

1. Gestione dei Dati Persistenti

1.1 Introduzione

Per memorizzare i dati persistenti abbiamo scelto un database relazionale in modo che possa essere gestito agevolmente l'accesso concorrente ai dati e l'utilizzo del DBMS garantisca la consistenza dei dati stessi. La scelta di un DBMS è stato il giusto compromesso per essere il più possibile coerenti con i design goals stabiliti, infatti, questa soluzione porta diversi vantaggi quali:

- **Privatezza dei dati:** Un DBMS permette un accesso protetto ai dati. Utenti diversi possono avere accesso a diverse porzioni della base di dati e possono essere abilitati a diverse operazioni su di esse.
- **Imposizioni di vincoli di integrità sui dati:** Un DBMS permette di specificare diversi tipi di vincoli per mantenere l'integrità dei dati e controlla che tali vincoli siano soddisfatti quando la base di dati cambia
- **Atomicità delle operazioni:** Un DBMS permette di effettuare sequenze di operazioni in modo atomico. Ciò significa che l'intera sequenza di operazioni viene eseguita con successo oppure nessuna di queste operazioni ha alcun effetto sui dati della base. L'atomicità delle transazioni permette di mantenere uno stato della base di dati consistente con la realtà modellata.
- **Affidabilità dei dati:** Un DBMS offre dei metodi per salvare copie dei dati e per ripristinare lo stato della base di dati in caso di guasti software e hardware.

2. Dizionario dei Dati

| Utente (Contiene i dati relativi ad un utente) | | |
|--|---|-------------|
| Campo | Vincoli | Tipo |
| E-mail | <ul style="list-style-type: none">Primary keyNot null | varchar(25) |
| Nome | <ul style="list-style-type: none">Not null | varchar(15) |
| Cognome | <ul style="list-style-type: none">Not null | varchar(15) |
| Titolo di Studio | | varchar(25) |
| Password | <ul style="list-style-type: none">Not null | varchar(10) |
| Tipo | <ul style="list-style-type: none">Not nullPuò assumere:<ul style="list-style-type: none">0: Coach non convalidato1: Coach2: Studente | int |
| MessaggiDaLeggere | <ul style="list-style-type: none">Not null | boolean |

| Amministratore (Contiene i dati relativi all'Amministratore) | | |
|--|--|-------------|
| Campo | Vincoli | Tipo |
| idAmministratore | <ul style="list-style-type: none">Primary KeyAuto_increment | int |
| E-mail | <ul style="list-style-type: none">Not null | varchar(25) |
| Password | <ul style="list-style-type: none">Not null | varchar(10) |

| Richiesta Coach (Contiene i dati relativi alla Richiesta Coach) | | |
|---|---|-------------|
| Campo | Vincoli | Tipo |
| E-mailUtente | <ul style="list-style-type: none">Primary Key | varchar(25) |
| Categoria | <ul style="list-style-type: none">Not null | varchar(20) |
| Argomento | <ul style="list-style-type: none">Not null | varchar(50) |

Thread (Contiene i dati relativi ad un Thread)

| Campo | Vincoli | Tipo |
|----------|--|--------------|
| idThread | <ul style="list-style-type: none">• Primary Key• Auto_increment | int |
| Oggetto | <ul style="list-style-type: none">• Not null | varchar(500) |
| Titolo | <ul style="list-style-type: none">• Not null | varchar(20) |

Partecipazione (Contiene i dati relativi alla partecipazione ad un thread)

| Campo | Vincoli | Tipo |
|--------------|---|--------------|
| idThread | <ul style="list-style-type: none">• Not null• Auto_increment | int |
| E-mailUtente | <ul style="list-style-type: none">• Not null | varchar(25) |
| Messaggio | <ul style="list-style-type: none">• Not null | varchar(500) |
| Ordine | <ul style="list-style-type: none">• Not null | int |

Messaggio (Contiene i dati relativi al Messaggio)

| Campo | Vincoli | Tipo |
|--------------|--|--------------|
| Ordine | <ul style="list-style-type: none">• Not null | int |
| Destinatario | <ul style="list-style-type: none">• Not null | varchar(25) |
| Mittente | <ul style="list-style-type: none">• Not null | varchar(25) |
| Oggetto | <ul style="list-style-type: none">• Not null | varchar(200) |

3. Schema E-R

