

Relazione Basi di Dati: Colonnine elettriche

Matteo Fasulo, Simone Flavio Paris, Matteo Sivoccia

Gennaio 2022

Indice

1	Requisito	2
2	Analisi del Requisito	2
3	Glossario dei Termini	2
4	Assunzioni	2
5	Diagramma E-R	3
6	Diagramma E-R v2.0	4
7	Diagramma E-R v3.0	5
8	Dizionario dei Dati ed elenco dei vincoli	6
8.1	Dizionario dei Dati	6
8.2	Elenco dei vincoli	7
9	Tavola delle Operazioni e Tavola dei Volumi	8
9.1	Tavola delle Operazioni (Frequenza e Accessi)	8
9.2	Tavola dei Volumi	8
10	Modello Logico	9
11	Modello Fisico	11
12	Codice SQL	12
12.1	Sorgente	12
12.1.1	Op. 1	12
12.1.2	Op. 2	12
12.1.3	Op. 3	12
12.1.4	Op. 4	12
12.1.5	Op. 5	12
12.1.6	Op. 6	12
12.1.7	Op. 7	12
12.1.8	Op. 8	13
12.1.9	Op. 9	13
12.1.10	Op. 10	14
12.1.11	Op. 11	14
12.1.12	Op. 12	14
12.1.13	Op. 13	15
12.1.14	Op. 14	15
12.1.15	Op. 15	15
12.1.16	Op. 16	15
12.1.17	Op. 17	15
12.1.18	Op. 18	16
12.1.19	Op. 19	16
12.1.20	Op. 20	16

1 Requisito

Si vuole progettare la base dati per la gestione di colonnine elettriche. La compagnia dispone di diverse aree di sosta di diverso tipo e capienza distribuiti sul territorio, e impiega diverse categorie di dipendenti con diversi ruoli che costituiscono lo staff. Alcune delle strutture sono in italia, altre all'estero. Per ogni colonnina sono disponibili diverse tipologie ricarica (normal, fast, ultra-fast). Quando viene effettuata una prenotazione, che va registrata, si possono prenotare posti auto per sé e per altri mezzi non propri nei limiti della capienza dell'area di sosta. I dipendenti possono esprimere preferenze sull'area da gestire.

2 Analisi del Requisito

Nella nostra analisi abbiamo individuato:

- Entità
- Attributi

Si vuole progettare la base dati per la gestione di colonnine elettriche. La compagnia dispone di diverse aree di sosta di diverso tipo e capienza distribuiti sul territorio, e impiega diverse categorie di dipendenti con diversi ruoli che costituiscono lo staff. Alcune delle strutture sono in italia, altre all'estero. Per ogni colonnina sono disponibili diverse tipologie ricarica (normal, fast, ultra-fast). Quando viene effettuata una prenotazione, che va registrata, si possono prenotare posti auto per sé e per altri mezzi non propri nei limiti della capienza dell'area di sosta. I dipendenti possono esprimere preferenze sull'area da gestire.

3 Glossario dei Termini

Tabella 1: Glossario dei Termini

Termine	Descrizione	Sinonimi	Collegamenti
Colonnina Elettrica	Dispositivo al quale si connettono le auto per la ricarica elettrica		Area di Sosta
Area di Sosta	Luogo fisico nel quale sono presenti le colonnine elettriche per la ricarica		Colonnina Elettrica, Dipendente, Prenotazione
Dipendente	Persona assunta dall'azienda		Area di Sosta
Prenotazione	Documento di sintesi che attesta la prenotazione, utile a identificare una transazione		Area di Sosta
Utente	Utilizzatore finale del servizio di prenotazione colonnine elettriche	Cliente	Prenotazione
Presenze	Storico delle presenze giornaliere dei dipendenti		Dipendente, Area di Sosta

4 Assunzioni

1. Staff = dipendenti di un'area
2. Prenotazione vincolante (no overbooking)
3. La prenotazione anche per altri solo se nella stessa area di sosta
4. I dipendenti vengono allocati nelle aree da loro preferite in base all'anzianità e non a un sistema di ranking che sarebbe migliore ma più complesso da implementare.

5. Durante la prenotazione posso sapere quante sono le colonnine disponibili in una area per tipo di ricarica
6. La prenotazione può essere fatta sul momento e sul luogo collegandosi alla colonnina (ISO15118) o tramite scansione QR CODE/NFC, se il posto era già prenotato sarà la colonnina a rifiutare la ricarica
7. Le prenotazioni sono aperte sul momento anche tramite internet (a distanza)
8. L'utente può effettuare una prenotazione solo in una finestra di 24 ore
9. Per comporre il nuovo staff per l'area di sosta (TASK 9) si fa affidamento oltre che alle operazioni di SQL anche a un altro linguaggio di programmazione. Descriveremo cosa svolgerà [QUI](#)
10. Se il ruolo del dipendente non richiede l'associazione con un'area (per esempio Help Desk), allora il dipendente non può esprimere la preferenza di un'area e nel database verrà inserito il valore NULL
11. L'indirizzo di fatturazione corrisponde all'indirizzo di domicilio

5 Diagramma E-R

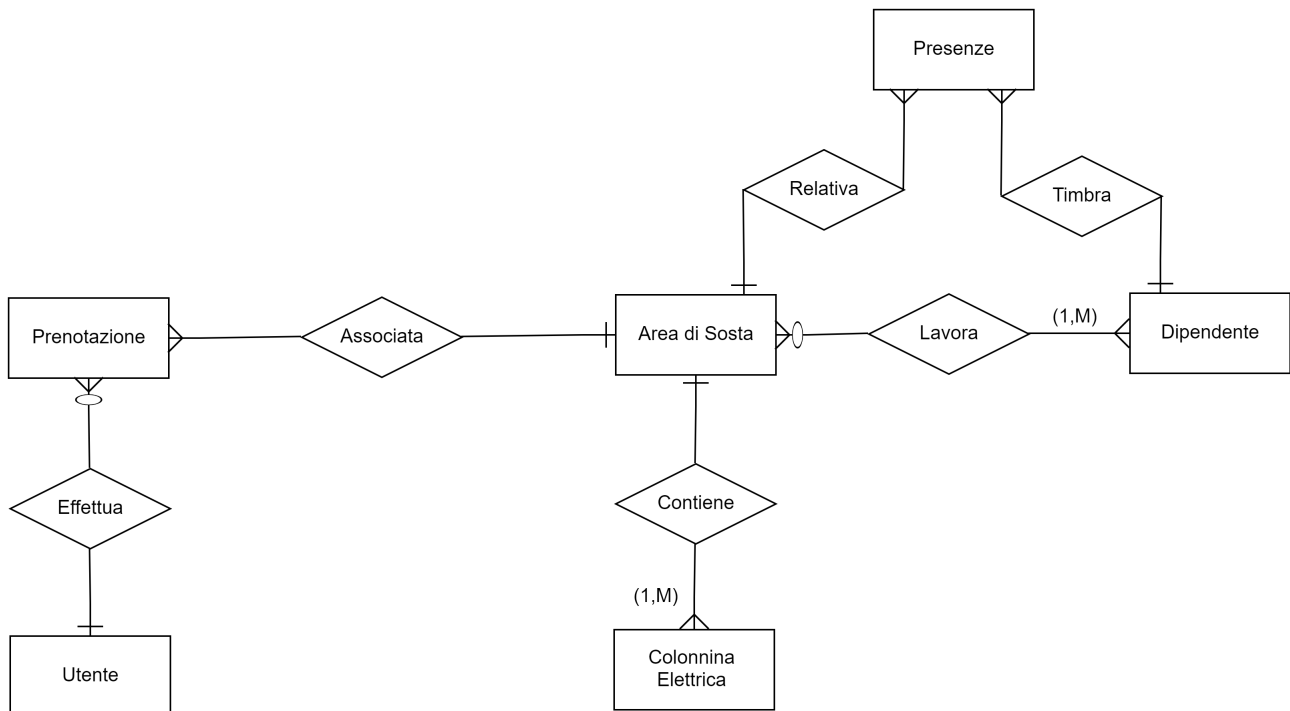
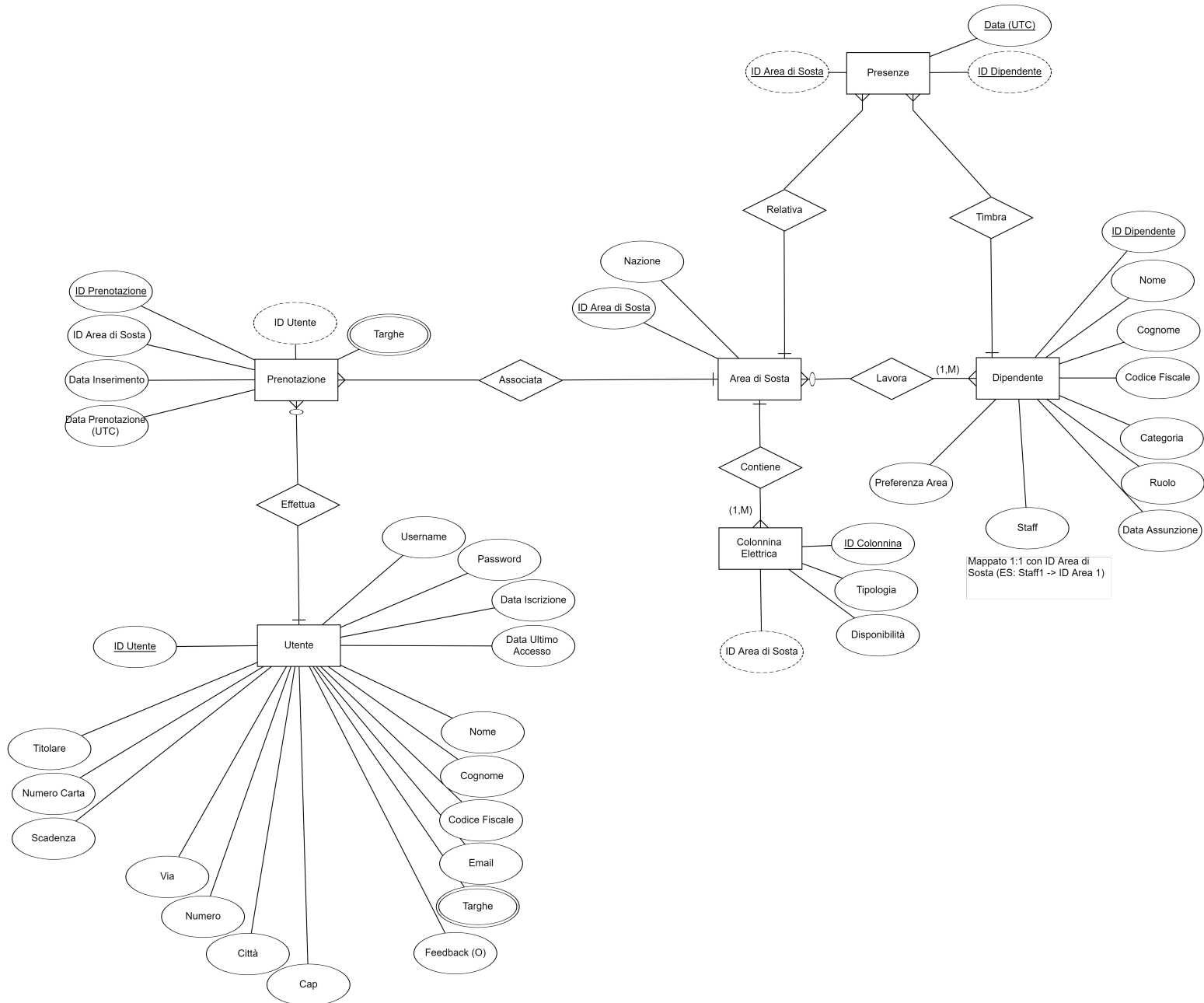


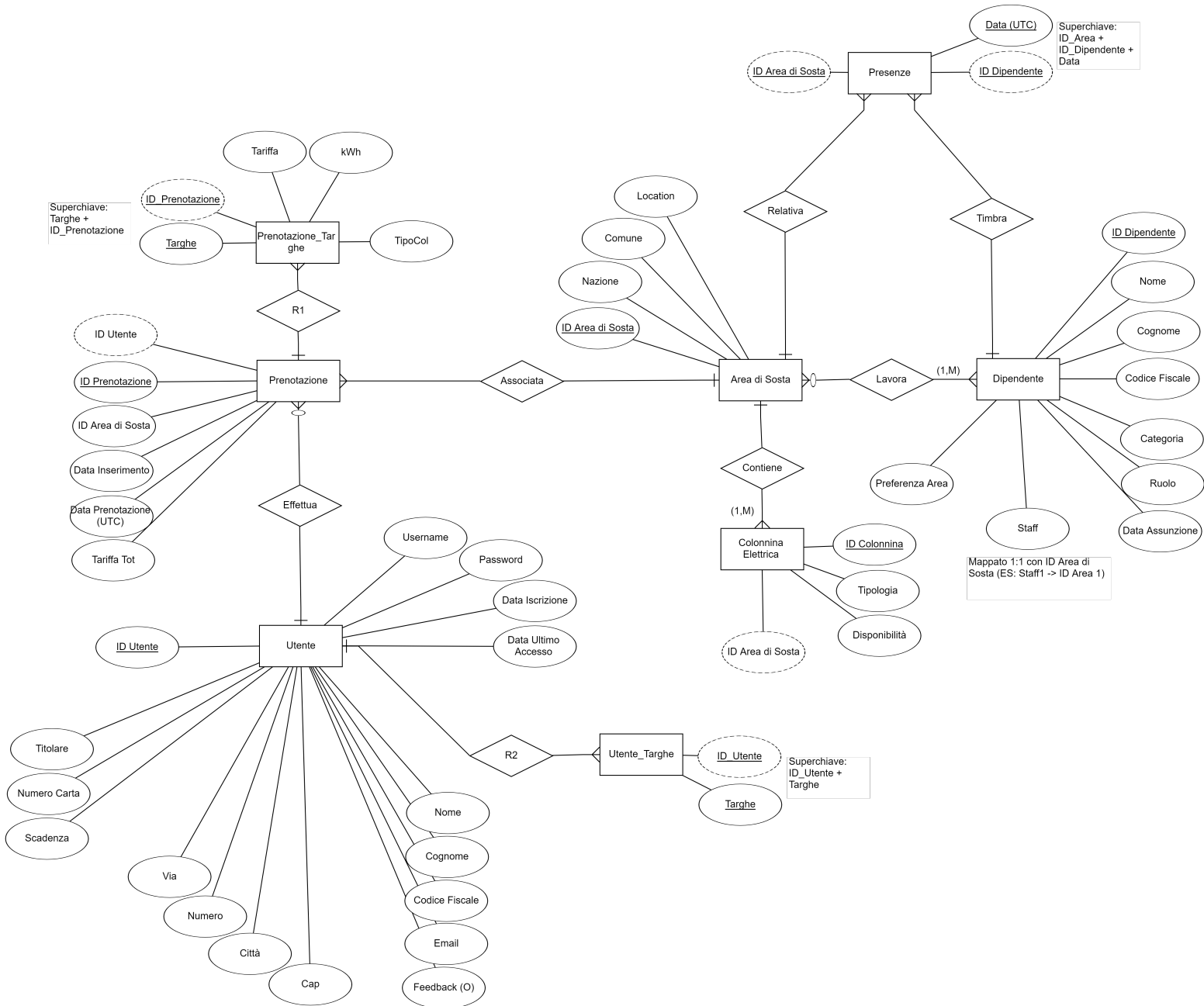
Diagramma ER contenente le entità principali con le relazioni e vincoli esatti.

6 Diagramma E-R v2.0



Nel secondo ER abbiamo aggiunto tutti gli attributi per ogni entità. Sia in **Utente** che in **Prenotazione** sono presenti attributi multivalore (**Targhe**) che sono rispettivamente le targhe memorizzate dall'utente nel suo account e le targhe inserite al momento della prenotazione per se stessi o per amici.

7 Diagramma E-R v3.0



In questa versione (finale) abbiamo esploso gli attributi multivalore in due relazioni R1, R2 con le loro entità. In aggiunta alcune entità presentano attributi aggiuntivi come la tariffa e il kWh (Prenotazione Targhe) oppure Location (Area di Sosta) che rappresenta le coordinate di tipo POINT dell'area.

8 Dizionario dei Dati ed elenco dei vincoli

8.1 Dizionario dei Dati

Entità	Attributi	Descrizione	Identificatori
Area di Sosta	ID Area, Nazione, Comune, Location	Luogo fisico nel quale sono presenti le colonnine elettriche per la ricarica	ID Area
Colonnina Elettrica	ID Colonnina, Tipologia, Disponibilità, ID Area	Dispositivo al quale si connettono le auto per la ricarica elettrica	ID Colonnina
Dipendente	ID_Dipendente, Nome, Cognome, Codice Fiscale, Categoria, Ruolo, Data Assunzione, Staff, Preferenza Area	Persona assunta dall'azienda	ID_Dipendente
Prenotazione	ID Prenotazione, Data Inserimento, Data Prenotazione, ID Area, ID Utente, Tariffa Tot.	Documento di sintesi che attesta la prenotazione, utile a identificare una transazione	ID Prenotazione
Prenotazione Targhe	ID_Prenotazione, Targhe, TipoCol, kWh, Tariffa	Tabella di mappatura tra prenotazione e targhe nella prenotazione	ID Prenotazione, Targhe
Presenze	Data Presenza, ID Area, ID Dipendente	Storico delle presenze giornaliere dei dipendenti	Data Presenza
Utente	ID_Utente, Nome, Cognome, Codice Fiscale, Email, Numero di Carta, Scadenza, Titolare, Via, Numero, Città, CAP, Feedback, Username, Password, Data iscrizione, Data ultimo accesso	Utilizzatore finale del servizio di prenotazione colonnine elettriche	ID_Utente
Utente Targhe	ID Utente, Targhe	Tabella di mappatura tra utente e targhe memorizzate	ID Utente, Targhe

8.2 Elenco dei vincoli

Associazione	Descrizione	Entità Coinvolte
R2	Creazione della relazione tra Utente_Targhe e Utente	Utente, Utente_Targhe
Effettua	Utente all'atto di creazione di una nuova prenotazione	Utente, Prenotazione
R1	Creazione della relazione tra Prenotazione e Prenotazione_Targhe	Prenotazione, Prenotazione_Targhe
Associata	Relazione tra una prenotazione e la relativa area di sosta	Prenotazione, Area di Sosta
Contiene	Associa una specifica colonnina elettrica ad un'area di sosta	Area di Sosta, Colonnina Elettrica
Relativa	Associa la presenza di un dipendente in un'area durante una giornata	Area di Sosta, Presenze
Timbra	Associa l'atto di timbrare il badge da parte dei dipendenti creando una presenza	Dipendente, Presenze
Lavora	Associa il dipendente all'area di sosta dove lavora	Dipendente, Area di Sosta

9 Tavola delle Operazioni e Tavola dei Volumi

9.1 Tavola delle Operazioni (Frequenza e Accessi)

Tabella 2: Tavola delle Operazioni

#	Operazione	Tipo di Operazione	Tipo di Accesso	Frequenza
1	Inserisci utente nel database	Interattiva	Scrittura	50/giorno
2	Inserisci area di ricarica	Interattiva	Scrittura	1/bimestre
3	Inserisci colonnina elettrica	Interattiva	Scrittura	15/mese
4	Assunzione nuovo dipendente	Interattiva	Scrittura	5/bimestre
5	Inserimento nuova prenotazione	Interattiva	Scrittura	200/giorno
6	Registrazione presenza dipendente	Interattiva	Scrittura	300/giorno
7	Area maggior numero di prenotazioni nell'anno	Batch	Lettura	1/settimana
8	Capienza totale per area di sosta	Batch	Lettura	300/giorno
9	Comporre nuovo staff con preferenza area di sosta	Interattiva	Lettura/Scrittura	10/giorno
10	Calcolare numero medio di auto per area di sosta	Batch	Lettura	1/mese
11	Trovare i nomi dei dipendenti che non hanno mai lavorato per due giorni consecutivi nella stessa area	Batch	Lettura	2/settimana
12	Trovare le aree su cui non si sono mai prenotate più di 100 auto	Batch	Lettura	1/mese
13	Trovare le aree con il maggior numero di auto prenotate	Batch	Lettura	1/mese
14	Inserimento preferenza area da gestire (dipendente)	Interattiva	Scrittura	1/trimestre
15	Trovare il numero di colonnine usate (in media) per area dato il tipo	Batch	Lettura	1/mese
16	Trovare l'area più utilizzata da ciascun utente	Batch	Lettura	1/mese
17	Inserimento feedback da parte di un utente	Interattiva	Scrittura	50/mese
18	Rimozione utente dal database	Interattiva	Scrittura	20/mese
19	Calcolare la media di ricarica per utente nel mese precedente	Batch	Lettura	1/mese
20	Trovare i dipendenti che hanno servito più di un'area di sosta nelle ultime 4 settimane	Batch	Lettura	3/mese

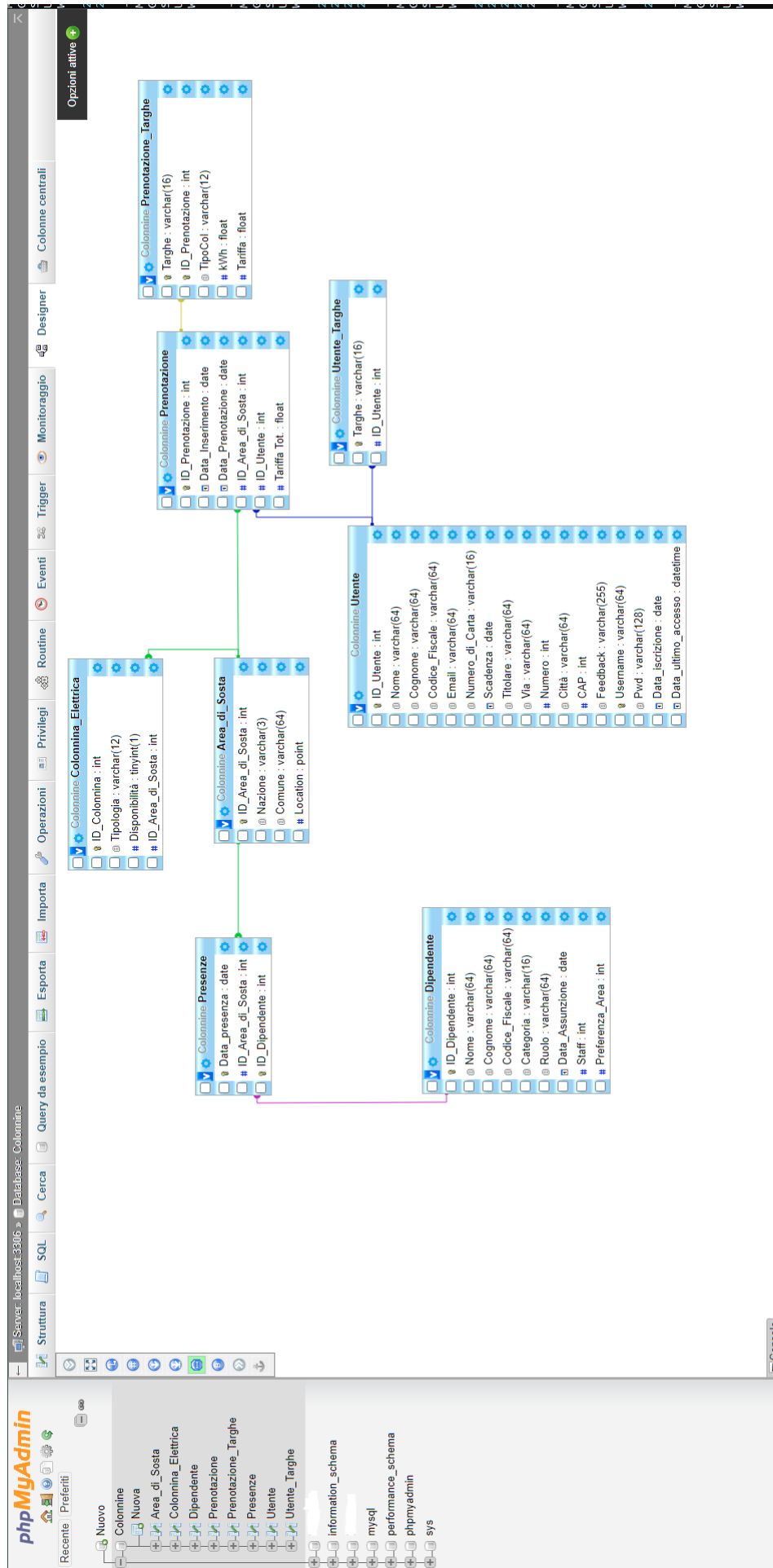
9.2 Tavola dei Volumi

Concetto	Tipo	Volume
Area di Sosta	Entità	1000
Colonnina Elettrica	Entità	10.000
Dipendente	Entità	2000
Prenotazione	Entità	100.000
Presenze	Entità	50.000
Utente	Entità	10.000
Effettua	Relazione	100.000
Associata	Relazione	100.000
Contiene	Relazione	10.000
Relativa	Relazione	100.000
Timbra	Relazione	50.000
Lavora	Relazione	2000

10 Modello Logico

Entità	Attributi	Foreign Key
Area di Sosta	ID_Area <i>int</i> , Nazione <i>varchar(3)</i> , Comune <i>varchar(64)</i> , Location <i>point</i>	
Colonnina Elettrica	ID_Colonnina <i>int</i> , Tipologia <i>varchar(12)</i> , Disponibilità <i>boolean</i> , ID_Area <i>int</i>	ID_Area <i>int</i>
Dipendente	ID_Dipendente <i>int</i> , Nome <i>varchar(64)</i> , Cognome <i>varchar(64)</i> , Codice_Fiscale <i>varchar(64)</i> , Categoria <i>varchar(16)</i> , Ruolo <i>varchar(64)</i> , Data_Assunzione <i>date</i> , Staff <i>int</i> , Preferenza_Area <i>int</i>	
Prenotazione	ID_Prenotazione <i>int</i> , Data_Inserimento, Data_Prenotazione, ID_Area <i>int</i> , ID_Utente <i>int</i> , Tariffa_Tot. <i>float</i>	ID_Area <i>int</i> , ID_Utente <i>int</i>
Prenotazione_Targhe	ID_Prenotazione <i>int</i> , Targhe <i>varchar(16)</i> , TipoCol <i>varchar(12)</i> , kWh <i>float</i> , Tariffa <i>float</i>	ID_Prenotazione <i>int</i>
Presenze	Data_Presenza <i>date</i> , ID_Area <i>int</i> , ID_Dipendente <i>int</i>	ID_Area <i>int</i> , ID_Dipendente <i>int</i>
Utente	ID_Utente <i>int</i> , Nome <i>varchar(64)</i> , Cognome <i>varchar(64)</i> , Codice_Fiscale <i>varchar(64)</i> , Email <i>varchar(64)</i> , Numero_di_carta <i>varchar(64)</i> , Scadenza <i>date</i> , Titolare <i>varchar(64)</i> , Via <i>varchar(64)</i> , Numero <i>int</i> , Città <i>varchar(64)</i> , CAP <i>int</i> , Feedback <i>varchar(255)</i> , Username <i>varchar(64)</i> , Pwd <i>varchar(128)</i> , Data_iscrizione <i>date</i> , Data_ultimo_accesso <i>datetime</i>	
Utente_Targhe	ID_Utente <i>int</i> , Targhe <i>varchar(16)</i>	ID_Utente <i>int</i>

11 Modello Fisico



12 Codice SQL

12.1 Sorgente

Il codice Sorgente del Modello Fisico è reperibile a questo [repository Github](#)

12.1.1 Op. 1

```
1 INSERT INTO `Utente` (`ID_Utente`, `Nome`, `Cognome`, `Codice_Fiscale`, `Email`,  
  ↳ `Numero_di_Carta`, `Scadenza`, `Titolare`, `CAP`, `Via`, `Numero`, `Città`,  
  ↳ `Feedback`, `Username`, `Pwd`, `Data_iscrizione`, `Data_ultimo_accesso`)  
  ↳ VALUES (NULL, 'Janina', 'Engel', 'JNNNGL44R48A952M',  
  ↳ 'JaninaEngel@jourrapide.com', '4532497863930373', '2024-11-30', 'Janina  
  ↳ Engel', '12081', 'Via Francesco Saverio Correrà', '147', 'Beinette', NULL,  
  ↳ 'Hemed1979', SHA1('passwordEsempio2022!'), '2021-12-11', '2021-12-21  
  ↳ 10:34:10');
```

12.1.2 Op. 2

```
1 INSERT INTO `Area_di_Sosta` (`ID_Area_di_Sosta`, `Nazione`)  
2 VALUES (1, 'ITA')
```

12.1.3 Op. 3

```
1 INSERT INTO `Colonnina_Elettrica` (`ID_Colonnina`, `Tipologia`, `Disponibilità`,  
  ↳ `ID_Area_di_Sosta`) VALUES (1, 'NORMAL', 1, 1)
```

12.1.4 Op. 4

```
1 INSERT INTO `Dipendente` (`ID_Dipendente`, `Nome`, `Cognome`, `Codice_Fiscale`,  
  ↳ `Categoria`, `Ruolo`, `Data_Assunzione`, `Staff`, `Preferenza_Area`)  
2 VALUES (1, 'Mario', 'Rossi', 'RSSMRA80A01H501U', 'Operaio', 'Spazzino',  
  ↳ '2021-11-17', 1, 2)
```

12.1.5 Op. 5

```
1 INSERT INTO `Prenotazione` (`ID_Prenotazione`, `Data_Inserimento`,  
  ↳ `Data_Prenotazione`, `ID_Area_di_Sosta`, `ID_Utente`)  
2 VALUES (1, '2022-01-02', '2022-01-03', 2, 1),
```

12.1.6 Op. 6

```
1 INSERT INTO `Presenze` (`Data_presenza`, `ID_Area_di_Sosta`, `ID_Dipendente`)  
2 VALUES ('2021-11-09', 1, 5),
```

12.1.7 Op. 7

```
1 SELECT `ID_Area_di_Sosta` AS `Area`, COUNT(*) AS `N. Prenotazioni`  
2 FROM `Prenotazione`  
3 WHERE YEAR(Data_Prenotazione) = YEAR(CURRENT_DATE)  
4 GROUP BY `ID_Area_di_Sosta` ORDER BY COUNT(*) DESC
```

12.1.8 Op. 8

```
1 SELECT COUNT(*) AS `Capienza Totale`, `ID_Area_di_Sosta` AS `AREA`,  
  ↳ `Disponibilità`  
2 FROM `Colonnina_Elettrica`  
3 WHERE `Disponibilità` = 1  
4 GROUP BY AREA
```

12.1.9 Op. 9

PSEUDOCODICE Python:

```
1 def Set_Personale():  
2     Personale = list()  
3     for ruolo in lista_ruoli_necessari_in_ogni_area_di_ricarica:  
4         Personale_Area = list()  
5         for ID_Area in Lista_Aree_di_ricarica:  
6             lista_ID = Query_SQL01          #Nella query dovrà essere cambiata la  
              ↳ Preferenza_Area e il Ruolo con quelli presi dal ciclo for  
7             if lista_ID is empty:  
8                 lista_ID = Query_SQL02  
9             if lista_ID is empty:          # I DIPENDENTI SONO FINITI  
10                return (Personale, False) # Restituisce una tupla dove il primo  
              ↳ elemento è una lista di liste con il personale per ogni area  
              ↳ in ordine crescente e il secondo elemento indica se l'ultima  
              ↳ area settata è al completo  
11                Personale_Area.append(lista_ID[0])  
12                Query_SQL03 #Assegna nel DB il ruolo al dipendente  
13                Personale.append(Personale_Area)  
14    return (Personale, True)              # Restituisce una tupla dove il primo  
      ↳ elemento è una lista di liste con il personale per ogni area in ordine  
      ↳ crescente e il secondo elemento indica se l'ultima area settata è al  
      ↳ completo
```

Query_SQL01 d'esempio con vincoli 'Preferenza_Area' = '2' e 'Ruolo' = 'Capo Staff'

```
1 SELECT `ID_Dipendente`  
2 FROM `Dipendente`  
3 WHERE `Staff` IS NULL AND `Preferenza_Area` IS NOT NULL AND `Preferenza_Area` =  
  ↳ '2' AND `Ruolo` = 'Capo Staff'  
4 ORDER BY `Data_Assunzione` ASC
```

Query_SQL02 d'esempio con vincolo 'Ruolo' = 'Capo Staff'

```
1 SELECT `ID_Dipendente`  
2 FROM `Dipendente`  
3 WHERE `Staff` IS NULL AND `Preferenza_Area` IS NOT NULL AND `Ruolo` = 'Capo  
  ↳ Staff'  
4 ORDER BY `Data_Assunzione` DESC
```

Query_SQL03 d'esempio per settare lo staff dove il dipendente lavorerà.

```
1 UPDATE `Dipendente` SET `Staff` = 'ID_Area' WHERE `Dipendente`.`ID_Dipendente` =  
  ↳ 'lista_ID[0]';
```

12.1.10 Op. 10

Calcolare numero medio di auto presenti in un'area di sosta durante il mese precedente, operazione batch con frequenza 1 volta al mese

```
1 SELECT SUM(cnt) / DATEDIFF(CURRENT_DATE(), (DATE_ADD(CURRENT_DATE(), INTERVAL -1
   ↳ MONTH))) as `Avg. car in month`
2 FROM (SELECT count(*) AS cnt, MONTH(`Data_Prenotazione`)
3       FROM (SELECT `Prenotazione`.`Data_Prenotazione`,
   ↳ `Prenotazione`.`ID_Prenotazione`, `Prenotazione_Targhe`.`Targhe`
4       FROM `Prenotazione`
5       RIGHT JOIN `Prenotazione_Targhe`
6       ON `Prenotazione`.`ID_Prenotazione` =
   ↳ `Prenotazione_Targhe`.`ID_Prenotazione`) as a
7 WHERE MONTH(Data_Prenotazione) = MONTH(DATE_ADD(CURRENT_DATE(), INTERVAL -1
   ↳ MONTH))
8 GROUP BY ID_Prenotazione) as b
```

12.1.11 Op. 11

```
1 SELECT `ID_Dipendente`, `Nome`, `Cognome`
2 FROM (SELECT *,
3       LEAD(`ID_Area_di_Sosta`) OVER (partition by `ID_Dipendente` order by
   ↳ `Data_presenza`) as total_1
4 FROM (SELECT `Presenze`.`ID_Dipendente`, `Presenze`.`Data_presenza`,
   ↳ `Presenze`.`ID_Area_di_Sosta`, `Dipendente`.`Nome`,
   ↳ `Dipendente`.`Cognome`
5 FROM `Presenze`
6 RIGHT JOIN `Dipendente`
7 ON `Presenze`.`ID_Dipendente` = `Dipendente`.`ID_Dipendente`) AS a
8 WHERE `ID_Area_di_Sosta` IS NOT NULL
9 ORDER BY `ID_Dipendente`, `Data_presenza` ASC) as s
10 WHERE `total_1` IS NOT NULL AND `total_1` != `ID_Area_di_Sosta`
11 GROUP BY `ID_Dipendente`
```

12.1.12 Op. 12

```
1 SELECT *
2 FROM (SELECT `ID_Area_di_Sosta` AS `Area`, count(*) AS nAuto
3       FROM (SELECT `Prenotazione`.`ID_Prenotazione`,
   ↳ `Prenotazione_Targhe`.`Targhe`, `Prenotazione`.`ID_Area_di_Sosta`
4       FROM `Prenotazione`
5       RIGHT JOIN `Prenotazione_Targhe`
6       ON `Prenotazione`.`ID_Prenotazione` =
   ↳ `Prenotazione_Targhe`.`ID_Prenotazione`) as a
7 GROUP BY ID_Area_di_Sosta) as b
8 WHERE nAuto < 100
```

12.1.13 Op. 13

```
1 SELECT *
2 FROM (SELECT `ID_Area_di_Sosta` AS `Area`, count(*) AS nAuto
3      FROM (SELECT `Prenotazione`.`ID_Prenotazione`,
4            ↪ `Prenotazione_Targhe`.`Targhe`, `Prenotazione`.`ID_Area_di_Sosta`
5            FROM `Prenotazione`
6            RIGHT JOIN `Prenotazione_Targhe`
7            ON `Prenotazione`.`ID_Prenotazione` =
8            ↪ `Prenotazione_Targhe`.`ID_Prenotazione`) as a
9      GROUP BY ID_Area_di_Sosta) as b
10 ORDER BY nAuto DESC
```

12.1.14 Op. 14

```
1 UPDATE `Dipendente`
2 SET `Preferenza_Area` = '2'
3 WHERE `Dipendente`.`ID_Dipendente` = N;
```

12.1.15 Op. 15

```
1 SELECT COUNT(`Prenotazione_Targhe`.`TipoCol`) AS `nCol`,
2      ↪ COUNT(`Prenotazione_Targhe`.`TipoCol`)/
3      ↪ DATEDIFF(CURRENT_DATE(),(DATE_ADD(CURRENT_DATE(),INTERVAL -1 MONTH))) AS `Avg`
4      ↪ Col`, `Prenotazione`.`ID_Area_di_Sosta`, `Prenotazione_Targhe`.`TipoCol`
5 FROM `Prenotazione`
6 RIGHT JOIN `Prenotazione_Targhe`
7 ON `Prenotazione`.`ID_Prenotazione` = `Prenotazione_Targhe`.`ID_Prenotazione`
8 WHERE MONTH(Data_Prenotazione) = MONTH(DATE_ADD(CURRENT_DATE(),INTERVAL -1
9      ↪ MONTH))
10 GROUP BY `ID_Area_di_Sosta`, `TipoCol`
11 ORDER BY `ID_Area_di_Sosta`, `TipoCol`
```

12.1.16 Op. 16

```
1 SELECT COUNT(`ID_Area_di_Sosta`) AS 'Volte', `ID_Area_di_Sosta`, `ID_Utente`
2 FROM (SELECT `Prenotazione`.`ID_Area_di_Sosta`, `Prenotazione`.`ID_Utente`,
3      ↪ `Area_di_Sosta`.`Comune`
4      FROM `Prenotazione`
5      LEFT JOIN `Area_di_Sosta` ON `Prenotazione`.`ID_Area_di_Sosta` =
6      ↪ `Area_di_Sosta`.`ID_Area_di_Sosta`) AS a
7 GROUP BY `ID_Utente`, `ID_Area_di_Sosta`
8 ORDER BY `ID_Utente` ASC
```

12.1.17 Op. 17

```
1 UPDATE `Utente`
2 SET `Feedback` = 'FEEDBACK'
3 WHERE `Utente`.`ID_Utente` = N;
```

12.1.18 Op. 18

```
1 DELETE FROM `Utente`
2 WHERE `Utente`.`ID_Utente`;
```

12.1.19 Op. 19

```
1 SELECT `ID_Utente`, COUNT(`ID_Utente`) AS `N. Ricariche effettuate`,
   ↪ COUNT(`ID_Utente`)/DATEDIFF(CURRENT_DATE(),(DATE_ADD(CURRENT_DATE(),INTERVAL
   ↪ -1 MONTH))) AS `AVG`
2 FROM (SELECT `Prenotazione`.`Data_Prenotazione`, `Prenotazione`.`ID_Utente`,
   ↪ `Prenotazione_Targhe`.`Targhe`, `Prenotazione_Targhe`.`kWh`
3       FROM `Prenotazione`
4       RIGHT JOIN `Prenotazione_Targhe`
5       ON `Prenotazione`.`ID_Prenotazione` =
   ↪ `Prenotazione_Targhe`.`ID_Prenotazione`
6       WHERE `kWh` IS NOT NULL) AS a
7 WHERE MONTH(Data_Prenotazione) = MONTH(DATE_ADD(CURRENT_DATE(),INTERVAL -1
   ↪ MONTH))
8 GROUP BY `ID_Utente`
```

12.1.20 Op. 20

```
1 SELECT `ID_Dipendente` AS `ID_Dipendente (che ha servito aree di sosta differenti
   ↪ nelle 4 ultime settimane)`
2 FROM (SELECT `ID_Dipendente`, COUNT(DISTINCT `ID_Area_di_Sosta`) AS `N. aree
   ↪ servite`
3       FROM (SELECT `ID_Dipendente`, `Data_presenza`, `ID_Area_di_Sosta`
4             FROM `Presenze`
5             WHERE `Data_presenza`>= DATE_ADD(CURRENT_DATE(),INTERVAL -4 WEEK)
   ↪ AND `ID_Area_di_Sosta` IS NOT NULL
6             ORDER BY `ID_Dipendente`) as a
7       GROUP BY `ID_Dipendente`) AS b
8 WHERE `N. aree servite` > 1
```
