

Palestra StrongHouse

Progetto di Basi di Dati

Bressan Stefano - 1201199

Uderzo Marco - 1201290

a.a. 2019/2020

1. Abstract

La palestra “StrongHouse” è un centro fitness che si pone come obiettivo un’esperienza di allenamento all’avanguardia. Non solo crede nel duro allenamento con i pesi, quelli con cui tutti i migliori atleti si sono allenati fin ora, ma anche nell’ottimizzazione di esso attraverso l’uso di nuove tecnologie di monitoraggio della performance.

I suoi 500 clienti sono quasi tutti atleti di alto livello nel campo del Bodybuilding, del Powerlifting e del Weightlifting olimpionico, e prendono molto seriamente il loro percorso in palestra. Ogni atleta fino ad ora si è annotato rigorosamente ogni propria sessione, a mano, su un quadernino. Ora, StrongHouse vuole automatizzare, migliorare e rendere più semplice questo processo.

Si vuole creare un Database che gestisca i dati dei clienti della palestra, in particolare gli accessi, le schede e le sessioni di allenamento di ogni cliente, in modo da garantirgli una programmazione dell’allenamento studiata e monitorata nel lungo periodo. Questo attraverso macchinari e attrezzi di ultima generazione, muniti di sensori e accelerometri integrati ad alta precisione, con cui vengono registrati i dati delle sessioni consentendo al sistema di elaborare grafici della performance e dello stato fisico del cliente nel corso del tempo. In questo modo, il cliente potrà consultare in qualsiasi momento, nell’app dedicata o nell’area personale del sito web della palestra, tutte le informazioni relative alle sue sessioni di allenamento e alla sua costituzione corporea.

2. Analisi dei requisiti

2.1 Requisiti

Il Database che si vuole creare dovrà gestire una serie di dati relativi ai clienti della palestra, ai loro accessi e alle loro sessioni di allenamento.

Quando un cliente si iscrive vengono salvati tutti i suoi dati anagrafici, la sua email e numero di telefono. Inoltre, si registrano Username e Password del cliente, che potrà poi utilizzare per accedere al sito/app della palestra, l'altezza in centimetri e la data di scadenza dell'ultimo certificato medico di buona salute presentato. Se non ne fosse provvisto, il cliente può presentare il certificato entro un mese dalla sua iscrizione.

StrongHouse propone i seguenti tipi di abbonamenti:

- Abbonamento temporale, con una certa durata prestabilita, determinata dalla data di inizio e di fine.
- Abbonamento a entrate singole, con un numero stabilito di accessi, determinato dal numero di entrate rimanenti.

Dopo la registrazione, il database inizia a raccogliere varie informazioni sul cliente. In particolare permette di:

- Registrare tutti i Body Check, che comprendono la misurazione del peso e, tramite test BIA, delle percentuali di grasso, massa muscolare e acqua. Questo grazie ad un nuovissimo dispositivo bioimpedenziometrico: il cliente sale sulla bilancia ove ci sono degli elettrodi che rilasceranno una carica elettrica minima attraverso il suo corpo e, osservando la resistenza di questa, potrà dedurre la composizione fisica.
- Memorizzare tutte le Schede di Allenamento personalizzate, create appositamente per lui da un Personal Trainer (PT).
- Registrare tutti i dati delle Sessioni di allenamento che vengono raccolti durante le Serie da:
 - Attrezzi di Resistenza, che registreranno il numero delle ripetizioni e i carichi
 - Macchinari Cardio, i quali salveranno Potenza Media, Heart Rate Medio e Heart Rate Massimo

Le serie registrate dalle due tipologie di macchinari saranno identificate univocamente dall'ID della sessione e dalla loro ora di inizio.

Il Volume, l'Inizio e la Fine di una sessione di allenamento vengono calcolati con un trigger quando essa viene conclusa, per questo quando una sessione viene creata ed è ancora in atto, questi dati sono assenti. Una sessione viene creata quando viene terminata la prima serie. Il trigger considera conclusa una sessione d'allenamento dopo che è trascorsa un'ora dalla fine dell'ultima serie.

Tramite l'App della palestra, il cliente potrà accedere a tutte queste informazioni, quindi potrà visualizzare le proprie sessioni di allenamento, dei grafici elaborati su queste, i suoi body check, l'ultima misurazione della propria altezza, l'andamento delle sue performance e tutti gli accessi fatti.

2.2. Glossario dei termini

Termine	Spiegazione
Cliente	Iscritto alla palestra con Abbonamento
Abbonamento Temporale	Abbonamento a durata stabilita
Abbonamento ad Entrate Singole	Abbonamento composto da N entrate singole
Body Check	Analisi dello stato fisico del cliente con Bioimpedenziometria
Esercizio	Sequenza di movimenti atti a stimolare fasce muscolari
Scheda di Allenamento	Insieme di N esercizi
Accesso	Entrata in palestra del cliente
Serie	Singolo gruppo di ripetizioni di un esercizio
Sessione	Insieme di serie
Attrezzature e Macchinari	Vari attrezzi e macchinari dotati di sensori

2.3. Operazioni

- 1) Visualizzare lo storico degli abbonamenti di un cliente
- 2) Visualizzare clienti con abbonamento scaduto non rinnovato
- 3) Visualizzare gli accessi di un cliente
- 4) Visualizzare la media degli accessi dei clienti nelle diverse fasce orarie durante la settimana in un determinato periodo di tempo
- 5) Visualizzare una scheda di allenamento di un cliente
- 6) Visualizzare una sessione di allenamento di cliente
- 7) Visualizzare un body check di un cliente
- 8) Visualizzare il volume di tutte le sessioni di allenamento di un cliente in un determinato periodo di tempo

3. Progettazione Concettuale

3.1 Descrizione Entità e Relazioni

Entità:

Entità	Descrizione	Attributi	Identificatore
Abbonamento	Abbonamento alla palestra	ID : int	ID
Entrate_Singole	Abbonamento ad entrate limitate alla palestra	ID : int Entrate_Rimaste : smallint	ID
Temporale	Abbonamento a tempo determinato alla palestra	ID : int Inizio : date Fine : date	ID
Accesso	Entrata in palestra tramite tessera	CF : varchar(16) Entrata : datetime Uscita : datetime	CF, Entrata
Body_Check	Misurazione bioimpedenziometrica della composizione corporea del cliente	CF: varchar(16) DataOra : datetime Peso: float Body_Fat_pct: float Muscle_Mass_pct: float Water_pct: float FFMI: float	CF, DataOra
Cliente	Abbonato alla palestra con abbonamento temporale o ad entrate singole	CF : varchar(16) Nome : varchar(40) Cognome : varchar(40) Telefono : varchar(15) Email : varchar (190) Scad_Cert_Med : date Altezza : float User_App : varchar(16) Psw_App : varchar(32)	CF
Esercizio	Esercizio ginnico	ID : int Nome : varchar(40) Attrezzo : varchar(20) Serie : tinyint Ripetizioni : tinyint Recupero : smallint RPE : float	ID
Scheda	Insieme di esercizi suddivisi in varie sedute di allenamento atti ad allenare tutti i fasci	CF : varchar(16) Numero : smallint Obiettivo : string(20)	CF, Numero

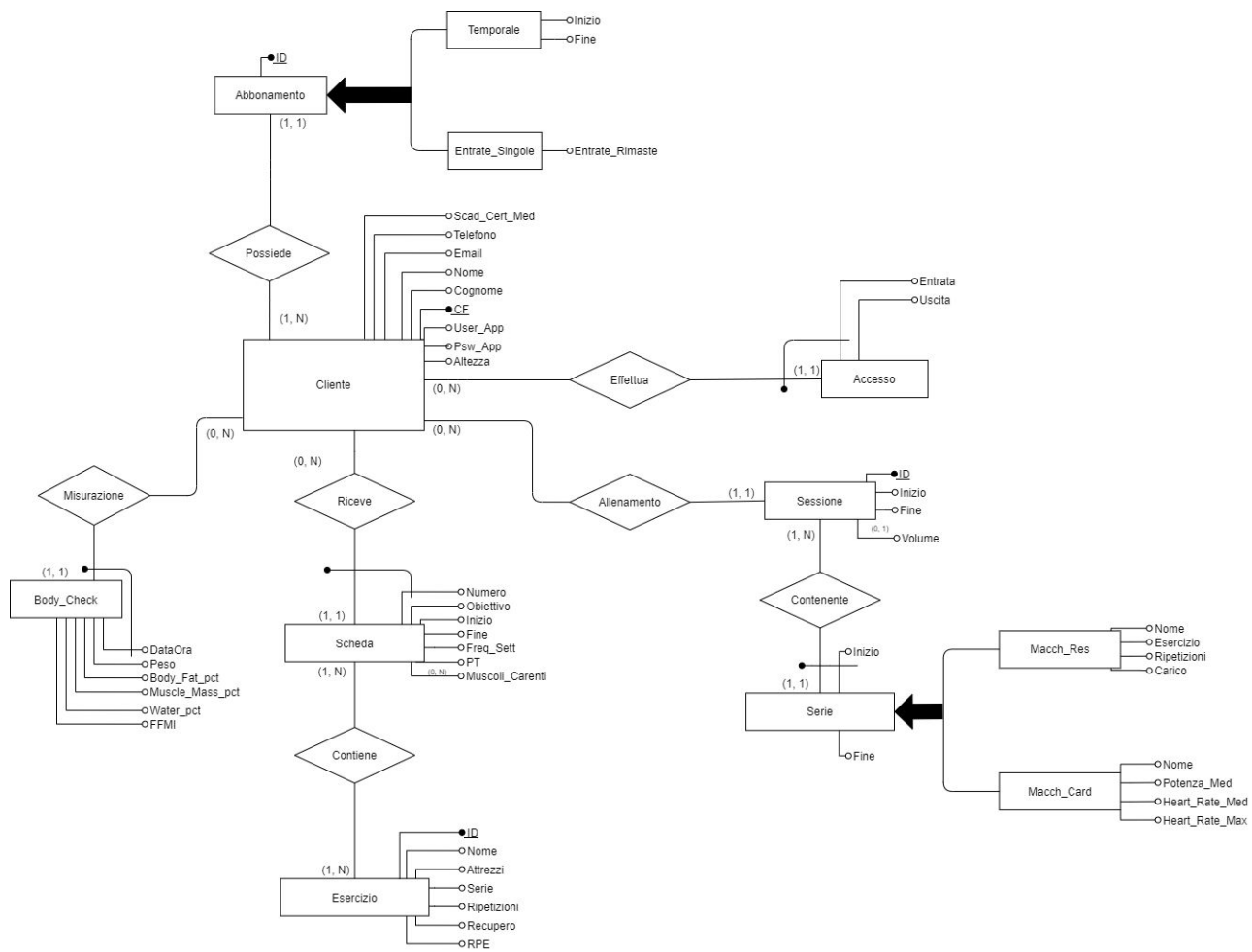
	muscolari	Inizio : date Fine : date Freq_Set : tinyint PT : varchar(40) Muscolo_Carente : varchar(20)	
Serie	Raggruppamento di una o più ripetizioni di un determinato esercizio o movimento	ID : int Inizio: datetime Fine: datetime	ID, Inizio
Macchinario_Res	Attrezzatura per l'allenamento della forza	ID : int Inizio : datetime Fine: datetime Nome : varchar(20) Esercizio : varchar(40) Ripetizione : tinyint Carico : float	ID, Inizio
Macchinario_Card	Macchinario per l'allenamento di tipo cardio	ID : int Inizio : datetime Fine : datetime Nome : varchar(20) Potenza_Med : float Heart_Rate_Med : smallint Heart_Rate_Max : smallint	ID, Inizio
Sessione	Insieme Serie effettuate durante un allenamento	ID : int Inizio : datetime Fine : datetime Volume: float	ID

Relazioni:

Relazione	Descrizione	Cardinalità
Allenamento	Un Cliente effettua una Sessione di allenamento in palestra	Cliente (0, N) Sessione (1, 1)
Contenente	Una Sessione contiene un insieme di Serie	Sessione (1, N) Serie (1, 1)
Contiene	Una Scheda contiene un insieme di Esercizi	Scheda (1, N) Esercizio (1, N)
Effettua	Un Cliente entra ed esce dalla palestra	Cliente (0, N) Accesso (1, 1)
Misurazione	Un cliente effettua sulla bilancia una misura del proprio Body_Check	Cliente (0, N) Body_Check(1, 1)
Possiede	Il cliente possiede un Abbonamento	Cliente (1, N)

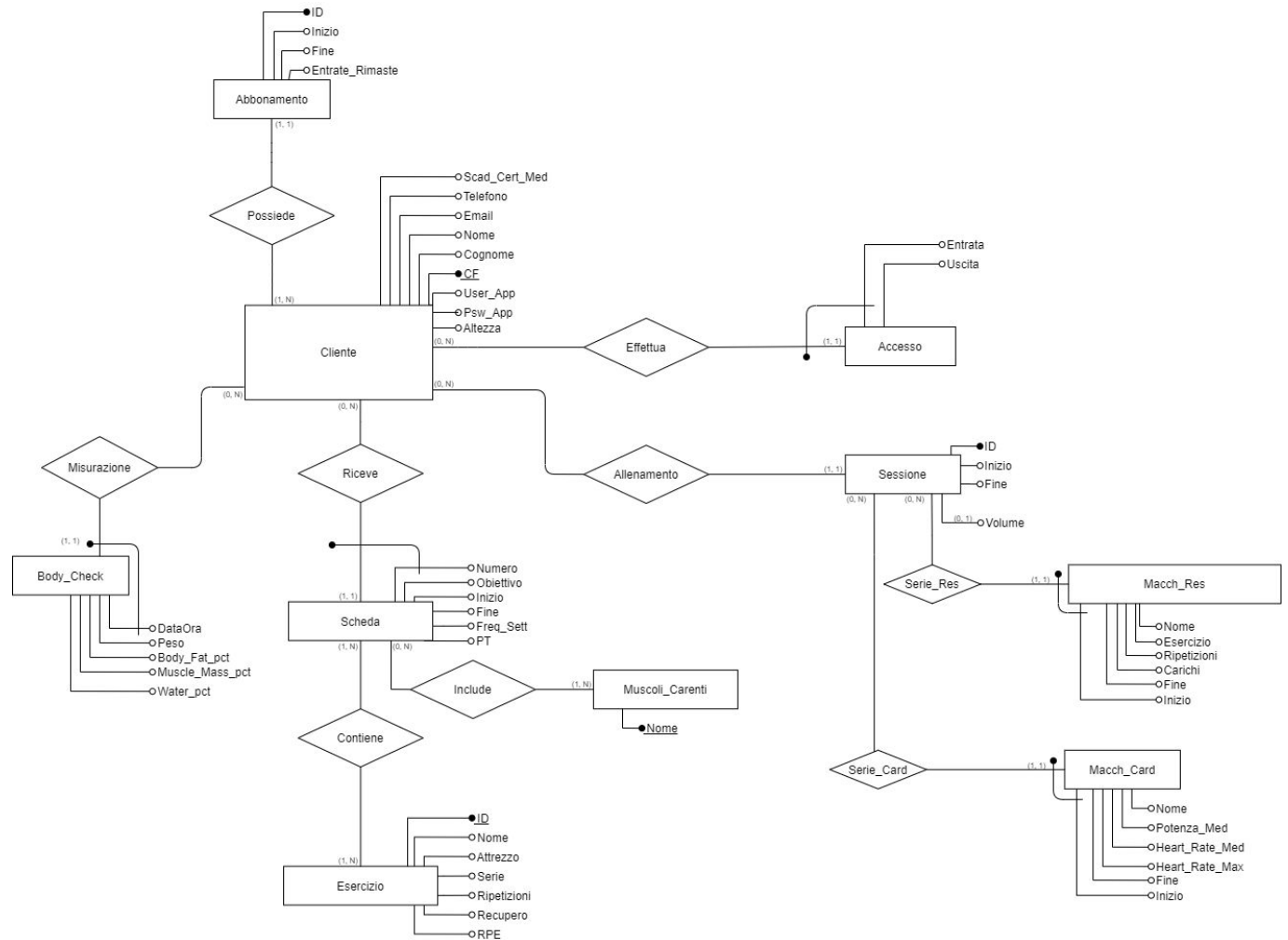
		Abbonamento (1, 1)
Riceve	Il cliente riceve dal Personal Trainer delle Schede di allenamento	Cliente (0, N) Scheda (1, N)

3.2. Schema Concettuale (E-R)



4. Progettazione Logica

4.1. Ristrutturazione dello Schema



L'entità **Scheda di Allenamento** ha un attributo multivalore "Gruppi_Muscolari_Carenti", con cardinalità (0,N). Per poterlo salvare nel database è stato necessario trasformarlo in un'entità con attributo:

- Nome

4.2. Analisi Ridondanze

FFMI

L'FFMI è un attributo di Body_Check che si calcola usando gli attributi Peso e Body_Fat_pct di Body_Check e l'attributo Altezza di Cliente. Conviene tenerlo salvato o calcolarlo ogni volta che viene richiesto?

Operazione 1 - Memorizzare un FFMI

Operazione 2 - Visualizzare FFMI di un cliente

Dati:

500 clienti

2 richieste di visualizzazione FFMI mediamente per cliente al giorno

1 registrazione di FFMI al mese

Presenza di ridondanza:

Operazione 1 - Memorizzare un FFMI

Concetto	Costrutto	Accessi	Tipo	Frequenza
Clienti	Entità	1	L	x 17 volte/giorno
Misurazione	Relazione	1	L	x 17 volte/giorno
Body_Check	Entità	1	S	x 17 volte/giorno

Operazione 2 - Visualizzare FFMI di un cliente

Concetto	Costrutto	Accessi	Tipo	Frequenza
Clienti	Entità	1	L	x 1000 volte/giorno
Misurazione	Relazione	1	L	x 1000 volte/giorno
Body_Check	Entità	1	L	x 1000 volte/giorno

- Costi:
 - Operazione 1: 34 accessi in lettura e 17 in scrittura al giorno
 - Operazione 2: 3000 accessi in lettura al giorno
- Assumendo costo doppio per gli accessi in scrittura
- Il costo giornaliero è $34 + 3000 + 17 \times 2 = 3068$

Assenza di ridondanza:

Operazione 2 - Visualizzare FFMI di un cliente

Concetto	Costrutto	Accessi	Tipo	Frequenza
Clienti	Entità	1	L	x 1000 volte/giorno
Misurazione	Relazione	1	L	x 1000 volte/giorno
Body_Check	Entità	1	L	x 1000 volte/giorno

- Costi:
- Operazione 2: 3000 accessi in lettura
- Il costo giornaliero è 3000

Salvare i Bodycheck dei clienti costerebbe ogni anno:

$4\text{Byte} \times 17 \text{ volte/giorno} \times 365 \text{ giorni} = 24\text{KB}$

Si conclude di eliminare la ridondanza perché calcolare l'FFMI al momento della richiesta, invece che salvarlo nel DataBase, oltre ad essere meno costoso, risparmia memoria.

Volume di una Sessione

L'entità Sessione possiede un attributo chiamato Volume, che rappresenta la mole di lavoro effettuata. Il Volume di una Sessione si ottiene sommando il Volume di tutte le serie che la compongono. Il Volume di una singola serie si calcola così:

Volume serie = Numero di Ripetizioni * Carico espresso in KG

Il volume è direttamente calcolabile tramite le entità Macch_Res appartenenti ad una Sessione; salvandolo su Sessione si creerebbe una ridondanza, ma come si può vedere è un calcolo oneroso in termini di accessi, poiché bisogna accedere ad ogni singola serie di resistenza della sessione di allenamento.

Operazione 1 - Memorizzare un Volume di una sessione

Operazione 2 - Visualizzare Volume di una sessione

Dati:

500 clienti

10 richieste per Volume di una sessione mediamente per cliente al giorno

35 Serie mediamente per sessione d'allenamento

285 Sessioni d'allenamento salvate mediamente al giorno

Presenza di ridondanza:

Operazione 1 - Memorizzare un Volume di una sessione

Concetto	Costrutto	Accessi	Tipo	Frequenza
Sessione	Entità	1	S	x 285 volte/giorno
Serie_Res	Relazione	35	L	x 285 volte/giorno
Macch_Res	Entità	35	L	x 285 volte/giorno

Operazione 2 - Visualizzare Volume di una sessione

Concetto	Costrutto	Accessi	Tipo	Frequenza
Sessione	Entità	1	L	x 5000 volte/giorno

- Costi:
 - Operazione 1: 19'950 accessi in lettura e 285 in scrittura al giorno
 - Operazione 2: 5000 accessi in lettura al giorno
- Assumendo costo doppio per gli accessi in scrittura
- Il costo giornaliero è $19'950 + 5000 + 285 \times 2 = 25'520$

Assenza di ridondanza:

Operazione 2 - Visualizzare Volume di una sessione

Concetto	Costrutto	Accessi	Tipo	Frequenza
Sessione	Entità	1	L	x 5000 volte/giorno

Serie_Res	Relazione	35	L	x 5000 volte/giorno
Macch_Res	Entità	35	L	x 5000 volte/giorno

- Costi:
- Operazione 2: 355'000 accessi in lettura
- Il costo giornaliero è 355'000

Salvare il Volume degli allenamenti dei clienti costerebbe ogni anno:
 $4\text{Byte} * 285 \text{ volte/giorno} * 365 \text{ giorni} = 416\text{KB}$

Si sceglie mantenere la ridondanza, nonostante consumi 416KB di memoria ogni anno, perché alleggerisce il costo delle richieste al DB di 12 volte e mezzo.

Visualizzare Ora di Inizio e Fine di una Sessione

L'entità Sessione ha come attributi un orario di inizio e di fine. Questi attributi possono essere ricavati, rispettivamente, dall'ora di inizio della prima serie e dall'ora di fine dell'ultima serie di una Sessione d'allenamento. Si ricorda che vi sono sia serie di Resistenza che Cardio.

Operazione 1 - Memorizzare Inizio e Fine di una Sessione

Operazione 2 - Visualizzare Inizio e Fine di una Sessione

Dati:

500 clienti

10 richieste per Inizio e Fine di una sessione mediamente per cliente al giorno

35 serie di resistenza mediamente per sessione d'allenamento

2 serie cardio mediamente per sessione d'allenamento

285 Sessioni d'allenamento salvate mediamente al giorno

Presenza di ridondanza:

Operazione 1 - Memorizzare Inizio e Fine di una Sessione

Concetto	Costrutto	Accessi	Tipo	Frequenza
Sessione	Entità	1	S	x 285 volte/giorno
Serie_Res	Relazione	35	L	x 285 volte/giorno
Macch_Res	Entità	35	L	x 285 volte/giorno
Serie_Card	Relazione	2	L	x 285 volte/giorno
Macch_Card	Entità	2	L	x 285 volte/giorno

Operazione 2 - Visualizzare Inizio e Fine di una Sessione

Concetto	Costrutto	Accessi	Tipo	Frequenza
Sessione	Entità	1	L	x 5000 volte/giorno

- Costi:
 - Operazione 1: 21'090 accessi in lettura e 285 in scrittura al giorno

- Operazione 2: 5000 accessi in lettura al giorno
- Assumendo costo doppio per gli accessi in scrittura
- Il costo giornaliero è $21'090 + 5000 + 285 \times 2 = 26'660$

Assenza di ridondanza:

Operazione 2 - Visualizzare Inizio e Fine di una Sessione

Concetto	Costrutto	Accessi	Tipo	Frequenza
Sessione	Entità	1	L	x 5000 volte/giorno
Serie_Res	Relazione	35	L	x 5000 volte/giorno
Macch_Res	Entità	35	L	x 5000 volte/giorno
Serie_Card	Relazione	2	L	x 5000 volte/giorno
Macch_Card	Entità	2	L	x 5000 volte/giorno

Costi:

- Operazione 2: 375'000 accessi in lettura
- Il costo giornaliero è 375'000

Salvare il l'Inizio e la Fine degli allenamenti dei clienti costerebbe ogni anno:

$$2 * 5\text{Byte} * 285 \text{ volte/giorno} * 365 \text{ giorni} = 1\text{MB}$$

Si sceglie di tenere la ridondanza, nonostante il peso dei dati, per mantenere leggere le richieste al Database.

4.2. Eliminazione delle generalizzazioni

Dato che in un normale DBMS non è possibile rappresentare delle generalizzazioni, abbiamo eliminato le due presenti usando due tecniche diverse:

- **Generalizzazione Abbonamento:**

Per questa generalizzazione si è deciso di accorpare le figlie nel padre. Con questa tecnica si trasferiscono al padre gli attributi dei figli, che diventano opzionali. Il campo Entrate_Rimaste verrà utilizzato per distinguere i due tipi di abbonamento: se l'istanza è null significa che è un abbonamento temporale, altrimenti si tratta di un abbonamento a entrate singole.

Quando viene creato un Abbonamento ad entrate singole le date di inizio e fine vengono messe a NULL, poi, al primo utilizzo, viene scritta la data d'inizio e quando viene usata l'ultima entrata si aggiorna da data di fine.

- **Generalizzazione Serie:**

Per questa generalizzazione si è deciso di accorpare il padre nelle figlie. Con questa tecnica si trasferiscono per ereditarietà gli attributi del padre alle figlie.

4.3. Vincoli di integrità referenziale

Relazione	Vincoli
Contiene	Contiene.Cliente -> Cliente.CF Contiene.Scheda -> Scheda_Allenamento.Numero Contiene.IDEsercizio -> Esercizio.ID
Include	Include.Cliente -> Cliente.CF Include.Scheda -> Scheda_Allenamento.Numero Include.Muscolo_Carente -> Muscolo_Carente.Nome

4.4. Integrità dei dati

Un cliente non può iniziare una nuova sessione di allenamento se ce n'è già una in corso.

5. Implementazione dello schema logico

Abbonamento(CF_Cliente, ID_Abb, Inizio, Fine, Entrate_Rimaste)

Accesso(CF_Cliente, Entrata, Uscita)

BodyCheck(CF_Cliente, DataOra, Peso, Body_Fat_pct, Muscle_Mass_pct, Water_pct)

Cliente(CE, Nome, Cognome, Telefono, Email, Scad_Cert_Med, Altezza, User_App, Psw_App)

Sessione(CF_Cliente, ID, Inizio, Fine, Volume)

Macch_Res(ID_Ses, Inizio, Fine, Nome, Esercizio, Ripetizioni, Carico)

Macch_Card(ID_Ses, Inizio, Fine, Nome, Potenza_Med, Heart_Rate_Med, Heart_Rate_Max)

Scheda(CF_Cliente, Numero, Obiettivo, Inizio, Fine, Freq_Set, PT)

Muscolo_Carente(Nome)

Esercizio(ID, Nome, Attrezzo, Serie, Ripetizioni, Recupero, RPE)

Contiene(CF_Cliente, Num_Scheda, ID_Esercizio)

Include(CF_Cliente, Num_Scheda, Muscolo_Carente)

6. Query e indici

6.1 Query

Query 1: Volume Competition

La query permette di vedere in una certa fascia oraria quale cliente si è allenato più intensamente ottenendo un il volume di allenamento maggiore, ovvero la somma di Ripetizioni*Carico di ogni serie della sessione.

```

1  SELECT C.Nome, C.Cognome, SUM(Ripetizioni*Carico) AS Volume
2  FROM Cliente C, Sessione S, Macch_Res MR
3  WHERE C.CF = S.CF_Cliente
4         AND S.ID = MR.ID_Ses
5         AND MR.Inizio > '2020-02-07 18:00:00'
6         AND MR.Fine < '2020-02-07 19:00:00'
7         GROUP BY C.CF
8  ORDER BY `Volume` DESC

```

Nome	Cognome	Volume ▾ 1
Jeff	Seid	2360
Marco	Marostegan	1370
Eddie	Hall	1130
Johnny	Facchinetti	450

Query 2: FFMI-Based Anti-Doping

Grazie alla misurazione dell'indice di massa magra (Fat Free Mass Index), StrongHouse può scoprire possibili casi di doping. E' dimostrato, infatti, che è statisticamente improbabile, eccetto per soggetti geneticamente dotati, che un atleta con FFMI maggiore di 25 non abbia mai fatto uso di sostanze anabolizzanti.

```

11 SELECT C.Nome, C.Cognome, (BC.Peso*((100-BC.Body_Fat_pct)/100)) / ((C.Altezza/100)*(C.Altezza/100)) AS FFMI
12 FROM Cliente C, Body_Check BC
13 WHERE C.CF = BC.CF_Cliente
14        HAVING FFMI > 25
15        ORDER BY FFMI

```

Nome	Cognome	FFMI ▴ 1
Eddie	Hall	30.51400115966797
Hafthor	Bjornsson	33.38140536976509

Query 3: Fat Loss Journey

Per ogni cliente che ha fatto almeno due Body_Check, la query mostra quanti KG di grasso ha perso. Se il cliente ha, nel complesso, aumentato i KG di grasso il numero visualizzato è negativo.

```

17 SELECT Nome, Cognome, BC1.DataOra AS Giornata1,
18        BC2.DataOra AS Giornata2,
19        BC1.Peso*BC1.Body_Fat_pct/100 AS KGGrassoPrima,
20        BC2.Peso*BC2.Body_Fat_pct/100 AS KGGrassoDopo,
21        ((BC2.Peso*BC2.Body_Fat_pct/100)-(BC1.Peso*BC1.Body_Fat_pct/100))*-1 AS KGGrassoPerso
22 FROM Cliente C, Body_Check BC1, Body_Check BC2
23 WHERE C.CF = BC1.CF_Cliente
24        AND BC1.CF_Cliente = BC2.CF_Cliente
25        AND BC1.DataOra < BC2.DataOra

```

Nome	Cognome	Giorno1	Giorno2	KGGrassoPrima	KGGrassoDopo	KGGrassoPerso
Venere	Botticelli	2020-01-19 17:01:01	2020-02-19 18:00:01	13.025999603271485	9.197999725341797	3.827999877929688
Damiano	Arnoldo	2020-01-19 17:32:55	2020-02-03 17:30:55	12.762000274658202	13.038999481201172	-0.2769992065429694
Jeff	Seid	2020-02-06 18:09:00	2020-02-23 18:09:00	6.313999786376953	8.316000137329102	-2.002000350952149

Query 4: Statistiche sugli obiettivi dei clienti

La query conta il numero di clienti che hanno un certo obiettivo (Massa, Forza, Definizione, Tonificazione, Ricomposizione), e ne calcola la percentuale.

```
29 SELECT S.Obiettivo, COUNT(*) AS Quantità, (COUNT(*)/(
30     SELECT COUNT(*)
31     FROM Cliente C, Scheda S
32     WHERE C.CF = S.CF_Cliente)*100) AS Percentuale
33 FROM Cliente C, Scheda S
34 WHERE C.CF = S.CF_Cliente
35 GROUP BY S.Obiettivo
36 ORDER BY Percentuale DESC
```

Obiettivo	Quantità	Percentuale ▾ 1
Massa	6	40.0000
Tonificazione	4	26.6667
Forza	3	20.0000
Definizione	1	6.6667
Ricomposizione	1	6.6667

Query 5: Affluenza Pomeridiana

Questa query permette alla palestra di visualizzare l'affluenza pomeridiana, al fine di ottimizzare i servizi. Ad esempio concentrare i corsi in quella fascia oraria.

```
39 SELECT COUNT(*) as NumeroAccPom, COUNT(*)/(SELECT COUNT(*) FROM Accesso)*100 as PercentualeAccessiPom
40 FROM Accesso
41 WHERE EXTRACT(HOUR FROM Entrata) >= 12 AND EXTRACT(HOUR FROM Entrata) <20
42
```

NumeroAccPom	PercentualeAccessiPom
7	87.5000

Query 6: Clienti abbonati ma che non hanno mai fatto un accesso

La query mostra i clienti che, pur essendo abbonati, non hanno mai fatto un accesso alla palestra. Tra questi si considerano solo i clienti che hanno un abbonamento temporale ancora attivo o un abbonamento ad entrate ancora da esaurire.

```
43 SELECT C.Nome, C.Cognome, C.CF
44 FROM Abbonamento as A JOIN Cliente as C ON A.CF_Cliente = C.CF
45 WHERE (A.Fine > CURDATE() OR A.Fine IS NULL)
46 AND A.CF_Cliente NOT IN (SELECT CF_Cliente
47 FROM Accesso)
```

Nome	Cognome	CF
Haflthor	Bjornsson	BJRHFT40A01O840I
Chiara	Bassani	BSSCHR30A01X842E
Venere	Botticelli	BTTVNR20A01G840O
Romina	Casonina	CSNRMN99M42A703P
Cristina	Luce	LCECRS30A01B840U
Rebecca	Murphy	MRPRBC80A01W840E
Stefano	Massello	MSSSTF80A01V521I
Jeff	Nippard	NPPJFF80A01T360E
Giovanni	Neve	NVEGVN60A01Y840O
Damiano	Arnoldo	RNDDRR96A01L846U

6.2 Indici

Si è scelto di non mettere nessun indice: sarebbe troppo dispendioso in quanto nuove entità, come Sessione, Accesso, Macch_Res, Macch_Card, vengono create con un'alta frequenza.

Non viene inserito un indice neanche su Body_Check perché viene acceduto con poca frequenza e nemmeno su entità come Cliente, Scheda, Esercizio e Muscoli_Carenti perché sono poco numerose.