

Elaborato per il corso di Basi di dati.
A.A. 2022-2023
Progetto di una base di dati per la gestione di
ordini in un ristorante.

Tomas Ventrucchi
tomas.ventrucchi@studio.unibo.it
0001043183

17 agosto 2023

Indice

1	Analisi dei requisiti	3
1.1	Intervista	3
1.2	Estrazione dei concetti principali	4
2	Progettazione concettuale	6
2.1	Schema scheletro	6
2.2	Schema concettuale finale	8
3	Progettazione logica	9
3.1	Stima volumetrica dei dati	9
3.2	Descrizione delle operazioni principali e stima della loro frequenza	10
3.3	Schemi di navigazione e tabelle degli accessi	10
3.4	Raffinamento dello schema	13
3.4.1	Eliminazione di identificatori esterni	13
3.4.2	Eliminazione delle gerarchie	14
3.5	Analisi delle ridondanze	14
3.6	Traduzione di entità ed associazioni in relazioni	14
3.7	Schema relazionale finale	15
3.8	Traduzione delle operazioni in query	15
3.8.1	Op. 1 Inserimento prodotto	15
3.8.2	Op. 2 Rimozione prodotto	15
3.8.3	Op. 3 Visualizzare dipendenti di turno	15
3.8.4	Op. 4 Associare cameriere a turno	16
3.8.5	Op. 5 Visualizzare spesa media per persona	16
3.8.6	Op. 6 Visualizzare totale coperti	16
3.8.7	Op. 7 Visualizzare media persone per tavolo	16
3.8.8	Op. 8 Visualizzare 5 prodotti più venuti	17
3.8.9	Op. 9 Visualizzare cameriere che ha permesso più incasso	17
3.8.10	Op. 10 Creare un ordine	17

3.8.11 Op. 11 Visualizzare scontrino di un tavolo	17
3.8.12 Op. 12 Visualizzare tavoli aperti	18
3.8.13 Op. 13 Inserire categoria	18
4 Progettazione dell'applicazione	19

Capitolo 1

Analisi dei requisiti

L'obiettivo è quello di realizzare un database atto alla gestione della presa ordini da parte dei camerieri all'interno di un ristorante.

1.1 Intervista

All'interno di un ristorante è utile avere un'applicazione che gestisce gli ordini dei clienti in modo da verificare le statistiche relative ai prodotti e usufruire del calcolo automatico dello scontrino. Quando i clienti entrano nel ristorante, il cameriere li fa sedere e in quel momento crea un nuovo ordine inserendo i vari prodotti che i clienti scelgono. Ogni prodotto è associato ad una categoria in modo da poterne verificare le statistiche di vendita. L'ordine è associato al tavolo in cui sono seduti i clienti. Al termine del pasto i clienti richiederanno lo scontrino che sarà visualizzato al cameriere. Si devono poter gestire gli utenti del database. Ad ogni cameriere sono associate delle credenziali per poter accedere al sistema e per poter tener traccia delle operazioni svolte. Dovrà essere presente una sezione di back office nella quale si potranno gestire i turni dei camerieri, i prodotti, le categorie, e varie statistiche relative ad una giornata di lavoro.

1.2 Estrazione dei concetti principali

Termine	Descrizione	Sinonimi
Prodotto	Pietanza, bevanda o altro venduto dal ristorante	
Categoria	Tipologia a cui appartiene il prodotto	
Ordine	Elenco dei prodotti scelti dai clienti	Comanda
Tavolo	Oggetto che identifica la posizione dei clienti all'interno del ristorante	
Scontrino	Oggetto che elenca tutti i prodotti venduti in un unico ordine mostrandone il totale	
Utente	Identifica il cameriere all'interno del sistema	Cameriere
Turno	Identifica il turno di lavoro del cameriere	

Di seguito viene riassunta la richiesta del cliente in modo da chiarire i concetti e le richieste:

Per tutti i **prodotti** venduti nel ristorante viene memorizzato un codice identificativo, il prezzo e una descrizione. Ad ogni prodotto è associata una **categoria** la quale viene gestita tramite il suo nome. Il **tavolo** sarà identificato dal suo numero e un valore che rappresenta il numero massimo di clienti che riesce ad ospitare. Ogni **ordine** sarà identificato da data e ora che corrispondono al momento in cui il cameriere invia l'ordine. Avrà un campo che indica se è già stato chiuso. Ogni ordine sarà legato ad un **dettaglio ordine** che tiene traccia dei prodotti che un determinato tavolo ha ordinato con le relative quantità. Quando i clienti hanno terminato il pasto, richiedono lo **scontrino** in cui saranno riepilogati tutti i prodotti presenti nell'ordine, il totale da pagare, la data e l'ora.

Il cameriere è colui che fa da intermediario tra il cliente e l'applicazione e verrà gestito tramite username che deve essere univoco all'interno del sistema e password al fine di permettere l'autenticazione e di tenere traccia delle sue azioni. Inoltre saranno presenti i campi per nome e cognome. Il cameriere all'interno del sistema sarà gestito come un **utente**. tra cui la spesa media dei clienti, il totale dei coperti, la media delle persone per tavolo, il cameriere che ha permesso più incasso e la lista dei prodotti più venduti. L'ultima sezione è quella dei **turni** di lavoro che permette di associare un cameriere ad un turno lavorativo composto da data e momento della giornata (Pranzo o Cena).

- Inserire un dipendente
- Inserire una categoria

- Inserire un prodotto
- Rimuovere un prodotto
- Visualizzare dipendenti di turno
- Associare un cameriere ad un turno
- Visualizzare la spesa media per persona
- Visualizzare il totale dei coperti
- Visualizzare la media di persone per tavolo
- Ottenere un elenco dei cinque prodotti più venduti
- Ottenere il nome del cameriere che ha consentito un maggiore incasso
- Creare un nuovo ordine
- Stampare lo scontrino di un tavolo
- Visualizzare i tavoli aperti
- Inserire categoria

Capitolo 2

Progettazione concettuale

2.1 Schema scheletro

Ogni **Categoria** è identificata da un nome che deve essere univoco all'interno del sistema. A ogni categoria sono associati dei **prodotti** che sono identificati da un id progressivo gestito in fase di creazione, un nome ed il relativo prezzo. Quando il cameriere compila l'ordine, vengono associati i prodotti all'ordine tramite il **dettaglio ordine** il quale contiene anche le quantità scelte di ogni prodotto. Ogni dettaglio ordine è associato al proprio **ordine** che è identificato dalla data e l'ora di creazione. L'ordine ha anche l'attributo che identifica l'orario di chiusura dell'ordine che corrisponde al momento in cui viene stampato lo scontrino. Ciò è necessario per poter utilizzare lo stesso tavolo più volte in una giornata. Ogni **tavolo** è identificato dal proprio numero e come attributo ha il numero massimo di persone che possono sedervi. Ad ogni tavolo può essere associato uno **scontrino** che è identificato dagli attributi ID (incrementale) e data è inoltre presente l'ora. Dagli ordini passiamo agli utenti. Gli **utenti** sono stati pensati come una gerarchia totale e disgiunta tra amministratori e dipendenti identificata dall'username univoco all'interno del sistema. Nella parte di gestione dei turni abbiamo che: il **turno** è identificato dalla data e dal momento della giornata (pranzo o cena) e vengono associati agli utenti tramite la tabella **assegnamento turni**.

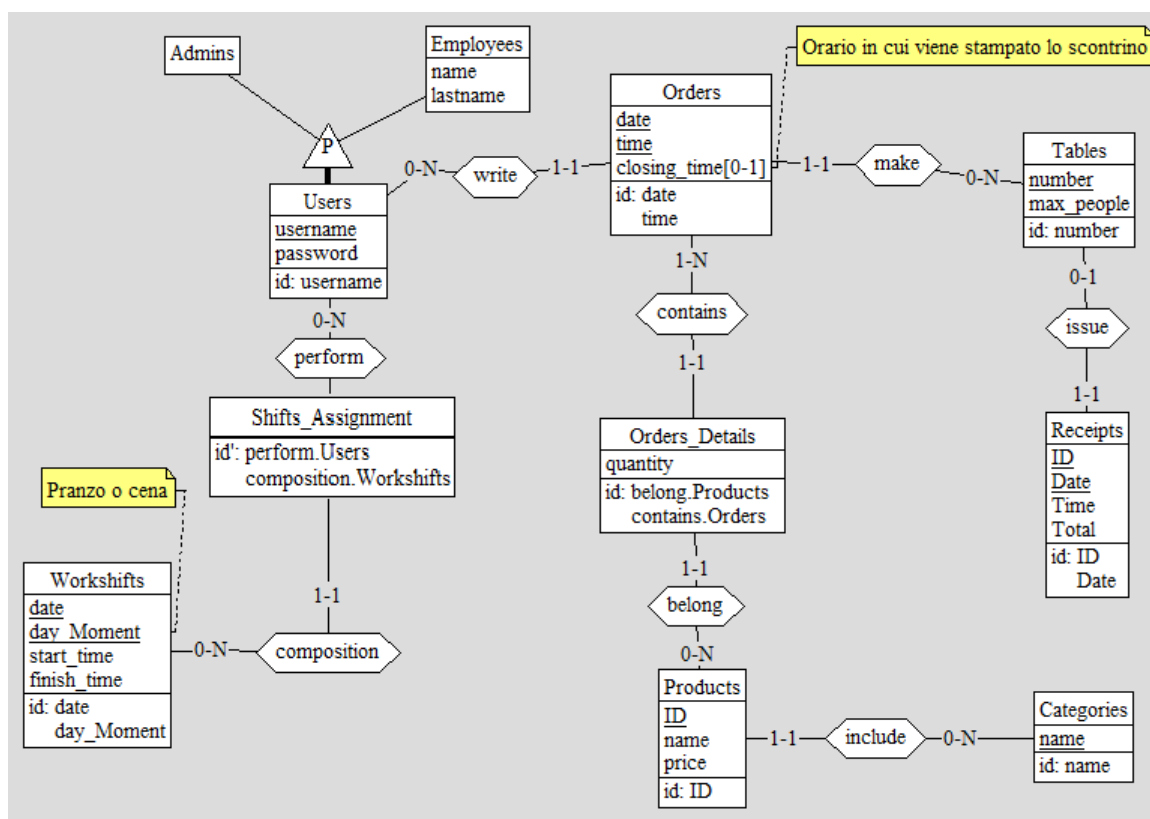


Figura 2.1: Schema scheletro.

2.2 Schema concettuale finale

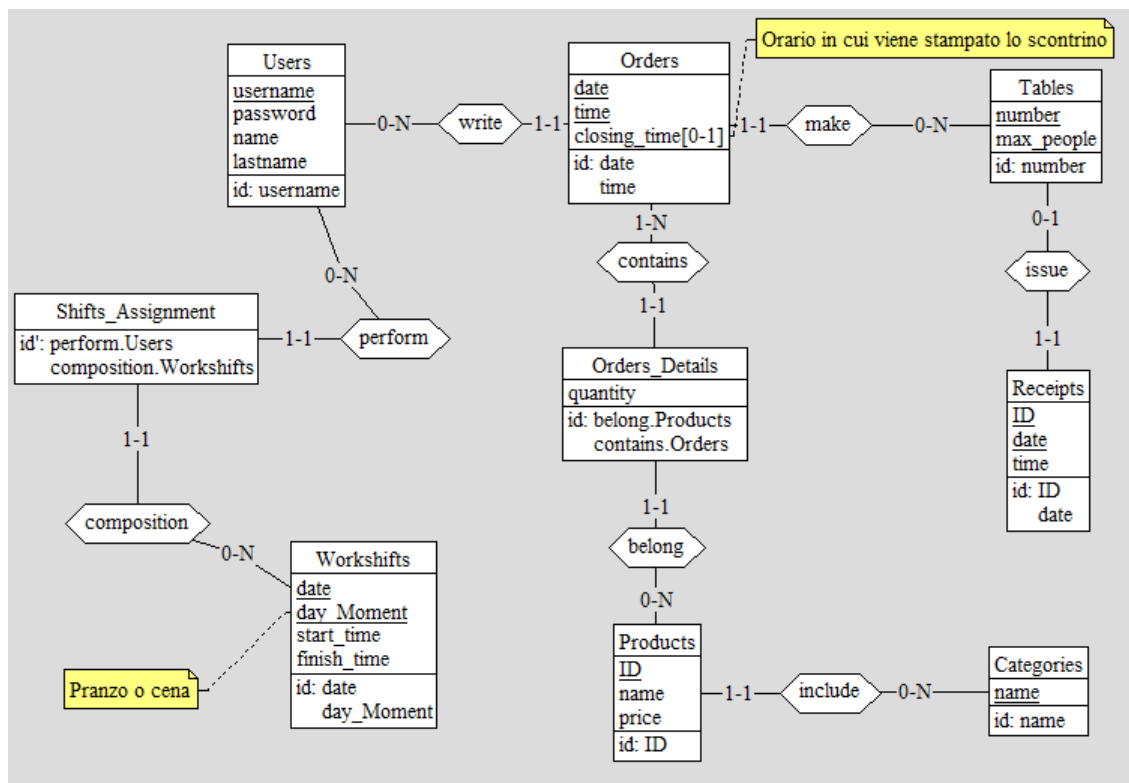


Figura 2.2: Schema concettuale finale.

Capitolo 3

Progettazione logica

3.1 Stima volumetrica dei dati

La stima dei dati è stata effettuata considerando un totale massimo di capienza del ristorante di 400 persone al giorno. Stima volumetrica dei dati all'interno del database dopo un anno.

Concetto	Costrutto	Volume
Prodotto	E	50
Categoria	E	7
Ordine	E	40.000
Dettaglio_Ordine	E	40.000
Utente	E	4
Tavolo	E	20
Scontrino	E	40.000
Turno	E	730
Assegnamento_Turno	E	1400
Inclusione	R	50
Appartenenza	R	40.000
Contiene	R	40.000
Scrivo	R	40.000
Esecuzione	R	360
Composizione	R	360
Fa	R	40.000
Emissione	R	30.000

3.2 Descrizione delle operazioni principali e stima della loro frequenza

Le operazioni da effettuare sono quelle già elencate nella fase di analisi. Segue una tabella riportante la loro descrizione e relativa frequenza:

Codice	Descrizione	Frequenza
1	Inserimento prodotto	2 all'anno
2	Rimozione prodotto	1 all'anno
3	Visualizzare dipendenti di turno	1 al giorno
4	Associare cameriere a turno	14 a settimana
5	Visualizzare spesa media per persona	1 al giorno
6	Visualizzare totale dei coperti	1 al giorno
7	Visualizzare media persone per tavolo	1 al giorno
8	Visualizzare 5 prodotti più venduti	1 al giorno
9	Visualizzare cameriere che ha permesso più incasso	3 a settimana
10	Creare un nuovo ordine	130 al giorno
11	Visualizzare scontrino di un tavolo	130 al giorno
12	Visualizzare tavoli aperti	15 al giorno

3.3 Schemi di navigazione e tabelle degli accessi

Sono riportate di seguito le tabelle degli accessi delle operazioni viste sopra. Al fine del calcolo dei costi, si considerano di peso doppio gli accessi in scrittura rispetto a quelli in lettura.

Op. 1 Inserimento prodotto			
Concetto	Costrutto	Accessi	Tipo
Prodotto	E	1	S
Totale: $1S = 2 \rightarrow 4$ all'anno			

Op. 2 Rimozione prodotto			
Concetto	Costrutto	Accessi	Tipo
Prodotto	E	1	S
Totale: $1S = 2 \rightarrow 2$ all'anno			

Op. 3 Visualizzazione dipendenti di turno			
Concetto	Costrutto	Accessi	Tipo
Utente	E	1	L
Assegnazione turno	E	1	L
Turno	E	1	L
Esecuzione	R	90	L
Composizione	R	3	L
Totale: 96L ->35.040 all'anno			

Op. 4 Associazione cameriere turno			
Concetto	Costrutto	Accessi	Tipo
Assegnamento turni	E	1	S
Totale: 1S = 2 ->756 all'anno			

Op. 5 Visualizzare spesa media per persona			
Concetto	Costrutto	Accessi	Tipo
Dettaglio ordine	E	2	L
Prodotto	E	2	L
Ordine	E	2	L
Contiene	R	2	L
Appartiene	E	1600	L
Totale: 1608L = 1608 ->1608 al giorno			

Op. 6 Visualizzare totale dei coperti			
Concetto	Costrutto	Accessi	Tipo
Dettaglio ordine	E	1	L
Prodotto	E	1	L
Ordine	E	1	L
Contiene	R	1	L
Appartiene	E	800	L
Totale: 804L = 804 ->804 al giorno			

Op. 7 Visualizzare media delle persone per tavolo			
Concetto	Costrutto	Accessi	Tipo
Dettaglio ordine	E	2	L
Prodotto	E	1	L
Ordine	E	2	L
Contiene	R	2	L
Appartiene	E	800	L
Totale: 809L = 809 ->809 al giorno			

Op. 8 Visualizzare 5 prodotti più venduti			
Concetto	Costrutto	Accessi	Tipo
Prodotto	E	1	L
Dettaglio ordine	E	1	L
Appartiene	R	800	L
Totale: 802L = 802 ->802 al giorno			

Op. 9 Visualizzare cameriere che ha permsso più incasso			
Concetto	Costrutto	Accessi	Tipo
Dettaglio ordine	E	1	L
Prodotto	E	1	L
Ordine	E	1	L
Contiene	R	1	L
Appartiene	R	800	L
Totale: 804L = 804 ->2412 a settimana			

Op. 10 Creare nuovo ordine			
Concetto	Costrutto	Accessi	Tipo
Ordine	E	1	S
Dettaglio ordine	E	1	S
Totale: 2S = 4 ->520 al giorno			

Op. 11 Visualizzare scontrino di un tavolo			
Concetto	Costrutto	Accessi	Tipo
Prodotto	E	1	L
Dettaglio ordine	E	1	L
Appartiene	R	1	L
Scontrino	E	1	S
Totale: $1S + 3L = 4 \rightarrow 520$ al giorno			

Op. 12 Visualizzare tavoli aperti			
Concetto	Costrutto	Accessi	Tipo
Tavolo	E	1	L
Ordine	E	1	L
Fa	E	2.000	L
Totale: $2002L = 2002 \rightarrow 30.030$ al giorno			

Op. 13 Inserire categoria			
Concetto	Costrutto	Accessi	Tipo
Cameriere	E	1	L
Cameriere	E	1	S
Totale: $1S + 1L = 3 \rightarrow 6$ all'anno			

3.4 Raffinamento dello schema

3.4.1 Eliminazione di identificatori esterni

Sono state rimosse le seguenti relazioni:

- inclusione, importando il nome della categoria dentro il prodotto
- appartenenza, importando l'ID del prodotto dentro il dettaglio ordine
- contiene, importando data e ora dentro il dettaglio ordine
- fa, importando il numero del tavolo dentro l'ordine
- emissione, importando il numero del tavolo dentro lo scontrino
- scrittura, importando l'username del cameriere dentro l'ordine
- esecuzione, importando l'username del cameriere dentro assegnazione turni

- composizione, importando data e momento della giornata dentro assegnazione turni

3.4.2 Eliminazione delle gerarchie

Le entità amministratori e dipendenti vengono collassate dentro utenti perchè per semplicità di gestione, si amministratori che dipendenti hanno gli stessi privilegi.

3.5 Analisi delle ridondanze

3.6 Traduzione di entità ed associazioni in relazioni

- prodotto(id_prodotto, nome, prezzo, categoria:(nome))
- categoria(nome)
- dettaglio_ordini(ordini:(data, ora), prodotto:(ID), quantità)
- ordine(data, ora, utente:(username), tavolo:(numero), orario_chiusura*)
- tavolo(numero, max_persone)
- scontrino(id, data, ora, tavolo:(numero))
- utente(username, password, nome, cognome)
- associazione_turni(utente:(username), turno:(data,momento))
- turno(data, momento, ora_inizio, ora_fine)

3.7 Schema relazionale finale

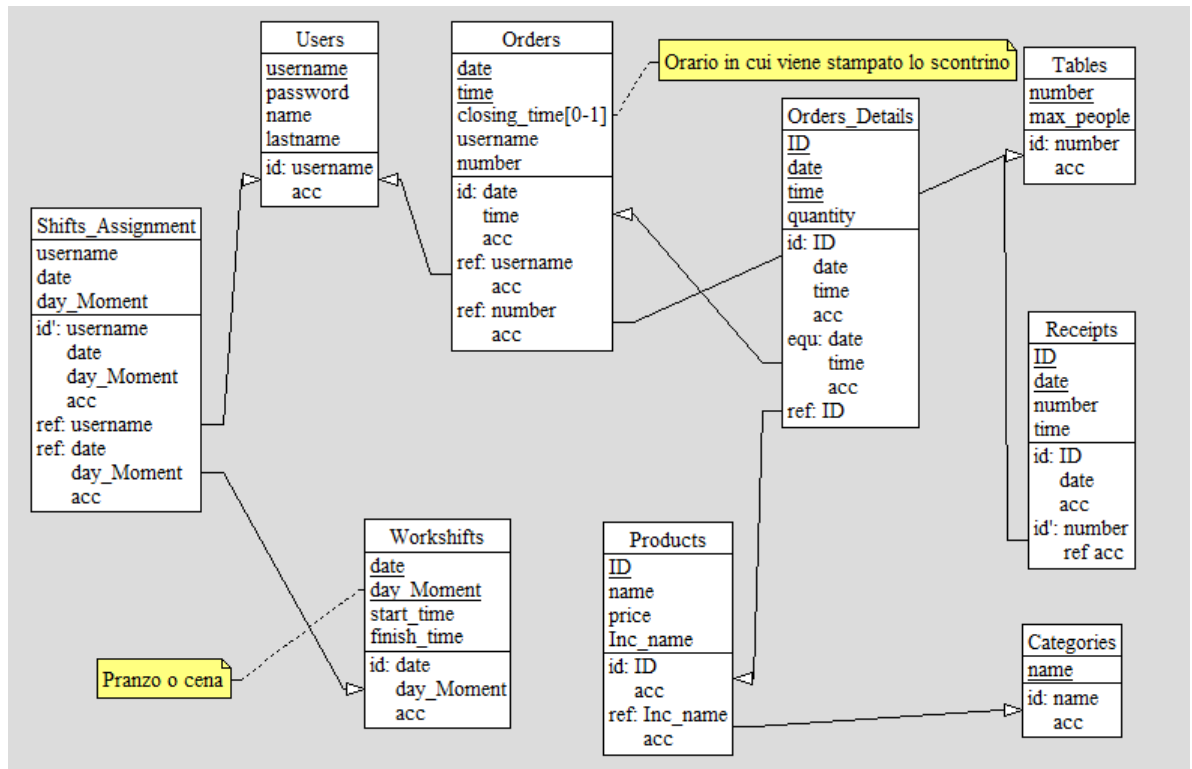


Figura 3.1: Schema relazionale finale.

3.8 Traduzione delle operazioni in query

3.8.1 Op. 1 Inserimento prodotto

```
INSERT INTO Products (name, price, Inc_name)
VALUES (?, ?, ?)
```

3.8.2 Op. 2 Rimozione prodotto

```
DELETE FROM Products WHERE id = ?
```

3.8.3 Op. 3 Visualizzare dipendenti di turno

```
SELECT u.username, u.name, u.lastname, u.password from Users as u
join shifts.assignment s on s.username = u.username
```



```
join workshifts w on w.date = s.date AND w.day_Moment = s.day_Moment
where s.date = ? and s.day_Moment = ?
```

3.8.4 Op. 4 Associare cameriere a turno

```
INSERT INTO Shifts_assignment
VALUES (?, ?, ?)
```

3.8.5 Op. 5 Visualizzare spesa media per persona

```
select (select sum(quantity * price) from orders_details od
join products p on od.product_id = p.id
join orders o on o.date = od.date and o.time = od.time
where o.date = ?)
/
(select sum(quantity) as tot from orders_details od
join products p on od.product_ID = p.id
join orders o on od.date = o.date and o.time = od.time
where od.date = ? and p.name = ? and o.closing_time is not null) as media
```

3.8.6 Op. 6 Visualizzare totale coperti

```
select od.date, sum(quantity) as tot from orders_details od
join products p on od.product_ID = p.id
join orders o on od.date = o.date and o.time = od.time
where od.date = ? and p.name = ? and o.closing_time is not null
group by od.date;
```

3.8.7 Op. 7 Visualizzare media persone per tavolo

```
select (select sum(quantity) as tot from orders_details od
join products p on od.product_ID = p.id
join orders o on od.date = o.date and o.time = od.time
where od.date = ? and p.name = ? and o.closing_time is not null)
/
(select sum(giriPerTavolo) as tot
from( select count(distinct o.closing_time) as giriPerTavolo
from orders_details od
join orders o on o.date = od.date and o.time = od.time
where o.date = '2023-08-12') as subQuery) as media;
```

3.8.8 Op. 8 Visualizzare 5 prodotti più venuti

```
select p.name, sum(od.quantity) as total from orders_details od
join products p on od.product_ID = p.id
where p.name != ?
group by p.name
order by total DESC
limit 5
```

3.8.9 Op. 9 Visualizzare cameriere che ha permesso più incasso

```
select o.username, sum(od.quantity * p.price) as total from orders o
join orders_details od on o.date=od.date and o.time=od.time
join products p on od.product_ID = p.id
where o.date = ? and closing_time is NOT null
group by o.username
order by total DESC
limit 1
```

3.8.10 Op. 10 Creare un ordine

Creo prima un ordine
INSERT INTO Orders (date,time,username,number)
VALUES (?, ?, ?, ?)
Poi aggiungo un dettaglio ordine
INSERT INTO Orders_Details VALUES (?, ?, ?, ?)

3.8.11 Op. 11 Visualizzare scontrino di un tavolo

Visualizzo il totale dello scontrino
select SUM(price*quantity) as total from orders_details as od
join products as p
on p.id = od.product_id
where od.date in (select o.date from Orders as o
where o.number = ?)
and od.time in (select o.time from Orders as o
where o.number = ? and o.closing_time is NULL)
Visualizzo l'elenco dei prodotti ordinati
select p.name, p.price, SUM(od.quantity) as quantity from orders_details as
od

```
join products as p
on p.id = od.product_id
where od.date in (select o.date from Orders as o
where o.number = ?)
and od.time in (select o.time from Orders as o
where o.number = ? and o.closing_time is NULL)
group by p.name, p.price
```

3.8.12 Op. 12 Visualizzare tavoli aperti

```
select t.number, t.max_people from tables t
where t.number in (select t.number from tables t
join orders o on o.number = t.number
where o.closing_time is NULL)
```

3.8.13 Op. 13 Inserire categoria

```
INSERT INTO Categories (name)
VALUES (?)
```

Capitolo 4

Progettazione dell'applicazione

Il linguaggio principale che ho utilizzato per sviluppare l'applicazione è Java combinato a JavaFX per la parte grafica. Per la gestione dei dati ho utilizzato un database memorizzato localmente. Il DBMS è MySQL. L'app è divisa in cinque finestre differenti, in modo da separare le principali azioni da eseguire, e ogni videata utilizza un file FXML e un controller dedicato.

La prima schermata che viene visualizzata è quella dedicata al login che utilizza la classe `DatabaseConnection` per collegarsi alla base di dati; tramite il nome utente e la password inseriti negli appositi campi dà accesso all'applicazione. Se i dati d'accesso inseriti risultano errati viene visualizzato un messaggio d'errore e non si potrà accedere alla pagina successiva. L'accesso può essere effettuato in due modalità: front office e backoffice. Il frontoffice permette la visione dei tavoli aperti e l'emissione degli scontrini. Tramite un bottone dedicato consente anche di accedere alla gestione degli ordini dei clienti. Il backoffice permette di gestire prodotti, categorie e vedere le statistiche. Da qui è possibile anche accedere ad un'altra videata che permette la gestione dei turni dei camerieri.

In tutte le finestre è quindi possibile visualizzare, interrogare o modificare tramite form e bottoni il database in base alle funzionalità descritte nella fase di progettazione.

L'interfaccia è divisa in più schermate in modo logico e intuitivo in modo che sia di facile utilizzo per l'utente finale. Inoltre le varie videate sono organizzate con dei divisori in modo da suddividere ulteriormente le viste per ambiti di utilizzo.

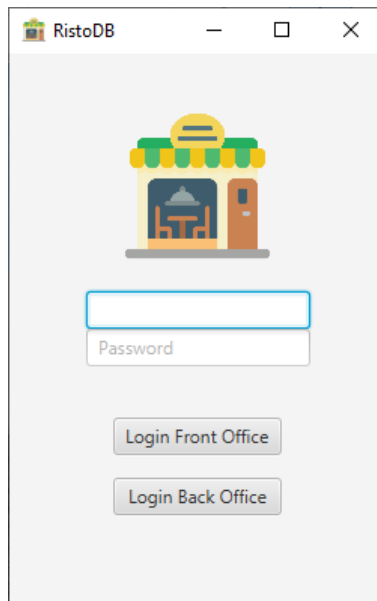


Figura 4.1: Schermata di login.

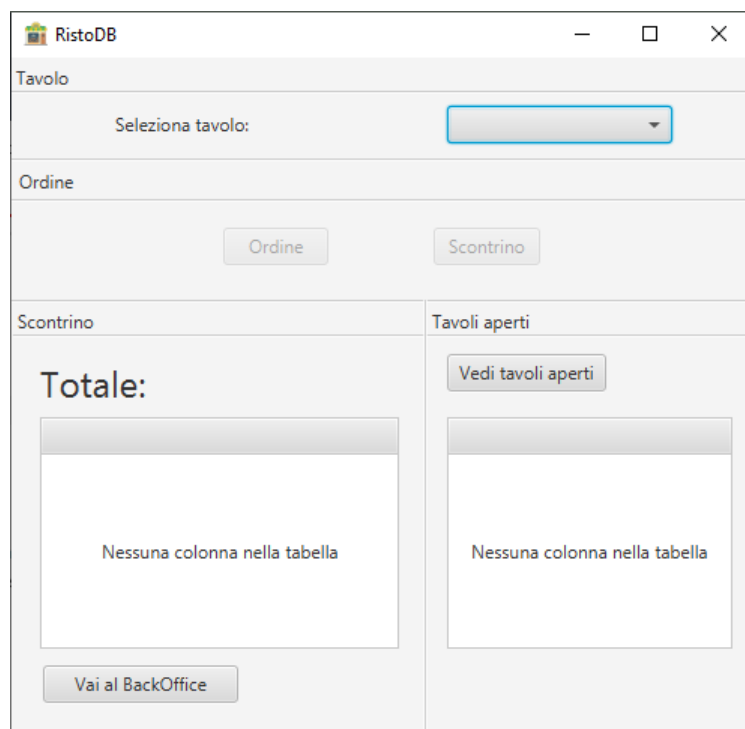
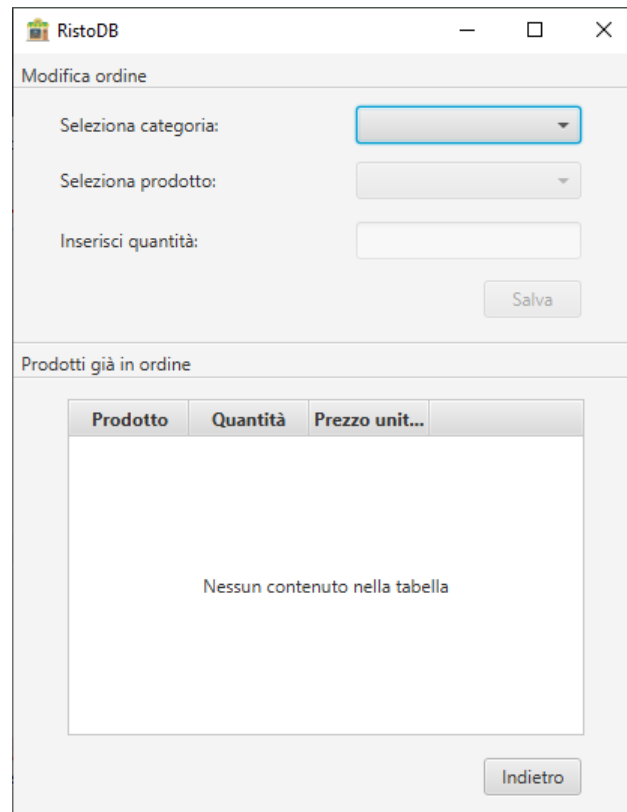


Figura 4.2: Schermata frontOffice.



The screenshot shows a window titled "RistoDB" with standard window controls. The main content is divided into two sections. The top section, "Modifica ordine", contains three input fields: "Seleziona categoria:" with a dropdown menu, "Seleziona prodotto:" with a dropdown menu, and "Inserisci quantità:" with a text input field. A "Salva" button is located to the right of the quantity field. The bottom section, "Prodotti già in ordine", features a table with three columns: "Prodotto", "Quantità", and "Prezzo unit...". The table is currently empty, displaying the message "Nessun contenuto nella tabella". An "Indietro" button is positioned at the bottom right of the window.

Prodotto	Quantità	Prezzo unit...
Nessun contenuto nella tabella		

Figura 4.3: Schermata di gestione ordini.

RistoDB

—

□

×

Turni e camerieri

Gestisci turni e camerieri

Inserisci categoria

Aggiungi

Inserisci prodotto

nome prodotto

prezzo

Categoria

Aggiungi

Rimuovi prodotto

Categoria

Prodotto

Rimuovi

Statistiche

Seleziona data:

Visualizza spesa media

Spesa media:

Visualizza tot coperti

Tot coperti:

Media persone/tavolo

Media pers/tav:

Cameriere che ha permesso più incasso

Nome e Incasso

prodotti + venduti

Nessuna colonna nella tabella

Vai al FrontOffice

Figura 4.4: Schermata di backoffice.

22

The screenshot shows a web application window titled "RistoDB". The main section is titled "Visualizza dipendenti di turno". It contains a dropdown menu, a "Turno" dropdown, and a "Visualizza" button. To the right is a large box with the text "Nessuna colonna nella tabella". Below this, there are two columns: "Inserisci cameriere" with input fields for "nome", "cognome", "utente", and "password", and an "Aggiungi" button; and "Associa cameriere al turno" with three dropdowns (one empty, one "Turno", one "Utente") and an "Associa" button. An "Indietro" button is located at the bottom right.

RistoDB

Visualizza dipendenti di turno

Turno

Visualizza

Nessuna colonna nella tabella

Inserisci cameriere

nome

cognome

utente

password

Aggiungi

Associa cameriere al turno

Turno

Utente

Associa

Indietro

Figura 4.5: Schermata di gestione turni e camerieri.