

Progetto per il Corso di
Basi di dati e sistemi informativi
–A.A. 2020/2021 –

Gestione delle prenotazioni di un ristorante

Autore: Nabiollah Tavakkoli

E-mail: nabiollah.tavakkoli@stud.unifi.it

Numero di matricola: 5953800

Data di consegna: 06/01/2022

Abstract

- Il progetto consiste nella realizzazione di una base di dati per la gestione delle prenotazioni di un ristorante.
- Ogni cliente potrà prenotare uno o più tavoli con le sue informazioni personali
- Ogni cliente può ordinare uno o più piatti dal Menu, e ogni piatto avrà le sue caratteristiche proprie
- Ogni Cameriere deve avere un proprio profilo contenente le sue informazioni personali, il tipo del lavoro e le sue responsabilità
- Sarà necessario tenere traccia della fascia oraria e data delle prenotazioni effettuate e i piatti ordinati dai clienti

Analisi dei requisiti

- Il progetto consiste nella realizzazione di una base di dati per la gestione delle prenotazioni di un ristorante. In particolare, si vogliono memorizzare le informazioni relative ai clienti, agli ordini, ai tavoli, ai camerieri
- Per ogni cliente che effettua una prenotazione è necessario memorizzare
 - **nome**,
 - **cognome**, ed
 - **il codice fiscale** che lo identifica univocamente
- Per ogni tavolo è necessario memorizzare
 - **il codice del tavolo**,
 - **numero dei posti disponibili** ad ogni tavolo.
- Ogni piatto presente nel menu è identificato
 - dal **nome dell'alimento**,
 - dal **prezzo** dei singoli piatti o bevande,
 - dalla **descrizione del piatto** (cioè, cosa contiene oppure come è stato preparato) e
 - dalla **quantità di calorie** presenti in ciascun/a piatto/bevanda.

I piatti vengono suddivisi in cinque gruppi di portate: Antipasto, primo, secondo, bevanda e dessert

- Per ogni cameriere è necessario memorizzare
 - **Id** del cameriere che lo/la identifica univocamente,
 - **Nome** e
 - **cognome** del cameriere,
 - lo **stipendio** attuale del cameriere ed
 - **il tipo del lavoro** (part-time / full-time).

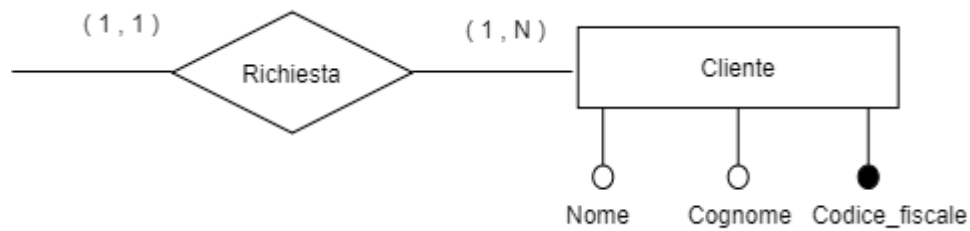
Il cameriere deve essere in possesso di uno o più **numeri di telefono**.

Il cameriere può servire uno o più tavoli non serviti dai altri

- Ad ogni prenotazione vengono associati
 - una serie di piatti che corrispondono alle ordinazioni fatte dai clienti,
 - la **data** e la **fascia oraria** per le quali il cliente riserva il tavolo.
- Ogni ordine conterrà
 - la **quantità** degli alimenti ordinati (per poter ricavare il prezzo coperto unitario)
 - l'aggiuntiva **nota** dei clienti (su ogni alimento)
- Si progetti la base di dati in maniera tale che:
 - Non possono esistere due prenotazioni distinte per lo stesso tavolo nella stessa data ed ora
 - Un cliente possa prenotare più tavoli per la stessa data ed ora

1. Costruzione dello schema concettuale (E-R)

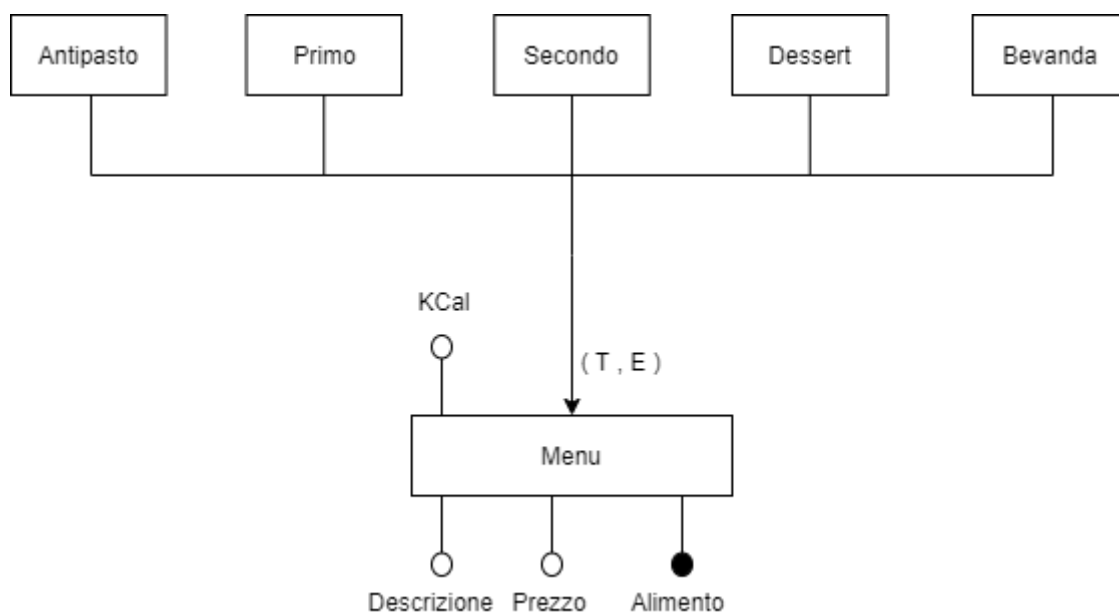
- 1.1. Per ogni cliente che effettua una prenotazione è necessario memorizzare nome, cognome e il codice fiscale che lo identifica univocamente



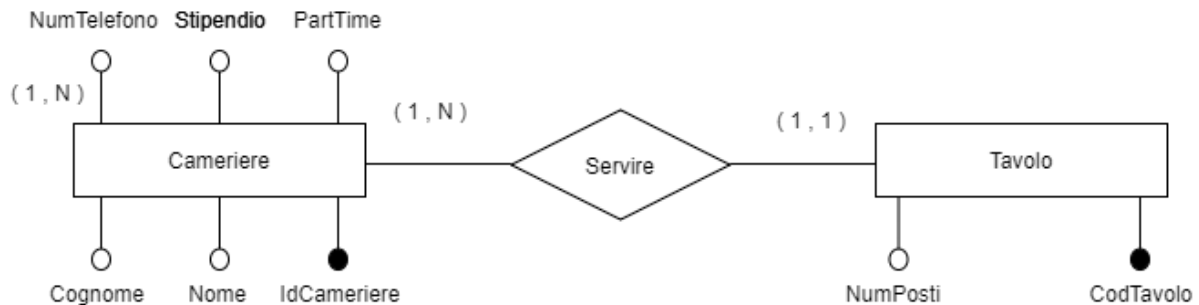
- 1.2. Per ogni tavolo è necessario memorizzare il codice del tavolo, numero dei posti disponibili ad ogni tavolo.



- 1.3. Ogni piatto presente nel menu è identificato dal nome dell'alimento, dal prezzo dei singoli piatti o bevande, dalla descrizione del piatto (cioè, cosa contiene oppure come è stato preparato) e dalla quantità di calorie presenti in ciascun/a piatto/bevanda. I piatti vengono suddivisi in cinque gruppi di portate: Antipasto, primo, secondo, bevanda e dessert



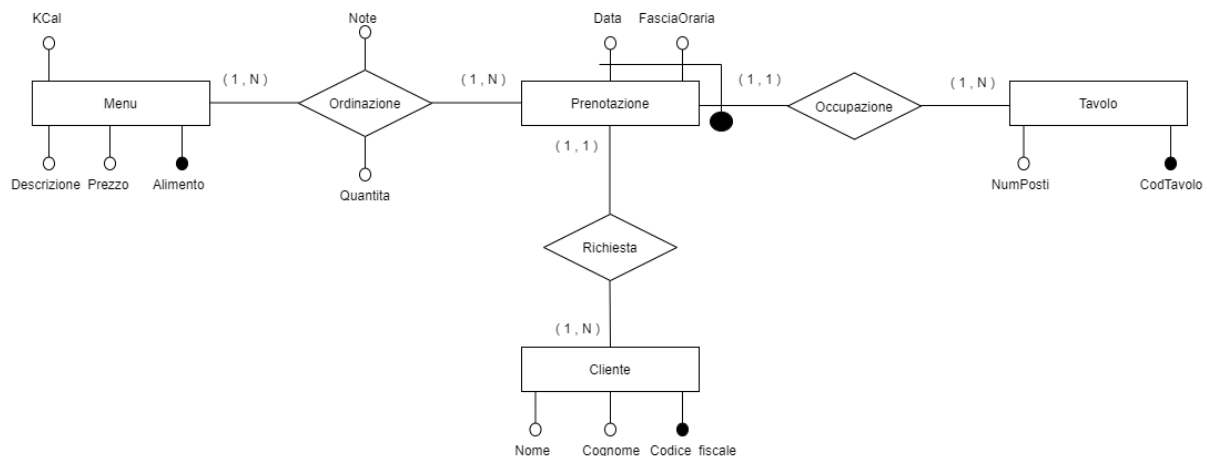
1.4 Per ogni cameriere è necessario memorizzare Id del cameriere che lo/la identifica univocamente, nome e cognome del cameriere, lo stipendio attuale del cameriere ed il tipo del lavoro (part-time / full-time). Il cameriere deve essere in possesso di uno o più numeri di telefono. Il cameriere può servire uno o più tavoli non serviti dai altri



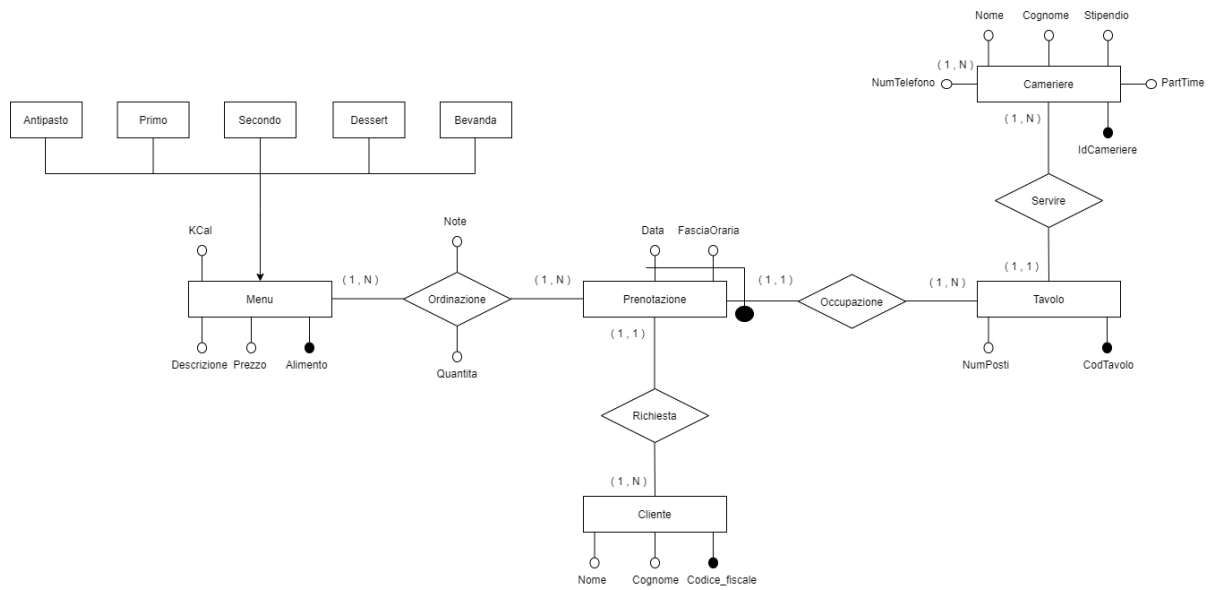
1.5 Ad ogni prenotazione vengono associati una serie di piatti che corrispondono alle ordinazioni fatte dai clienti, la data e la fascia oraria per le quali il cliente riserva il tavolo. Ogni ordine conterrà la quantità degli alimenti ordinati (per poter ricavare il prezzo coperto unitario) e l'aggiuntiva nota dei clienti (su ogni alimento)

1.5.1 Si progetti la base di dati in maniera tale che:

- Non possono esistere due prenotazioni distinte per lo stesso tavolo, nella stessa data ed ora
- Un cliente possa prenotare più tavoli per la stessa data ed ora



1.6 Lo schema E-R finale:



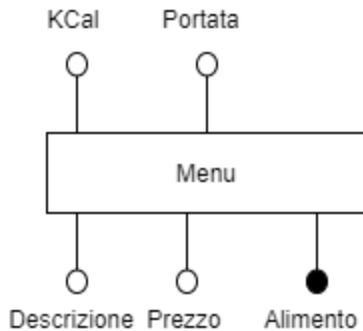
2. Ristrutturazione dello schema E-R

2.1 Analisi delle ridondanze:

- non sono presenti eventuali ridondanze nello schema per poter eliminare o mantenere, e non è necessario introdurne di nuove

2.2 Eliminazione delle generalizzazioni:

- Dobbiamo eliminare la generalizzazione dell'entità Menu ed aggiungere un nuovo attributo, detto: **Portata**, per indicare che il piatto sarà Primo, secondo, Antipasto, Dessert oppure una bevanda.

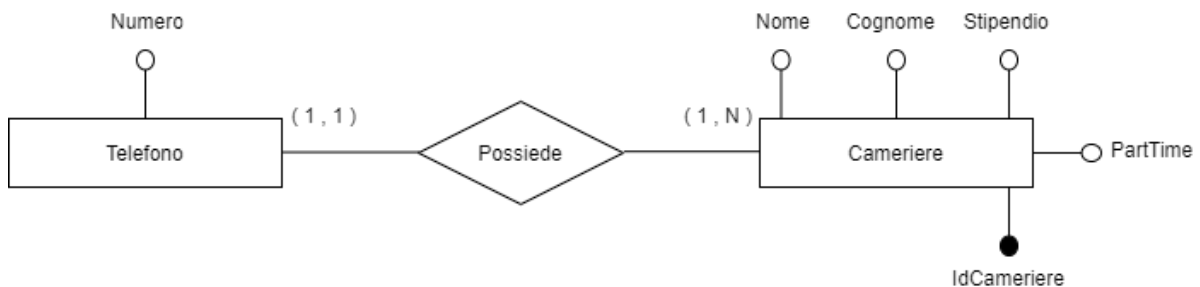


2.3 Partizionamento / accorpamento di entità / associazione:

- Non sono necessari partizionamenti / Accorpamenti di entità / associazione

2.4 Eliminazione degli attributi multi-valore:

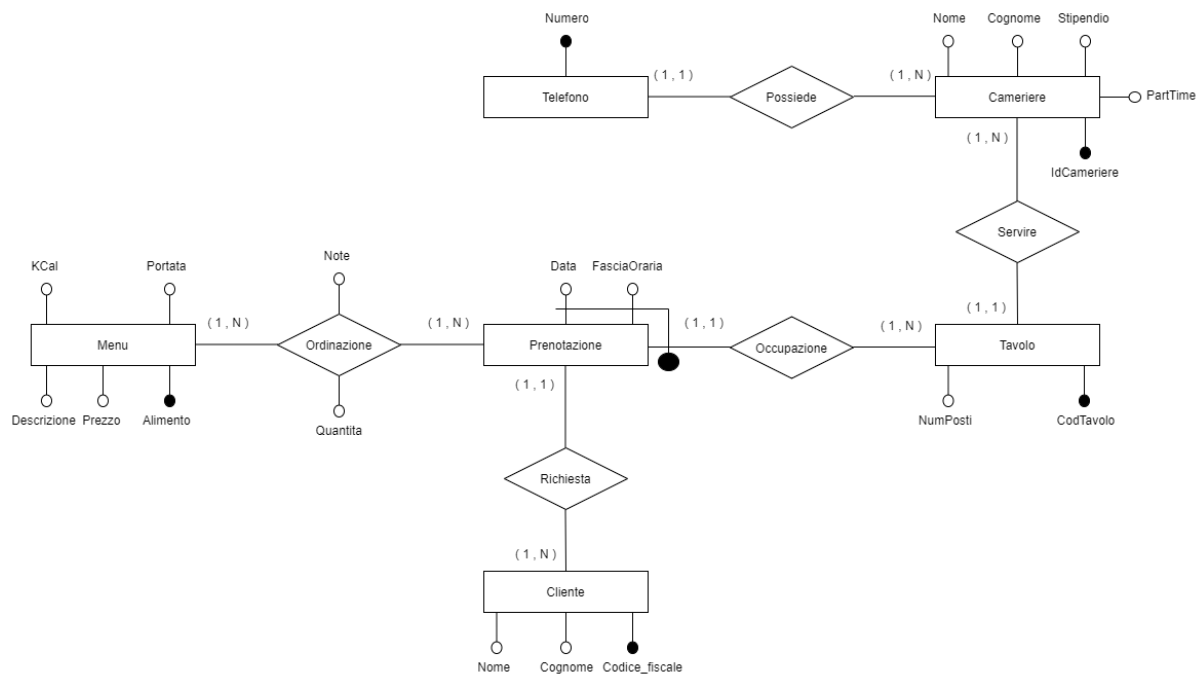
- soltanto dobbiamo eliminare l'attributo multi-valore **Telefono** dell'entità Cameriere e da aggiungere una nuova entità (Telefono) ed una nuova associazione (Possiede) che correla l'entità Cameriere all'entità Telefono



2.5 scelta dell'identificatore principale:

- Sono già scelte

2.6 Lo schema E-R ristrutturato



3. Traduzione verso il modello relazionale:

3.1 entità Cliente

Cliente (Codice_fiscale, nome, cognome)

3.2 entità Menu

Menu (Alimento, Prezzo, Kcal, Descrizione, Portata)

3.3 associazione Ordinazione

Ordinazione (Alimento, Data, FasciaOraria, CodTavolo, Quantita, Note)

Ordinazione.Alimento -> Menu.Alimento

(Ordinazione.Data, Ordinazione.FasciaOraria) -> (Prenotazione.Data, Prenotazione.FasciaOraria)

Ordinazione.CodTavolo -> Tavolo.CodTavolo

3.4 entità Telefono - associazione Possiede - entità Cameriere

Telefono (Numero, IdCameriere)

Cameriere (IdCameriere, Nome, Cognome, Stipendio, PartTime)

Telefono.IdCameriere -> Cameriere.IdCameriere

3.5 entità Tavolo

Tavolo (CodTavolo, NumPosti, IdCameriere)

Tavolo.IdCameriere -> Cameriere.IdCameriere

3.6 entità Prenotazione

Prenotazione (Data, FasciaOraria, CodTavolo, Codice_fiscale)

Prenotazione.CodTavolo -> Tavolo.CodTavolo

Prenotazione.Codice_fiscale -> Cliente.Codice_fiscale

4. Le tabelle della base di dati

Tabella Cliente			
Attributi	Dominio	Lunghezza	Vincoli
<u>Codice_fiscale</u>	VARCHAR	20	
Nome	VARCHAR	20	NOT NULL
Cognome	VARCHAR	20	NOT NULL

Tabella Menu			
Attributi	Dominio	Lunghezza	Vincoli
<u>Alimento</u>	VARCHAR	30	
Prezzo	DOUBLE		
Kcal	INT		
Descrizione	VARCHAR	60	
Portata	ENUM	Antipasto, Primo, Secondo, Contorno, Dessert, Bevanda	

Tabella Cameriere			
Attributi	Dominio	Lunghezza	Vincoli
<u>IdCameriere</u>	INT	2	AUTO_INCREMENT
Nome	VARCHAR	20	
Cognome	VARCHAR	20	
Stipendio	DOUBLE		
PartTime	BOOLEAN		

Tabella Telefono			
Attributi	Dominio	Lunghezza	Vincoli
<u>Numero</u>	INT	15	
IdCameriere	INT	2	

Tabella Tavolo			
Attributi	Dominio	Lunghezza	Vincoli
<u>CodTavolo</u>	CHAR	2	
NumPosti	INT		
IdCameriere	INT	2	

Tabella Prenotazione			
Attributi	Dominio	Lunghezza	Vincoli
<u>CodTavolo</u>	CHAR	2	
<u>Data</u>	DATE		
<u>FasciaOraria</u>	ENUM	12:00-13:00, 13:30-14:30, 20:00-21:00, 21:30-22:30, 23:00-24:00	
Codice_fiscale	VARCHAR	20	NOT NULL

Tabella Ordinazione			
Attributi	Dominio	Lunghezza	Vincoli
<u>CodTavolo</u>	CHAR	2	
<u>Alimento</u>	VARCHAR	30	
<u>Data</u>	DATE		
<u>FasciaOraria</u>	ENUM	12:00-13:00, 13:30-14:30, 20:00-21:00, 21:30-22:30, 23:00-24:00	
Quantita	INT		
Note	VARCHAR	40	

4.1 Fra le tabelle del database esistono i seguenti vincoli di integrità referenziale:

Ordinazione.Alimento -> Menu.Alimento

(Ordinazione.Data, Ordinazione.FasciaOraria) -> (Prenotazione.Data, Prenotazione.FasciaOraria)

Ordinazione.CodTavolo -> Tavolo.CodTavolo

Telefono.IdCameriere -> Cameriere.IdCameriere

Tavolo.IdCameriere -> Cameriere.IdCameriere

Prenotazione.CodTavolo -> Tavolo.CodTavolo

Prenotazione.Codice_fiscale -> Cliente.Codice_fiscale

Interrogazioni

1. Contare il numero dei 'Primi' Ordinati dai clienti
2. Trovare Codice del tavolo, il nome dell'alimento, Quantità di un ordine con il Prezzo coperto unitario più alto
3. Tramite l'uso di una Procedura -- Dato il '**codice Fiscale**' del Cliente stampare lo SCONTRINO contenente il codice del tavolo, gli Alimenti ordinati con quantità e prezzo (per ciascuno), e il Prezzo Totale da pagare
4. Tramite l'uso di una Procedura -- Dato il '**codice Fiscale**' del Cliente stampare lo SCONTRINO contenente il codice del tavolo, gli Alimenti ordinati (separati dalla virgola) con quantità e il Prezzo Totale da pagare
5. Tramite l'uso di una Procedura -- Stampare tutti gli Alimenti con **Kcal < 'n'** e **Prezzo < 'pre'**
6. Tramite l'uso di una Funzione -- Dato l'Id del cameriere, lo stipendio del cameriere corrispondente deve essere aumentato di 10% se è part-time, altrimenti di 20%
7. Tramite l'uso di una Procedura -- Trovare la lista degli Alimenti non ordinati e ordinati in un dato giorno
8. Tramite l'uso di una vista -- trovare i piatti che sono stati ordinati in maggiore quantità (con e senza usare le funzioni aggregate)
9. Tramite l'uso di TRIGGER -- Se viene aggiunto un nuovo cameriere ed il suo stipendio è NULL (zero) -- allora, calcolare lo stipendio medio ed assegnarlo a tale cameriere
10. Tramite l'uso di TRIGGER -- impedire l'inserimento degli alimenti che possono scatenare alcune reazioni allergiche, cioè gli alimenti contenenti 'glutine', 'arachidi', 'soia'