menù.
Garanzia di successo (post-condizioni)	Nella finestra vengono visualizzati tutti gli elementi necessari per poter iniziare il gioco: il mazzo (stock) coperto, le sette colonne di carte, nelle quali la prima (quella più a sinistra) sarà sempre scoperta, le quattro basi inizialmente vuote.

Scenario principale di successo	<ol style="list-style-type: none"> 1. Il giocatore avvia il programma per iniziare una nuova partita del gioco <i>Solitario</i>. 2. Il programma mostra il tavolo da gioco con il mazzo (stock) inizialmente coperto, le carte distribuite sulle sette colonne in modo che ci sia una carta coperta nella prima colonna, due nella seconda colonna, tre sulla terza e così via; inoltre, vengono mostrate anche le quattro basi inizialmente vuote.
Estensioni (o flussi alternativi)	<p>*a. In qualsiasi momento, l'applicazione termina in modo anomalo:</p> <p>per consentire il ripristino, bisogna garantire che il sistema possa essere ripristinato, a partire da qualsiasi passo dello scenario.</p> <ol style="list-style-type: none"> 1. Il giocatore riavvia l'applicazione e richiede il ripristino dello stato precedente. 2. L'applicazione ricostruisce lo stato precedente. <p>2a. L'applicazione rileva delle anomalie che impediscono il ripristino:</p> <ol style="list-style-type: none"> 1. L'applicazione segnala un errore al giocatore, registra l'errore, e passa in uno stato pulito. 2. Il giocatore inizia una nuova partita.
Requisiti speciali	
Elenco delle varianti tecnologiche e dei dati	
Frequenza di ripetizione	potrebbe essere quasi ininterrotta.
Varie	

3 Analisi Orientata agli Oggetti

L'analisi orientata agli oggetti si basa sulla creazione di una descrizione del dominio da un punto di vista ad oggetti. Vengono utilizzati diversi strumenti per fornire tale descrizione: Modello di Dominio, SSD (Sequence System Diagram) e Contratti delle operazioni.

3.1 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 considerati significativi.

Relativamente al caso d'uso scelto (UC1), dopo un'attenta valutazione dello scenario principale di successo, è stato possibile identificare le seguenti classi concettuali:

Player: Rappresenta l'attore primario, che interagisce col sistema per eseguire le operazioni

SolitaireGame: Rappresenta l'applicazione del gioco Solitario

Deck: Rappresenta un mazzo di carte francesi costituito da 52 carte

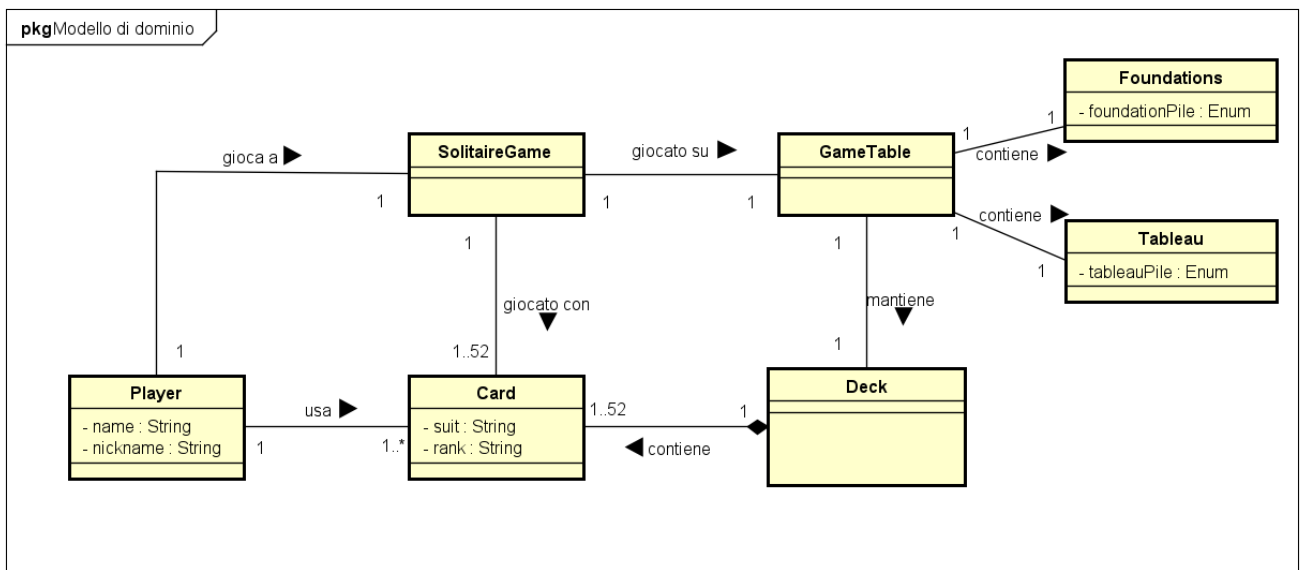
Card: Rappresenta il concetto di carta da gioco, caratterizzata da un valore e da un seme

GameTable: Rappresenta il luogo in cui viene svolto il gioco

Foundations: Rappresenta la zona del tavolo da gioco in cui sono presenti quattro pile di carte inizialmente vuote (basi)

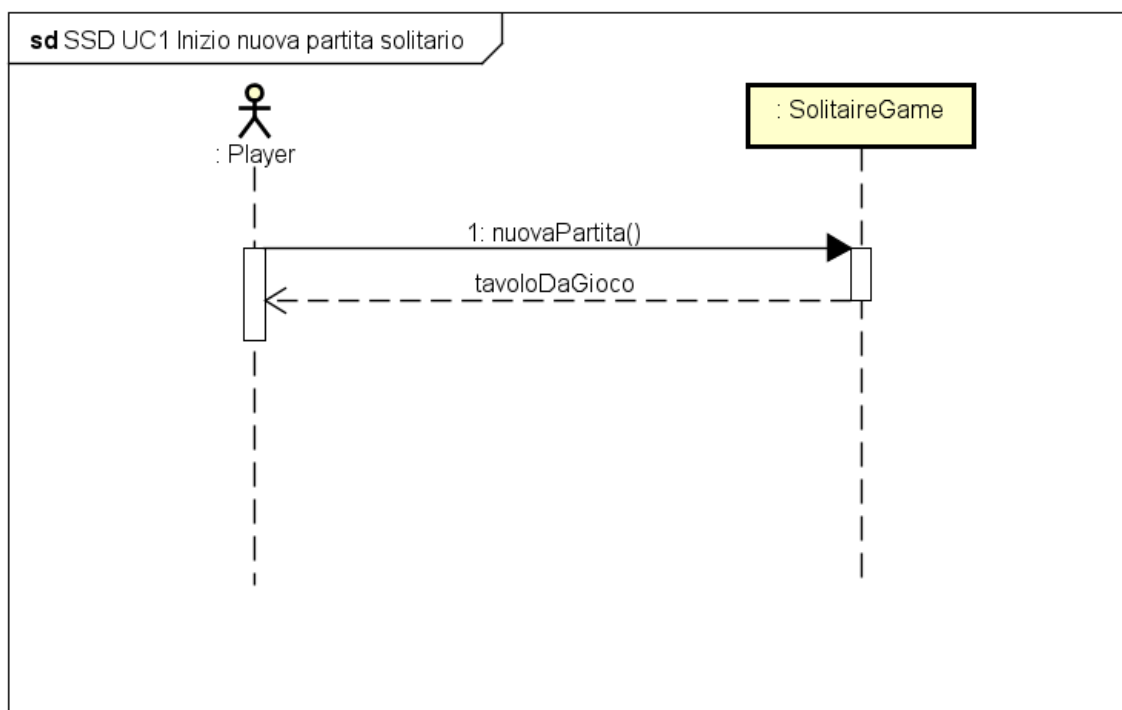
Tableau: Rappresenta la zona del tavolo da gioco in cui sono presenti sette colonne di carte. Su queste verranno distribuite le carte in modo che ogni colonna ne abbia un numero uguale alla sua posizione (la prima colonna una, la seconda due, ecc..), per un totale di 28 carte presenti nel Tableau all'inizio del gioco

Tenendo conto di associazioni e attributi tra queste classi, il modello di dominio che ne viene fuori è il seguente:



3.2 Diagramma di sequenza di sistema

Procediamo ora con il secondo step dell'analisi orientata agli oggetti, con la creazione del diagramma di sequenza di sistema (SSD), al fine di illustrare il corso degli eventi di input e output costituenti il caso d'uso in analisi UC1, e nello specifico come già detto lo scenario principale di successo. Avremo allora:



3.3 Contratti delle operazioni

Il prossimo passo è quello della descrizione delle operazioni individuate all'interno del SSD tramite i contratti delle operazioni.

Contratto CO1: Nuova Partita

Operazione: nuovaPartita()

Riferimenti: Caso d'uso: Inizio nuova partita Solitario

Pre-condizioni: -

Post-condizioni:

- è stata creata un'istanza singleton *sg* di SolitaireGame;
- gli attributi di *sg* sono stati inizializzati;
- è stata creata un'istanza *gt* di GameTable;
- gli attributi di *gt* sono stati inizializzati;
- è stata creata un'istanza *deck* di Deck;
- gli attributi di *deck* sono stati inizializzati;
- *gt* è stata associata all'istanza *sg* tramite l'associazione "giocato su";
- *deck* è stata associata all'istanza *gt* tramite l'associazione "mantiene";

4 Progettazione

La progettazione orientata agli oggetti è la disciplina di UP interessata alla definizione degli oggetti software, delle loro responsabilità e a come questi collaborano per soddisfare i requisiti individuati nei passi precedenti. 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 (Diagrammi di Interazione) che da un punto di vista statico (Diagramma delle Classi). Seguono dunque i diagrammi di Interazione più significativi e il diagramma delle Classi relativi al caso d'uso UC1, determinati a seguito di un attento studio degli elaborati scritti in precedenza.

➤ **Nuova partita solitario**

[illegible]