

# TRAIN

PROFILO REGISTRATO: [f.dimaio@registrato.it](mailto:f.dimaio@registrato.it) password: test

PROFILO ESERCIZIO: [train.station@esercizio.it](mailto:train.station@esercizio.it) password: test

PROFILO AMMINISTRATIVO: [carlo.rossi@amministrativo.it](mailto:carlo.rossi@amministrativo.it) password: test

il Servizio è stato sviluppato utilizzando PHP 8.2.4 e HTML5, le semplificazioni apportate sono le seguenti:

- Pagamenti e prenotazioni biglietti per treni illimitati
- I posti a sedere vengono inseriti in maniera randomica in base alla disponibilità dei posti del treno composto
- I treni viaggiano a tutti gli orari
- Si possono creare treni in base alle istruzioni della prova in itinere
- La registrazione avviene senza nessun tipo di verifica lato utente (telefono, credenziali ecc....)
- La registrazione su Train Station crea di default un'utenza su Pay Stream con l'apertura di un conto corrente per il pagamento dei biglietti acquistati

## **Funzionalità:**

1. Registrazione utente con associazione del conto corrente su Pay stream
2. Possibilità di prenotare treni per tutte le stazioni disponibili
3. Ad ogni utente viene assegnato un posto sul treno
4. Il profilo esercizio può comporre/eliminare  $n$  treni in base al limite giornaliero
5. Il profilo amministrativo può creare/eliminare treni extra e può visualizzare la redditività del servizio Train Station
6. Puoi cercare treni per data
7. Puoi visualizzare i biglietti prenotati
8. Puoi effettuare il pagamento dei biglietti direttamente dal servizio Train Station

## **Elenco Entità:**

1. **Table: carrozza**
  - id\_carrozza (int)
  - serie\_carrozza (varchar)
  - tipo\_carrozza (varchar)
  - numero\_posti (int)

### **Table: composizione\_treno**

- id\_treno (int)
- id\_carrozze (varchar)
- id\_locomotive (varchar)
- numero\_posti\_totale (int)
- data\_inizio\_servizio (datetime)
- data\_fine\_servizio (datetime)

### **2. Table: locomotiva**

- id\_locomotiva (int)
- tipo\_locomotiva (varchar)

### **3. Table: stazione**

- id\_stazione (int)
- nome\_stazione (varchar)
- posizione\_km (decimal)

### **4. Table: tratta**

- id\_tratta (int)
- distanza\_km (decimal)
- costo\_biglietto (decimal)
- id\_stazione\_partenza (int)
- id\_stazione\_arrivo (int)
- data\_orario\_partenza (datetime)
- id\_utente (int)
- posto (int)

### **5. Table: utenti\_registrati\_train**

- id\_utente (int)
- nome (varchar)
- cognome (varchar)
- codice\_fiscale (varchar)
- data\_di\_nascita (datetime)
- e-mail (varchar)
- password (varchar)
- ruolo (varchar)

Queste tabelle sono interconnesse attraverso relazioni di chiave esterna, stabilendo associazioni tra di loro:

La tabella carrozza:

- composizione\_treno.id\_carrozze fa riferimento a carrozza.id\_carrozza.

La tabella composizione\_treno:

- tratta.id\_carrozze fa riferimento a carrozza.id\_carrozza.

- tratta.id\_locomotive fa riferimento a locomotiva.id\_locomotiva.

La tabella tratta:

- tratta.id\_stazione\_partenza fa riferimento a stazione.id\_stazione.
- tratta.id\_stazione\_arrivo fa riferimento a stazione.id\_stazione.
- tratta.id\_utente fa riferimento a utenti\_registrati\_train.id\_utente.

## Operazioni:

### 1. Gestione dei Treni:

- Creazione di un nuovo treno nella tabella composizione\_treno, specificando le carrozze e le locomotive coinvolte.

### 2. Gestione delle Tratte:

- Creazione di nuove tratte nella tabella tratta, specificando la stazione di partenza e di arrivo, la distanza, il costo del biglietto e il posto a sedere.
- Associazione di una tratta a un utente registrato nella tabella utenti\_registrati\_train.

### 3. Gestione delle Stazioni:

- Inserimento o eliminazione delle informazioni sulle stazioni nella tabella stazione.

### 4. Gestione degli Utenti Registrati:

- Creazione di nuovi utenti registrati nella tabella utenti\_registrati\_train.

### 5. Utente Amministrativo:

- Creazione/eliminazione treni extra.
- Visualizzazione rendita del servizio Train Station

## Schema ER:

