

Cloudnine Manager

Software di gestione per ristoranti in formula all-you-can-eat



Luca Venturini

Federico Bagattoni

27 agosto 2024

Indice

1	Analisi	3
1.1	Requisiti	3
1.2	Operazioni richieste	4
1.2.1	Operazioni degli amministratori	4
1.2.2	Operazioni dei Camerieri	5
1.2.3	Operazioni dei Cuochi	6
1.2.4	Operazioni di tutti gli agenti	7
2	Progettazione concettuale	8
2.1	Schema iniziale	8
2.2	Raffinamenti proposti	9
2.2.1	Membri del personale	9
2.2.2	Menù e categorie delle vivande	9
2.2.3	Inventario e ricette	10
2.2.4	Gestione delle prenotazioni	11
2.3	Schema finale	11
2.4	Vincoli inespressi	13
3	Progettazione logica	14
3.1	Stima del volume dei dati	14
3.2	Descrizione delle operazioni principali e stima della loro frequenza	15
3.3	Schemi di navigazione e tabelle degli accessi	16
3.4	Raffinamento dello schema	20
3.5	Analisi delle ridondanze	20
3.5.1	Analisi della ridondanza di conto finale	20
3.5.2	Analisi della ridondanza di quantità totale	23
3.6	Traduzione di entità e associazioni in relazioni	25
3.7	Schema relazionale finale	25
3.8	Traduzione delle operazioni in query SQL	26

A Guida Utente	32
A.1 Log in e utenti	33

Capitolo 1

Analisi

1.1 Requisiti

Si vuole realizzare un software per la gestione di un ristorante in formula all-you-can-eat ed alla carta. Dall'interrogazione degli agenti del dominio si ricava la seguente descrizione in linguaggio naturale.

L'amministratore del locale richiede di poter gestire diversi aspetti del locale: i piatti nel menù appartenenti a varie categorie (antipasti, vari tipi di sushi rolls, primi piatti caldi, crudi...), l'inventario delle materie prime presenti in cella, gli ordini e le comande create dal personale di sala ed infine la chiusura del conto ed un resoconto dei ricavi di fine turno. Il menù deve poter essere modificato, aggiungendo o rimuovendo nuovi piatti o bevande appartenenti ad una sola categoria. Viene tenuto conto di quali e quante materie prime sono disponibili nell'inventario del locale, al fine di sapere quali piatti possono essere cucinati, così che il personale di sala possa comunicare al cliente se un piatto non è disponibile. Inoltre si controlla quali materie prime e confezioni siano in scadenza per poterle gettare oppure quali siano quelle che scarseggiano per ordinarle dal fornitore in anticipo, evitando di esaurirle completamente. Il personale della cucina è in grado di modificare le quantità nell'inventario togliendo quelle che ha usato durante un servizio, oltre che poter visualizzare le. Il personale del locale apre una comanda ogni volta che dei clienti si siedono ad un tavolo, ogni volta che i clienti eseguono un ordine (es. 4 pezzi del n. 75 e 5 pezzi del n. 80) questo viene aggiunto alla comanda e inviato alla cucina, poi servito. All'inizio del pasto i clienti scelgono tra il menù fisso all-you-can-eat ed il menù alla carta, infatti al momento della chiusura della comanda se è stata scelta la prima opzione verrà

calcolato il conto sommando un menù fisso per ogni commensale oppure scegliendo la seconda opzione verranno sommati tutti i prezzi dei singoli piatti ordinati, in entrambi i casi si aggiungono i prezzi delle bevande. La scelta del menù è uguale per tutti i clienti di uno stesso tavolo, non si possono avere clienti che scelgono due menù diversi presso lo stesso tavolo. Al fine di facilitare il servizio il gestore ha deciso di dividere la sala in zone ognuna assegnata ad un membro del personale di sala che gestisce sempre gli stessi tavoli. Infine le prenotazioni vengono segnate nell'agenda del ristorante ogni volta che un cliente contatta il locale. Il locale ha a disposizione un numero di coperti fisso, potendo disporli in qualsiasi combinazione di tavoli possibile.

1.2 Operazioni richieste

Nel DB vi sono tre classi di utenza:

- Amministratori
- Camerieri
- Cuochi

Notiamo che gli *Amministratori* hanno la possibilità di eseguire le azioni di tutti gli altri ruoli mentre i *Camerieri* ed i *Cuochi* sono limitati alle operazioni della loro mansione. *E' importante notare che le operazioni menzionate successivamente non comprendono tutte le operazioni che l'applicativo svolge.*

1.2.1 Operazioni degli amministratori

Un amministratore deve poter:

- Aggiungere, rimuovere e modificare il personale
- Aggiungere nuove materie prime all'inventario
- Visualizzare le materie prime la cui quantità è sotto una certa soglia critica
- Visualizzare i ricavi della giornata
- Aggiungere, rimuovere e modificare i menu
- tutte le operazioni segnate ai punti 1.2.2 e 1.2.3

Le operazioni per gli amministratori sono quindi definite nel seguente modo:

- A1 "AGGIUNGI STAFF": aggiunge un nuovo membro del personale
- A2 "RIMUOVI STAFF": rimuove un nuovo membro del personale
- A3 "MODIFICA STAFF": modifica uno o più dati riguardo ad un membro del personale
- A4 "RICAVI GIORNATA": visualizza ricavi della giornata
- A5 "NUOVO MENU": aggiunge un nuovo menù
- A6 "RIMUOVI MENU": rimuove un menù
- A7 "AGGIUNGI VIVANDA A MENU": aggiunge un vivanda ad un menù
- A8 "RIMUOVI VIVANDA DA MENU": rimuove un vivanda dal menù
- A9 "NUOVA VIVANDA": aggiunge una nuova vivanda al sistema
- A10 "CANCELLA VIVANDA": rimuove una vivanda dal sistema

1.2.2 Operazioni dei Camerieri

Un cameriere deve poter:

- Creare e modificare le comande
- Aggiungere, rimuovere o modificare gli ordini di una comanda
- Aggiungere, rimuovere o modificare le prenotazioni
- Visualizzare i piatti che non è possibile cucinare in un dato momento
- Chiudere una comanda, visualizzando il conto finale

Le operazioni per i camerieri sono quindi definite nel seguente modo:

- S1 "AGGIUNGI COMANDA": crea una nuova comanda generando il relativo codice ed aggiungendo il numero di coperti ed il tavolo
- S2 "MODIFICA COMANDA": modifica uno o più parametri di una comanda
- S3 "RIMUOVI COMANDA": elimina una comanda dal sistema

- S4 "AGGIUNGI ORDINE": aggiunge un ordine ad una comanda, contenente piatti o bevande e relative quantità
- S5 "RIMUOVI VIVANDA": elimina una vivanda da un ordine già esistente
- S6 "RIMUOVI ORDINE": rimuove un ordine da una comanda, se la cucina non l'ha ancora iniziato a cucinare
- S7 "CHIUDI COMANDA": chiudi una comanda, calcolando una sorta di scontrino finale
- S8 "AGGIUNGI VIVANDA": aggiunge una vivanda ad un ordine già esistente

1.2.3 Operazioni dei Cuochi

Un cuoco deve poter:

- Rimuovere quantità di materie prime dall'inventario quando se ne fa utilizzo
- Visualizzare le materie prime scadute
- Aggiungere una materia prima nuova o già esistente all'inventare
- Visualizzare le materie prime insufficienti

Le operazioni per i cuochi sono quindi definite nel seguente modo:

- K1 "RIMUOVI MATERIA PRIMA": Rimuovere una certa quantità di materia prima
- K2 "MATERIE PRIMA SCADUTE": Visualizzare le materie prime scadute
- K3 "AGGIUNGI MATERIA PRIMA": aggiungere una certa quantità di materia prima
- K4 "MATERIE PRIMA INSUFFICIENTI": Visualizzare le materie prime insufficienti

1.2.4 Operazioni di tutti gli agenti

Le operazioni che tutti gli agenti del dominio possono svolgere sono:

- Prendere prenotazioni

Che si traducono in:

- S9 "AGGIUNGI PRENOTAZIONE": aggiunge una nuova prenotazione

Capitolo 2

Progettazione concettuale

2.1 Schema iniziale

Di seguito viene riportato uno schema scheletro che verrà espanso nelle sezioni successive. Questo include le entità fondamentali dello schema:

- una vivanda
- un ordine
- una comanda
- un membro del personale

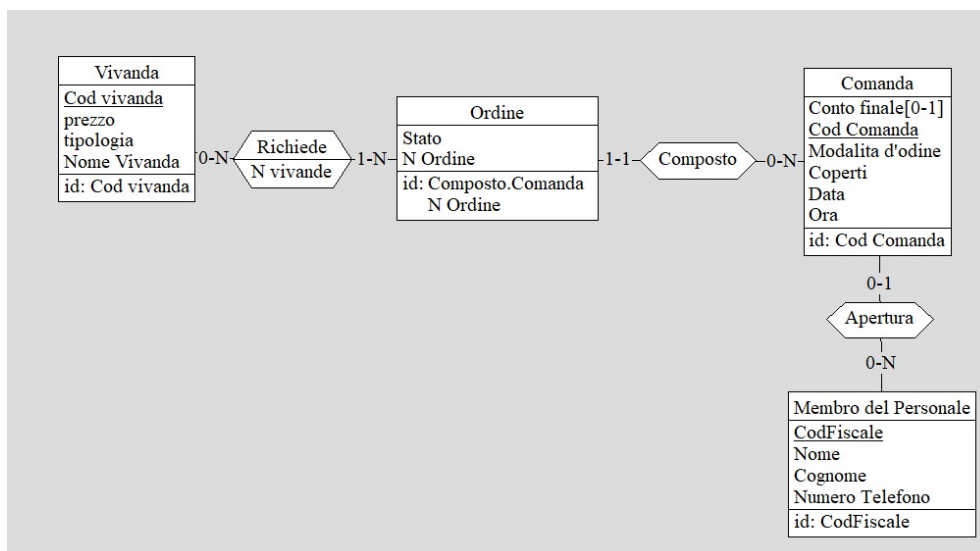


Figura 2.1: lo schema E-R iniziale.

2.2 Raffinamenti proposti

2.2.1 Membri del personale

Al fine di specificare i ruoli all'interno del ristorante si propone la seguente gerarchia:

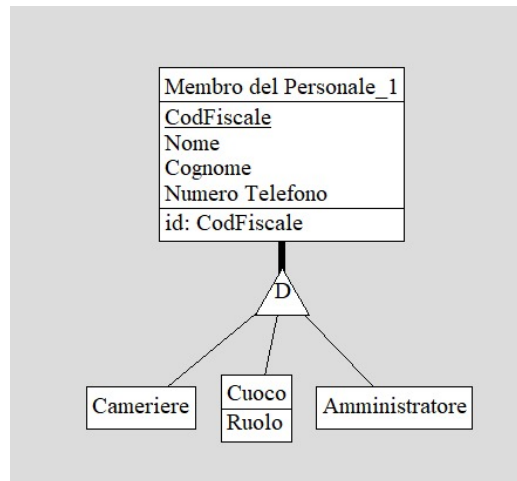


Figura 2.2: differenziazione dei membri del personale.

2.2.2 Menù e categorie delle vivande

Visto che la peculiarità del ristorante è la diversità in termini di piatti e categorie il sistema deve permettere l'esistenza di diversi menù ognuno proposto in giornate e servizi diversi.

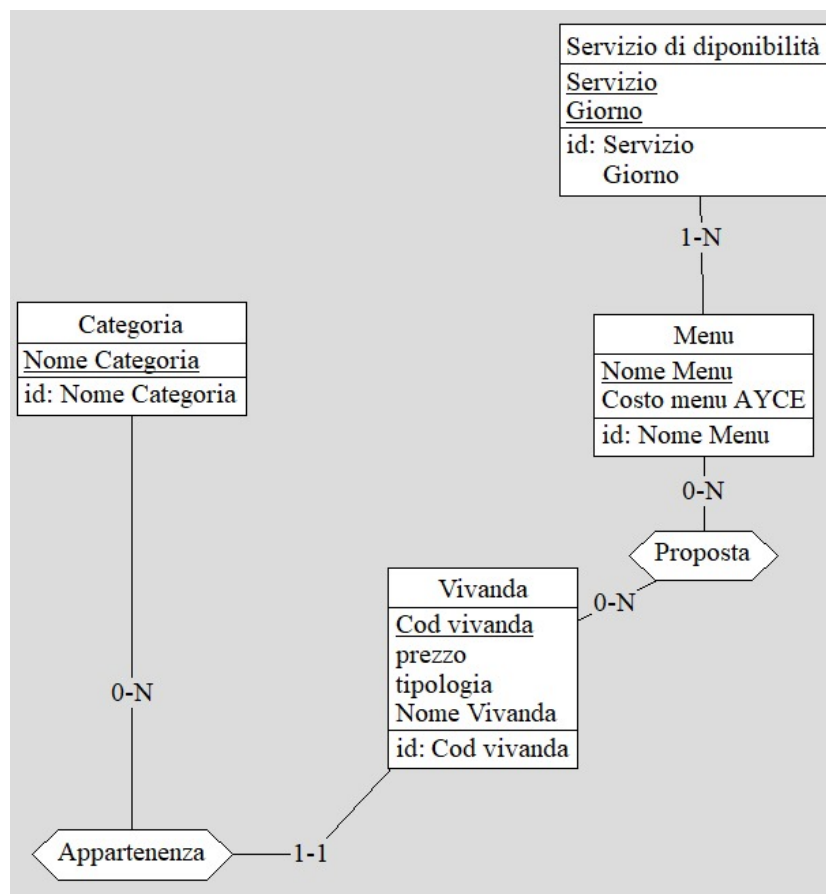


Figura 2.3: modellazione del menù e delle categorie di piatti al suo interno.

2.2.3 Inventario e ricette

Un requisito del sistema deve essere quello di tenere conto della quantità di materie prime disponibili in cucina, questo è reso possibile modellando le materie prime necessarie a cucinare un piatto.

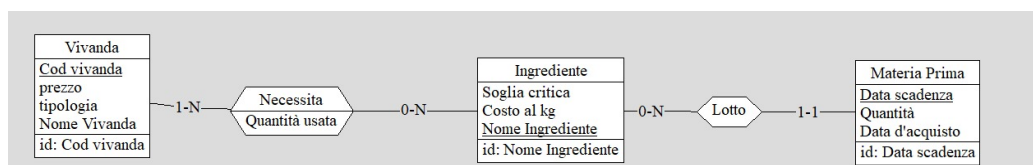


Figura 2.4: modellazione dell'inventario e della gestione delle materie prime.

2.2.4 Gestione delle prenotazioni

La gestione delle prenotazioni per una determinata data ed un certo turno è così modellata.

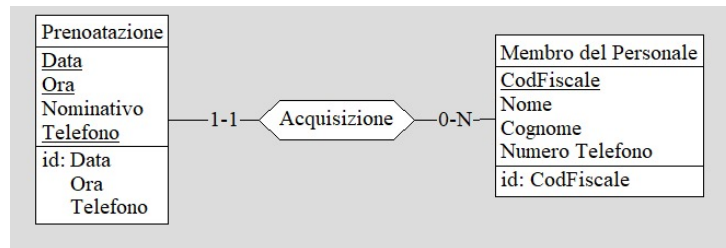


Figura 2.5: gestione delle prenotazioni.

2.3 Schema finale

Lo schema dopo l'applicazione dei raffinamenti è il seguente.

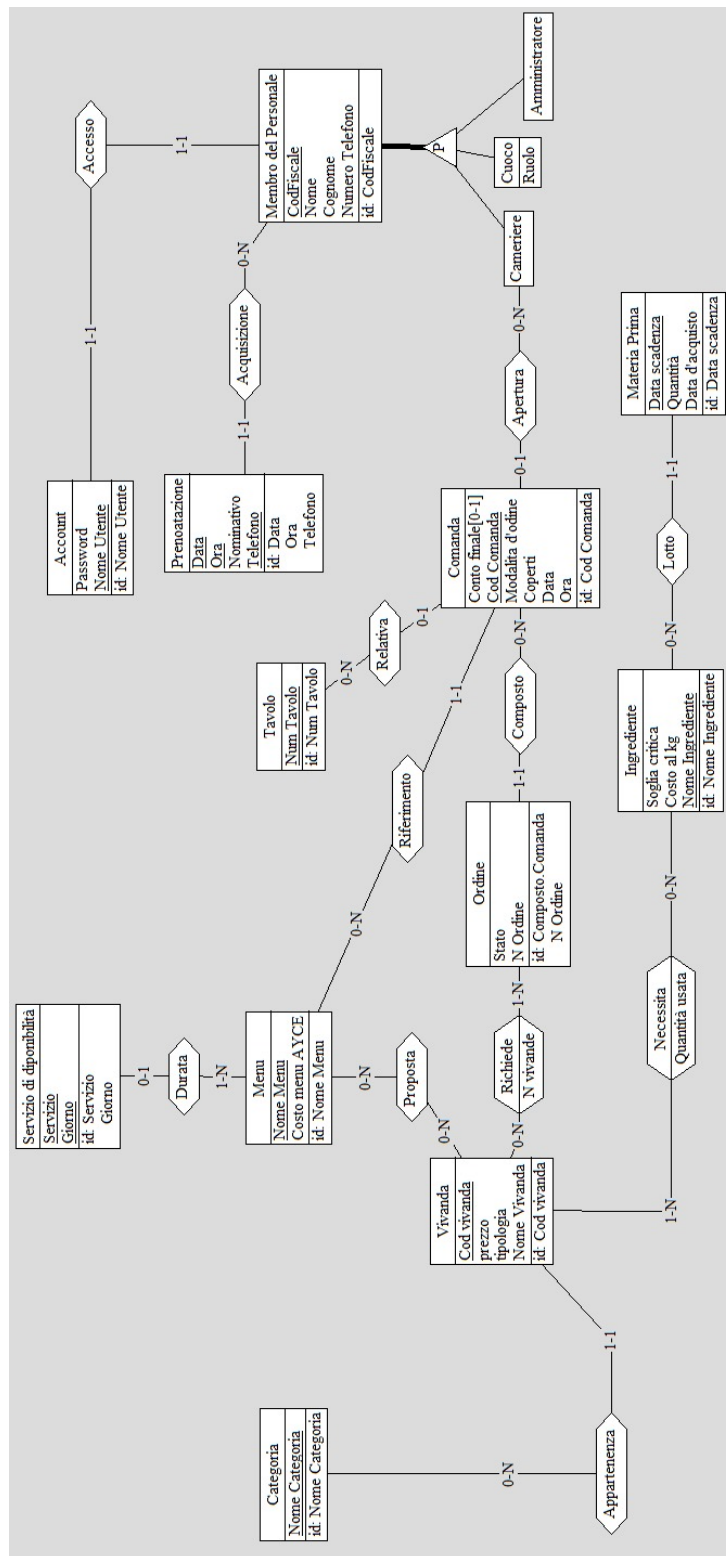


Figura 2.6: Schema relazionale dopo l'applicazione dei raffinamenti.

2.4 Vincoli inespressi

Bisogna osservare che lo schema ha dei vincoli inespressi che devono essere verificati tramite query:

- non possono esistere più menu con lo/gli stesso/i giorni di disponibilità
- la data e l'ora di una comanda devono essere coerenti con il giorni di disponibilità del menu da cui sono stati ordinati i piatti contenuti negli ordini relativi alla suddetta comanda
- data ed ora di una prenotazione devono essere relativi a data e servizio del tavolo con cui quella prenotazione è legata

Capitolo 3

Progettazione logica

3.1 Stima del volume dei dati

In tabella 3.1 viene riportata la stima del volume dei dati per ogni entità e relazione presente nello schema. Si stima l'utilizzo del sistema da parte di una modesta impresa di 8 persone e 50 coperti.

Concetto	Tipologia	Volume
Membro del personale	E	8
Cameriere	E	4
Cuoco	E	3
Amministratore	E	1
Account	E	8
Accesso	R	8
Prenotazioni	E	35
Acquisizioni	R	5
Tavolo	E	12
Agenda	R	35
Comanda	E	2700
Riferimento	E	2700
Apertura	R	2600
Ordine	E	5400
Composto	R	2700
Richiede	E	32400
Vivande	R	100
Bevanda	E	20
Consumo	R	20
Piatto	R	80
Necessita	R	320
Ingrediente	E	60
Lotto	R	150
Materia prima	E	150
Categorie	E	9
Appartenenza	R	100
Menu	E	3
Proposta	R	240
Divisione	R	27
Periodo di disponibilità	E	3

Tabella 3.1: stime del volume dei dati

3.2 Descrizione delle operazioni principali e stima della loro frequenza

In tabella 3.2 sono riportate le frequenze delle operazioni previste dal sistema. Fare riferimento alla sezione 1.2 per una descrizione più accurata delle

operazioni.

Operazione	Descrizione	Tipologia	Frequenza
A1	Aggiunge un nuovo membro del personale	I	2 all'anno
A2	Rimuove un membro del personale	I	2 all'anno
A3	Modifica i dati di un utente	I	3 all'anno
A4	Visualizza ricavi della giornata	I	1 al giorno
A5	Aggiunge un nuovo menù	I	4 all'anno
A6	Rimuove un menù	I	4 all'anno
A7	Aggiunge una vivanda ad un menù	I	10 all'anno
A8	Rimuove una vivanda dal menù	I	10 all'anno
A9	Aggiunge una nuova vivanda al sistema	I	10 all'anno
A10	Rimuove una vivanda dal sistema	I	10 all'anno
S1	Crea una nuova comanda	I	40 al giorno
S2	Modifica una comanda	I	20 al giorno
S3	Elimina una comanda dal sistema	I	5 al giorno
S4	Aggiunge un ordine ad una comanda	I	100 al giorno
S5	Rimuovi una vivanda da un ordine	I	10 al giorno
S6	Rimuove un ordine da una comanda	I	10 al giorno
S7	Chiudi una comanda	I	90 al giorno
S8	Aggiungi una vivanda ad un ordine	I	10 al giorno
S9	Aggiungi prenotazione	I	10 al giorno
K1	Rimuove una certa quantità di una materia prima	I	200 al giorno
K2	Visualizza le materie prime non più utilizzabili	I	1 al giorno
K3	Aggiunge una materia prima all'inventario	I	30 volte al giorno
K4	Visualizza materie prima critica	I	1 al giorno

Tabella 3.2: stima della frequenza delle operazioni

3.3 Schemi di navigazione e tabelle degli accessi

Certe operazioni non vengono prese in esame in quanto operano su poche entità/relazioni ed hanno costo minimo. Altre operazioni vengono direttamente calcolate e valutate nella sezione 3.5 sulle ridondanze.

A1: aggiunge un membro del personale

Concetto	Costrutto	Accesso	Tipo
Membro del personale	E	1	S
Accesso	R	1	S
Account	E	1	S

Per un costo di:

$$3 S = 9 \cdot 2 = 18/\text{anno} \quad (3.1)$$

E' contestualmente simile l'operazione A2.

S3: elimina una comanda

Concetto	Costrutto	Accesso	Tipo
Comanda	E	1	S
Comanda	E	1	L
Relativa	R	1	L
Relativa	R	1	S
Tavolo	R	1	L
Tavolo	R	1	S
Composto	E	2	L
Composto	E	2	S
Ordine	E	2	L
Ordine	E	2	S
Richiede	E	$2 \cdot 12$	L
Richiede	E	$2 \cdot 12$	S
Riferimento	E	1	S
Riferimento	E	1	L

Per un costo di:

$$(8 + 2 \cdot 12)S + (8 + 2 \cdot 12)L = 96 \cdot 5 = 480/\text{giorno} \quad (3.2)$$

S4: aggiungi un ordine ad una comanda

Concetto	Costrutto	Accesso	Tipo
Comanda	E	1	L
Composto	A	1	S
Ordine	E	1	S
Vivanda	E	12	L
Richiede	A	12	S

Per un costo di:

$$((1 + 12)L + (1 + 1 + 12)S) = 41 \cdot 100 = 4100/\text{giorno} \quad (3.3)$$

S5: rimuovi vivanda dall'ordine

Concetto	Costrutto	Accesso	Tipo
Ordine	E	1	L
Richiede	R	1	L
Richiede	R	1	S
Vivanda	E	1	L

$$((1 + 1)L + (1)S) = 4 \cdot 10 = 40/\text{giorno} \quad (3.4)$$

S6: rimuove un ordine da una comanda

Concetto	Costrutto	Accesso	Tipo
Comanda	E	1	L
Composto	R	1	L
Composto	R	1	S
Ordine	E	1	L
Ordine	E	1	S
Richiede	R	12	L
Richiede	R	12	S

$$((1 + 1 + 1 + 12)L + (1 + 1 + 12)S) = 43 \cdot 10 = 430/\text{giorno} \quad (3.5)$$

S7: visualizza i piatti che non è possibile cucinare

Concetto	Costrutto	Accesso	Tipo
Piatto	E	80	L
Necessita	R	320	L
Ingrediente	E	60	L

Per un costo di:

$$(100 + 320 + 60)L = 480 \cdot 10 = 4800/\text{giorno} \quad (3.6)$$

S7-bis: visualizza le bevande non disponibili

Concetto	Costrutto	Accesso	Tipo
Bevande	E	20	L
Consumo	R	20	L
Mageria prima	R	20	L

Per un costo di:

$$60L = 60 \cdot 10 = 600/\text{giorno} \quad (3.7)$$

S9: aggiungi una vivanda ad un ordine

Concetto	Costrutto	Accesso	Tipo
Ordine	E	1	L
Richiede	R	1	S
Vivanda	R	1	L

Per un costo di:

$$2L + 1S = 4 \cdot 10 = 40/\text{giorno} \quad (3.8)$$

S10: aggiungi una prenotazione

Concetto	Costrutto	Accesso	Tipo
Prenotazione	E	1	S
Prenotazione	R	1	S
Membro del personale	E	1	L
Acquisizione	R	1	S

Per un costo di:

$$1L + 3S = 7 \cdot 10 = 7/\text{giorno} \quad (3.9)$$

3.4 Raffinamento dello schema

Si propongono i seguenti raffinamenti:

- Collasso verso l'alto della gerarchia sul personale con la creazione del vincolo inespresso per cui solo un cameriere può aprire una comanda
- Collasso verso l'alto della gerarchia riguardo alle vivande con l'introduzione dell'attributo **tipo**. D'ora in poi una bevanda può fare utilizzo di più ingredienti.

3.5 Analisi delle ridondanze

Le ridondanze presenti in questo schema sono: l'attributo **quantità** nella entità **Ingrediente** il quale potrebbe essere calcolato contando **Materia prima** e l'attributo **conto finale** di **Comanda** che potrebbe essere ricavato dagli ordini eseguiti per una certa comanda quando si va a cercare il ricavo a fine serata.

3.5.1 Analisi della ridondanza di conto finale

Le operazioni interessate sono **S8** e **A4**.

Caso con ridondanza : per quanto riguarda il calcolo dei ricavi sarà sufficiente leggere l'attributo **conto finale** da tutte le comande mentre nell'operazione del calcolo del totale di una comanda si deve

A4

Questa operazione viene svolta a fine servizio per determinare i ricavi della giornata.

Concetto	Costrutto	Accesso	Tipo
Comanda	E	90	L

Per un costo di:

$$90/\text{giorno} \quad (3.10)$$

S8

In caso di un menu all you can eat:

Concetto	Costrutto	Accesso	Tipo
Comanda	E	1	L
Comanda	E	1	S
Riferimento	R	1	L
Menù	E	1	L
Composto	R	2	L
Ordine	E	2	L
Richiede	R	4	L
Vivanda	E	4	L

$$(1 + 1 + 1 + 2 + 2 + 4 + 4)L + 1S = 17 \cdot 80 = 1360/\text{giorno} \quad (3.11)$$

In caso di un menu alla carta:

Concetto	Costrutto	Accesso	Tipo
Comanda	E	1	L
Comanda	E	1	S
Composto	R	2	L
Ordine	E	2	L
Richiede	R	24	L
Vivanda	E	24	L

$$(1 + 2 + 2 + 24 + 24)L + 1S = 55 \cdot 10 = 550/\text{giorno} \quad (3.12)$$

Caso senza ridondanza : calcolando i ricavi ora bisogna calcolare gli importi delle comande.

A4

Ora è necessario calcolare tutte le comande, similmente al punto successivo, infatti se non si mantiene l'attributo allora si dovrà calcolare nuovamente tutte le comande:

$$1200 + 530 = 1730/\text{giorno} \quad (3.13)$$

S8

Per un menu all you can eat:

Concetto	Costrutto	Accesso	Tipo
Comanda	E	1	L
Riferimento	R	1	L
Menù	E	1	L
Composto	R	2	L
Ordine	E	2	L
Richiede	R	4	L
Vivanda	E	4	L

Per un costo di:

$$(1 + 1 + 1 + 2 + 2 + 4 + 4)L = 15 \cdot 80 = 1200/\text{giorno} \quad (3.14)$$

Per un menu alla carta:

Concetto	Costrutto	Accesso	Tipo
Comanda	E	1	L
Composto	R	2	L
Ordine	E	2	L
Richiede	R	24	L
Vivanda	E	24	L

Per un costo di:

$$(1 + 2 + 2 + 24 + 24)L = 53 \cdot 10 = 530/\text{giorno} \quad (3.15)$$

Quindi questa operazione costa:

$$1200 + 530 = 1730/\text{giorno} \quad (3.16)$$

Conclusione

Il caso senza ridondanza costa:

$$1730 + 1730 = 3460/\text{giorno} \quad (3.17)$$

Mentre nel caso con ridondanza queste due operazioni costano:

$$90 + 1360 + 550 = 2020/\text{giorno} \quad (3.18)$$

Dati i numeri delle equazioni 3.17 e 3.18 si conclude che il caso con ridondanza è quello più conveniente.

3.5.2 Analisi della ridondanza di quantità totale

Le operazioni interessate sono K1, K3, K4.

Caso con ridondanza

K1: rimuove una certa quantità di una materia prima

Concetto	Costrutto	Accesso	Tipo
Materia prima	E	1	L
Materia prima	E	1	S
Ingrdiente	E	1	L
Ingrediente	E	1	S
Lotto	R	1	L

$$(1 + 1)S + (1 + 1 + 1)L = 7 \cdot 200 = 1400/\text{giorno} \quad (3.19)$$

K3: aggiunge una certa quantità di una materia prima

Concetto	Costrutto	Accesso	Tipo
Materia prima	E	1	S
Ingrdiente	E	1	L
Ingrediente	E	1	S
Lotto	R	1	S

$$1L + (1 + 1 + 1)S = 7 \cdot 30 = 210/\text{giorno} \quad (3.20)$$

K4: visualizza materia prima critica

Concetto	Costrutto	Accesso	Tipo
Ingrediente	E	60	L

Per un costo di:

$$60 L = 60 \cdot 1 = 60/\text{giorno} \quad (3.21)$$

Caso senza ridondanza

K1: rimuove una certa quantità di materia prima

Concetto	Costrutto	Accesso	Tipo
Materia prima	E	1	L
Materia prima	E	1	S

Per un costo di:

$$1S + 1L = 3 \cdot 200 = 600/\text{giorno} \quad (3.22)$$

K3: aggiungi una materia prima all'inventario

Concetto	Costrutto	Accesso	Tipo
Materia prima	E	1	S
Lotto	E	1	S
Ingrediente	E	1	L

Per un costo di:

$$1 L + 2 S = 5 \cdot 30 = 150/\text{giorno} \quad (3.23)$$

K4: visualizza materia prima critica

Concetto	Costrutto	Accesso	Tipo
Materia prima	E	150	L
Lotto	E	150	L
Ingrediente	E	60	L

Per un costo di:

$$150 + 150 + 60 L = 360 \cdot 1 = 360/\text{giorno} \quad (3.24)$$

Conclusione

Il costo delle operazioni mantenendo l'attributo è:

$$210 + 1400 + 60 = 1670 / \text{giorno} \quad (3.25)$$

mentre il costo delle operazioni eliminando l'attributo è:

$$600 + 360 + 150 = 1110 / \text{giorno} \quad (3.26)$$

Quindi visti i risultati delle operazioni 3.26 e 3.25 si conclude che il caso senza ridondanza è più conveniente.

3.6 Traduzione di entità e associazioni in relazioni

Account(Nome utente, Password, Codice fiscale: Membro del personale)
Membro del personale(Codice fiscale, Nome, Cognome, Professione, Ruolo cuoco*, Numero telefono)
Cameriere(Codice fiscale: Membro del personale)
Prenotazione(Data, Ora, Telefono, Nominativo, Codice fiscale: Membro del personale)
Tavolo(Numero tavolo)
Comanda(Codice comanda, Conto finale*, Modalità d'ordine, Data, Ora, Coperti, Nome menù: Menù, Numero tavolo*: Tavolo, Codice fiscale: Cameriere)
Ordine(Codice comanda: Comanda, Numero ordine, Stato)
Richiede(Codice vivanda: Vivanda, Codice comanda: Comanda, Numero ordine: Ordine, Numero vivande)
Vivanda(Codice Vivanda, Nome, Prezzo, Nome categoria: Categoria)
Categoria(Nome categoria)
Necessita(Codice vivanda: Vivanda, Nome: Ingrediente, Quantità usata)
Ingrediente(Nome, Codice vivanda: Vivanda, Costo al kg, Soglia critica)
Materia prima(Data scadenza, Nome: Ingrediente, Quantità, Data acquisto)
Proposta(Nome: Menù, Codice vivanda: Vivanda)
Menù(Nome, Costo AYCE)
Servizio di disponibilità(Servizio, Giorno, Nome*: Menù)

3.7 Schema relazionale finale

Segue lo schema relazionale ristrutturato come dai punti precedenti.

3.8 Traduzione delle operazioni in query SQL

A1: Inserimento di un membro del personale

```
1 INSERT INTO `Membro_del_Personale` (`CodFiscale`, `Ruolo_cuoco`,  
  ↪ `Professione`, `Nome`, `Cognome`, `Numero_Telefono`)  
2 VALUES ('BGTFR03M18D704C', NULL, 'Amministratore', 'Federico',  
  ↪ 'Bagattoni', '3803797762');
```

A2: Rimozione di un membro del personale

```
1 DELETE FROM membro_del_personale  
2 WHERE CodFiscale = ?
```

A3: modifica i dati di un utente

```
1 UPDATE `account` SET `Nome_Utente` = ?, `Password` = ?  
2 WHERE account.CodFiscale = ?
```

A4: visualizza i ricavi della giornata

```
1 SELECT SUM(comanda.Conto_finale)  
2 FROM comanda  
3 WHERE comanda.Data = CURRENT_DATE
```

A5: aggiungi nuovo menù

```
1 INSERT INTO `Menu` (`Nome_Menu`, `Costo_menu_AYCE`)  
2 VALUES ('?', '?');
```

A6: rimuovi un menù

```
1 DELETE
2 FROM menù
3 WHERE Nome_Menù = "Menù elegante";
```

A7: aggiungi un piatto ad un menù

Si assume che la vivanda sia stata aggiunta al sistema tramite l'operazione A10 ed il menù già aggiunto tramite operazione A6.

```
1 INSERT INTO `Proposta` (`Nome_Menu`, `Cod_vivanda`)
2     VALUES ('Menù elegante', '1');
```

A8: rimuovi una vivanda dal menù

```
1 DELETE FROM proposta
2 WHERE proposta.Nome_Menu = ?
3     AND proposta.Cod_vivanda = ?
```

A9: aggiungi una vivanda al sistema

```
1 INSERT INTO `vivanda` (`Cod_vivanda`, `prezzo`, `tipologia`,
2     ↪ `Nome_Vivanda`, `Nome_Categoria`)
3     VALUES (NULL, ?, ?, ?, ?)
```

A10: rimuovi una vivanda al sistema

```
1 DELETE FROM Vivanda WHERE `Cod_vivanda` = ?;
```

S1: crea una nuova comanda

```
1 INSERT INTO `Comanda` (`Conto_finale`, `Cod_Comanda`, `Modalita_d_ordine`,
   ↪ `Coperti`, `Data`, `Ora`, `Nome_Menu`, `Num_Tavolo`, `CodFiscale`)
2 VALUES (NULL, NULL, 'AYCE', '3', current_timestamp(), curtime(), 'Menù
   ↪ elegante', '14', 'BGTFR03M18D704C')
```

S2: modifica una comanda

```
1 UPDATE `Comanda` SET `Coperti` = '4' WHERE `Comanda`.`Cod_Comanda` = 1;
```

S3: elimina una comanda dal sistema

```
1 DELETE FROM Comanda WHERE `Comanda`.`Cod_Comanda` = 1
```

S4: aggiungi un ordine ad una comanda

Nell'applicativo le query sono separate per separare le query di set da quelle di get, ma qui sono nella stessa operazione.

```
1 SELECT IF(MAX(N_Ordine) IS NULL, 1, MAX(N_Ordine) + 1) INTO @NORDINE
2 FROM Ordine
3 WHERE Cod_Comanda = "1";
4
5 INSERT INTO Ordine(`Cod_Comanda`, `Stato`, `N_Ordine`)
6     VALUES("1", "Inserito", @NORDINE);
```

S5: rimuovi una vivanda da un ordine

```
1 DELETE FROM Richiede
2     WHERE `Richiede`.`Cod_Comanda` = 1
3         AND `Richiede`.`N_Ordine` = 1
4         AND `Richiede`.`Cod_vivanda` = 12
```

S6: rimuovi un ordine da una comanda

```
1 DELETE FROM Ordine WHERE `Ordine`.`Cod_Comanda` = 1 AND
   ↳ `Ordine`.`N_Ordine` = 1
```

S7: chiudi una comanda

```
1 UPDATE Comanda AS C
2 SET C.Conto_finale = IF(Modalita_d_odine = 'CARTA',
3     (
4         SELECT SUM(V.Prezzo * R.N_vivande)
5     FROM Ordine AS O
6     JOIN Richiede AS R ON O.Cod_Comanda = R.Cod_Comanda AND O.N_Ordine =
7         ↳ R.N_Ordine
8     JOIN Vivanda AS V ON R.Cod_vivanda = V.Cod_vivanda
9     WHERE O.Cod_Comanda = C.Cod_Comanda),
10    (
11        SELECT SUM(V.Prezzo * R.N_vivande) + (SELECT Costo_menu_AYCE from menu
12        ↳ WHERE Nome_Menu = C.Nome_Menu)
13    FROM Ordine AS O
14    JOIN Richiede AS R ON O.Cod_Comanda = R.Cod_Comanda AND O.N_Ordine =
15        ↳ R.N_Ordine
16    JOIN Vivanda AS V ON R.Cod_vivanda = V.Cod_vivanda
17    WHERE O.Cod_Comanda = C.Cod_Comanda AND V.tipologia = 'Bevanda'
18    )
19 )
20 WHERE C.Cod_Comanda = '2';
```

S8: aggiungi una vivanda ad un ordine

```
1 INSERT INTO `Richiede` (`Cod_Comanda`, `N_Ordine`, `Cod_vivanda`,
   ↳ `N_vivande`)
2 VALUES ('1', '1', '12', '2');
```

S9: aggiungi una prenotazione

```
1 INSERT INTO `Prenotazione` (`Data`, `Ora`, `Nominativo`, `Coperti`,  
  ↳ `Telefono`, `CodFiscale`)  
2 VALUES ('2024-07-23', '13:00:00', 'luca', '2', '1234567899',  
  ↳ 'BGTFR03M18D704C');
```

K1: rimuovi una certa quantità di materia prima

```
1 UPDATE Materia_Prima AS M1  
2 SET M1.Quantita = M1.Quantita - 1  
3 WHERE M1.Nome_Ingrediente = 'Costata Manzo'  
4 AND M1.Data_scadenza = (SELECT MIN(Materia_Prima.Data_scadenza)  
5                           FROM Materia_Prima  
6                           WHERE M1.Nome_Ingrediente =  
9                               ↳ Materia_Prima.Nome_Ingrediente  
7                               AND  
                               ↳ Materia_Prima.Data_scadenza  
                               ↳ > NOW())
```

K2: visualizza le materie prime non utilizzabili

```
1 SELECT *  
2 FROM Materia_Prima AS M  
3 WHERE M.Data_scadenza < NOW();
```

K3: aggiungi materia prima all'inventario

```
1 INSERT INTO `Materia_Prima`(`Data_scadenza`, `Nome_Ingrediente`,  
  ↳ `Quantita`, `Data_d_acquisto`)  
2 VALUES ('2024-07-23', 'Costata Manzo', 13, '2024-07-23')
```

K4: visualizza ingrediente in quantità critica

```
1 SELECT I.Nome_Ingrediente
2 FROM Ingrediente AS I JOIN (SELECT M.Nome_Ingrediente, SUM(M.Quantita) as
   ↪  Quantita_tot
3                               FROM Materia_Prima AS M
4                               GROUP BY M.Nome_Ingrediente) AS M
5 ON I.Nome_Ingrediente = M.Nome_Ingrediente
6 WHERE I.Nome_Ingrediente = M.Nome_Ingrediente
7 AND Quantita_tot < I.Soglia_critica;
```

Appendice A

Guida Utente

L'applicativo è diviso in pagine, come un browser di internet. Ogni pagina concerne un aspetto specifico del sistema. Le pagine possono essere chiuse ed aggiornate (in caso l'aggiornamento automatico non abbia effetto oppure ci siano state modifiche da parte di altri utenti).

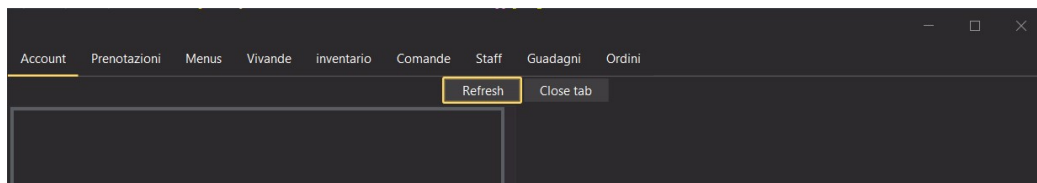


Figura A.1: Le pagine dell'interfaccia grafica.

La maggior parte delle pagine sono divise a metà, a sinistra si trova la lista degli elementi che concerne la pagina mentre a destra un form che si può usare per modificare o aggiungere elementi. Inoltre ogni casella di elementi ha dei tasti per operare sul singolo elemento.

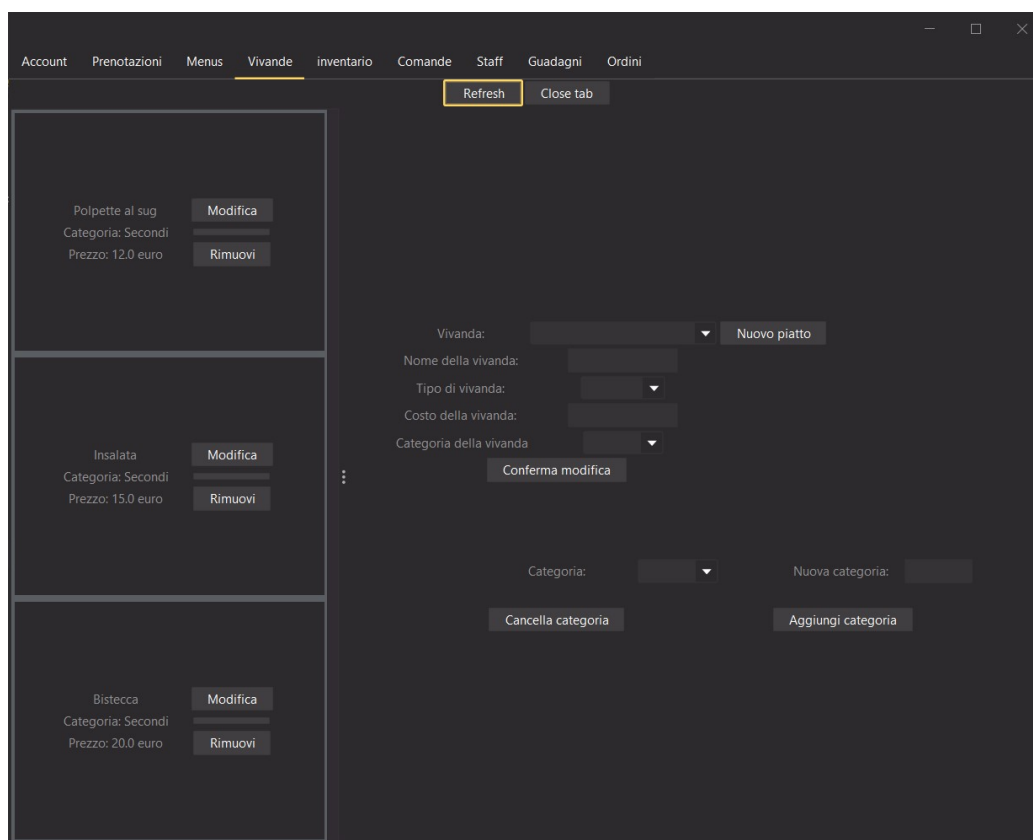


Figura A.2: La pagina riguardante le vivande.

Vediamo in figura A.2 che sulla sinistra i tasti "Modifica" e "Rimuovi" possono essere usati per rimuovere il singolo piatto o modificarlo. Alla pressione del tasto "Modifica" il form a fianco verrà riempito con le informazioni della vivanda e l'utente potrà modificarle. Inoltre è possibile anche aggiungere una nuova categoria di piatti o rimuovere quelle presenti.

A.1 Log in e utenti

Ogni utente ha un ruolo specifico deciso al momento della creazione dell'account. Il ruolo permette di fare uso di certi componenti del sistema come i menù, l'inventario... etc. Al momento del login l'utente vedrà aprire le schermate di suo interesse.

Gli utenti che sono amministratori possono operare su tutte le finestre previste.

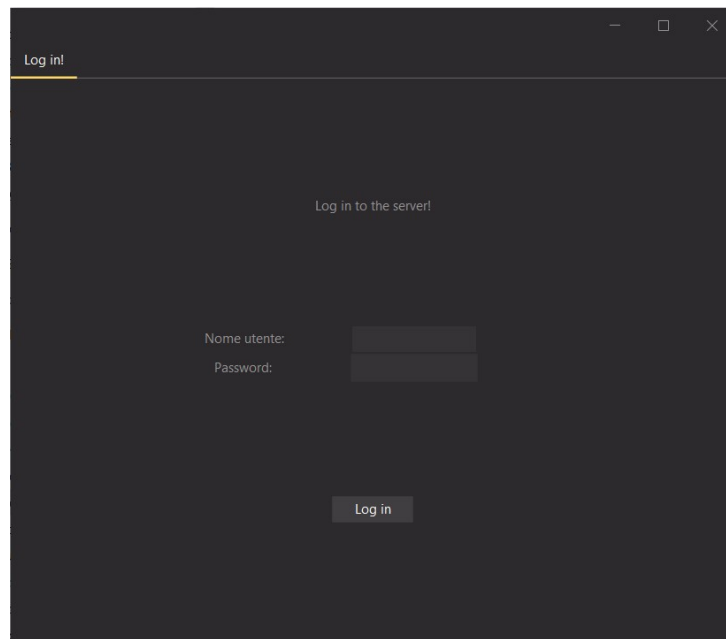


Figura A.3: La schermata di login.