

Report Database

Federico Diotallevi

2023/2024

Contents

1	Analisi dei requisiti	3
1.1	Intervista	3
1.2	Estrazione dei concetti principali	4
1.3	Elaborazione dei concetti principali	4
2	Progettazione concettuale	6
2.1	Schema Scheletro	6
3	Progettazione logica	10
3.1	Stima del volume di dati	10
3.2	Descrizione delle operazioni principali e stima della loro frequenza	11
3.3	Schemi di navigazione e tabelle degli accessi	11
3.3.1	Operazione 1 - Aggiornare prodotti	12
3.3.2	Operazione 2 - Aggiungere Tavoli	12
3.3.3	Operazione 3 - Compilare Comande	12
3.3.4	Operazione 4 - Mostrare le comande non completate in ordine di arrivo	12
3.3.5	Operazione 5 - Mostrare il numero di tavoli e clienti serviti in una sera	13
3.3.6	Operazione 6 - Ricerca prodotti per categorie e sottocategorie	13
3.3.7	Operazione 7 - Ricerca prodotti per nome	13
3.3.8	Operazione 8 - Visualizzare prodotti non pagati in un tavolo	13
3.3.9	Operazione 9 - Compilare scontrini	13
3.3.10	Operazione 10 - Visualizzare i guadagni in un dato periodo	14
3.3.11	Operazione 11 - Compilare gli ordini del magazzino	14
3.3.12	Operazione 12 - Visualizzare le spese per i rifornimenti in un dato periodo	14
3.3.13	Operazione 13 - Visualizzare le prenotazioni per un dato giorno	14
3.3.14	Operazione 14 - Visualizzare la ricetta di un prodotto	15
3.3.15	Operazione 15 - Aggiornare staff del locale	15
3.4	Riepilogo dei costi stimati per operazione	15
3.5	Raffinamento dello schema	16
3.5.1	Eliminazione delle gerarchie	16
3.5.2	Eliminazione attributi composti	16
3.5.3	Eliminazione degli identificatori esterni	16
3.6	Analisi delle ridondanze	17
3.7	Traduzione di entità e associazioni in relazioni	18

1 Analisi dei requisiti

Si vuole realizzare un database a supporto dell'automazione della gestione di un esercizio commerciale attivo nella ristorazione (bar/ristoranti). Il database dovrà quindi memorizzare le informazioni riguardanti i tavoli con le relative comande, prodotti e scontrini. I dipendenti del bar potranno visualizzare le comande associate ai tavoli e compilare gli scontrini, mentre l'amministratore potrà visualizzare i dati relativi al fatturato.

1.1 Intervista

Un primo testo ottenuto dall'intervista è il seguente:

Si vuole tenere traccia delle comande e della gestione del magazzino di un bar/ristorante memorizzando i prodotti e le comande. Ogni prodotto è caratterizzato da un nome, una descrizione, un prezzo di vendita o, eventualmente, un prezzo di acquisto per la gestione del magazzino.

Al momento della creazione di una comanda, vengono registrati i prodotti associati, la quantità di ciascun prodotto e il tavolo a cui la comanda è destinata. I tavoli possono essere identificati sia da un indice numerico sia da un nome scelto al momento (ad esempio "Tavolo 1" o "Tavolo Pippo").

Il sistema permette di redigere uno o più scontrini per ogni tavolo, consentendo di dividere la spesa tra più clienti se necessario. Gli scontrini riportano il prezzo di ogni prodotto e il totale della spesa. Si tenga presente che anche se in futuro si decidesse di eliminare con una certa periodicità lo storico dei tavoli serviti, gli scontrini non devono comunque essere eliminati.

Il database mantiene uno storico di tutti i pagamenti riscossi, consentendo la generazione di grafici per monitorare il fatturato sia annuale che mensile, o l'andamento delle vendite di ciascun prodotto.

Il bar/ristorante dispone di uno o più amministratori con la possibilità di aggiungere nuovi utenti e visualizzare i dati sul fatturato. Gli utenti possono essere suddivisi nei seguenti ruoli:

- Camerieri: possono redigere le comande per i tavoli.
- Baristi/Cuochi: possono visualizzare le comande in arrivo e segnalarle come pronte una volta preparate.
- Magazzinieri: possono stilare la lista dei prodotti presenti in magazzino, annotare i prodotti da ricomprare e gestire il loro prezzo di acquisto.

Ogni comanda viene registrata con la data e l'ora in cui è stata creata e può essere modificata solo da utenti autorizzati fino a quando non viene chiusa con l'emissione di uno scontrino.

Gli amministratori possono anche visualizzare report dettagliati che comprendono:

- Elenco dei prodotti più venduti.
- Analisi del fatturato su base giornaliera, settimanale, mensile e annuale.
- Monitoraggio delle scorte di magazzino, inclusi avvisi per i prodotti che stanno per terminare.

In sintesi, il compito dello staff del bar/ristorante è assicurarsi che le comande vengano redatte e gestite correttamente, che i prodotti siano sempre disponibili in magazzino, e che le vendite e il fatturato vengano monitorati e analizzati per migliorare la gestione complessiva del locale.

1.2 Estrazione dei concetti principali

Termine	Breve descrizione	Eventuali Sinonimi
Prodotto	Sono gli articoli venduti dall'esercizio, ognuno con un proprio costo	
Tavolo	Oggetto a cui è possibile attribuire un nome e una data, rappresenta il gruppo dei clienti a cui si riferiscono comande e scontrini	Clienti
Comanda	Una comanda è un ordine effettuato in un qualsiasi momento dai clienti di un tavolo, contenente uno o più prodotti	
Scontrino	Oggetto in cui viene riportata una spesa effettuata da un tavolo, possono esserci uno o più scontrini associati ad un tavolo	Spesa
Cameriere	Colui che compila comande e scontrini	Cassiere
Preparatore	Colui che segna le comande come completate	Cuoco, Bartender
Magazziniere	Colui che segna la disponibilità dei prodotti nel magazzino	

1.3 Elaborazione dei concetti principali

A seguito della lettura e comprensione dei requisiti si redige un testo che ne riassume tutti i concetti, con particolare attenzione a quelli principali ed eliminando le ambiguità:

Per ogni **prodotto**¹ nel menù vengono memorizzati *nome*, *descrizione*, *prezzo di vendita*, inoltre ogni prodotto possiede un *codice* univoco assegnatogli quando viene aggiunto al menù. Se il prodotto è un semplice prodotto da magazzino allora deve riportare anche il proprio *costo*.

Ogni **comanda** possiede *data e ora*, può essere in preparazione o completata ed ha un *codice univoco* all'interno del tavolo a cui è collegata. Ad ogni comanda sono associati uno o più prodotti insieme alle *quantità* richieste. Deve essere possibile verificare che tutti i prodotti appartenenti ad una comanda sono stati pagati. Un **tavolo** è caratterizzato da: *un codice univoco*, *un nome (scelto al momento della creazione)*, *un numero di persone* sedute al tavolo e *una data di creazione*.

Uno **scontrino** rappresenta la somma dei costi di uno o più prodotti presenti in una comanda associata ad un tavolo, l'insieme degli scontrini erogati in un tavolo rappresenta la spesa totale. Ogni scontrino ha un *codice univoco* indipendente dal tavolo, *una data e un orario*, *una spesa* e *una modalità di pagamento*².

Lo staff è composto da:

- **amministratori**: hanno la possibilità di aggiungere membri dello staff e visualizzare info sulle vendite
- **preparatori**: preparano le comande e le segnano come completate e aggiornano i prodotti in menu.

¹Un prodotto può essenzialmente essere un prodotto preparato (es: un drink), un prodotto da magazzino (es: farina) oppure entrambi (es: bottiglia di vino).

²i.e. contante, bancomat...

- **magazzinieri**: ordinano prodotti del magazzino e ne segnano la disponibilità.
- **camerieri**: aggiungono tavoli e compilano comande e scontrini.

Segue un elenco delle principali azioni richieste:

1. *Aggiornare prodotti*
2. *Aggiungere tavoli*
3. *Compilare comande*
4. *Mostrare le comande non completate in ordine di arrivo*
5. *Mostare il numero di tavoli e clienti serviti in una sera*
6. *Ricerca prodotti per categorie e sottocategorie*
7. *Ricerca prodotti per nome*
8. *Visualizzare prodotti non pagati in un tavolo*
9. *Compilare scontrini*
10. *Visualizzare i guadagni in un dato periodo*
11. *Compilare ordini del magazzino*
12. *Visualizzare le spese per i rifornimenti in un dato periodo*
13. *Visualizzare le prenotazioni per un dato giorno*
14. *Visualizzare la ricetta di un prodotto*
15. *Aggiornare staff del locale*

2 Progettazione concettuale

2.1 Schema Scheletro

L'entità **prodotto** viene identificata tramite un codice univoco, dall'analisi si evince che ne esistono due tipologie:

- prodotti da magazzino
- prodotti in menu

queste tipologie sono specializzazioni *sovrapponibili* dell'entità prodotto.

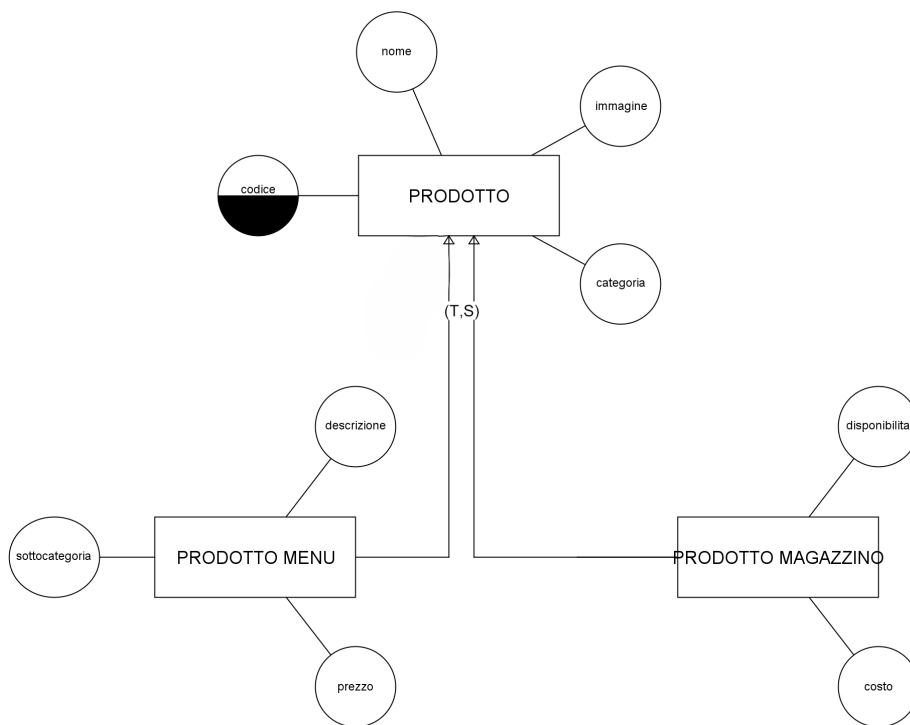


Figure 1: Schema parziale sulle specializzazioni di prodotto

Uno stesso **prodotto** può presentarsi nella stessa **comanda** con richieste e quantità differenti (si pensi ad esempio ad una stessa bevanda, nello stesso tavolo, ordinata allo stesso momento, una con ghiaccio, una senza). E' quindi necessario reificare l'entità *prodotto in comanda*, identificata tramite un ordine nella comanda e il codice della comanda.

Ad ogni comanda corrisponde un **tavolo**, identificato anch'esso tramite un *codice univoco*. Ad ogni tavolo è possibile assegnare un *nome* e un *numero di clienti* seduti.

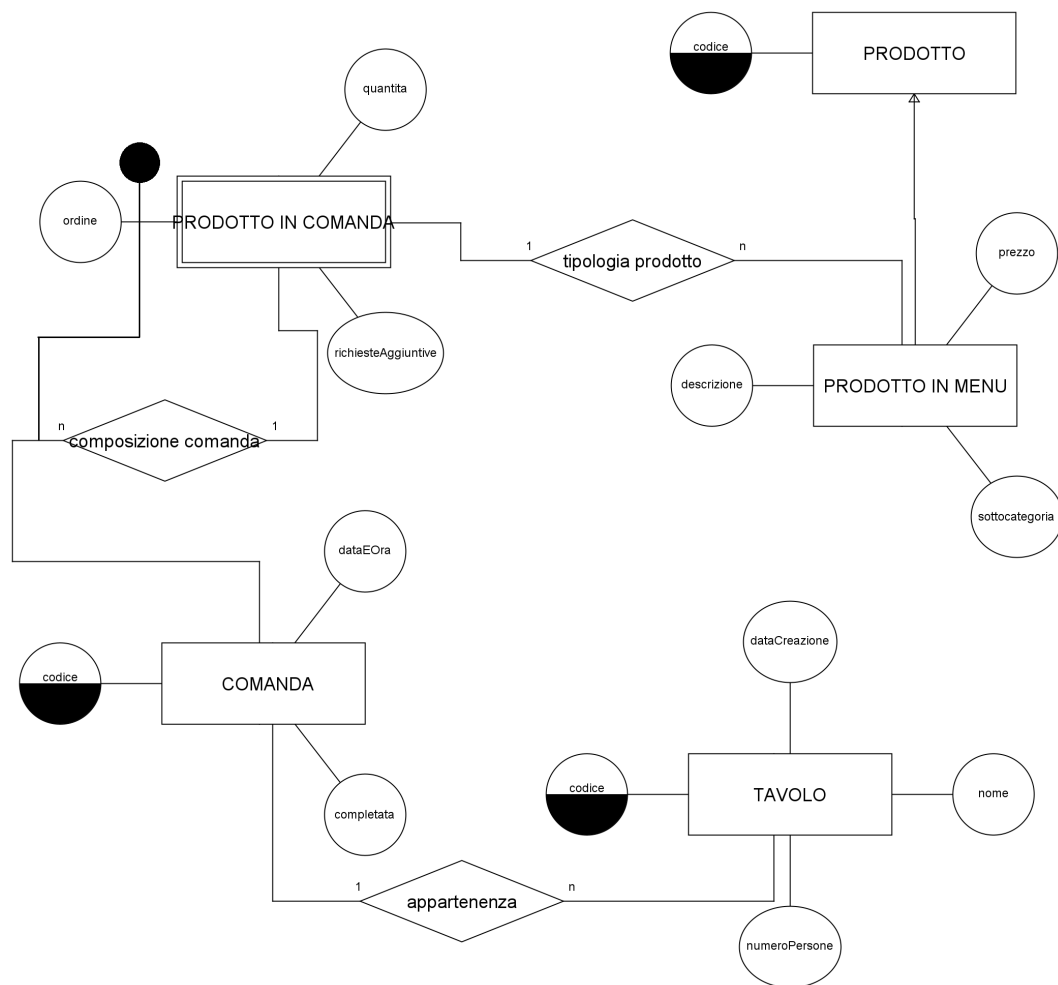


Figure 2: Schema parziale sulle relazioni tra tavoli, comande e prodotti

Le entità *cameriere*, *preparatore*, *amministratore* e *magazziniere* non sono altro che specializzazioni dell'entità *persona* (o dipendente), ognuna identificata tramite *codice fiscale*. Di ogni dipendente si vuole mantenere nel databse il ruolo all'interno del locale per conoscere le operazioni che possono effettuare (ad esempio la visualizzazione dei dati sul fatturato è un'operazione che spetta esclusivamente ad un amministratore). In particolare per ogni comanda si vuole tenere nel database il cameriere che l'ha compilata.

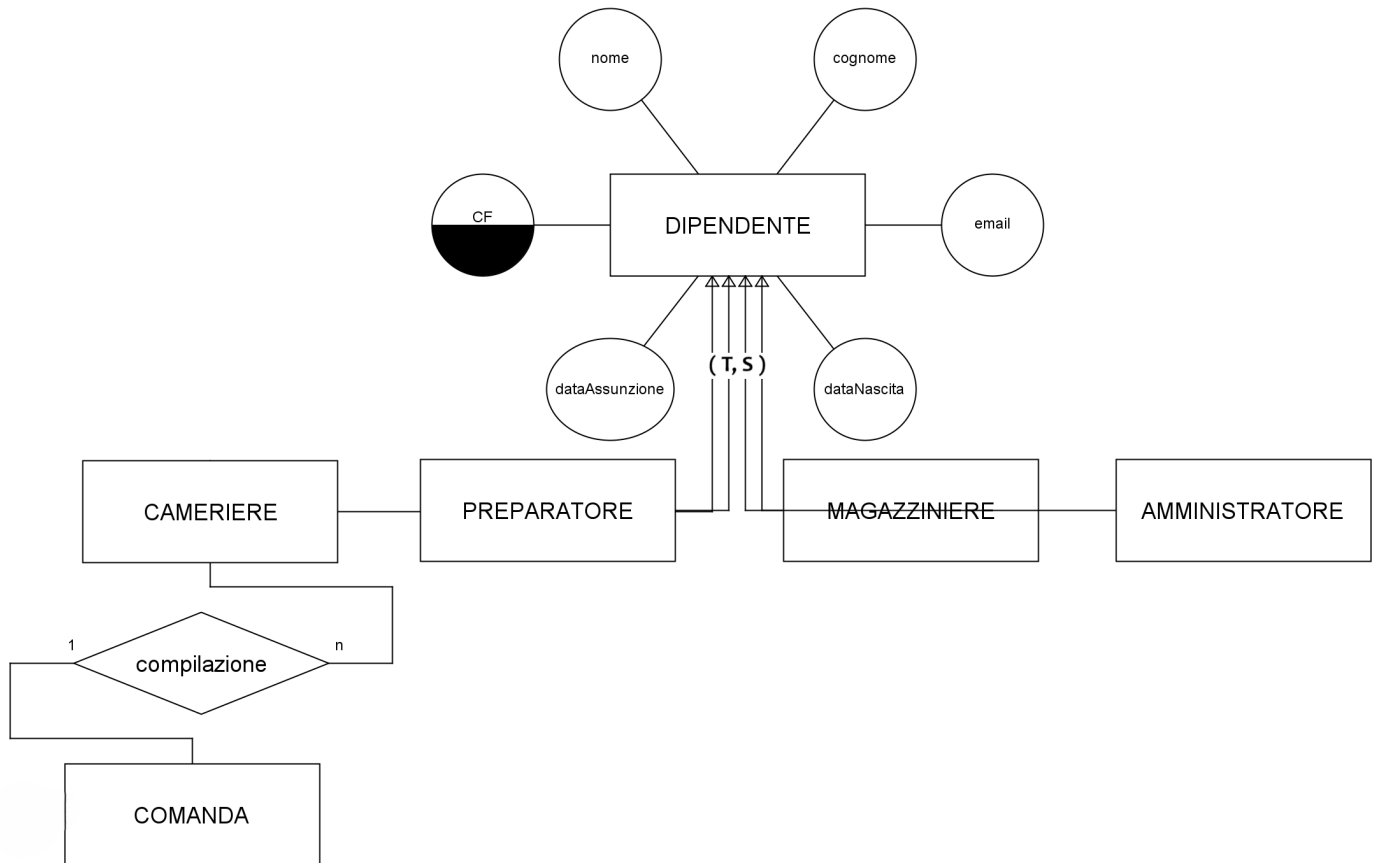


Figure 3: Schema parziale sulle specializzazioni dei dipendenti

Uno stesso *prodotto* può presentarsi in uno stesso *ordine* al più una volta, perciò la relazione di *ordinazione* possiede l'attributo *quantità* esprimendo il vincolo per il quale le stesse istanze di ordine e prodotto non possono presentarsi con molteplici quantità, creando confusione nell'ordine. Per la *disponibilità* in magazzino del prodotto si è aggiunto un semplice attributo alla specializzazione *prodotto in magazzino*.

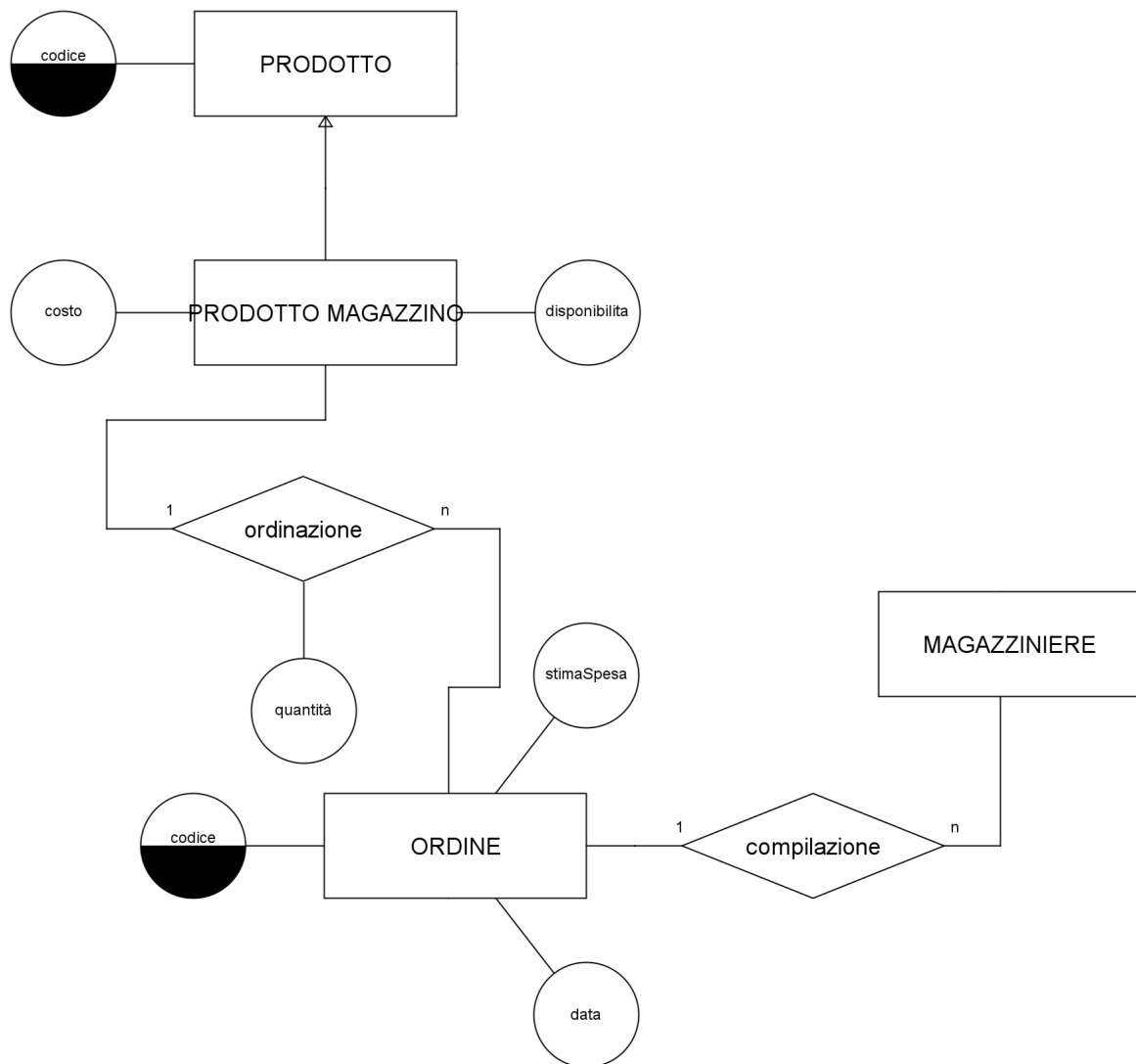


Figure 4: Schema parziale sugli ordini dei prodotti

Di seguito si allega lo schema ER nel suo complesso, importato su [DB-Main](#).

3 Progettazione logica

3.1 Stima del volume di dati

Di seguito la stima dei volumi richiesti per entità e relazioni:

Concetto	Costrutto	Volume
Menu Product	E	80
Stocked Up Product	E	250
Ingredient	R	320 ³
Variation	E	480 ⁴
customer choice	R	480

Concetto	Costrutto	Volume
Table	E	150.000
Product In Table	E	750.000 ⁵
Customer Order	E	300.000 ⁶
based on	R	750.000
additional request	R	750.000 ⁷
Receipt	E	300.000 ⁸
Paid Product	E	1.000.000 ⁹
product payment	R	1.000.000
receipt composition	R	1.000.000
table payment	R	300.000
order composition	R	750.000
assignment	R	300.000
order compilation	R	300.000
Reservation	E	5.000
request	R	5.000

Concetto	Costrutto	Volume
Supplier	E	10
supply	R	750

³Considerando una media di 4 ingredienti per prodotto in menu

⁴Considerando 6 variazioni per ogni prodotto in menu

⁵Considerando circa 5 diversi prodotti ordinati in un tavolo

⁶Considerando circa 2 comande per tavolo

⁷Considerando che ogni prodotto ordinato presenta una variazione

⁸Considerando circa 2 scontrini per tavolo

⁹Considerando che per ogni prodotto al tavolo deve esistere almeno un corrispondente prodotto pagato e che alcuni potrebbero essere pagati in scontrini diversi

Concetto	Costrutto	Volume
provision	R	10.000
type	R	10.000
Supply Item	E	10.000
inclusion	R	10.000
Stock Order	E	1.000

Concetto	Costrutto	Volume
Waiter	E	25
order compilation	R	100.000
Kitchen Staff	E	25
Admin	E	5
Storekeeper	E	10
order planning	R	1.000

3.2 Descrizione delle operazioni principali e stima della loro frequenza

Le operazioni da effettuare sono quelle precedentemente elencate nella fase di analisi. Di seguito vengono elencate insieme alla relativa frequenza:

Numero operazione	Operazione	Frequenza
1.	Aggiornare prodotti	50 all'anno
2.	Aggiungere tavoli	200 a settimana
3.	Compilare comande	800 a settimana
4.	Mostrare le comande non completate in ordine di arrivo	30 al giorno
5.	Mostare il numero di tavoli e clienti serviti in una sera	2 al giorno
6.	Ricerca prodotti per categorie e sottocategorie	10 al giorno
7.	Ricerca prodotti per nome	4000 a settimana
8.	Visualizzare prodotti non pagati in un tavolo	2400 a settimana
9.	Compilare scontrini	600 a settimana
10.	Visualizzare i guadagni in un dato periodo	10 al mese
11.	Compilare ordini del magazzino	3 a settimana
12.	Visualizzare le spese per i rifornimenti in un dato periodo	10 al mese
13.	Visualizzare le prenotazioni per un dato giorno	100 a settimana
14.	Visualizzare la ricetta di un prodotto	20 al giorno
15.	Aggiornare staff del locale	5 all'anno

3.3 Schemi di navigazione e tabelle degli accessi

Sono riportate in seguito le tabelle degli accessi delle operazioni elencate sopra. Per il calcolo dei costi le operazioni in scrittura sono considerate con peso doppio rispetto a quelle in lettura.

3.3.1 Operazione 1 - Aggiornare prodotti

L'aggiornamento di un *prodotto* comporta anche l'aggiornamento della ricetta, possiamo considerare che in media ogni prodotto ha 3 *ingredienti* e ogni *prodotto di magazzino* è *ingrediente* di 3 *prodotti da menù*.

Concetto	Costrutto	Accessi	Tipo
Product	E	1	L
Product	E	1	S
Ingredient	R	3	L
Ingredient	R	3	S
Totale:		$4S + 4L \rightarrow 500$ all'anno	

3.3.2 Operazione 2 - Aggiungere Tavoli

Concetto	Costrutto	Accessi	Tipo
Table	E	1	S
Totale:		$1S \rightarrow 400$ a settimana	

3.3.3 Operazione 3 - Compilare Comande

La compilazione delle *comande* richiede delle ricerche di *prodotti di menu*, che in media sono 3 a comanda, successivamente occorrono degli accessi in lettura alle *variazioni*, degli aggiornamenti nei *prodotti al tavolo* e degli accessi in scrittura nelle *richieste aggiuntive*.

Concetto	Costrutto	Accessi	Tipo
Menu Product	E	3	L
Variation	E	3	L
additional request	R	3	S
Product In Table	E	3	L
Product In Table	E	3	S
ordination	R	3	S
Customer Order	E	1	S
Totale:		$10S + 9L \rightarrow 23.200$ a settimana	

3.3.4 Operazione 4 - Mostrare le comande non completate in ordine di arrivo

Considerando 3 prodotti per comanda bisogna accedere in lettura anche alle *ordinazioni* per ricostruire l'intera comanda.

Concetto	Costrutto	Accessi	Tipo
Customer Order	E	1	L
ordination	R	3	L
Totale:		$4L \rightarrow 120$ al giorno	

3.3.5 Operazione 5 - Mostrare il numero di tavoli e clienti serviti in una sera

Concetto	Costrutto	Accessi	Tipo
Table	E	1	L
Totale:		1L → 2 al giorno	

3.3.6 Operazione 6 - Ricerca prodotti per categorie e sottocategorie

Concetto	Costrutto	Accessi	Tipo
Menu Product	E	1	L
Stocked Up Product	E	1	L
Totale:		2L → 20 al giorno	

3.3.7 Operazione 7 - Ricerca prodotti per nome

Concetto	Costrutto	Accessi	Tipo
Menu Product	E	1	L
Stocked Up Product	E	1	L
Totale:		2L → 8000 a settimana	

3.3.8 Operazione 8 - Visualizzare prodotti non pagati in un tavolo

Per visualizzare i prodotti non pagati è sufficiente accedere prima al *tavolo* e successivamente ai *prodotti nel tavolo*, che in media in un *tavolo* sono 5.

Concetto	Costrutto	Accessi	Tipo
Table	E	1	L
Product In Table	E	5	L
Totale:		6L → 14.400 a settimana	

3.3.9 Operazione 9 - Compilare scontrini

Per compilare uno *scontrino* è necessario innanzitutto una visualizzazione dei prodotti non pagati in un tavolo. Successivamente bisogna aggiungere dei *prodotti pagati*, in media 3 per *scontrino*, e aggiornare l'attributo *numPaid* nei relativi *prodotti ordinati*.

Concetto	Costrutto	Accessi	Tipo
Receipt	E	1	S
Table	E	1	L
Product In Table	E	5	L
Paid Product	E	3	S
Product In Table	E	3	S

Concetto	Costrutto	Accessi	Tipo
Totale: 7S + 8L → 13.200 a settimana			

3.3.10 Operazione 10 - Visualizzare i guadagni in un dato periodo

E' sufficiente accedere ai dati dei pagamenti degli *scontrini*.

Concetto	Costrutto	Accessi	Tipo
Receipt	E	1	L
Totale: 1L → 10 al mese			

3.3.11 Operazione 11 - Compilare gli ordini del magazzino

Per compilare un *ordine di magazzino* è necessario accedere ai *prodotti di magazzino*, in media 20 per ordine, scegliere il *fornitore*, in media sono 4, confrontando il *costo*. Occorre poi creare gli *elementi di fornitura* con le quantità scelte.

Concetto	Costrutto	Accessi	Tipo
Stocked Up Product	E	20	L
supply	R	4	L
Supply Item	E	20	S
Supply Item	E	20	L
Stock Order	E	1	S
Totale: 21S + 44L → 258 a settimana			

3.3.12 Operazione 12 - Visualizzare le spese per i rifornimenti in un dato periodo

Considerando come riferimento una settimana¹⁰:

Concetto	Costrutto	Accessi	Tipo
Stock Order	E	3	L
Supply Item	E	60	L
Totale: 63L ¹¹ → 189 a settimana			

3.3.13 Operazione 13 - Visualizzare le prenotazioni per un dato giorno

Si possono considerare in media 5 prenotazioni al giorno, per ogni prenotazione si vuole visualizzare il *tavolo* previsto se esiste.

¹⁰Ci si attiene alle stime fatte precedentemente, che erano su base settimanale. Il costo può essere moltiplicato per trovare la stima mensile e/o annuale.

¹¹Se si volesse ottenere il costo di una lettura su base mensile fatta 3 volte a settimana il costo diventerebbe:

$Totale = 63L * 4 * 3 = 756 \text{ a settimana}$

Concetto	Costrutto	Accessi	Tipo
Prenotation	E	1	L
Table	E	1	L
Totale:		2L → 200 a settimana	

3.3.14 Operazione 14 - Visualizzare la ricetta di un prodotto

Per ogni *prodotto in menu* consideriamo una media di 3 *ingredienti*, come già fatto in precedenza.

Concetto	Costrutto	Accessi	Tipo
Menu Product	E	1	L
Stocked Up Product	E	3	L
ingredient	R	3	L
Totale:		7L → 140 al giorno	

3.3.15 Operazione 15 - Aggiornare staff del locale

Concetto	Costrutto	Accessi	Tipo
Employee	E	1	S
Totale:		1S → 10 all'anno	

3.4 Riepilogo dei costi stimati per operazione

Di seguito viene stilata una tabella riassuntiva coi costi sopraelencati:

Indice	Operazione	Frequenza	Costo stimato
1.	Aggiornare prodotti	50 all'anno	500 all'anno
2.	Aggiungere tavoli	200 a settimana	400 a settimana
3.	Compilare comande	800 a settimana	23.200 a settimana
4.	Mostrare le comande non completate in ordine di arrivo	30 al giorno	120 al girono
5.	Mostare il numero di tavoli e clienti serviti in una sera	2 al giorno	2 al giorno
6.	Ricerca prodotti per categorie e sottocategorie	10 al giorno	20 al giorno
7.	Ricerca prodotti per nome	4000 a settimana	8000 a settimana
8.	Visualizzare prodotti non pagati in un tavolo	2400 a settimana	14.400 a settimana

Indice	Operazione	Frequenza	Costo stimato
9.	Compilare scontrini	600 a settimana	13.200 a settimana
10.	Visualizzare i guadagni in un dato periodo	10 al mese	10 al mese
11.	Compilare ordini del magazzino	3 a settimana	258 a settimana
12.	Visualizzare le spese per i rifornimenti in un dato periodo	10 al mese	189 a settimana
13.	Visualizzare le prenotazioni per un dato giorno	100 a settimana	200 a settimana
14.	Visualizzare la ricetta di un prodotto	20 al giorno	140 al giorno
15.	Aggiornare staff del locale	5 all'anno	10 all'anno

3.5 Raffinamento dello schema

3.5.1 Eliminazione delle gerarchie

Per la gerarchia dei *dipendenti*, poiché la copertura è sovrapposta e le diverse specializzazioni non presentano attributi aggiuntivi, ma si configurano più come delle specie di permessi per effettuare determinate operazioni, si è scelto il collasso verso l'alto, aggiungendo all'entità *dipendente* tanti attributi booleani quante sono le specializzazioni.

Per la gerarchia dei *prodotti* si è invece scelto un collasso verso il basso, replicando gli attributi di *prodotto* in *prodotto in menu* e *prodotto in magazzino*. Si adotta questa strategia in quanto la ridondanza, essendo non esclusiva, è presente, ma effettivamente trascurabile. La quantità di prodotti che sono sia prodotti da magazzino che prodotti in menù è infatti scarsa (ad esempio bevande e pochi altri prodotti). Inoltre le operazioni svolte sui prodotti in menù vengono effettuate molto più spesso di quanto venga fatto per i prodotti in magazzino.

3.5.2 Eliminazione attributi composti

Nello schema è presente l'attributo composto *indirizzo* nell'entità *dipendente*. Tale attributo è stato diviso nelle sue sotto componenti, si renderà necessario poi, a livello applicativo, che tali valori siano impostati in modo coerente tra loro.

3.5.3 Eliminazione degli identificatori esterni

Nello schema E/R sono eliminate le seguenti relazioni:

- **ingredient**: reificata importando *menuProdId* da *Menu Product* e *ingredientId* da *Stocked Up Product*
- **based on**: eliminata importando *menuProdId* da *Menu Product* a *Product In Table*
- **customer choice**: eliminata importando *prodId* da *Menu Product* a *Variation*
- **additional request**: reificata importando *menuProdId* da *Menu Product*, *tableId*, *orderedProdId* da *Product in Table* e *variationId* da *Variation*
- **ordination**: reificata importando *menuProdId* da *Menu Product*, *orderNum* da *Customer Order*, *orderedProdId* da *Product In Table* e *tableId*

- **assignment**: eliminata importando *tableId* da *Table* a *Customer Order*
- **order composition**: eliminata importando *tableId* in *Product in Table*
- **product payment**: eliminata importando *orderedProdId* da *Product In Table*, *menuProdId* da *Menu Product* e *tableId* in *Paid Product*
- **order compilation**: eliminata importando *waiterId* da *Waiter* a *Customer Order*
- **receipt composition**: eliminata importando *receiptId* da *Receipt* a *Paid Product*
- **table payment**: eliminata importando *tableId* in *Receipt*
- **request**: eliminata importando *tableId* da *Table* a *Reservation*
- **supply**: reificata importando *prodId* da *Stocked Up Product* e *supplierName* da *Supplier*
- **type**: eliminata importando *prodId* da *Stocked Up Product* a *Supply Item*
- **provision**: eliminata importando *supplierName* da *Supplier* a *Supply Item*
- **inclusion**: eliminata importando *orderDate* da *Stock Order* a *Supply Item*
- **order planning**: eliminata importando *storekeeperId* da *Storekeeper* a *Stock Order*

3.6 Analisi delle ridondanze

Nell'entità *Product In Table* gli attributi *numPaid*, *quantity* e *finalPrice* sono ridondanze. Tuttavia, a discapito di un piccolo overhead in termini di memoria si ha vantaggio nelle operazioni più eseguite, infatti:

- Per reperire *numPaid* occorrerebbero, in media, 2 accessi in lettura a *Paid Product*
- Per ottenere *quantity* occorrerebbero, in media, 2 accessi in lettura a *ordination*
- Per conoscere la quantità dei **non pagati** occorrerebbero quindi almeno 4 accessi in lettura in più ogni volta
- Per ottenere *final price* occorrerebbe per *Product In Table* un accesso in lettura a *Menu Product*, *additional request* e *Variation*, quindi circa 3 accessi in lettura in più per prodotto considerando una sola variazione

Operazione 8:

con ridondanza:

Concetto	Costrutto	Accessi	Tipo
Table	E	1	L
Product In Table	E	5	L
Totale:		6L → 14.400 a settimana	

senza ridondanza:

Concetto	Costrutto	Accessi	Tipo
Table	E	1	L
Product In Table	E	5	L
Paid Product	E	10	L
ordination	E	10	L
Totale:		26L → 62.400 a settimana	

Operazione 9:

con ridondanza:

Concetto	Costrutto	Accessi	Tipo
Receipt	E	1	S
Table	E	1	L
Product In Table	E	5	L
Paid Product	E	3	S
Product In Table	E	3	S
Totale:		$7S + 8L \rightarrow 13.200$ a settimana	

senza ridondanza:

Concetto	Costrutto	Accessi	Tipo
Receipt	E	1	S
Table	E	1	L
Product In Table	E	5	L
Paid Product	E	10	L
ordination	E	10	L
Paid Product	E	3	S
Variation	E	3	L
additional request	E	3	L
Totale:		$4S + 32L \rightarrow 24.000$ a settimana	

Si può notare come in entrambe le operazioni più frequenti la ridondanza semplifichi di tanto il costo complessivo stimato.

3.7 Traduzione di entità e associazioni in relazioni

- EMPLOYEES(employeeId, email, password, cf, name, surname, birthday, hiringDate, city, zipCode, streetName, isAdmin, isWaiter, isKitchenStaff, isStoreKeeper)
- MENU_PRODUCTS(prodId, name, imgFile, category, description, subcategory, price)
- STOCKED_UP_PRODUCTS(prodId, name, imgFile, category, subcategory, availability)
- INGREDIENTS(menuProdId, ingredientId, portion)

FK: menuProdId REFERENCES MENU_PRODUCTS

FK: menuProdId REFERENCES MENU_PRODUCTS

- VARIATIONS(variationId, menuProdId, additionalRequest, additionalPrice*)

FK: menuProdId REFERENCES MENU_PRODUCTS

- TABLES(tableId, creationTimestamp, name, seats)
- RESERVATIONS(cellNumber, dateAndTime, clientName, seats, tableId*)

FK: tableId REFERENCES TABLES

- RECEIPTS(receiptId, dateAndTime, total, paymentMethod, givenMoney*, change*, tableId*)

FK: tableId REFERENCES TABLES

- PRODUCTS_IN_TABLE(orderedProdId, menuProdId, tableId, quantity, finalPrice, hasVariations, numPaid)

FK: tableId REFERENCES TABLES

FK: menuProdId REFERENCES MENU_PRODUCTS

- PAID_PRODUCTS(orderedProdId, menuProdId, tableId, receiptId, quantity)

FK: receiptId REFERENCES RECEIPTS

FK: (orderedProdId, menuProdId, tableId) REFERENCES PRODUCTS_IN_TABLE

- ADDITIONAL_REQUESTS(variationId, tableId, orderedProdId, menuProdId)

FK: (variationId, menuProdId) REFERENCES VARIATIONS

FK: (orderedProdId, menuProdId, tableId) REFERENCES PRODUCTS_IN_TABLE

- CUSTOMER_ORDERS(orderNum, tableId, timestamp, inPreparation, delivered, waiterId)

FK: tableId REFERENCES TABLES

FK: waiterId REFERENCES EMPLOYEES

- ORDINATIONS(orderNum, menuProdId, tableId, orderedProdId, quantity)

FK: (orderNum, tableId) REFERENCES CUSTOMER_ORDERS

FK: (orderedProdId, menuProdId, tableId) REFERENCES PRODUCTS_IN_TABLE

- SUPPLIERS(companyName, email)

- SUPPLY_COSTS(prodId, companyName, cost)

FK: prodId REFERENCES STOCKED_UP_PRODUCTS

FK: companyName REFERENCES SUPPLIERS

- STOCK_ORDERS(creationTimestamp, sent, estimatedCost, storekeeperId)

FK: storekeeperId REFERENCES EMPLOYEES

- SUPPLY_ITEMS(prodId, orderDate, supplierName, quantity)

FK: supplierName REFERENCES SUPPLIERS

FK: orderDate REFERENCES STOCK_ORDERS

FK: prodId REFERENCES STOCKED: