Relazione per il corso di Basi di Dati

Anno Accademico 2018/2019

Progetto di una base di dati per un’azienda di distribuzione bevande



Autori:

Alpi Davide [davide.alpi@studio.unibo.it](mailto:davide.alpi@studio.unibo.it) 0000831238

Penazzi Paolo [paolo.penazzi@studio.unibo.it](mailto:paolo.penazzi@studio.unibo.it) 0000825618

Università di Bologna - Campus di Cesena

Corso di Laurea in Ingegneria e Scienze Informatiche

**Sommario**

[Analisi dei requisiti 3](#_Toc11809780)

[Descrizione in linguaggio naturale 3](#_Toc11809781)

[Estrazione concetti fondamentali 4](#_Toc11809782)

[Progettazione concettuale 5](#_Toc11809783)

[Venditori 5](#_Toc11809784)

[Progettazione schema E/R 5](#_Toc11809785)

[Raffinamenti proposti 5](#_Toc11809786)

[Schema concettuale parziale 6](#_Toc11809787)

[Magazzinieri 6](#_Toc11809788)

[Progettazione schema E/R 6](#_Toc11809789)

[Raffinamenti proposti 7](#_Toc11809790)

[Schema concettuale parziale 7](#_Toc11809791)

[Amministrativi 8](#_Toc11809792)

[Progettazione schema E/R 8](#_Toc11809793)

[Raffinamenti proposti 9](#_Toc11809794)

[Schema concettuale parziale 9](#_Toc11809795)

[Schema concettuale finale 9](#_Toc11809796)

[Specifiche Funzionali 11](#_Toc11809797)

[Operazioni sui dati 11](#_Toc11809798)

[Operazioni comuni 11](#_Toc11809799)

[Operazioni Venditori 11](#_Toc11809800)

[Operazioni Magazzinieri 11](#_Toc11809801)

[Operazioni Amministrativi 12](#_Toc11809802)

[Progettazione Logica 13](#_Toc11809803)

[Stima del volume dei dati 13](#_Toc11809804)

[Descrizione delle operazioni principali e stima della loro frequenza 14](#_Toc11809805)

[Tavole degli accessi e costi 15](#_Toc11809806)

[Analisi delle ridondanze 18](#_Toc11809807)

[Traduzione delle entità e associazioni in relazioni 19](#_Toc11809808)

[Schema relazionale finale 20](#_Toc11809809)

[Costruzione delle tabelle in linguaggio SQL 21](#_Toc11809810)

# Analisi dei requisiti

## Descrizione in linguaggio naturale

L’azienda richiede la creazione di un sistema informatico per la gestione della vendita, della distribuzione di bevande a vari clienti sparsi sul territorio nazionale. L’ufficio acquisti stabilisce degli accordi con i fornitori per determinati prodotti. Prima di accordarsi sulla fornitura si deve verificare che i dati del fornitore siano registrati nel database, nel caso non lo siano, vanno aggiunti prima di concludere l’accordo. Questi prodotti verranno inseriti in un catalogo, contenente tutti i prodotti in vendita. I contratti con i fornitori hanno durata variabile. L’azienda ha a disposizione un magazzino principale, e diversi magazzini secondari. Ogni magazzino ha una giacenza minima di prodotti per fornire un adeguato servizio ai clienti. L’ufficio riordino si occupa di effettuare gli acquisti dei prodotti dai fornitori. La spedizione dei prodotti è gestita dal fornitore, che fornisce un numero di telefono per accordarsi sulla data di consegna col magazzino al quale verrà spedita la merce (il deposito va specificato in fase di acquisto). Per ogni acquisto di merce, dovrà essere emessa una fattura di acquisto. All’arrivo della merce, il magazzino dovrà aggiornare la lista dei prodotti in giacenza, memorizzando la zona del magazzino in cui sono riposti. L’azienda dispone di una rete di vendita composta da agenti. L’agente è un libero professionista che viene messo sotto contratto dall’azienda per un determinato periodo di tempo. Gli agenti vengono pagati in base a quanti prodotti vendono. I clienti acquistano la merce dai venditori, che emetteranno un ordine di vendita, che specifica la quantità, la tipologia dei prodotti venduti, e il cliente al quale recapitare la merce. Se il cliente non è ancora registrato nell’anagrafica cliente, l’agente ha il compito di inserirlo nel database prima di completare la vendita. I venditori devono inoltre emettere una fattura di vendita, contenente tutte le informazioni relative alla vendita. Il reparto distribuzione si occupa di organizzare i trasporti merce. I trasporti merce prelevano la merce da un magazzino e la portano al cliente che l’ha acquistata. I clienti posso effettuare reclami al servizio clienti nel caso siano scontenti del servizio. Il reclamo può essere di varie tipologie. L’amministrazione si occupa di registrare le fatture, sia di acquisto che di vendita. Per le fatture di vendita è importante memorizzare il codice del venditore, in modo da poter calcolare la provvigione da pagare agli agenti.

Il gestionale si svilupperà su 3 viste in base al ruolo che l’impiegato ricopre nell’azienda: Amministrativo, Venditore, Magazziniere.

## Estrazione concetti fondamentali

L’azienda richiede la creazione di un sistema informatico per la gestione della vendita, della distribuzione di bevande a vari clienti sparsi sul territorio nazionale. L’**ufficio acquisti** stabilisce degli **accordi** con i **fornitori** per determinati **prodotti**. Prima di accordarsi sulla fornitura si deve verificare che i dati del fornitore siano registrati nel database, nel caso non lo siano, vanno aggiunti prima di concludere l’accordo. Questi prodotti verranno inseriti in un **catalogo**, contenente tutti i prodotti in vendita. I contratti con i fornitori hanno durata variabile. L’azienda ha a disposizione un **magazzino principale**, e diversi **magazzini secondari**. Ogni magazzino ha una **giacenza** minima di prodotti per fornire un adeguato servizio ai clienti. L’**ufficio riordino** si occupa di effettuare gli **acquisti** dei prodotti dai fornitori. La spedizione dei prodotti è gestita dal fornitore, che fornisce un numero di telefono per accordarsi sulla data di consegna col magazzino al quale verrà spedita la **merce** (il deposito va specificato in fase di acquisto). Per ogni acquisto di merce, dovrà essere emessa una fattura di acquisto. All’arrivo della merce, il magazzino dovrà aggiornare la lista dei prodotti in giacenza, memorizzando la zona del magazzino in cui sono riposti. L’azienda dispone di una rete di vendita composta da **agenti**. L’agente è un libero professionista che viene messo sotto contratto dall’azienda per un determinato periodo di tempo. Gli agenti vengono pagati in base a quanti prodotti vendono. I **clienti** acquistano la merce dai venditori, che emetteranno un ordine di vendita, che specifica la quantità, la tipologia dei prodotti venduti, e il cliente al quale recapitare la merce. Se il cliente non è ancora registrato nell’anagrafica cliente, l’agente ha il compito di inserirlo nel database prima di completare la vendita. I venditori devono inoltre emettere una fattura di vendita, contenente tutte le informazioni relative alla vendita. Il **reparto distribuzione** si occupa di organizzare i **trasporti** merce. I trasporti merce prelevano la merce da un magazzino e la portano al cliente che l’ha acquistata. I clienti posso effettuare **reclami** al **servizio clienti** nel caso siano scontenti del servizio. Il reclamo può essere di varie tipologie. L’**amministrazione** si occupa di registrare le fatture, sia di acquisto che di vendita. Per le fatture di vendita è importante memorizzare il codice del venditore, in modo da poter calcolare la provvigione da pagare agli agenti.

Riteniamo che la descrizione sia già per sé esaustiva, ma vorremmo fare alcune precisazioni:

* Merce e Prodotto nel nostro contesto assumono lo stesso significato.
* Tutti i prodotti (bevande) hanno una data di scadenza trascurabile.
* Essendoci un solo Catalogo, non verrà trattato come un’entità.
* Gli acquisti dai fornitori verranno d’ora in poi chiamati **Ordini**.
* Agente e Venditore assumono lo stesso significato.
* Ogni magazzino ha il suo reparto distribuzione. Quando si farà riferimento ad un magazzino, comprenderà anche quest’ultimo.
* Amministrazione, Servizio clienti, Ufficio acquisti sono tutte tipologie di **Ufficio**.

# Progettazione concettuale

## Venditori

### Progettazione schema E/R

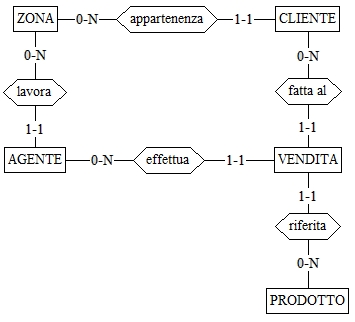


Figura - Schema scheletro

### Raffinamenti proposti

L’**AGENTE** è identificato da un codice agente. Oltre a memorizzare i dati di ogni agente (Nome, Cognome…) è importante conoscere la sua Partita IVA, e la provvigione (Percentuale stabilita all’inizio della collaborazione che indica il guadagno dell’agente sull’incasso totale).

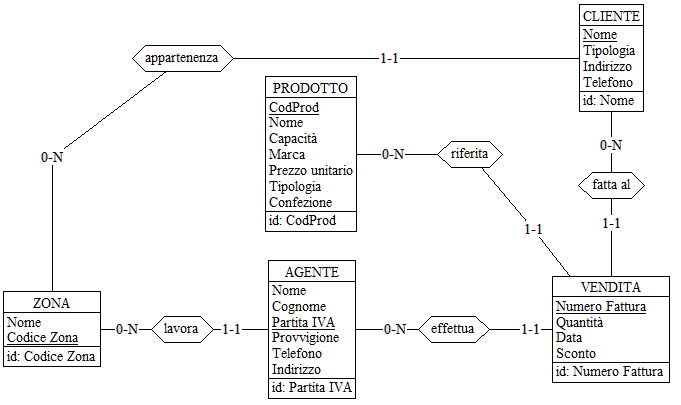
La **VENDITA** è identifica dall’ID della fattura relativa. Vanno inoltre memorizzati dettagli come lo sconto effettuato e la quantità di merce venduta.

Ogni vendita sarà associata ad un **CLIENTE**, che è identificato dal nome (ragione sociale).

La vendita è sempre riferita ad uno e un solo **PRODOTTO**, identificato da un codice.

Per evitare che il lavoro di più agenti si sovrapponga, viene introdotta l’entità **ZONA**.  
Ogni agente opera in una zona, in una zona possono operare più agenti. Ogni zona ha un nome ed è identificata da un ‘Codice Zona’.

### Schema concettuale parziale



## Magazzinieri

### Progettazione schema E/R

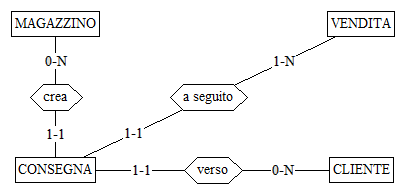


Figura - Consegne

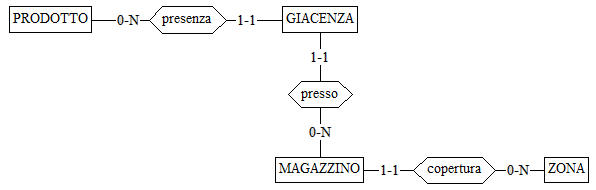


Figura - Scorte



Figura – Riordino

### Raffinamenti proposti

Lo schema riguardante i magazzinieri è particolarmente complesso, in quanto si devono occupare di diversi ambiti: Consegne, Scorte, Riordino.

Ogni **MAGAZZINO** è identificato da un nome, e dal codice della zona che copre.

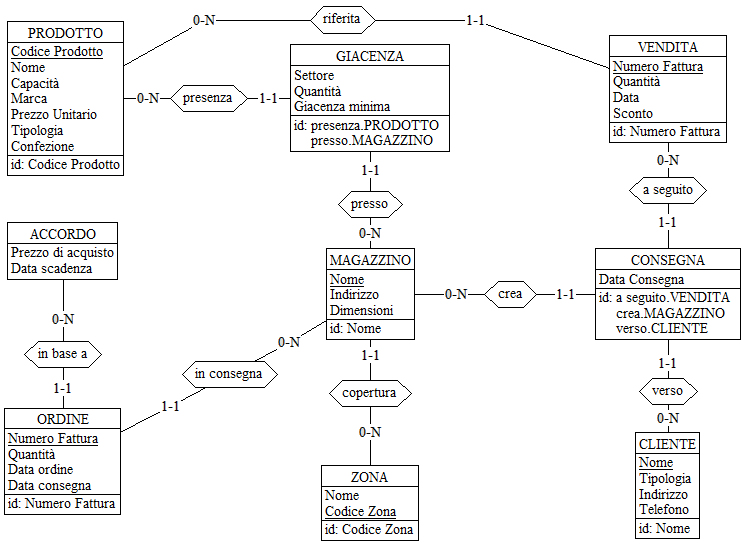
Il magazzino si occupa di effettuare le consegne per le vendite nella sua zona. La **CONSEGNA** è identificata dalla data di consegna, ed è associata ad una vendita e ad un cliente.

Ogni magazzino avrà anche una lista di prodotti in **GIACENZA**. Le giacenze sono associate ad un prodotto e ad un magazzino. Inoltre, viene salvato per ogni giacenza il settore del magazzino in cui si trova la merce e la giacenza minima.

Infine, il magazzino si preoccupa di riordinare i prodotti quando questi finiscono o la quantità scende sotto la giacenza minima. Quando viene effettuato un **ORDINE**, va emessa una fattura che lo identifica. Ricordiamo che le consegne degli ordini sono a carico dei fornitori.

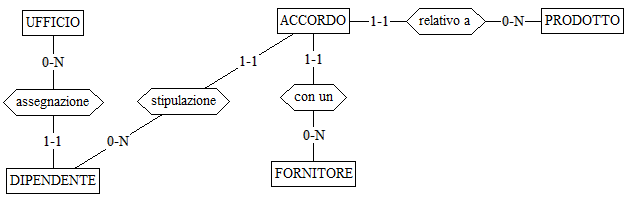
### Schema concettuale parziale

Questo sarà lo schema parziale per quanto riguarda il magazzino. I 3 ambiti sono stati collegati e dove necessario sono state aggiunte entità (già trattate in precedenza o che verranno trattate nelle prossime viste) per rendere più chiaro lo schema. Questo schema potrebbe subire ulteriori modifiche con l’avanzare del progetto.



## Amministrativi

### Progettazione schema E/R



### Raffinamenti proposti

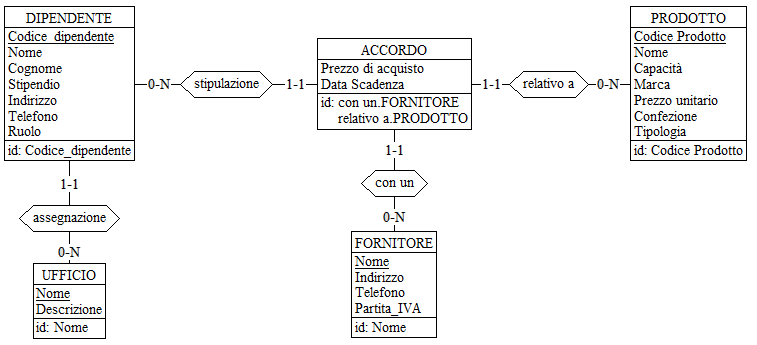
Per ogni **DIPENDENTE** vogliamo memorizzare alcune informazioni base come nome, cognome, indirizzo e stipendio. Come identificativo è stato scelto un codice dipendente univoco all’interno di tutta l’azienda.

Ogni dipendente sarà assegnato ad un **UFFICIO**.

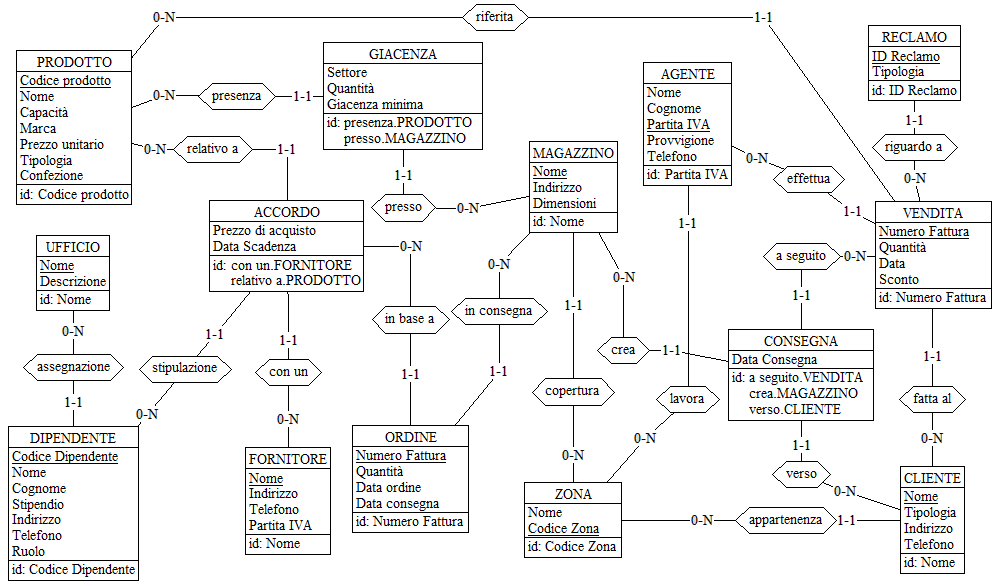
Per acquistare un prodotto da un fornitore è necessario stabilire un **ACCORDO**. Gli accordi sono una specie di contratti tra l’azienda e i fornitori. Ogni accordo è relativo ad un solo prodotto e ne stabilisce il prezzo di acquisto. Per ogni accordo è necessario memorizzare la data di inizio e quella di fine.

Per quanto riguarda i **FORNITORI** ci interessa memorizzare il nome (Identificativo), l’indirizzo, il telefono e la Partita IVA.

### Schema concettuale parziale



### Schema concettuale finale



# Specifiche Funzionali

## Operazioni sui dati

Prendiamo ora in considerazione le operazioni sui dati che devono essere svolte. In base a queste ci potranno essere modifiche allo schema concettuale. Inizialmente elencheremo le operazioni elementari comuni alle tre categorie di dipendenti, successivamente entreremo più nel dettaglio elencano le operazioni che possono essere effettuate solo da una determinata categoria di persone all’interno dell’azienda.

### Operazioni comuni

Queste operazioni saranno in tutte e 3 le viste del nostro gestionale. Si tratta di operazioni elementari come quelle di ricerca.

***Ricerca Cliente***

Dato un ‘Nome’, permette di controllare se il cliente è già nel database. Questa operazione è effettua sia dai Venditori che dagli Amministrativi.

**Ricerca Prodotto**

Dato un ‘Codice prodotto’ visualizza le informazioni relative a quest’ultimo. Questa operazione è utile per tutti i reparti.

### Operazioni Venditori

***Vendita Prodotto***

Dopo aver effettuato una vendita, quest’ultima va aggiunta al database inserendo anche i dettagli relativi ad essa.

***Aggiunta Cliente***

Se non è ancora stata effettuata nessuna vendita ad un cliente, va aggiunto al database.

### Operazioni Magazzinieri

***Crea giacenza***

Aggiunge una nuova giacenza al magazzino.

***Visualizza giacenze***

Dato un prodotto e un magazzino, ritorna il settore del magazzino nella quale il prodotto è depositato.

**Registra ordine**

Ordina un prodotto.

***Visualizza giacenze sotto la soglia minima***

Ritorna tutti i prodotti la cui giacenza è sotto la soglia minima.

***Registra consegna***

Quando una consegna viene effettuata, la marca come effettuata, inserendo i dati relativi ad essa.

***Aggiorna Giacenza***

Dopo una consegna aggiorna la giacenza relativa al prodotto venduto.

### Operazioni Amministrativi

***Calcolo provvigioni***

Calcola il fatturato di ogni venditore e in seguito applica la provvigione prevista dal suo contratto, in questo modo viene calcolato quanto dovrà essere pagato ciascun venditore.

***Registra prodotto***

Inserisce un prodotto nel catalogo

***Registra Fornitore***

Aggiunge un fornitore all’anagrafica fornitori.

***Aggiorna Accordo***

Aggiorna le informazioni di un accordo

***Registra Accordo***

Registra un nuovo accordo stipulato con un fornitore.

***Visualizza Dipendenti***

Visualizza la lista dei dipendenti

***Visualizza Agenti***

Visualizza la lista degli agenti.

**Registra Dipendente**

Aggiunge un dipendente alla lista dei dipendenti dell’azienda

**Registra Agente**

Aggiunge un agente alla lista degli agenti che lavora in cambusa

**Calcolo Provvigioni**

Dato un agente, calcola il suo fatturato e applica la sua provvigione.

# Progettazione Logica

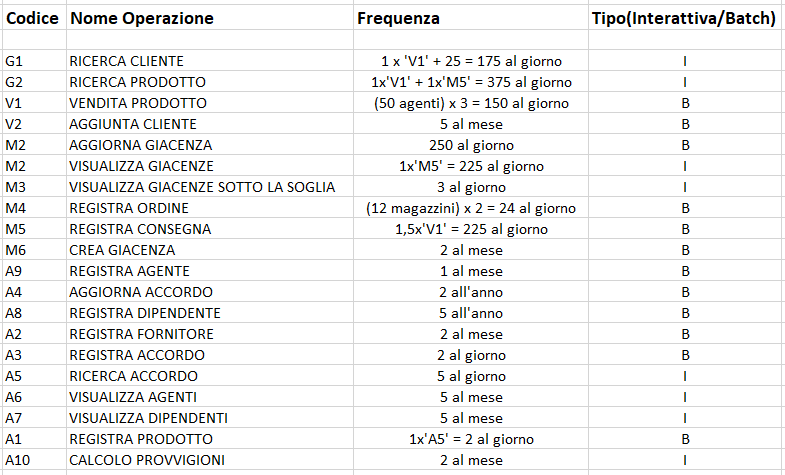
## Stima del volume dei dati

Di seguito viene fornita la tabella che mostra una stima del numero di istanze per ogni entità e per ogni relazione.



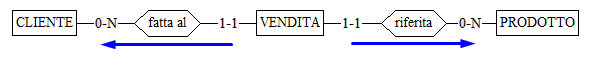
## Descrizione delle operazioni principali e stima della loro frequenza

Si vuole stimare ora la frequenza di ogni operazione. Alcune operazioni vengono eseguite obbligatoriamente prima di eseguirne un’altra. Viene inoltre specificato il tipo di operazione. Ricordiamo che le operazioni di scrittura hanno un costo maggiore delle operazioni di lettura.

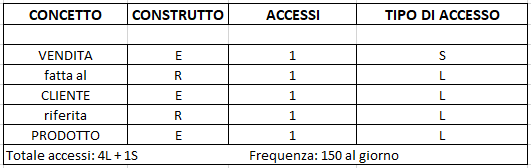


## Tavole degli accessi e costi

***V1 - VENDITA PRODOTTO***

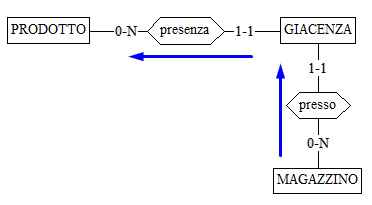


Registra la vendita di un determinato prodotto ad un cliente.

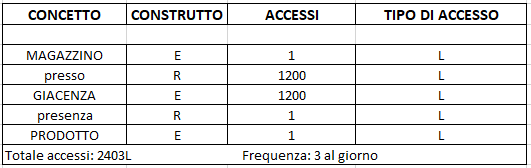


Costo Totale: 4L + 1S = 6 \* 150 = *900 accessi al giorno.*

***M2 - VISUALIZZA GIACENZE***

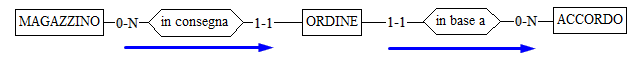


Dato un magazzino, ritorna le giacenze di ogni prodotto nel suddetto magazzino.

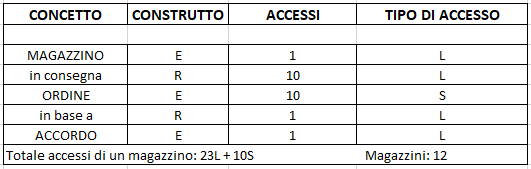


Costo Totale: 2403L = 2403 \* 3 = *7209 accessi al giorno.*

***M4 – REGISTRA ORDINE***

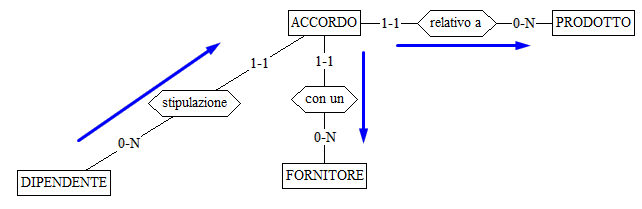


Quando la giacenza di un prodotto in un magazzino è sotto la soglia minima, il magazzino si occupa di effettuare l’ordine del prodotto.



Costo Totale: 43 accessi \* 12 = *516 accessi al giorno.*

***A5 – REGISTRA ACCORDO***

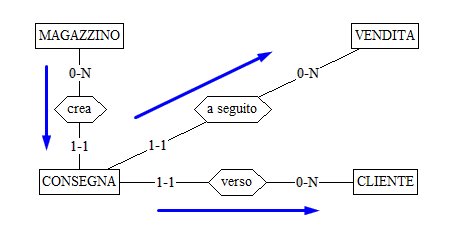


Viene effettuato un nuovo accordo con un fornitore, per la fornitura di un prodotto.

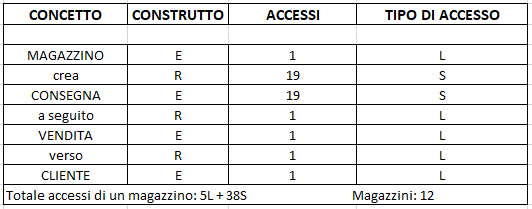


Costo Totale: 9 \* 2 = *45 accessi al giorno*

**M5 - REGISTRA CONSEGNA**



Un magazzino effettua la consegna di una merce ad un cliente.



Costo Totale: 81 \* 12 = *972 accessi al giorno.*

Le operazioni non analizzate hanno tavole di accesso e costi banali, o equivalenti alle operazioni sopra descritte.

## Analisi delle ridondanze

Nella tabella delle vendite è stato rimosso l’attributo ‘Costo Totale’ in quanto esso si può facilmente ricavare moltiplicando VENDITA.Quantità e PRODOTTO.Prezzo\_Unitario. L’operazione per ricavare il prezzo totale a dei costi irrisori, mentre inserire ogni volta il prezzo totale sarebbe risultato più oneroso.

Un prodotto, oltre che con l’attributo ‘ID\_prodotto’ è anche identificato univocamente dagli attributi (Nome, Capacità, Marca). Per comodità abbiamo però ritenuto opportuno usare un identificativo del prodotto come chiave. La terna (Nome, Capacità, Marca) rimane comunque una chiave di accesso alla tabella PRODOTTI.

## Traduzione delle entità e associazioni in relazioni

**ACCORDI**(*Nome\_fornitore*, *Codice\_prodotto*, Prezzo\_di\_acquisto, Data\_scadenza, Codice\_dipendente)

FK: Codice\_prodotto REFERENCES **PRODOTTI.Codice\_prodotto**

FK: Nome\_fornitore REFERENCES **FORNITORI.Nome**AK: Codice\_dipendente REFERENCES **DIPENDENTI**

**AGENTI**(*Partita\_IVA*, Nome, Cognome, Provvigione, Telefono, Codice\_zona) AK: Codice\_zona REFERENCES **ZONE.Codice\_zona**

**CLIENTI**(*Nome*, TIpologia, Indirizzo, Telefono, Codice\_zona) AK: Codice\_zona REFERENCES **ZONE.Codice\_zona**

**CONSEGNE**(*Numero\_fattura*, *Nome\_magazzino*, *Nome\_cliente*, *Data\_consegna*)

FK: Numero\_fattura REFERENCES **VENDITE.Numero\_fattura**

FK: Nome\_magazzino REFERENCES **MAGAZZINI.Nome**FK: Nome\_cliente REFERENCES **CLIENTI.Nome**

**DIPENDENTI**(*Codice\_dipendente*, Nome, Cognome, Stipendio, Indirizzo, Telefono, Ruolo, Ufficio) AK: Ufficio REFERENCES **UFFICI.Nome**

**FORNITORI**(*Nome*, Indirizzo, Telefono, Partita\_IVA)

**GIACENZE**(*Codice\_prodotto*, *Magazzino*, Settore, Quantità, Giacenza\_minima)

FK: Codice\_prodotto REFERENCES **PRODOTTI.Codice\_prodotto**

FK: Magazzino REFERENCES **MAGAZZINI.Nome**

**MAGAZZINI**(*Nome*, Indirizzo, Dimensioni, Codice\_zona)  
AK: Codice\_zona REFERENCES **ZONE.Codice\_zona**

**ORDINI**(*Numero\_fattura*, Quantità, Data\_ordine, Data\_consegna, Nome, Nome\_fornitore, Codice\_prodotto)

FK: Nome REFERENCES **MAGAZZINI.Nome**FK: Nome\_fornitore, Codice\_prodotto REFERENCES **ACCORDI (Nome\_fornitore, Codice\_prodotto)**

**PRODOTTI**(*Codice\_prodotto*, Nome, Capacità, Marca, Prezzo\_unitario, Tipologia, Confezione)

AK: (Nome, Capacità, Marca)

**RECLAMI**(*ID\_Reclamo*, Numero\_fattura, Tipologia)

AK: Numero\_fattura REFERENCES **VENDITE.Numero\_fattura**

**UFFICI**(*Nome*, Descrizione,)

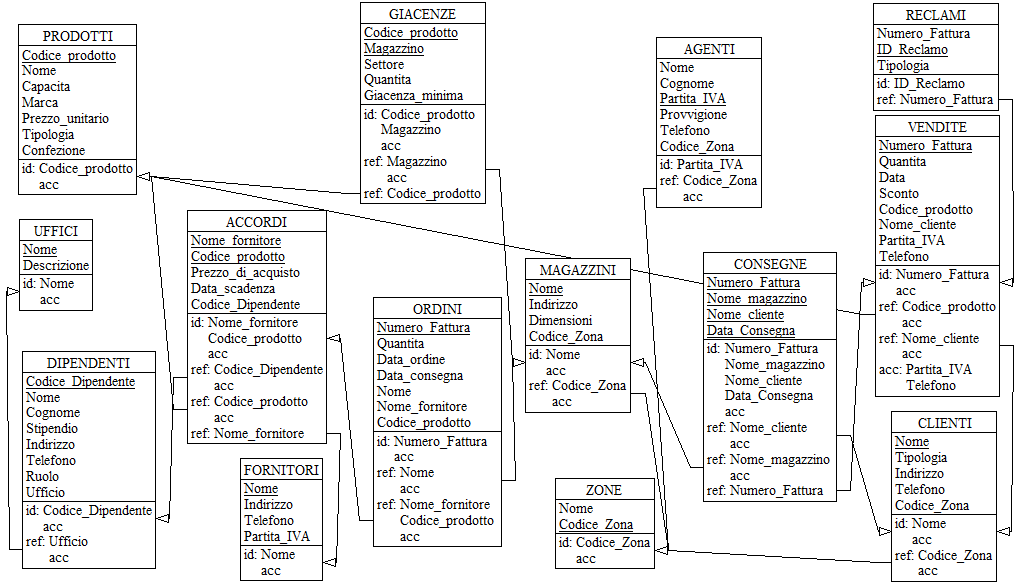
**VENDITE**(*Numero\_fattura*, Quantità, Data, Sconto, Codice\_prodotto, Nome\_cliente, Partita\_IVA, Telefono)

FK: Codice\_prodotto REFERENCES **PRODOTTI.Codice\_prodotto**

FK: Nome\_cliente REFERENCES **CLIENTI.Nome**AK: Partita\_IVA, Telefono

**ZONE**(*Codice\_zona*, Nome,)

## Schema relazionale finale



## Costruzione delle tabelle in linguaggio SQL

create table ACCORDI (

Nome\_fornitore varchar(20) not null,  
 Codice\_prodotto int not null,  
 Prezzo\_di\_acquisto decimal(4,2) not null,  
 Data\_scadenza date not null,  
 Codice\_Dipendente int not null,  
 constraint ID\_ACCORDO\_ID primary key (Nome\_fornitore, Codice\_prodotto));

create table AGENTI (

Nome varchar(20) not null,  
Cognome varchar(20) not null,  
Partita\_IVA varchar(20) not null,  
Provvigione decimal(4,2) not null,  
Telefono varchar(16) not null,  
Codice\_Zona int not null,  
constraint IDAGENTE primary key (Partita\_IVA));

create table CLIENTI (

Nome varchar(25) not null,  
Tipologia varchar(20) not null,  
Indirizzo varchar(40) not null,  
Telefono varchar(16) not null,  
Codice\_Zona int not null,  
constraint ID\_CLIENTE\_ID primary key (Nome));

create table CONSEGNE (

Numero\_Fattura int not null,  
Nome\_magazzino varchar(20) not null,  
Nome\_cliente varchar(25) not null,  
Data\_Consegna date not null,  
constraint ID\_CONSEGNA\_ID primary key (Numero\_Fattura, Nome\_magazzino, Nome\_cliente, Data\_Consegna));

create table DIPENDENTI (

Codice\_Dipendente int not null AUTO\_INCREMENT,  
Nome varchar(20) not null,  
Cognome varchar(20) not null,  
Stipendio int not null,  
Indirizzo varchar(40) not null,  
Telefono varchar(16) not null,  
Ruolo varchar(30) not null,  
Ufficio varchar(20) not null,  
constraint ID\_DIPENDENTE\_ID primary key (Codice\_Dipendente));

create table FORNITORI (

Nome varchar(20) not null,  
Indirizzo varchar(40) not null,  
Telefono varchar(16) not null,  
Partita\_IVA varchar(20) not null,  
constraint ID\_FORNITORE\_ID primary key (Nome));

create table GIACENZE (

Codice\_prodotto int not null,  
Magazzino varchar(20) not null,  
Settore int not null,  
Quantita int not null,  
Giacenza\_minima int not null,  
constraint ID\_GIACENZA\_ID primary key (Codice\_prodotto, Magazzino));

create table MAGAZZINI (

Nome varchar(20) not null,  
Indirizzo varchar(40) not null,  
Dimensioni int not null,  
Codice\_Zona int not null,  
constraint ID\_MAGAZZINO\_ID primary key (Nome));

create table ORDINI (

Numero\_Fattura int not null AUTO\_INCREMENT,  
Quantita int not null,  
Data\_ordine date not null,  
Data\_consegna date not null,  
Nome varchar(20) not null,  
Nome\_fornitore varchar(20) not null,  
Codice\_prodotto int not null,  
constraint ID\_ORDINE\_ID primary key (Numero\_Fattura));

create table PRODOTTI (

Codice\_prodotto int not null AUTO\_INCREMENT,  
Nome varchar(20) not null,  
Capacità int not null,  
Marca varchar(20) not null,  
Prezzo\_unitario decimal(4,2) not null,  
Tipologia varchar(15) not null,  
Confezione varchar(15) not null,  
constraint ID\_PRODOTTO\_ID primary key (Codice\_prodotto));

create table RECLAMI (

Numero\_Fattura int not null,  
ID\_Reclamo int not null AUTO\_INCREMENT,  
Tipologia varchar(30) not null,  
constraint ID\_RECLAMO\_ID primary key (ID\_Reclamo));

create table UFFICI (

Nome varchar(20) not null,  
Descrizione varchar(100) not null,  
constraint ID\_UFFICIO\_ID primary key (Nome));

create table VENDITE (

Numero\_Fattura int not null AUTO\_INCREMENT,  
Quantita int not null,  
Data date not null,  
Sconto decimal(4,2) not null,  
Codice\_prodotto int not null,  
Nome\_cliente varchar(25) not null,  
Partita\_IVA varchar(20) not null,  
Telefono varchar(16) not null,  
constraint ID\_VENDITA\_ID primary key (Numero\_Fattura));

create table ZONE (

Nome varchar(25) not null,  
Codice\_Zona int not null AUTO\_INCREMENT,  
constraint ID\_ZONA\_ID primary key (Codice\_Zona));

## Traduzioni delle operazioni in query SQL

***M1 – Aggiorna giacenza***update Giacenze  
set Quantità = ?

***M2 – Visualizza giacenze***

select \*  
from Giacenze g  
where g.Magazzino = ?

***M3 – Visualizza giacenze sotto la soglia minima***

select \*  
from Giacenze g  
where g.Magazzino = ? and g.Quantità < g.Giacenza\_minima

***M4 – Registra ordine***

insert into ordini(Numero\_fattura, Quantità, Data\_ordine, Data\_consegna, Nome, Nome\_fornitore, Codice\_prodotto)   
values (?,?,?,?,?,?,?)

***M5 – Registra consegna***

insert into consegne (Numero\_fattura, Nome\_magazzino, Nome\_cliente, Data\_Consegna) values (?,?,?,?)

***M6 – Crea giacenza***

insert into giacenze (Codice\_prodotto, Magazzino, Settore, Quantità, Giacenza\_minima) values (?,?,?,?,?)

***V1 – Vendita Prodotto***

insert into vendite (Numero\_Fattura, Quantita, Data, Sconto, Codice\_prodotto, Nome\_cliente, Partita\_IVA, Telefono)  
values (?,?,?,?,?,?,?,?)

***V2 – Aggiunta Cliente***

insert into clienti (Nome, Tipologia, Indirizzo, Telefono, Codice\_zona)   
values (?,?,?,?,?)

***G1 – Ricerca Cliente***

select \*  
from Clienti c  
where c.Nome = ?

***G2 – Ricerca Prodotto***

Select \*  
from Prodotti p  
where p.ID\_prodotto = ?

***A1 – Registra Prodotto***

insert into Prodotti (Codice\_prodotto, Nome, Capacità, Marca, Prezzo\_unitario, Tipologia, Confezione)  
values (?,?,?,?,?,?,?)

***A2 – Registra Fornitore***

insert into Fornitori (Nome, Indirizzo, Telefono, Partita\_IVA)  
values(?,?,?,?)

***A3 – Registra Accordo***

insert into Accordi (Nome\_fornitore, Codice\_prodotto, Prezzo\_di\_acquisto, Data\_scadenza, Codice\_dipendente)  
values (?,?,?,?,?)

***A4 – Aggiorna Accordo***

update Accordi  
set Prezzo\_di\_acquisto = ?, Data\_scadenza = ?

***A5 – Ricerca Accordo***

select \*  
from Accordi a  
where a.Nome\_Fornitore = ? and a.Codice\_prodotto = ?

***A6 – Visualizza Agenti***

select \*  
from Agenti

***A7 – Visualizza Dipendenti***

select \*  
from Dipendenti

***A8 – Registra Dipendente***

Insert into Dipendenti (Codice\_dipendente, Nome, Cognome, Stipendio, Indirizzo, Telefono, Ruolo, Ufficio)  
values (?,?,?,?,?,?,?,?)

***A9 – Registra Agente***

Insert into Agenti (Nome, Cognome, Partita\_IVA, Provvigione, Telefono, Codice\_zona)  
values (?,?,?,?,?,?)

***A10 – Calcolo Provvigioni***

Select p.prezzo\_unitario\*v.quantità\*a.provvigione  
from prodotti p, agenti a, vendite v  
where v.Partita\_IVA\_agente = a.Partita\_IVA  
and p.codice\_prodotto = v.codice\_prodotto  
and a.partita\_IVA = ?

# Progettazione dell’applicazione

L’applicazione Java che sviluppiamo sarà molto semplice, l’unico requisito richiesto è che sia possibile effettuare le operazioni necessarie e che sia possibile visualizzare correttamente i risultati delle query.

Il progetto utilizza la libreria JDBC, il DB risiede in loca ed è stato utilizza MySQL Workbench come DBMS.

Quando l’applicazione viene lanciata si apre una vista nella quale, attraverso un apposito menu a tendina, è possibile scegliere il tipo di mansione che si svolge (Magazziniere, Venditore, Amministrativo).

Una volta selezionato il tipo di utente, si aprirà una nuova schermata con un ulteriore menu a tendina, nel quale si deve selezionare l’operazione da effettuare.

Solo per i magazzinieri, prima di scegliere l’operazione, è necessario specificare a quale magazzino appartengono.