

# **Progetto Base di dati**

## ***Gestionale agenzia funebre***

**Rosario Scuderi**

***Università degli studi di Catania - Dipartimento di Matematica e Informatica***

1. Caratteristiche	pag.2
2. Analisi dei requisiti	pag.3
• 2.1 Glossario dei termini	pag.3
• 2.2 Analisi dei dati	pag.3
• 2.3 Operazioni	pag.5
3. Diagramma E-R	pag.6
• 3.1 Schema iniziale	pag.6
• 3.1 Schema Intermedio 1	pag.7
• 3.1 Schema Intermedio 2	pag.8
• 3.1 Schema Intermedio 3	pag.9
• 3.2 Schema finale	pag.10
• 3,3 Tabella entità	pag.11
• 3.4 Tabella relazioni	pag.12
• 3.5 Vincoli	pag.13
• 3.6 Tabella dei volumi	pag.13
• 3.7 Tabella operazioni	pag.14
• 3.8 Traduzione nel modello relazionale	pag.14
4. Implementazione SQL	pag.15
• 4.1 Tabelle	pag.15
• 4.2 Operazioni	pag.18
• 4.3 Diagramma EER	pag.21
5. Esempi di query, guida all'utilizzo e dettagli extra	pag.22-25

# 1. Caratteristiche

Si vuole progettare un database gestionale/informativo per un'agenzia funebre.

La base di dati dovrà tenere traccia delle peculiarità di ogni funerale organizzato dall'agenzia stessa, in modo tale da ottenere una visione globale dei processi, andamenti aziendali e status dell'azienda.

Per ogni funerale l'agenzia mette a disposizione due veicoli, un'auto e un furgone (ogni veicolo può essere usato solo per un funerale al giorno), due impiegati, che si occuperanno della gestione dell'evento stesso, e un'ampia varietà di bare e ornamenti funerari: dunque, sarà sempre necessario verificare la disponibilità di un veicolo o di un impiegato in una determinata data.

Per ogni dipendente si conoscono id, nome, cognome, disponibilità e stipendio mentre, per quanto riguarda le auto, sono noti targa, modello, colore e disponibilità.

Per le bare saranno presi in considerazione la tipologia di legno, che rappresenta la chiave primaria, quantità in magazzino, che verrà aggiornata continuamente, e il costo all'ingrosso dell'intero lotto di ogni tipologia (che può cambiare da un mese all'altro) : nelle info di alcuni funerali saranno richieste informazioni extra riguardanti alcune caratteristiche ornamentali (ad esempio incisioni o oggetti fissati) o particolari richieste da parte del cliente.

Le bare saranno rifornite una volta al mese e il loro prezzo all'ingrosso varia in base alle offerte proposte dal fornitore.

Per ogni funerale bisogna memorizzare: id del funerale che viene rilasciato dal Comune, codice fiscale della salma, data, cliente (per ogni cliente si considerano cognome, numero di telefono e codice fiscale), luogo dello svolgimento, luogo di sepoltura del defunto, costo del servizio, bara utilizzata, auto e furgone utilizzati e dipendenti presenti all' esequie.

I defunti dovranno essere salvati all'interno del database memorizzando i loro dati anagrafici.

Il database dovrà essere in grado di soddisfare alcune richieste da parte dell'utente di tipo informativo come ad esempio: analizzare le entrate e le uscite mensili dell'azienda, controllare quante volte un cliente si è rivolto all'agenzia, verificare la disponibilità di alcuni elementi e altro.

## 2. Analisi dei requisiti

### 2.1 Glossario dei termini

TERMINE	SINONIMO	DESCRIZIONE	CORRELATI
Funerale	Esequie, evento, servizio	Elemento cardine: costituisce la principale fonte di guadagno dell'azienda	Veicoli, dipendenti, bare, cliente, defunto
Veicoli	Auto, furgone	Mezzi di trasporto utilizzati dall'azienda	Funerale
Bare		Oggetto destinato a contenere la salma	Funerale
Dipendenti	Impiegato	Coloro che lavorano per l'agenzia	Funerale
Cliente		Colui che commissiona il servizio	Funerale
Defunto	Salma	Persona deceduta	Funerale

### 2.2 Analisi dei Dati

#### Generale:

Si vuole progettare un database gestionale/informativo per un agenzia funebre.

#### Dati sui funerali:

I funerali sono identificati da un codice fornito dal Comune e ne si conoscono tutti i dettagli a riguardo: veicoli usati, dipendenti presenti, bara scelta dal cliente, luogo dello svolgimento, cliente, dettagli sul defunto, costo e data, dettagli sulla bara e luogo di sepoltura.

**Dati sui veicoli:**

I veicoli costituiscono un elemento fondamentale per l'azienda. I veicoli si suddividono in due gruppi: auto e furgoni. Sono caratterizzati dal numero di targa, marca e colore, inoltre bisogna verificare la loro disponibilità continuamente. Per ogni funerale vengono usati due veicoli (un'auto e un furgone) e ogni veicolo può essere usato solo una volta al giorno.

**Dati sulle bare:**

Le bare sono identificate dal tipo di legno di cui sono fatte, dal prezzo che paga l'azienda al fornitore e dalla quantità in magazzino. Ad ogni funerale sarà associata una ed una sola bara.

**Dati sui dipendenti:**

Ai dipendenti viene affidata la gestione del servizio; Ad ogni funerale si associano due diversi dipendenti che saranno caratterizzati da id, nome, cognome e stipendio. Sarà necessario verificarne la disponibilità continuamente. Ogni dipendente può partecipare a un solo funerale al giorno.

**Dati sui clienti:**

I clienti, nel momento in cui si rivolgeranno all'agenzia, dovranno essere memorizzati all'interno del database, qualora non siano stati già inseriti, considerando cognome, numero di telefono e codice fiscale.

**Dati sui defunti:**

I defunti dovranno essere salvati all'interno del database memorizzando i loro dati anagrafici.

## 2.3 Operazioni

- **Op 1:** Inserimento funerale

*-Durante esecuzione di questa procedura avverranno una serie di operazioni che mirano ad aggiornare alcune tabelle che riferimento a 'funerale':*

- 1. Aggiornare la tabella relativa alle prenotazioni dei veicoli*
- 2. Aggiornare la tabella relativa alla disponibilità degli impiegati*
- 3. Aggiornare le caratteristiche del servizio*
- 4. Inserire I dati del defunto all'interno del database*
- 5. Aggiornare quantità delle bare*

*Sarà anche necessario verificare alcune condizioni:*

- 1. Verificare la disponibilità di un veicolo*
- 2. Verificare la disponibilità di un impiegato*
- 3. Verificare se un cliente è già presente all'interno della banca dati*  
*-Nel caso in cui non sia presente sarà necessario inserirlo*
- 4. Verificare se l'id del funerale inserito è unico*
- 5. Verificare se una tipologia di bara è presente in magazzino*

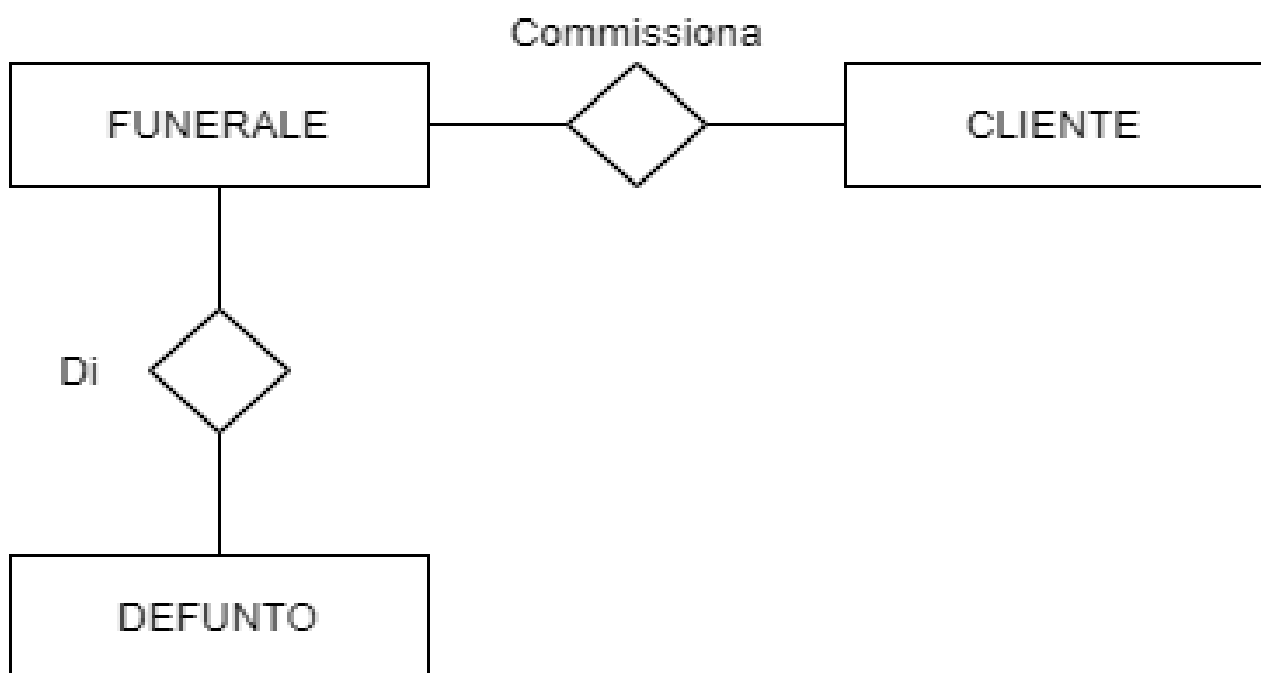
- **Op 2:** Contare quante volte un cliente si è riferito all'azienda.
- **Op 3:** Calcolare le entrate economiche mensili (guadagni dai funerali)
- **Op 4:** Calcolare le uscite mensili dovute ai rifornimenti e agli stipendi
- **Op 5:** Aggiornare recapiti telefonici dei clienti
- **Op 6:** Verificare disponibilità veicolo
- **Op 7:** Verificare la disponibilità degli impiegati
- **Op 8:** Mostrare tutte le caratteristiche di un servizio
- **Op 9:** Rifornire bare

### 3. Diagramma E-R

Per la progettazione del diagramma verrà utilizzato un approccio di tipo TOP-DOWN

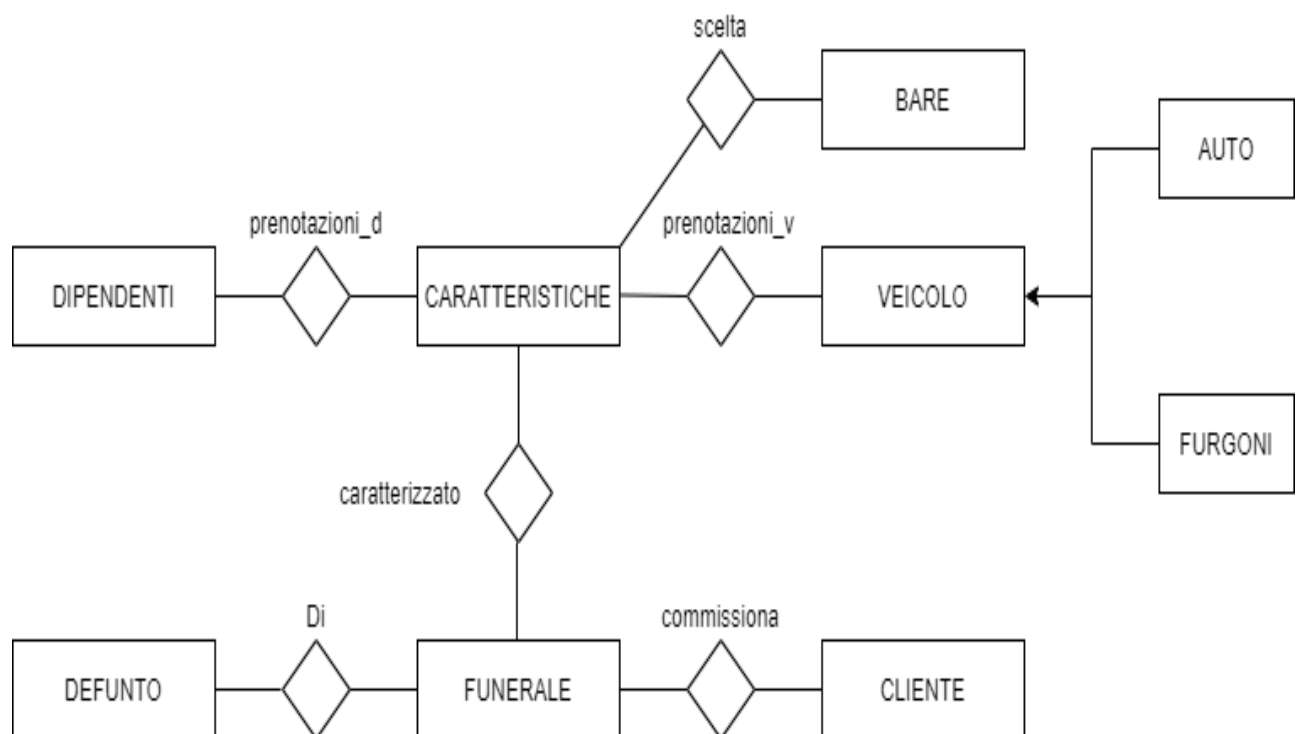
#### 3.1 Schema iniziale

*Rappresenta un'idea generale e minimale dell'intera base di dati.*



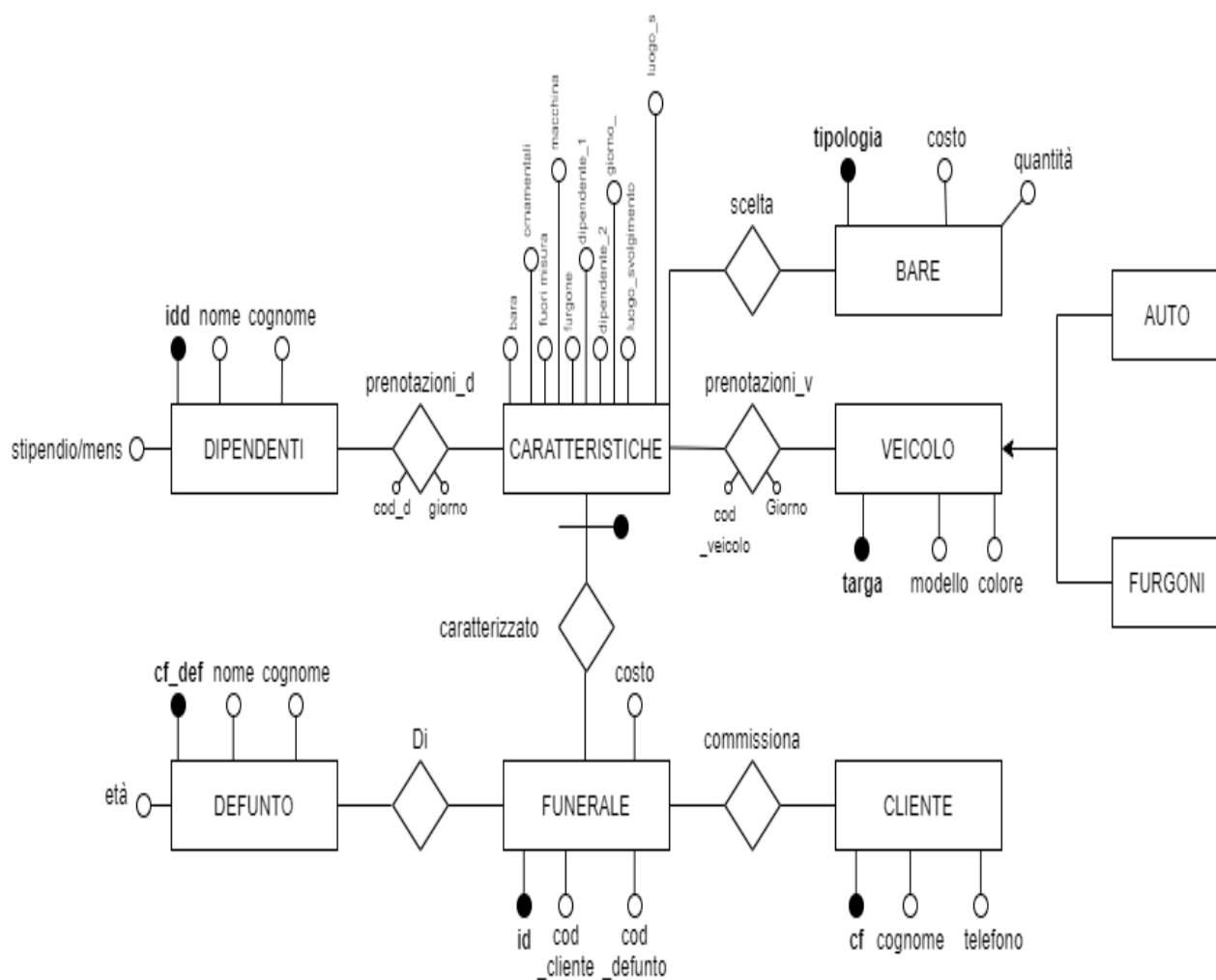
### 3.1 Schema Intermedio 1

*Vengono definite più entità e relazioni*



### 3.1 Schema Intermedio 2

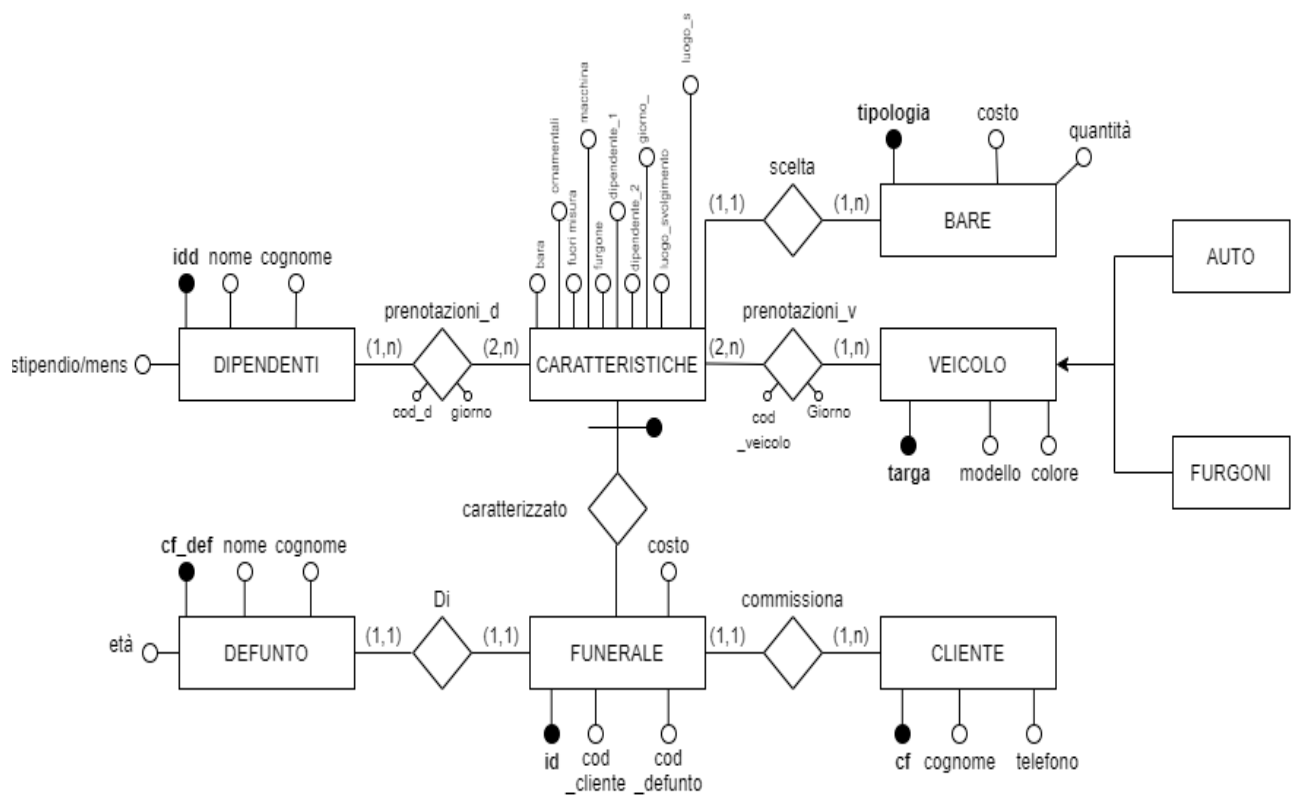
*Si inseriscono i dettagli di ogni elemento*



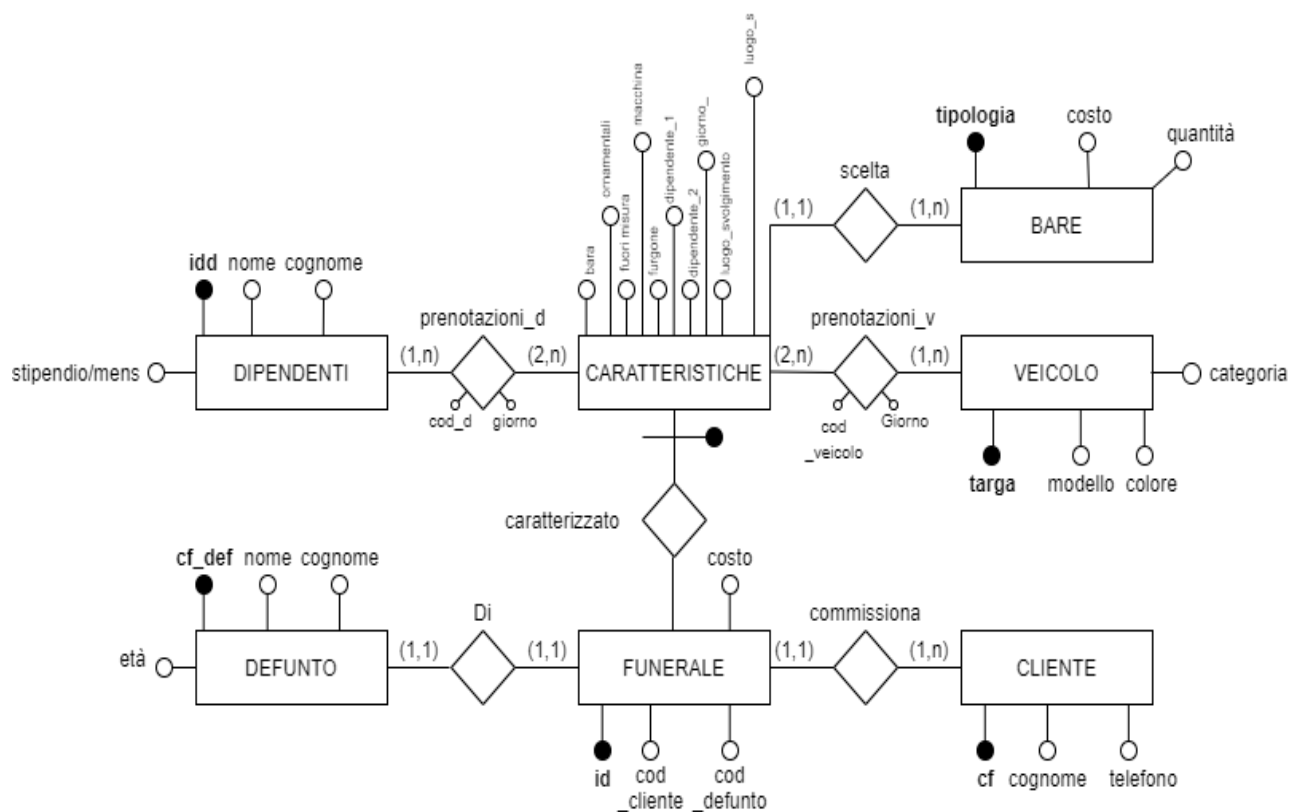


### 3.1 Schema Intermedio 3

*Si inseriscono le cardinalità*



### 3.2 Schema finale (post eliminazione gerarchie)



### 3.3 Entità

ENTITA'	ATTRIBUTI	DESCRIZIONE	PRIMARY_KEY
FUNERALE	ID, COD_CLIENTE, COD_DEFUNTO COSTO	Elemento centrale all'interno del contesto aziendale	ID
DEFUNTO	CF_DEF, NOME, COCOGNOME, ETA'	Persona deceduta per cui si svolge il funerale	CF_DEF
CARATTERISTICHE	BARA, ORNAMENTALI, FUORI_MISURA, MACCHINA, FURGONE, DIPENDENTE_1, DIPENDENTE_2, GIORNO_, LUOGO_SVOLGIMENTO LUOGO_SEPOLTURA	Entità, fortemente connessa a quasi tutte le altre, che contiene tutte le caratteristiche del funerale	ID (funerale)
BARE	TIPOLOGIA, COSTO, QUANTITA'	Oggetto destinato a contenere la salma	TIPOLOGIA
VEICOLO	TARGA, MODELLO, COLORE, CATEGORIA	Mezzi di trasporto utilizzati dall'azienda	TARGA
DIPENDENTI	IDD, NOME, COGNOME, STIPENDIO	Persone che collaborano all'interno dell'agenzia	IDD
CLIENTE	CF, COGNOME, TELEFONO	Persona che commissiona il funerale	CF

### 3.4 Relazioni

RELAZIONE	ENTITA'	DESCRIZIONE	ATTRIBUTI
PRENOTAZIONE_D	DIPENDENTI, CARATTERISTICHE	Serve a verificare la disponibilità di un dipendente	COD_D, GIORNO
PRENOTAZIONE_V	VEICOLO, CARATTERISTICHE	Serve a verificare la disponibilità di un veicolo	COD_VEICOLO, GIORNO
SCELTA	BARE, CARATTERISTICHE	Tipologia della bara utilizzata	
DI	DEFUNTO, FUNERALE	Il funerale del defunto	
COMMISSIONA	CLIENTE, FUNERALE	Il cliente commissiona il funerale	

### 3.5 Vincoli

- Ad ogni funerale dovranno partecipare esattamente due dipendenti
- Ad ogni funerale dovranno esserci esattamente una macchina e un furgone
- Prima di procedere all'inserimento del funerale dipendenti e veicoli dovranno essere disponibili
- All'interno del database, un singolo defunto comparirà una sola volta
- Il cliente verrà memorizzato una ed una sola volta all'interno del database
- I codici fiscali saranno di lunghezza massima di 16 caratteri.
- Ad ogni funerale corrisponde uno e un solo defunto
- La chiave primaria "targa" di veicolo avrà lunghezza di 7 caratteri
- Ogni veicolo può essere furgone oppure auto
- Ogni dipendente può essere presente a un solo funerale al giorno
- Ogni veicolo può essere usato solo una volta al giorno
- L'attributo "fuori\_misura" può essere solo 'si' o 'no'

### 3.6 Tabella dei volumi

*-Per 'N1' si intende un volume attuale in comune a più entità, poichè non è possibile assegnargli un numero preciso.*

ELEMENTO	TIPOLOGIA	VOLUME
Funerale	E	N1
Defunto	E	N1
Caratteristiche	E	N1
Bare	E	6
Veicolo	E	8
Dipendenti	E	12
Cliente	E	N
Prenotazioni_v	R	N1*2
Prenotazioni_d	R	N1*2

### 3.7 Tabella delle operazioni

*\*si tratta di stime approssimative*

Operazioni	Frequenza
Op. 1	3/giorno (in media)
Op. 2	3/giorno (in media)
Op. 3	1/mese
Op. 4	1/mese
Op. 5	1/anno
Op. 6	10/giorno
Op. 7	10/giorno
Op. 8	50/giorno

### 3.8 Traduzione nel modello relazionale

*In grassetto: chiavi primarie*

*Sottolineato: chiavi esterne*

Funerale (**id**, cod\_cliente, cod\_defunto, costo)

Caratteristiche (**cod\_funerale**, bara, ornamentali, fuori\_misura, macchina, furgone, luogo\_svolgimento, luogo\_sepolitura, giorno\_, dipendente\_1, dipendente\_2)

Cliente (**cf**, cognome, telefono)

Dipendenti (**idd**, nome, cognome, stipendio)

Prenotazioni\_d (**cod\_d**, **giorno**)

Prenotazioni\_v (**cod\_veicolo**, **Giorno**)

Veicolo (**targa**, modello, colore, categoria)

Defunto (**cd\_def**, nome, cognome, età)

Bare (**tipologia**, costo, quantità)

## 4. Implementazione SQL

### 4.1 Tabelle

Entità/Relazione	SQL
<b>Funerale</b>	<pre>CREATE TABLE `funerale` (   `id` int NOT NULL,   `cod_cliente` varchar(16) NOT NULL,   `cod_defunto` varchar(16) NOT NULL,   `costo` int DEFAULT NULL,   PRIMARY KEY (`id`),   UNIQUE KEY `defunto_UNIQUE` (`cod_defunto`),   KEY `cod_cliente_idx` (`cod_cliente`),   CONSTRAINT `cod_cliente` FOREIGN KEY (`cod_cliente`) REFERENCES `cliente` (`cf`) ON DELETE CASCADE ON UPDATE CASCADE,   CONSTRAINT `cod_defunto` FOREIGN KEY (`cod_defunto`) REFERENCES `defunto` (`cf_def`) ON DELETE CASCADE ON UPDATE CASCADE )</pre>
<b>Cliente</b>	<pre>CREATE TABLE `cliente` (   `cf` varchar(16) NOT NULL,   `cognome` varchar(45) NOT NULL,   `telefono` int unsigned NOT NULL,   PRIMARY KEY (`cf`),   UNIQUE KEY `cf_UNIQUE` (`cf`),   UNIQUE KEY `telefono_UNIQUE` (`telefono`) )</pre>
<b>Dipendenti</b>	<pre>CREATE TABLE `dipendenti` (   `idd` int NOT NULL AUTO_INCREMENT,   `nome` varchar(45) NOT NULL,   `cognome` varchar(45) NOT NULL,   `stipendio` int NOT NULL,   PRIMARY KEY (`idd`) )</pre>
<b>Prenotazioni_v</b>	<pre>CREATE TABLE `prenotazioni_v` (   `cod_veicolo` varchar(45) NOT NULL,   `Giorno` date NOT NULL,   PRIMARY KEY (`cod_veicolo`,`Giorno`),   CONSTRAINT `cod_veicolo` FOREIGN KEY (`cod_veicolo`) REFERENCES `veicolo` (`targa`) ON DELETE CASCADE ON UPDATE CASCADE )</pre>

<b>Prenotazioni_d</b>	CREATE TABLE `prenotazioni_d` ( `cod_d` int NOT NULL, `giorno` date NOT NULL, PRIMARY KEY (`cod_d`,`giorno`), CONSTRAINT `cod_d` FOREIGN KEY (`cod_d`) REFERENCES `dipendenti` (`idd`) )
<b>Veicolo</b>	CREATE TABLE `veicolo` ( `targa` varchar(7) NOT NULL, `modello` varchar(45) NOT NULL, `colore` varchar(45) NOT NULL, `categoria` varchar(45) NOT NULL, PRIMARY KEY (`targa`), UNIQUE KEY `targa_UNIQUE` (`targa`) )
<b>Defunto</b>	CREATE TABLE `defunto` ( `cf_def` varchar(16) NOT NULL, `nome` varchar(45) DEFAULT NULL, `cognome` varchar(45) DEFAULT NULL, `eta` int DEFAULT NULL, PRIMARY KEY (`cf_def`), UNIQUE KEY `cf_def_UNIQUE` (`cf_def`) )
<b>Bare</b>	CREATE TABLE `bare` ( `tipologia` varchar(45) NOT NULL, `costo_lotto` int NOT NULL, `quantità` int NOT NULL, PRIMARY KEY (`tipologia`), UNIQUE KEY `tipologia_UNIQUE` (`tipologia`) )
<b>Caratteristiche</b>	CREATE TABLE `caratteristiche` ( `cod_funerale` int NOT NULL, `bara` varchar(45) NOT NULL, `ornamentali` varchar(100) DEFAULT NULL, `fuori_misura` varchar(2) NOT NULL DEFAULT 'no', `macchina` varchar(45) NOT NULL, `furgone` varchar(45) NOT NULL, `luogo_svolgimento` varchar(45) NOT NULL, `luogo_sepolitura` varchar(45) NOT NULL, `giorno` date NOT NULL, `dipendente_1` int NOT NULL, `dipendente_2` int NOT NULL, PRIMARY KEY (`cod_funerale`), KEY `bara_idx` (`bara`), KEY `auto_idx` (`macchina`), KEY `furgone_idx` (`furgone`), KEY `dipendente_1_idx` (`dipendente_1`), KEY `dipendente_2_idx` (`dipendente_2`), CONSTRAINT `bara` FOREIGN KEY (`bara`) REFERENCES `bare` (`tipologia`),



	<pre> CONSTRAINT `cod_funerale` FOREIGN KEY (`cod_funerale`) REFERENCES `funerale` (`id`) ON DELETE CASCADE ON UPDATE CASCADE, CONSTRAINT `dipendente_1` FOREIGN KEY (`dipendente_1`) REFERENCES `dipendenti` (`idd`), CONSTRAINT `dipendente_2` FOREIGN KEY (`dipendente_2`) REFERENCES `dipendenti` (`idd`), CONSTRAINT `furgone` FOREIGN KEY (`furgone`) REFERENCES `veicolo` (`targa`), CONSTRAINT `macchina` FOREIGN KEY (`macchina`) REFERENCES `veicolo` (`targa`) ) </pre>
--	--

## #DETTAGLI SUGLI ELEMENTI DELLE TABELLE

- La tabella ‘bara’ dovrà essere aggiornata manualmente una volta al mese e il prezzo potrebbe variare da un mese all’altro. L’OP.4 potrà, dunque, essere eseguita dopo aver effettuato OP.9.
- Le date sono definite nel seguente modo: ‘ANNO-MESE-GIORNO’
- Dipendenti, Veicoli e Tipologie di bare saranno già inseriti all’interno del database

## 4.2 Operazioni

OPERAZIONE	SQL
<b>OP.1</b>	<pre> CREATE DEFINER=`root`@`localhost` PROCEDURE `ins_f` (in cod int,in c varchar(16),in d varchar(16), in co int,in loco varchar(45),in cell int,in cognC varchar(45),in auto varchar(7), in dt date, in nomed varchar(45), in cognomed varchar(45), in age int, in b varchar(45),in au2 varchar(7),in idd1 int, in idd2 int,in ls varchar(45),in fm varchar(2),in orn varchar(45)) BEGIN     declare x int;     declare y int;     declare z int;     declare t int;     declare y2 int;     declare dip int;     declare dip2 int;     set t=(select count(id)from funerale where cod=id);     set x=(select count(cf) from cliente where c=cf);     set y=(select count(cod_veicolo) from prenotazioni_v where cod_veicolo=auto and dt=Giorno);     set y2=(select count(cod_veicolo) from prenotazioni_v where cod_veicolo=au2 and dt=Giorno);     set z=(select count(cf_def)from defunto where cf_def=d);     set dip=(select count(cod_d)from prenotazioni_d where cod_d=idd1 and dt=giorno);     set dip2=(select count(cod_d)from prenotazioni_d where cod_d=idd2 and dt=giorno);     if(t=0) then         if(y=0 and y2=0 and dip=0 and dip2=0) then             if((select quantità from bare where b=tipologia)&lt;&gt;0)then                 if(z=0)then                     if(x=0) then                         insert into cliente(cf,cognome,telefono)values (c,cognC,cell);                     end if;                     insert into defunto(cf_def,nome,cognome,eta) values(d,nomed,cognomed,age);                     insert into funerale(id,cod_cliente,cod_defunto,costo) values(cod,c,d,co);                     insert into prenotazioni_v(cod_veicolo,Giorno)values(auto,dt);                     insert into prenotazioni_v(cod_veicolo,Giorno)values(au2,dt);                     insert into prenotazioni_d(cod_d,giorno)values(idd1,dt);                     insert into prenotazioni_d(cod_d,giorno)values(idd2,dt); </pre>

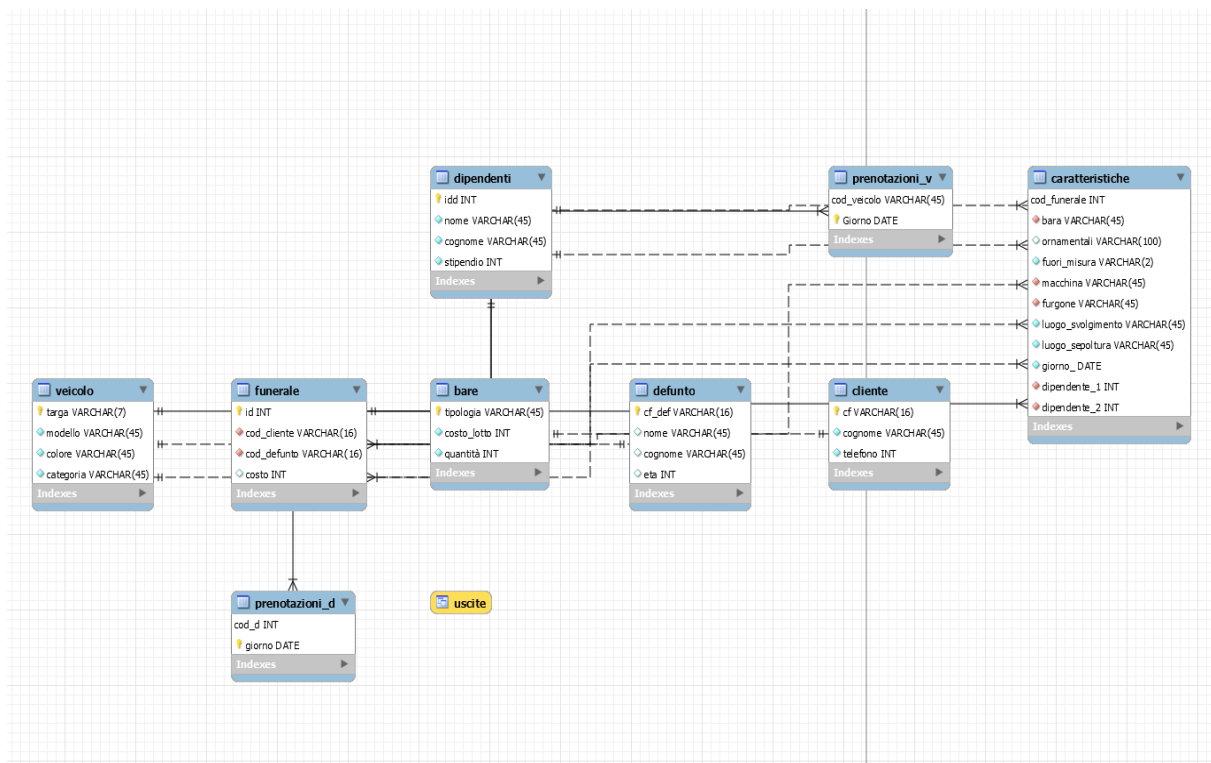
	<pre> insert into caratteristiche(cod_funerale,bara,ornamentali,fuori_misura,macchina,furg one,luogo_svolgimento,luogo_sepolitura,giorno_,dipendente_1,dipendente _2) values(cod,b,orn,fm,auto,au2,loco,ls,dt,idd1,idd2); update bare set quantità=quantità-1 where tipologia=b; select 'SUCCESSO: INSERIMENTO EFFETTUATO' AS "; else select 'ERRORE: Defunto già presente nel database' AS "; end if; else select 'ERRORE: Bara non presente in magazzino' AS "; end if; else select 'ERRORE: VEICOLO O DIPENDENTE NON DISPONIBILE -&gt; CAMBIARE INPUT E RIPROVARE' AS "; end if; else select 'ERRORE: ID FUNERALE ESISTENTE' AS "; end if; END </pre>
<b>OP.2</b>	<pre> CREATE DEFINER=`root`@`localhost` PROCEDURE `record_cliente`(in c varchar(16)) BEGIN select count(cod_cliente) from funerale where cod_cliente=c; END </pre>
<b>OP.3</b>	<pre> CREATE DEFINER=`root`@`localhost` PROCEDURE `entrate`(in dt1 date, in dt2 date) BEGIN select sum(costo)from funerale inner join caratteristiche on funerale.id=caratteristiche.cod_funerale where giorno_&gt;dt1 and giorno_&lt;dt2 ; END </pre>

<b>OP.4</b> Verrà implementata una vista	<pre>CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY DEFINER VIEW `uscite` AS select `bare`.`costo_lotto` AS `u` from `bare` union select `dipendenti`.`stipendio` AS `u` from `dipendenti`  -----  CREATE DEFINER=`root`@`localhost` PROCEDURE `uscite`() BEGIN     select sum(u) from uscite;  END</pre>
<b>OP.5</b>	<pre>CREATE DEFINER=`root`@`localhost` PROCEDURE `aggiorna_telefono`(in c varchar(16), in n int) BEGIN     update cliente set telefono=n where c=cf; END</pre>
<b>OP.6</b>	<pre>CREATE DEFINER=`root`@`localhost` PROCEDURE `veicolo_disponibile`(in t varchar(7), in dt date) BEGIN     declare x int;     set x=(select count(cod_veicolo) from prenotazioni_v where cod_veicolo=t and dt=Giorno);     if(x=1) then         select 'Disponibile' AS ";     else         select 'NON DISPONIBILE' AS ";     end if; END</pre>
<b>OP.7</b>	<pre>CREATE DEFINER=`root`@`localhost` PROCEDURE `dipendente_diponibile`(in d int, in dt date) BEGIN     declare dip int;     set dip=(select count(cod_d)from prenotazioni_d where cod_d=d and dt=giorno);     if(dip=1) then         select 'NON DISPONIBILE' AS ";     else         select 'DISPONIBILE' AS ";     end if; END</pre>
<b>OP.8</b>	<pre>CREATE DEFINER=`root`@`localhost` PROCEDURE `info`(in f int) BEGIN     select* from funerale inner join caratteristiche on cod_funerale=id where funerale.id=f; END</pre>

**OP.9**

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `rifornisci`(in t
varchar(45), in n int, in c int)
BEGIN
    update bare set quantità=n, costo_lotto=c where t=tipologia;
END
```

## 4.3 Diagramma EER



## 5. Esempi di query

*Questa sezione rappresenta una guida all'utilizzo del database*

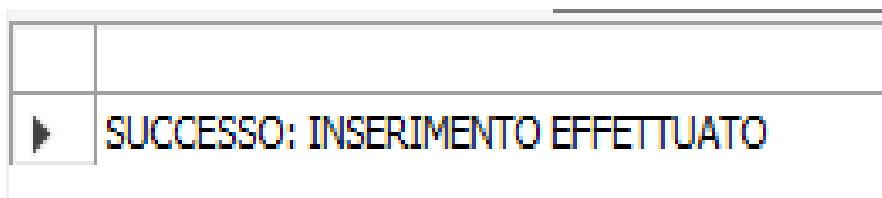
Si effettui l'inserimento di un nuovo funerale:

call ins\_f (id funerale, cf cliente, cf defunto, costo, luogo svolgimento, numero di telefono del cliente, cognome del cliente, targa auto, data, nome del defunto, cognome del defunto, età del defunto, tipologia bara, targa furgone, id impiegato 1, id impiegato 2, luogo sepoltura, fuori misura, dettagli aggiuntivi)

**ES:**

call ins\_f (7761, 'BOIKNB80A19G273B', 'BNDJMS43S10H501H', 3500, 'Chiesa San Giovanni', 555901, 'KENOBI', 'BJ999HT', '2021-03-10', 'JAMES', 'BOND', 78, 'NOCE', 'AA123BB', 1, 2, 'Cimitero Catania', 'NO', 'Crocifisso fissato sul fronte della bara');

**RISULTATO (SENZA ERRORI)**



*Durante l'inserimento possono verificarsi alcuni errori che annulleranno la procedura.*

### **ERRORI POSSIBILI:**

1. ID FUNERALE GIA' PRESENTE NEL DATABASE
2. BARA NON DISPONIBILE IN MAGAZZINO
3. DEFUNTO GIA' PRESENTE NEL DATABASE
4. VEICOLO O DIPENDENE NON DISPONIBILI PER LA DATA INSERITA

## CASO 2: ID FUNERALE GIA' PRESENTE NEL DATABASE

*Si provi ad inserire un nuovo funerale ma con un 'id funerale' già esistente:*

**id presente nel database: 7761**

INSERIMENTO:

```
call ins_f (7761,'SFFCLT80A01C351L', 'GNILND39R09C351K', 1700, 'Chiesa Cattedrale',  
555888, 'SOFFICINI', 'BJ999HT', '2021-11-05', 'GINO', 'LAVANDINO', 90, 'NOCE', 'AA123BB',  
1, 2, 'Cimitero Roma', 'si', '/n');
```

### RISULTATO

▶	ERRORE: ID FUNERALE ESISTENTE

## CASO 2: BARA NON DISPONIBILE IN MAGAZZINO

*Si assuma che l'attributo 'quantità' della tabella 'bare' della tipologia 'ABETE' sia uguale a zero.*

INSERIMENTO:

```
call ins_f (6405,'SFFCLT80A01C351L', 'GNILND39R09C351K', 1700, 'Chiesa Cattedrale',  
555888, 'SOFFICINI', 'BJ999HT', '2021-11-05', 'GINO', 'LAVANDINO', 90, 'ABETE',  
'AA123BB', 1, 2, 'Cimitero Roma', 'si', '/n');
```

### RISULTATO

▶	ERRORE: Bara non presente in magazzino

### CASO 3: DEFUNTO GIA' PRESENTE NEL DATABASE

**defunto presente nel database:** BNDJMS43S10H501H

INSERIMENTO:

```
call ins_f (6405,'SFFCLT80A01C351L', 'BNDJMS43S10H501H', 1700, 'Chiesa Cattedrale',  
555888, 'SOFFICINI', 'BJ999HT', '2021-11-05', 'GINO', 'LAVANDINO', 90, 'NOCE', 'AA123BB',  
1, 2, 'Cimitero Roma', 'si', '/n');
```

#### RISULTATO

▶	ERRORE: Defunto già presente nel database

### CASO 4: VEICOLO O DIPENDENE NON DISPONIBILI PER LA DATA INSERITA

**targa presente:** 'BJ999HT' in **data** '2021-03-10'

*Situazione analoga nel caso “dipendenti non disponibili”*

INSERIMENTO:

```
call ins_f (8300,'SFFCLT80A01C351L', 'GNILND39R09C351K', 1900, 'Chiesa Cattedrale',  
555888, 'SOFFICINI', 'BJ999HT', '2021-03-10', 'GINO', 'LAVANDINO', 90, 'NOCE', 'TT444ZZ',  
3, 4, 'Cimitero Roma', 'si', '/n');
```

#### RISULTATO

▶	ERRORE: VEICOLO O DIPENDENTE NON DISPONIBILE -> CAMBIARE INPUT E RIPROVARE



# STATUS INIZIALE DEL DATABASE

## DIPENDENTI

	idd	nome	cognome	stipendio
▶	1	Aldo	Baglio	1200
	2	Giovanni	Storti	1200
	3	Giacomino	Poretti	1500
	4	Carlo	Conti	1700
	5	Bill	Gates	5000
	6	Robert	De Niro	3000
*	NULL	NULL	NULL	NULL

## VEICOLI

	targa	modello	colore	categoria
*	AA123BB	FIAT	NERO	furgone
	BB321AA	MERCEDES	GRIGIO	furgone
	BJ999HT	MERCEDES	NERO	auto
	HH783YT	MASERATI	BLU SCURO	auto
	JB000BL	FERRARI	ROSSA	auto
	JS033DD	MAGGIOLINO	GRIGIO	auto
	KW163FG	PANDINO	BIANCO	auto
	TT444ZZ	MERCEDES	NERO	furgone
*	NULL	NULL	NULL	NULL

## BARE

	tipologia	costo_lotto	quantità
▶	ABETE	400	0
	FRASSINO	600	10
	LARICE	500	10
	MOGANO	700	10
	NOCE	900	8
	ROVERE	800	10
*	NULL	NULL	NULL