

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	Diagramma E-R	3
5	Diagramma E-R v2.0	4
6	Diagramma E-R v3.0	5
7	Tavola delle Operazioni e Tavola dei Volumi	6
8	Modello Logico	7
9	Modello Fisico	9
10	Codice SQL	10
10.1	Sorgente	10
10.1.1	Op. 1	10
10.1.2	Op. 2	10
10.1.3	Op. 3	10
10.1.4	Op. 4	10
10.1.5	Op. 5	10
10.1.6	Op. 6	10
10.1.7	Op. 7	10
10.1.8	Op. 8	11
10.1.9	Op. 9	11
10.1.10	Op. 10	11
10.1.11	Op. 11	12
10.1.12	Op. 12	12
10.1.13	Op. 13	12
10.1.14	Op. 14	12
10.1.15	Op. 15	13
10.1.16	Op. 16	13
10.1.17	Op. 17	13
10.1.18	Op. 18	13
10.1.19	Op. 19	13
10.1.20	Op. 20	14

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 Diagramma E-R

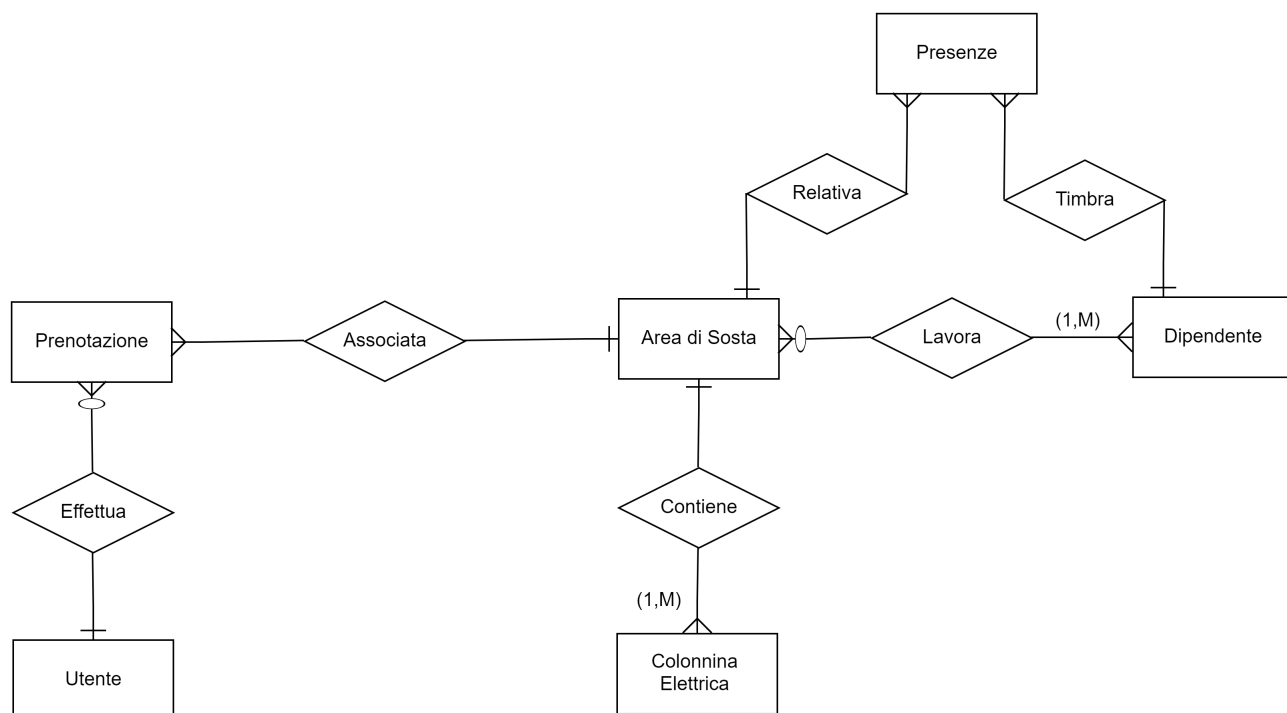
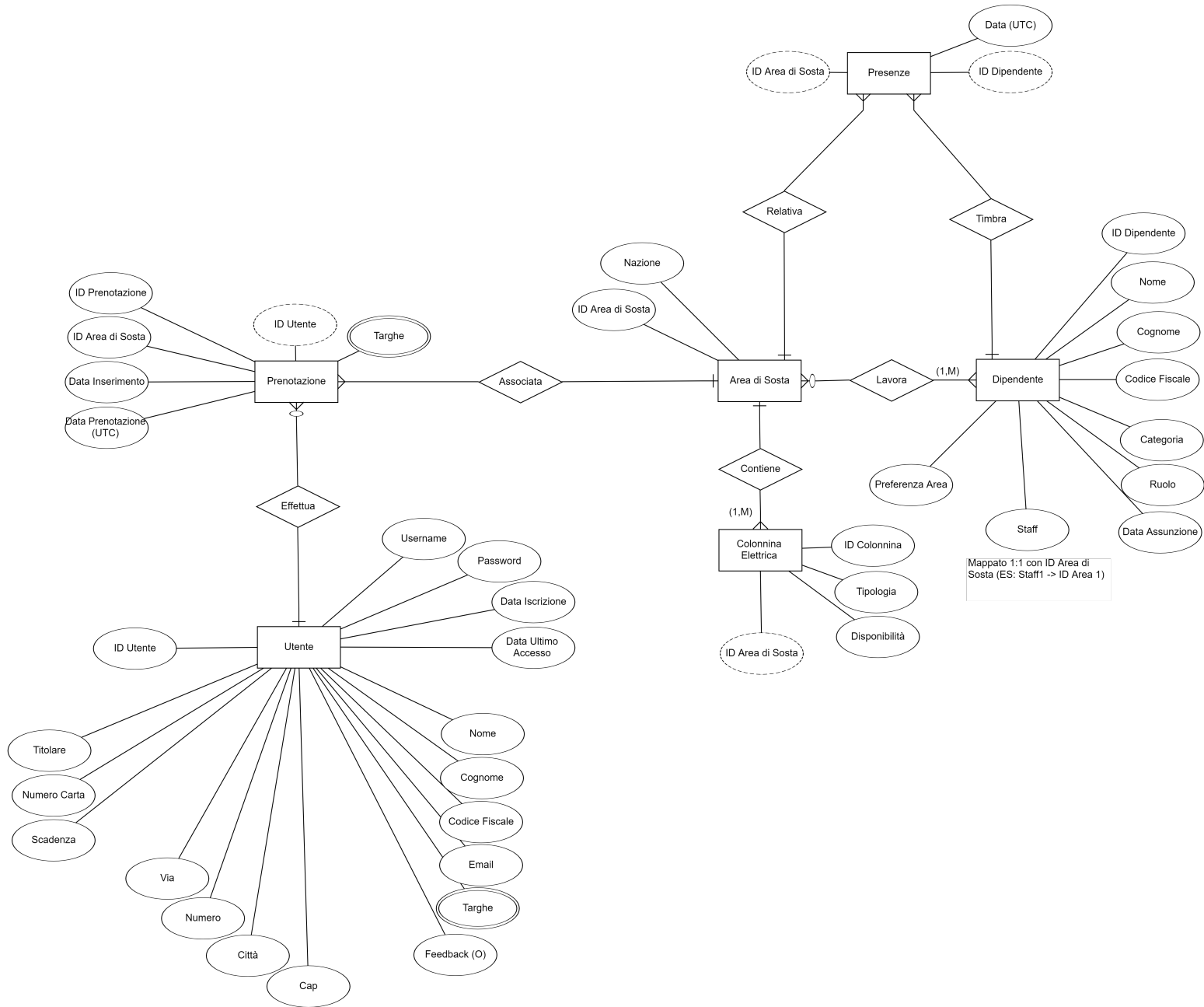


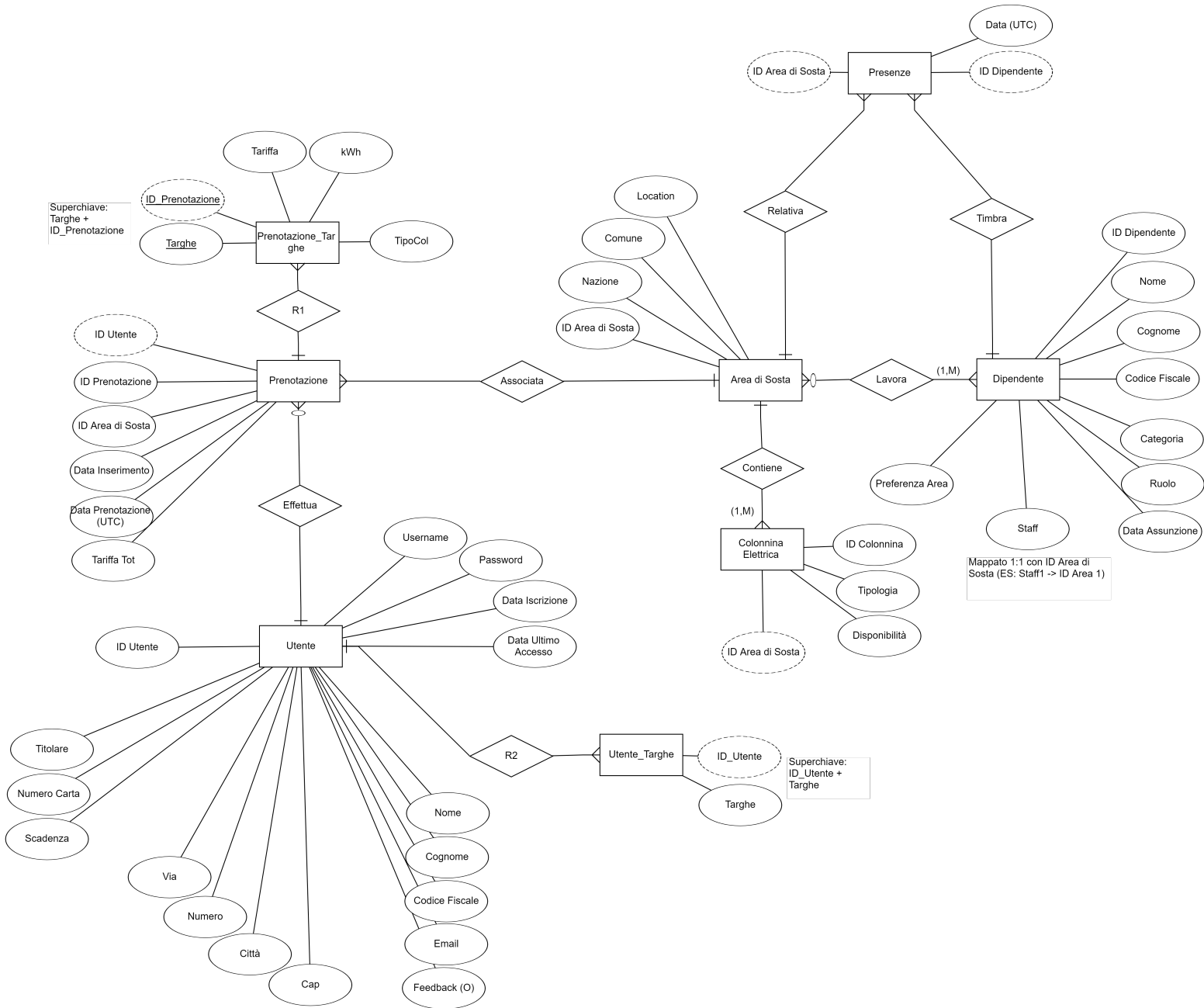
Diagramma ER contenente le entità principali con le relazioni e vincoli esatti.

5 Diagramma E-R v2.0



Nel secondo ER abbiamo aggiunto tutti gli attributi per ogni entità.

6 Diagramma E-R v3.0



7 Tavola delle Operazioni e Tavola dei Volumi

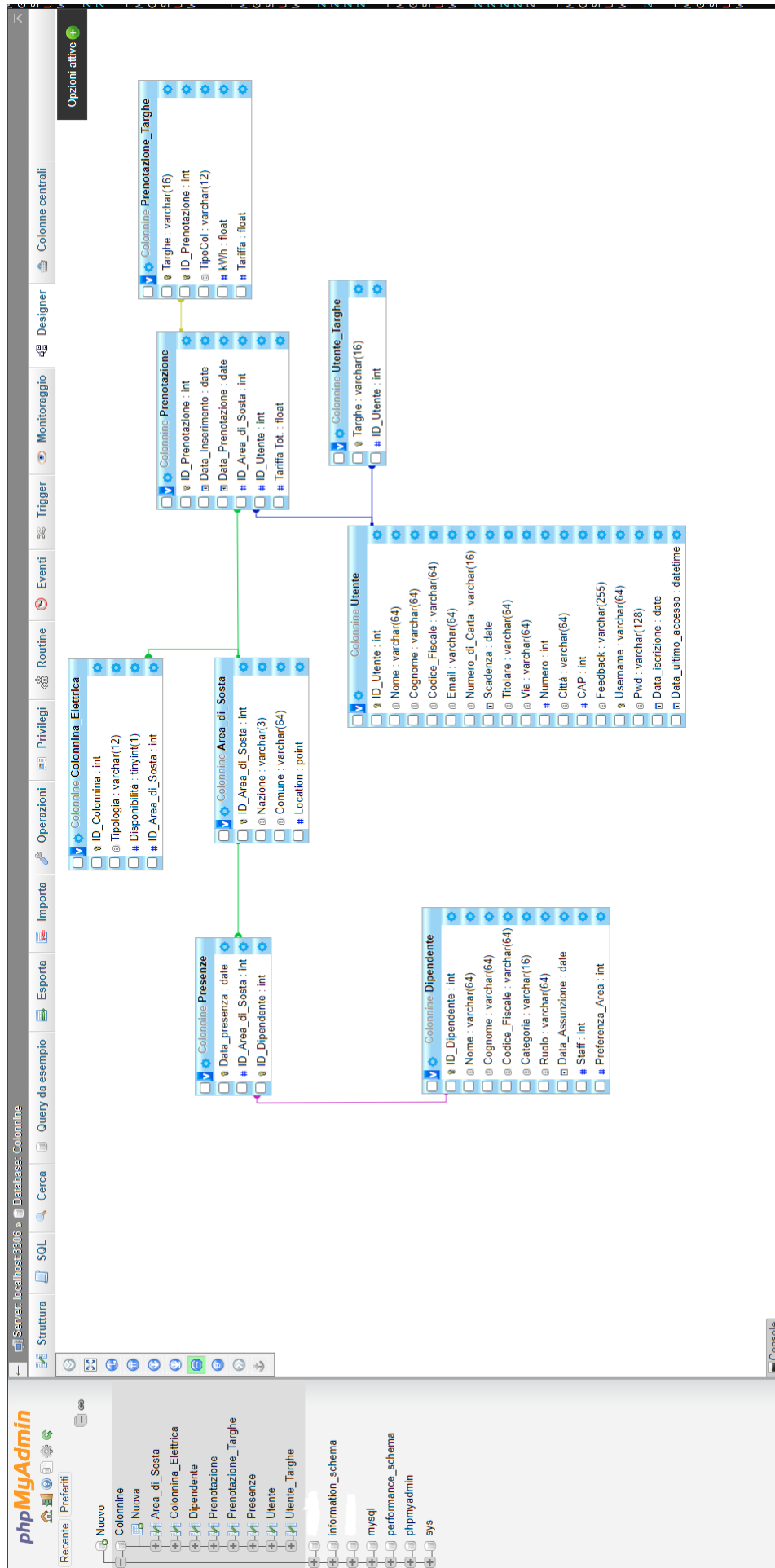
Tabella 2: Tavola delle Operazioni

#	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	1000/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	Interattiva	Lettura	1/mese
13	Trovare le aree con il maggior numero di auto prenotate	Interattiva	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	Interattiva	Lettura	3/mese

8 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_Targhe	ID_Utente <i>int</i> , Targhe <i>varchar(16)</i>	ID_Utente <i>int</i>

9 Modello Fisico



10 Codice SQL

10.1 Sorgente

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

10.1.1 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');
```

10.1.2 Op. 2

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

10.1.3 Op. 3

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

10.1.4 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)
```

10.1.5 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),
```

10.1.6 Op. 6

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

10.1.7 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
```

10.1.8 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
```

10.1.9 Op. 9

10.1.10 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
```

DEBUG:

```
1 SELECT COUNT(cnt) AS `Tot. Giorni`, SUM(cnt) as `Tot. Auto`, SUM(cnt) /  
  ↳ COUNT(cnt) as `Media` from (SELECT count(*) AS cnt,  
  ↳ MONTH(`Data_Prenotazione`) FROM (SELECT `Prenotazione`.`Data_Prenotazione`,  
  ↳ `Prenotazione`.`ID_Prenotazione`, `Prenotazione_Targhe`.`Targhe` FROM  
  ↳ `Prenotazione` RIGHT JOIN `Prenotazione_Targhe` ON  
  ↳ `Prenotazione`.`ID_Prenotazione` = `Prenotazione_Targhe`.`ID_Prenotazione`)as  
  ↳ a WHERE MONTH(Data_Prenotazione) = MONTH(CURRENT_DATE) GROUP BY  
  ↳ ID_Prenotazione) as b
```

Giorni nel mese corrente

```
1 SELECT DATEDIFF((DATE_ADD(CURRENT_DATE(),INTERVAL 1 MONTH)),CURRENT_DATE());
```

Giorni nel mese appena passato

```
1 SELECT DATEDIFF(CURRENT_DATE(),(DATE_ADD(CURRENT_DATE(),INTERVAL -1 MONTH)))
```

```
1 SELECT count(*) AS cnt, `Data_Prenotazione`, `ID_Area_di_Sosta`, `ID_Prenotazione`  
2 FROM (SELECT `Prenotazione`.`Data_Prenotazione`,  
  ↳ `Prenotazione`.`ID_Area_di_Sosta`, `Prenotazione`.`ID_Prenotazione`,  
  ↳ `Prenotazione_Targhe`.`Targhe`  
3   FROM `Prenotazione`  
4   RIGHT JOIN `Prenotazione_Targhe`  
5   ON `Prenotazione`.`ID_Prenotazione` =  
  ↳ `Prenotazione_Targhe`.`ID_Prenotazione`) as a
```

```

6 WHERE MONTH(Data_Prenotazione) = MONTH(DATE_ADD(CURRENT_DATE(),INTERVAL -1
  ↳ MONTH))
7 GROUP BY ID_Prenotazione

```

10.1.11 Op. 11

```

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

```

10.1.12 Op. 12

```

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 WHERE nAuto < 100

```

10.1.13 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

```

10.1.14 Op. 14

```

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

```

10.1.15 Op. 15

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

10.1.16 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`,
   ↪ `Area_di_Sosta`.`Comune`
3 FROM `Prenotazione`
4 LEFT JOIN `Area_di_Sosta` ON `Prenotazione`.`ID_Area_di_Sosta` =
   ↪ `Area_di_Sosta`.`ID_Area_di_Sosta`) AS a
5 GROUP BY `ID_Utente`, `ID_Area_di_Sosta`
6 ORDER BY `ID_Utente` ASC
```

10.1.17 Op. 17

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

10.1.18 Op. 18

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

10.1.19 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`
```

10.1.20 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
```
