# Elaborazione – Iterazione 2

## 2.1 Introduzione

Obiettivo dell’Iterazione 2 sarà quello di implementare lo scenario principale di successo e tutte le estensioni individuate nel caso d’uso UC2: “Inserimento e pagamento di una prenotazione” nella fase di ideazione.

## 2.2 Analisi Orientata agli Oggetti

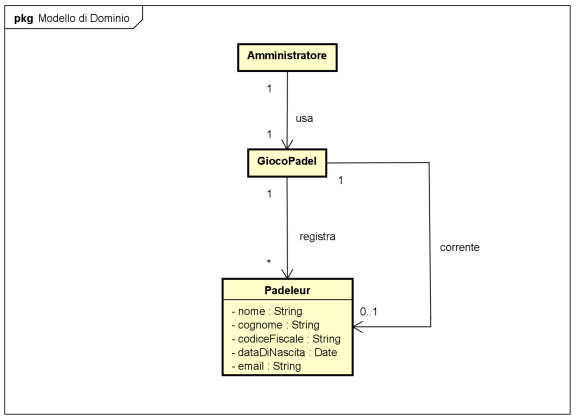
Di seguito vengono elencati il Modello di Dominio, SSD, e Contratti delle operazioni.

### 2.3.1 Modello di Dominio

Per il caso d’uso scelto UC2 sono stati identificati le seguenti classi concettuali:

* GiocoPadel: rappresenta il Sistema;
* Utente (Padeleur): rappresenta un giocatore che vuole utilizzare il Sistema per prenotare una partita di Padel.
* Prenotazione: contiene i dati relativi alla prenotazione di un campo di padel, in un giorno e fascia oraria specifica.
* Magazzino: al suo interno vi è l’attrezzatura che può essere noleggiata.
* CampoPadel: contiene i dati relativi ad un campo di padel.

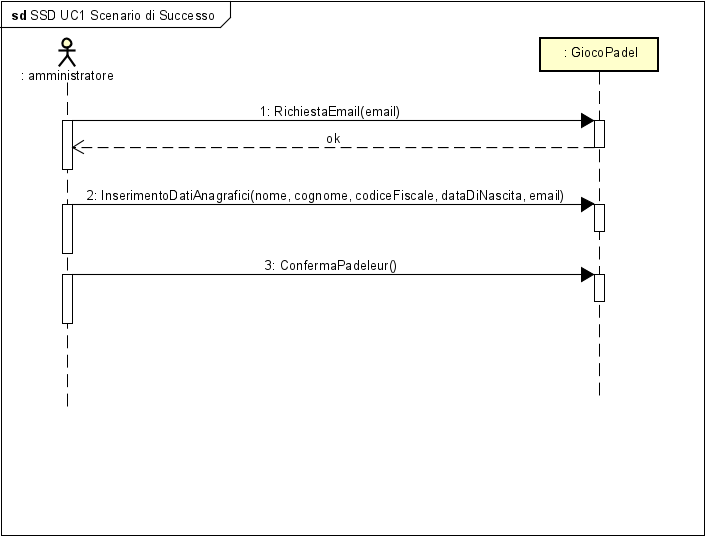
È stato ricavato il seguente Modello di Dominio:



Si nota che si è preferito, rispetto all’Iterazione 1, denominare il collegamento tra Amministratore e GiocoPadel con il termine “gestisce”.

### 2.3.2 Diagramma di sequenza di sistema

Procedendo con l’analisi Orientata agli Oggetti, andiamo a creare il Diagramma di Sequenza di Sistema (SSD) per lo scenario del caso d’uso scelto UC1. Avremo:



In questo diagramma, l'Amministratore richiede l’e-mail del Padeleur al sistema GiocoPadel, che a sua volta verifica se il Padeleur esiste già nel sistema o meno. Se il Padeleur non esiste ancora su GiocoPadel, l'amministratore inserisce i dati anagrafici del Padeleur nel sistema e conferma la registrazione. In caso contrario, il sistema segnala all'amministratore che il Padeleur è già stato registrato in precedenza e il processo di registrazione si interrompe.

### 2.3.3 Contratto delle operazioni

Di seguito viene indicato il Contratto delle operazioni per l’UC1:

|  |  |
| --- | --- |
| **Operazione:** | RichiestaEmail(email). |
| **Riferimenti:** | Caso d’uso UC1: Registrazione nuovo Padeleur. |
| **Pre-condizioni:** | - |
| **Post-condizioni:** | è stato restituito un Messaggio di Verifica dal Sistema. |

|  |  |
| --- | --- |
| **Operazione:** | InserimentoDatiAnagrafici (nome, cognome, codiceFiscale, dataDiNascita, email). |
| **Riferimenti:** | Caso d’uso UC1: Registrazione nuovo Padeleur. |
| **Pre-condizioni:** | Il Messaggio di Verifica ha dato esito positivo, cioè nel Sistema non esiste nessun Utente (Padeleur) con l’e-mail inserita. |
| **Post-condizioni:** | 1. È stata creata l’istanza nuovoPadelur di Padeleur; 2. Gli attributi nome, cognome, codiceFiscale, dataDiNascita, email di nuovoPadeleur sono stati inizializzati correttamente. 3. nuovoPadeleur è stato associato a GiocoPadel tramite l’associazione “corrente”. |

|  |  |
| --- | --- |
| **Operazione:** | ConfermaPadeleur() |
| **Riferimenti:** | Caso d’uso UC1: Registrazione nuovo Padeleur. |
| **Pre-condizioni:** | È in corso l’inserimento del nuovo Padeleur. |
| **Post-condizioni:** | nuovoPadeleur è stata aggiunto a GiocoPadel tramite l’associazione “registra”; |

## 2.4 Progettazione

Di seguito sono presenti i Diagrammi di Sequenza:

RichiestaEmail(email : String)

Immagine che contiene testo, schermata, diagramma, linea

Descrizione generata automaticamente

InserimentoDatiAnagrafici(nome : String, cognome : String, codiceFiscale : String, dataDiNascita : Date, email : String)Immagine che contiene testo, schermata, linea, diagramma

Descrizione generata automaticamente

ConfermaPadeleur()

Immagine che contiene testo, schermata, diagramma, linea

Descrizione generata automaticamente

## 2.4.1 Modello delle classi di Progetto

Immagine che contiene testo, schermata, linea, Carattere

Descrizione generata automaticamente