

# RELAZIONE PROGETTO

## DI BASI DI DATI

*Piattaforma di condivisione di ricette e  
gestione delle recensioni*

” ” ” ” ” ” ” ” ”



**Massimo Belfiore – X81001087**

## INDICE

RELAZIONE PROGETTO.....	1
DI BASI DI DATI.....	1
1 DESCRIZIONE DELLA BASE DI DATI E SPECIFICHE.....	3
2 SPECIFICHE SUI DATI.....	3
2.1 COSTRUZIONE DEL GLOSSARIO DEI TERMINI.....	5
3 PROGETTAZIONE CONCETTUALE.....	6
3.2 SCHEMA SCHELETRO.....	7
3.3 SCHEMA INTERMEDIO.....	8
3.4 SCHEMA E-R FINALE.....	9
3.5 DOCUMENTAZIONE DELLO SCHEMA E-R.....	10
3.5.1 DIZIONARIO DEI DATI.....	11
3.5.2 REGOLE DI VINCOLO E DI DERIVAZIONE.....	13
3.5.3 SPECIFICHE SULLE OPERAZIONI.....	14
4 PROGETTAZIONE LOGICA.....	15
4.1 Tavola dei volumi.....	16
4.2 Tavola delle operazioni.....	18
4.3 RISTRUTTURAZIONE DELLO SCHEMA E-R.....	19
4.3.1 Analisi delle ridondanze.....	20
4.3.2 Eliminazione delle generalizzazioni.....	26
4.3.3 Partizionamento di concetti: eliminazione di attributi multivalore.....	27
4.3.4 Scelta degli identificatori principali.....	28
4.4 SCHEMA E-R RISTRUTTURATO.....	29
4.5 TRADUZIONE VERSO IL MODELLO RELAZIONALE.....	32
5 PROGETTAZIONE FISICA.....	33
5.1 MODELLO FISICO.....	34
5.2 DATABASE.....	34
Riferimenti Bibliografici:.....	41
Sitografia:.....	41

# 1 DESCRIZIONE DELLA BASE DI DATI E SPECIFICHE

Si vuole realizzare il progetto di un database per la gestione di una piattaforma di condivisione di ricette online partendo da un insieme di requisiti.

Gli utenti che navigano sulla piattaforma possono consultare le varie ricette disponibili nell'elenco. Ogni utente può decidere di registrarsi per ottenere delle funzionalità aggiuntive, e; in fase di registrazione, può anche iscriversi alla newsletter per ottenere aggiornamenti periodici mirati al suo benessere riguardo delle ricette selezionate da provare. Per ogni utente registrato si tiene traccia delle sue ricette pubblicate e delle sue recensioni. Questi devono poter scrivere recensioni per le ricette pubblicate da altri autori. Ogni utente registrato ha anche la possibilità di specificare se è intollerante o allergico a dei prodotti contenuti nelle ricette. Le ricette vengono pubblicate dagli utenti registrati, contengono ingredienti e possono avere delle recensioni che rispecchiano l'apprezzamento complessivo da parte dei recensori. Ogni ricetta appartiene ad una categoria e può contenere o meno glutine e lattosio. Gli ingredienti possono essere contenuti nelle ricette in una certa quantità.

## 2 SPECIFICHE SUI DATI

- La piattaforma online è visitata dagli **utenti**:

«» Ogni utente ha un id univoco che lo identifica;

«» Ogni utente ha dei dati personali: nome, cognome.

- Ogni utente consulta le **ricette**:

«» Ogni ricetta è caratterizzata da un titolo;

«» Ogni ricetta ha un solo pubblicatore;

«» Ogni ricetta ha una categoria di appartenenza.

«» Ogni ricetta ha una preparazione;

«» Ogni ricetta ha un numero di recensioni;

«» Per ogni ricetta è indicata una media complessiva di apprezzamento;

«» Ogni ricetta ha delle dosi per persona;

«» Una ricetta può essere considerata o meno gourmet;

«» Per ogni ricetta è mantenuta o meno l'indicazione senza\_glutine;

«» Per ogni ricetta è mantenuta o meno l'indicazione senza\_lattosio;

- Ogni ricetta viene pubblicata da un **utente registrato**:

«» Ogni utente registrato ha un nickname che lo identifica;

«» Ogni utente registrato può indicare la sua data di nascita;

«» Ogni utente registrato può avere o meno uno o più contatto\_instagram;

«» Ogni utente registrato ha un numero di recensioni fatte.

- Ogni utente registrato può **recensire** le ricette:

«» Ogni recensione ha una data;

«» Ogni recensione ha una descrizione;

«» Ogni recensione può esprimere o meno un voto di apprezzamento.

- Ogni ricetta contiene degli **ingredienti**:

«» Ogni ingrediente ha un id che lo identifica;

«» Ogni ingrediente ha un nome.

«» Ogni ingrediente può contenere o meno glutine;

«» Ogni ingrediente può contenere o meno lattosio.

# 2.1 COSTRUZIONE DEL GLOSSARIO DEI TERMINI

Termine	Descrizione	Sininimi	Termini Collegati
Utente	Persona che visita la piattaforma di condivisione per consultare le ricette.		Ricetta Utente registrato
Utente registrato	Utente iscritto alla piattaforma di condivisione di ricette che può effettuare recensioni e pubblicare ricette.	Recensore	Ricetta Ingrediente
Ricetta	Cibo o bevanda composto/a da uno o più ingredienti.		Ingrediente Utente registrato
Ingrediente	Ingrediente che può far parte delle ricette.	Prodotto	Ricetta Utente registrato
Recensione	Descrizione qualitativa di una ricetta.		Utente registrato Ricetta

### 3 PROGETTAZIONE CONCETTUALE

Per andare a rappresentare la realtà d'interesse verrà utilizzato il modello E-R, il quale è un modello concettuale di dati che permette attraverso l'utilizzo di vari costrutti di costruire lo schema concettuale (schema E-R).

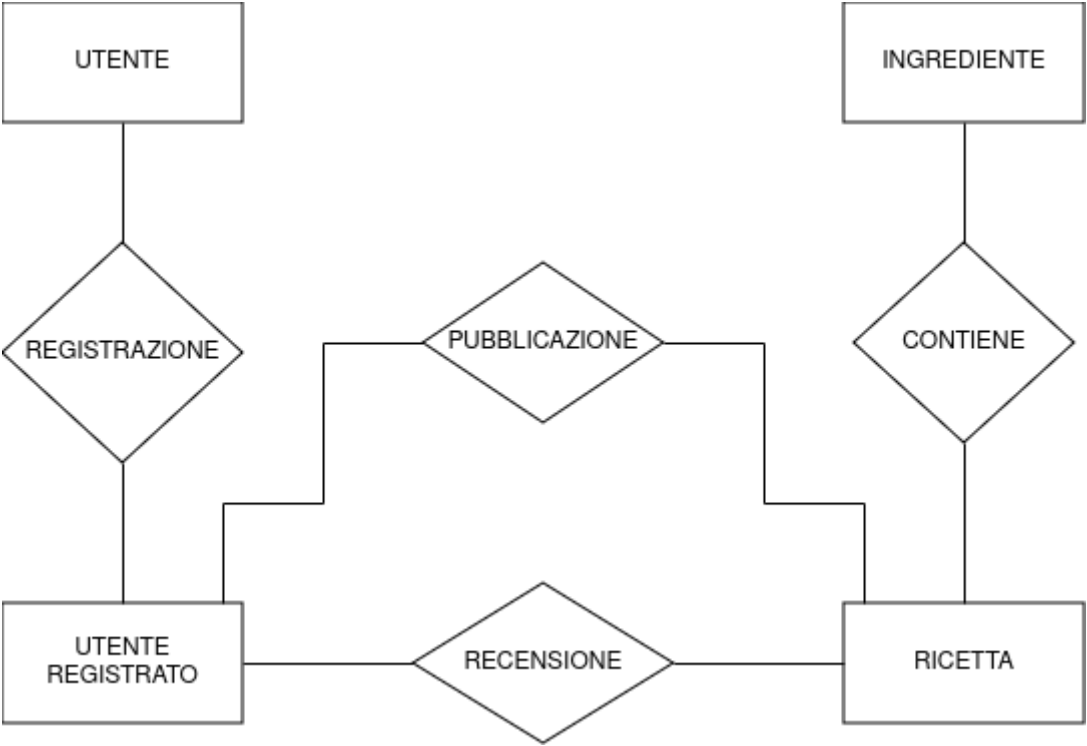
In questa fase verrà utilizzata una strategia di progetto Top-Down che permetterà di produrre uno schema concettuale passando da vari livelli di raffinamento tramite l'utilizzo delle primitive di trasformazione top-down.

Lo schema di partenza è lo schema scheletro, il quale attraverso una serie di raffinamenti verrà trasformato prima in uno schema intermedio e infine nello schema E-R finale. Successivamente lo schema E-R finale ottenuto verrà corredato da una documentazione.

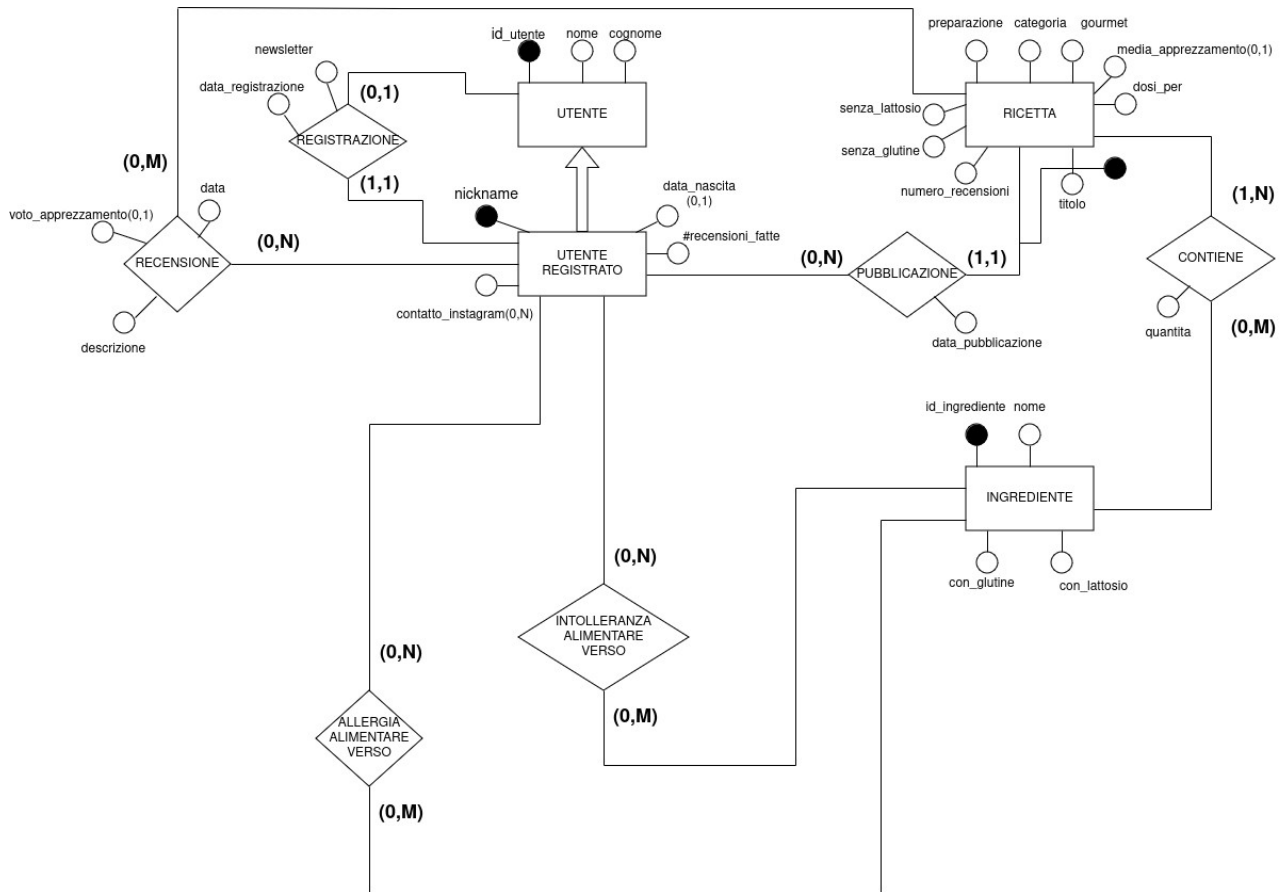
Primitive di trasformazione top-down:

- R1.** Si applica quando un'entità descrive due concetti diversi legati fra loro.
- R2.** Si applica quando un'entità è composta da sotto-entità distinte.
- R3.** Si applica quando un'associazione descrive due diverse associazioni tra le stesse entità.
- R4.** Si applica quando un'associazione esprime un concetto con esistenza autonoma.
- R5.** Si applica per aggiungere attributi ad entità.
- R6.** Si applica per aggiungere attributi ad associazioni.

3.2 SCHEMA SCHELETRO



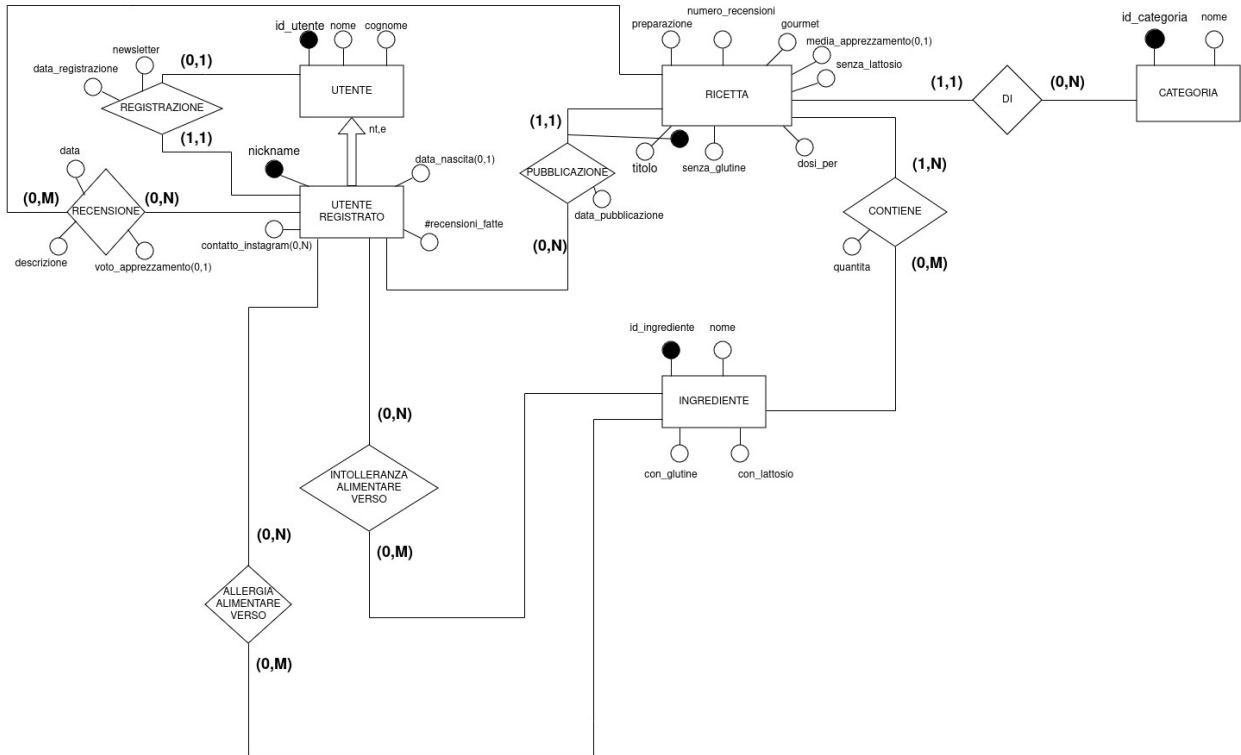
### 3.3 SCHEMA INTERMEDIO



«Ho applicato la regola R2 sull'entità Utente. Inoltre ho applicato anche la R5 e la R6.»



### 3.4 SCHEMA E-R FINALE



«Ho applicato la primitiva di trasformazione R1 sull'entità "ricetta". Ho applicato anche la R5.»

### 3.5 DOCUMENTAZIONE DELLO SCHEMA E-R

Al fine di produrre una documentazione precisa ed efficace dello **schema concettuale** ottenuto verranno adottate delle **business rules**.

- L'attributo voto\_apprezzamento dell'entità recensione rappresenta il giudizio dell'utente riguardo ad una determinata ricetta.
- L'attributo dosi\_per dell'entità ricetta rappresenta il numero di dosi per persona.

### 3.5.1 DIZIONARIO DEI DATI

«Gli identificatori composti da un insieme di attributi verranno rappresentati tra le parentesi graffe.»

Entità	Descrizione	Attributi	Identificatore
Utente	Persona che vuole consultare delle ricette sulla piattaforma di condivisione.	id_utente nome cognome	id_utente
Utente registrato	Utente che ha effettuato la registrazione alla piattaforma di condivisione di ricette online.	nickname data_nascita contatto_instagram #recensioni_fatte vedi Utente	nickname
Ricetta	Ricetta composta da un uno o più ingredienti.	titolo preparazione media_apprezzamento numero_recensioni dosi_per senza_lattosio senza_glutine gourmet	{titolo, nickname}
Ingrediente	Ingrediente di una o più ricette.	id_ingrediente nome con_glutine con_lattosio	id_ingrediente
Categoria	Categoria di appartenenza di una ricetta: antipasti, primi, ecc...	id_categoria nome	id_categoria

**N.B.** Nel dizionario appena sopra riportato l' identificatore esterno dell'entità Ricetta l' ho espresso come l' insieme degli attributi: titolo dell'entità Ricetta e nickname dell' entità Utente registrato). Questo perchè l'entità ricetta è identificata esternamente dall'entità Utente registrato.

<b>Relazione</b>	<b>Descrizione</b>	<b>Entità Coinvolte</b>	<b>Attributi</b>
Contiene	Associa una ricetta agli ingredienti che sono contenuti in essa.	Ricetta (1,N) Ingrediente (0,M)	quantita
Registrazione	Associa un utente ad un utente registrato sulla piattaforma di condivisione di ricette.	Utente (0,1) Utente registrato (1,1)	data_registrazione newsletter
Allergia_alimentare_verso	Associa un utente registrato agli ingredienti a cui è allergico.	Utente registrato (0,N) Ingrediente (0,M)	
Intolleranza_alimentare_verso	Associa un utente registrato agli ingredienti verso cui è intollerante.	Utente registrato (0,N) Ingrediente (0,M)	
Pubblicazione	Associa una ricetta all'utente che l'ha pubblicata	Ricetta (1,1) Utente registrato (0,N)	data_pubblicazione
Recensione	Associa un utente registrato alla ricetta da lui recensita.	Ricetta (0,M) Utente registrato (0,N)	data voto_apprezzamento(0,1) descrizione
Di	Associa una ricetta ad una categoria: primi piatti, secondi ecc...	Ricetta (1,1) Categoria (0,N)	

### 3.5.2 REGOLE DI VINCOLO E DI DERIVAZIONE

REGOLE DI VINCOLO
<p>(RV1) L'attributo voto_apprezzamento dell'entità "recensione" deve assumere un valore intero compreso tra 1 e 5.</p> <p>(RV2) L'attributo con_glutine dell'entità "ingrediente" è di tipo booleano.</p> <p>(RV3) L'attributo con_lattosio dell'entità "ingrediente" è di tipo booleano.</p> <p>(RV4) L'attributo newsletter della relazione "registrazione" è di tipo booleano.</p> <p>(RV5) L'attributo senza_glutine dell'entità "ricetta" è di tipo booleano.</p> <p>(RV6) L'attributo senza_lattosio dell'entità "ricetta" è di tipo booleano.</p> <p>(RV7) L'attributo gourmet della relazione "ricetta" è di tipo booleano.</p> <p>(RV8) Un utente registrato non deve recensire una ricetta che ha pubblicato.</p> <p>(RV9) Un'istanza dell'entità utente registrato non deve partecipare sia alla relazione "allergia_alimentare_verso" che alla relazione "intolleranza_alimentare_verso" con la stessa istanza di ingrediente.</p>
REGOLE DI DERIVAZIONE
<p>(RD1) numero_recensioni dell'entità "ricetta" si ottiene contando il numero di volte che una data istanza dell'entità ricetta partecipa all'associazione recensione con una o più istanze dell'entità "utente registrato".</p> <p>(RD2) #recensioni_fatte dell'entità "utente registrato" si ottiene tenendo conto del numero di volte che la stessa istanza dell'entità "utente registrato" è associata a un'istanza dell'entità "recensione".</p> <p>(RD3) La media_apprezzamento dell'entità "ricetta" si ottiene sommando i k valori dell'attributo voto_apprezzamento (diversi da NULL) delle m istanze dell'entità "recensione" a cui partecipa una particolare istanza dell'entità ricetta, diviso m.</p>

### 3.5.3 SPECIFICHE SULLE OPERAZIONI

Di seguito vengono riportate per la nostra applicazione una lista di operazioni definite sui dati descritti dallo schema E-R ottenuto dalla progettazione concettuale:

- **Operazione 1:** Aggiunta di una ricetta (25 volte al giorno).
- **Operazione 2:** Visionare tutti i dati di una ricetta (200 volte al giorno).
- **Operazione 3:** Ricerca di una ricetta in base agli ingredienti (300 volte al giorno).
- **Operazione 4:** Assegna un ingrediente ad una ricetta (100 volte al giorno).
- **Operazione 5:** Inserire un nuovo ingrediente (100 volte al giorno).
- **Operazione 6:** Stampare tutti gli ingredienti che fanno parte di una ricetta (150 volte al giorno).
- **Operazione 7:** Visionare tutti gli ingredienti con lattosio verso cui un utente registrato è intollerante (200 volte al giorno).
- **Operazione 8:** Stampare il nickname degli utenti registrati che non hanno mai pubblicato ricette di categoria “Primi piatti” (20 volte al giorno).
- **Operazione 9:** Inserire una nuova recensione (25 volte al giorno).
- **Operazione 10:** Stampare il numero di recensioni totali di una ricetta (100 volte al giorno).
- **Operazione 11:** Visionare il numero di recensioni fatte da un utente registrato (30 volte al giorno).
- **Operazione 12:** Inserire un utente registrato (100 volte al giorno).
- **Operazione 13:** Stampare l’elenco di tutte le ricette pubblicate di un utente registrato con votazione maggiore di tre (20 volta alla settimana).
- **Operazione 14:** Inserire un nuovo utente (200 volte al giorno).
- **Operazione 15:** Cancellazione di un ingrediente (5 volte al settimana).
- **Operazione 16:** Visionare tutte le ricette senza glutine (250 volte al giorno).
- **Operazione 17:** Effettuare una statistica sugli utenti registrati che hanno pubblicato almeno 10 ricette e sugli ingredienti di cui esse sono composte. (1 volta al mese).
- **Operazione 18:** Invio aggiornamenti tramite newsletter agli utenti registrati (1 volta a settimana).

## 4 PROGETTAZIONE LOGICA

In questa fase viene riorganizzato lo schema E-R risultante dalla progettazione concettuale attraverso uno step di ristrutturazione, e successivamente si applica uno step di traduzione verso il modello logico scelto (in questo caso il modello relazionale).

-Ristrutturazione dello schema E-R: prende in input lo schema concettuale (risultato della progettazione concettuale) e le informazioni sul carico applicativo previsto, in termini di dimensione dei dati e caratteristiche delle operazioni, e dà come output uno schema E-R ristrutturato.

- Traduzione verso il modello logico: prende in input lo schema E-R ristrutturato e il modello logico scelto (in questo progetto è stato scelto il modello relazionale) e produce come risultato lo schema logico della base di dati per la piattaforma di condivisione di ricette online.

- Come prima cosa vado a studiare il carico applicativo in termini di dimensione dei dati e caratteristiche delle operazioni da effettuare sulla base di dati.

Sebbene mi stia concentrando su un'analisi delle prestazioni che fa riferimento ad un set ristretto di operazioni, la quale potrebbe risultare riduttiva a confronto del carico reale della base di dati che verrà progettata, va ricordato che le operazioni seguono la famosa **regola “ottanta venti”**, la quale è applicata anche in svariati contesti diversi dalla progettazione di una base di dati per la sua efficacia.

## 4.1 Tavola dei volumi

Concetto	Tipo	Volume
Ingrediente	E	140
Utente	E	400
Utente registrato	E	150
Ricetta	E	40
Categoria	E	10
Contiene	R	$\leq 350$
Registrazione	R	150
Recensione	R	120
Allergia_alimentare_verso	R	$\leq 750$
Intolleranza_alimentare_verso	R	$\leq 900$
Pubblicazione	R	40
Di	R	40

N.B. «le intolleranze alimentari sono più comuni delle allergie alimentari».

### Cardinalità relazione Allergia\_alimentare\_verso:

Ipotizziamo arbitrariamente che un utente registrato è allergico in media a 5 ingredienti.

Ipotizziamo quindi che la cardinalità della relazione molti-a-molti “Allergia\_alimentare\_verso” sia:

$$|\text{Allergia\_alimentare\_verso}| \leq 5 * 150 = 750 \text{ istanze.}$$



**Cardinalità relazione Intolleranza\_alimentare\_verso:**

Ipotizziamo arbitrariamente che un utente registrato è intollerante in media a 6 ingredienti.

- Assumo che la cardinalità della relazione “Intolleranza\_alimentare\_verso” multi-a-molti sia:

$$|\text{Allergia\_alimentare\_verso}| \leq 6 * 150 = 900 \text{ istanze.}$$

**Cardinalità relazione Recensione:**

Supponiamo arbitrariamente che in media ci siano 3 recensioni su ogni ricetta.

Ipotizziamo arbitrariamente quindi che la cardinalità della relazione multi-a-molti “recensione” sia:

$$|\text{Recensione}| \leq 40 * 3 = 120 \text{ record}$$

## 4.2 Tavola delle operazioni

Operazione	Tipo	Frequenza
O1	I	25 volte al giorno
O2	I	200 volte al giorno
O3	I	300 volte al giorno
O4	I	100 volte a settimana
O5	I	100 volte al giorno
O6	I	150 volte al giorno
O7	I	200 volte al giorno
O8	I	20 volte al giorno
O9	I	25 volte al giorno
O10	I	100 volte al giorno
O11	I	30 volte al giorno
O12	I	100 volte al giorno
O13	I	20 volte a settimana
O14	I	200 volte al giorno
O15	I	5 volte a settimana
O16	I	250 volte al giorno
O17	B	2 volte al mese
O18	B	1 volta a settimana

### **4.3 RISTRUTTURAZIONE DELLO SCHEMA E-R**

- ANALISI DELLE RIDONDANZE
- ELIMINAZIONE DELLE GENERALIZZAZIONI
- PARTIZIONAMENTO/ACCORPAMENTO DI CONCETTI
- SCELTA DEGLI IDENTIFICATORI PRINCIPALI

### 4.3.1 Analisi delle ridondanze

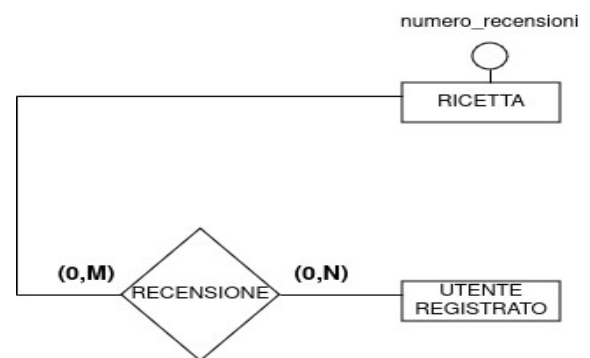
«Gli attributi ridondanti **a volte** è utile mantenerli per velocizzare l'esecuzione di certe operazioni (ad esempio le Query). Il dato ridondante invece pesa di più per le operazioni di scrittura, quindi per quelle di modifica.»

Nello schema prodotto in fase di progettazione concettuale sono presenti delle ridondanze. In particolare gli attributi **numero\_recensioni**, **#recensioni\_fatte**, **media\_apprezzamento** sono tutti dei dati derivabili e quindi delle ridondanze.

«» L'attributo **numero\_recensioni** dell'entità “**Ricetta**” è un dato derivabile, e quindi una ridondanza. Bisogna quindi valutare se conviene mantenere tale ridondanza o meno in termini di occupazione di memoria (quindi il numero di byte richiesto per memorizzare il dato ridondante) e di costo delle operazioni che interessano il dato ridondante.

Andiamo quindi a valutare gli indici di prestazione in caso di presenza e assenza del dato ridondante.

- Costo memoria per l'attributo **numero\_recensioni**: il dato ridondante richiede 4 byte \* 40 = 160 byte, quindi mantenerlo memorizzato costa per la precisione **0.16 kB**.



Le **operazioni 9 e 10** sono legate a questo dato ridondante, e faremo le valutazioni delle prestazioni in base ad esse.

Ricordando che  $1S = 2L$ .

**Con la ridondanza)**

**O9**

Costrutto	Tipologia	Operazione
Utente registrato	E	1L
Recensione	R	1S

Ricetta	E	1L + 1S (update)
---------	---	------------------

$$O9 = 2S + 2L = 6L * 25 \text{ v/gg} = 150 \text{ L/gg.}$$

**O10**

Costrutto	Tipologia	Operazione
Ricetta	E	1L

$$O10 = 1L * 100 \text{ v/gg} = 100 \text{ L/gg.}$$

$$\text{TOT con la ridondanza} = O9 + O10 = 150 \text{ L/gg} + 100 \text{ L/gg} = 250 \text{ accessi al giorno.}$$

**Senza la ridondanza)**

**O9**

Costrutto	Tipologia	Operazione
Recensione	R	1S

$$O9 = 1S = 2L * 25 \text{ v/gg} = 50 \text{ L/gg.}$$

**O10**

Costrutto	Tipologia	Operazione
Recensione	R	L

step di setup: essendo una relazione molti a molti procedo così:

|recensione| <= 120

volume di ricetta = 40

per trovare il numero magico faccio  $120/40 = 3$ .

$$O10 = 3L * 100 \text{ v/gg} = 300 \text{ L/gg.}$$

$$\text{TOT senza la ridondanza} = O9 + O10 = 50 \text{ L/gg} + 300 \text{ L/gg} = 350 \text{ accessi al giorno.}$$

TOT con la ridondanza < Tot senza la ridondanza.

**Mantengo quindi la ridondanza, l'attributo ridondante.**

Oppure potevo procedere per questa strada:  
considero x come incognita

$$O_{10} = x * 100 \text{ v/gg} = 100x$$

$$\text{TOT senza ridondanza} = 100x + 50 \text{ L/gg}$$

$$\text{TOT con ridondanza} = 250 \text{ accessi al giorno}$$

$$250 < 100x + 50$$

$$200 < 100x$$

per  $x > 2$  conviene mantenere la ridondanza.

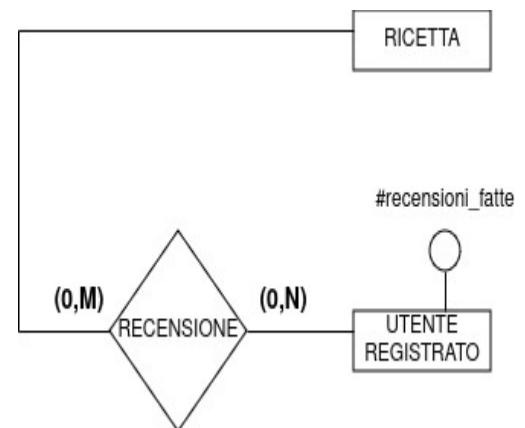
Si conclude che in termini di spazio e di efficienza delle operazioni che interessano quel dato mi conviene mantenere la ridondanza.

**Mantengo quindi l'attributo ridondante numero\_recensioni.**

«» L'attributo **#recensioni\_fatte** dell'entità “Ricetta” è un dato derivabile, e quindi una ridondanza. Bisogna quindi valutare se conviene mantenere tale ridondanza o meno in termini di occupazione di memoria (quindi il numero di byte richiesto per memorizzare il dato ridondante) e di costo delle operazioni che interessano il dato ridondante.

Andiamo quindi a valutare gli indici di prestazione in caso di presenza e assenza del dato ridondante.

- Costo memoria per l'attributo **#recensioni\_fatte**: il dato ridondante richiede  $4 \text{ byte} * 150 = 600 \text{ byte}$ , quindi mantenerlo memorizzato costa **0.6 kB**.



Le operazioni legate a questo dato che andrò a considerare nel calcolo e su cui farò le analisi delle prestazioni sono l'**operazione 9**, l'**operazione 11**. Andiamo a studiare i casi con e senza il dato ridondante.

**Con la ridondanza)**

**O9**

Costrutto	Tipologia	Operazione
Recensione	R	1S
Utente registrato	E	1L + 1S

$$O_9 = 2S + 1L = 5L * 25 \text{ v/gg} = 125 \text{ L/gg}.$$

**O11**

Costrutto	Tipologia	Operazione
Utente registrato	E	1L

$$O11 = 1L * 30 \text{ v/gg} = 30 \text{ L/gg.}$$

$$\text{TOT con la ridondanza} = O9 + O11 = 125 \text{ L/gg} + 30 \text{ L/gg} = 155 \text{ accessi al giorno.}$$

**Senza la ridondanza)**

**O9**

Costrutto	Tipologia	Operazione
Recensione	R	1S

$$O9 = 1S = 2L * 25 \text{ v/gg} = 50 \text{ L/gg.}$$

**O11**

Costrutto	Tipologia	Operazione
Recensione	R	L
Utente registrato	E	L

prima avevamo supposto che la cardinalità per la relazione “recensione” fosse  $\leq 120$ .  
 $|recensione| / \text{volume dell'entità ricetta} = 120 / 40 = 3$ .

$$O11 = 3L * 30 \text{ v/gg} = 90 \text{ L/gg.}$$

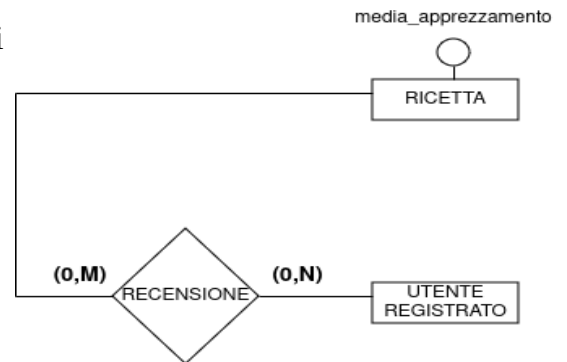
$$\text{TOT senza la ridondanza} = O9 + O11 = 50 \text{ L/gg} + 90 \text{ L/gg} = 140 \text{ accessi al giorno.}$$

TOT con la ridondanza > TOT senza la ridondanza => **conviene eliminare il dato ridondante.**

**Decido di eliminare l'attributo ridondante #recensioni\_fatte nello schema.  
 Elimino la ridondanza.**

«» L'attributo **media\_apprezzamento** dell'entità "Ricetta" è un dato derivabile, e quindi una ridondanza. Bisogna quindi valutare se conviene mantenere tale ridondanza o meno in termini di occupazione di memoria (quindi il numero di byte richiesto per memorizzare il dato ridondante) e di costo delle operazioni che interessano il dato ridondante.

- Costo memoria per l'attributo **media\_apprezzamento**:  
il dato ridondante richiede  $4 \text{ byte} * 40 = 160 \text{ byte}$ , quindi mantenerlo memorizzato costa **0.16 kB**.



N.B.  $1S = 2L$ .

Le operazioni che coinvolgono il dato ridondante per cui valuteremo le prestazioni sono:  
l'**operazione 9** e l'**operazione 13**.

**Con la ridondanza)**

**O9**

Costrutto	Tipologia	Operazione
Recensione	R	1S
Utente Registrato	E	1L
Ricetta	E	1L + 1S (update)

**O9** =  $2S + 1L + 1L = 6L * 25 \text{ v/gg} = 150 \text{ L/gg}$ . (costo giornaliero)  
costo settimanale(5gg) =  $150 \text{ L/gg} * 5\text{gg} = 750 \text{ L/sett}$ .

**O13**

Costrutto	Tipologia	Operazione
Ricetta	E	1L

**O13** =  $1L * 20 \text{ v/sett} = 20L/\text{sett}$ .

**TOT con la ridondanza** =  $O10 + O14 = 750 \text{ L/sett} + 20 \text{ L/sett} = 770 \text{ accessi a settimana}$ .



## Senza la ridondanza)

### O9

Costrutto	Tipologia	Operazione
Recensione	R	1S

$O9 = 1S = 2L * 25 \text{ v/gg} = 50 \text{ L/gg.}$  (Costo giornaliero)  
costo settimanale (5gg) =  $50 \text{ L/gg} * 5\text{gg} = 250 \text{ L/sett.}$

### O13

Costrutto	Tipologia	Operazione
Recensione	R	L
Ricetta	E	L

seguendo la seguente strada senza fare nessuna supposizione avremo che:  
x è un'incognita che rappresenta il numero di record che ci servono per il calcolo

$O13 = x * 20 \text{ v/sett} = 20x$   
TOT senza la ridondanza =  $20x + 250$   
 $770 < 20x + 250$   
 $520 < 20x$   
 $x > 26$

quando x è maggiore di 26 conviene mantenere la ridondanza.

**Mantengo quindi il dato ridondante nello schema.**

### - Risultato dell'analisi delle ridondanze:

**ridondanze da mantenere:** numero\_recensioni, media\_apprezzamento.

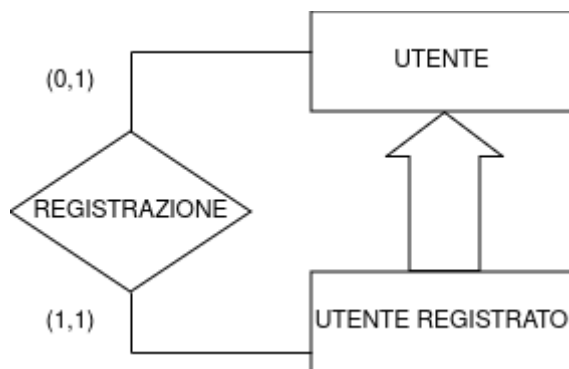
**ridondanze da togliere dallo schema:** #recensioni\_fatte.

### 4.3.2 Eliminazione delle generalizzazioni

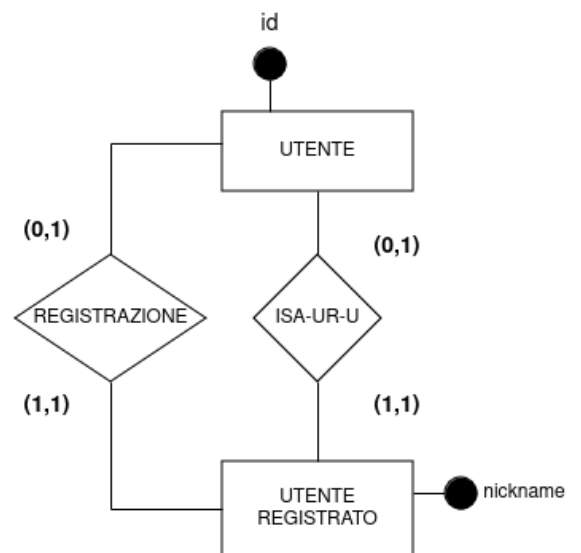
- Eliminazione della generalizzazione fra l'entità "utente" e il sottoinsieme "utente registrato".

«Attenzione quando diciamo eliminiamo non è che viene eliminata la generalizzazione, ma viene tradotta con qualcosa senza perdere la semantica»

#### Prima



#### Dopo



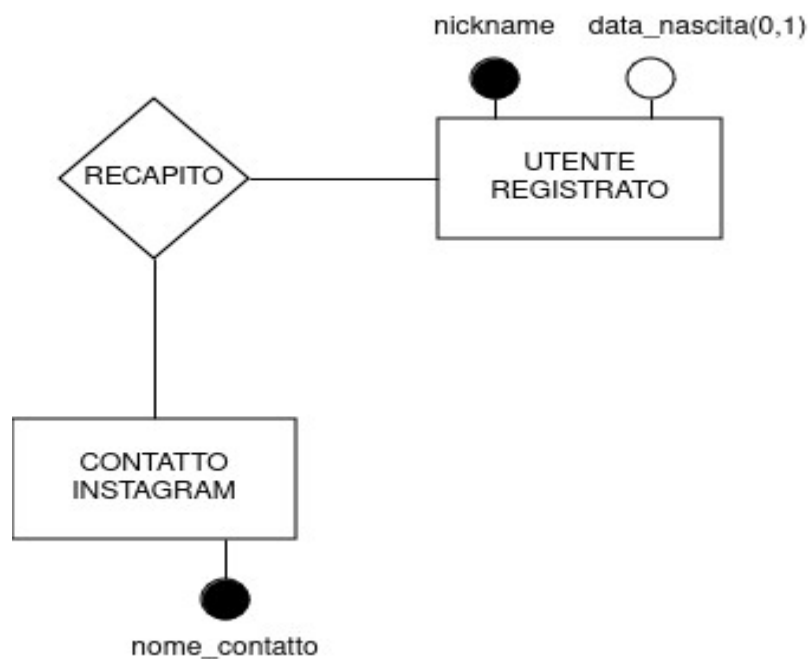
### 4.3.3 Partizionamento di concetti: eliminazione di attributi multivalore

Nello schema E-R non-ristrutturato prodotto sono presenti degli attributi multivalore.

Questo rappresenta un caso particolare di partizionamento di concetti.

In particolare l'attributo **contatto\_instagram** dell'entità "utente registrato" ha cardinalità (0,N), e quindi è multivalore e bisogna ristrutturare lo schema di conseguenza.

Eliminazione dell'attributo multivalore **contatto\_instagram**



#### 4.3.4 Scelta degli identificatori principali

«Un solo identificatore di ogni entità verrà utilizzato come chiave primaria. Su ogni chiave primaria verranno costruiti delle strutture ausiliarie (gli indici) per il reperimento efficiente dei dati»

- Faccio ora delle valutazioni sull'entità "ricetta", la quale è identificata dall'attributo **titolo** e dall'entità **utente registrato**. Essendo **titolo** e **nickname** stringhe (cioè insiemi di caratteri), quindi due attributi che probabilmente dovremo rappresentare come varchar ne risulta un identificatore pesante, che in una rappresentazione relazionale deve essere usato per rappresentare le due relazioni "Pubblicazione" e "Contiene"; le quali hanno molte istanze; e sicuramente in un'applicazione molto grande ed esigente le prestazioni ne risentirebbero abbastanza.

Quindi **decido di introdurre un nuovo identificatore per l'entità "ricetta"** costituito da un solo attributo, che chiamo **id\_ricetta**, il quale rimpiazza l'identificatore esterno dell'entità ricetta. Questo lo faccio per questioni di convenienza, applicando questa ristrutturazione allo schema E-R.

**id\_ricetta** sarà quindi l'**identificatore interno** dell'entità **ricetta**, quindi l'attributo che diventerà chiave primaria della tabella ricetta.

Di seguito viene riportata la lista degli identificatori principali degli attributi che verranno tradotti in chiave primaria dell'entità a cui appartengono durante la fase di traduzione nello schema relazionale:

**id\_ricetta** (dell'entità ricetta)

**nickname** (dell'entità utente registrato)

**id\_utente** (dell'entità utente)

**id\_ingredient** (dell'entità ingrediente)

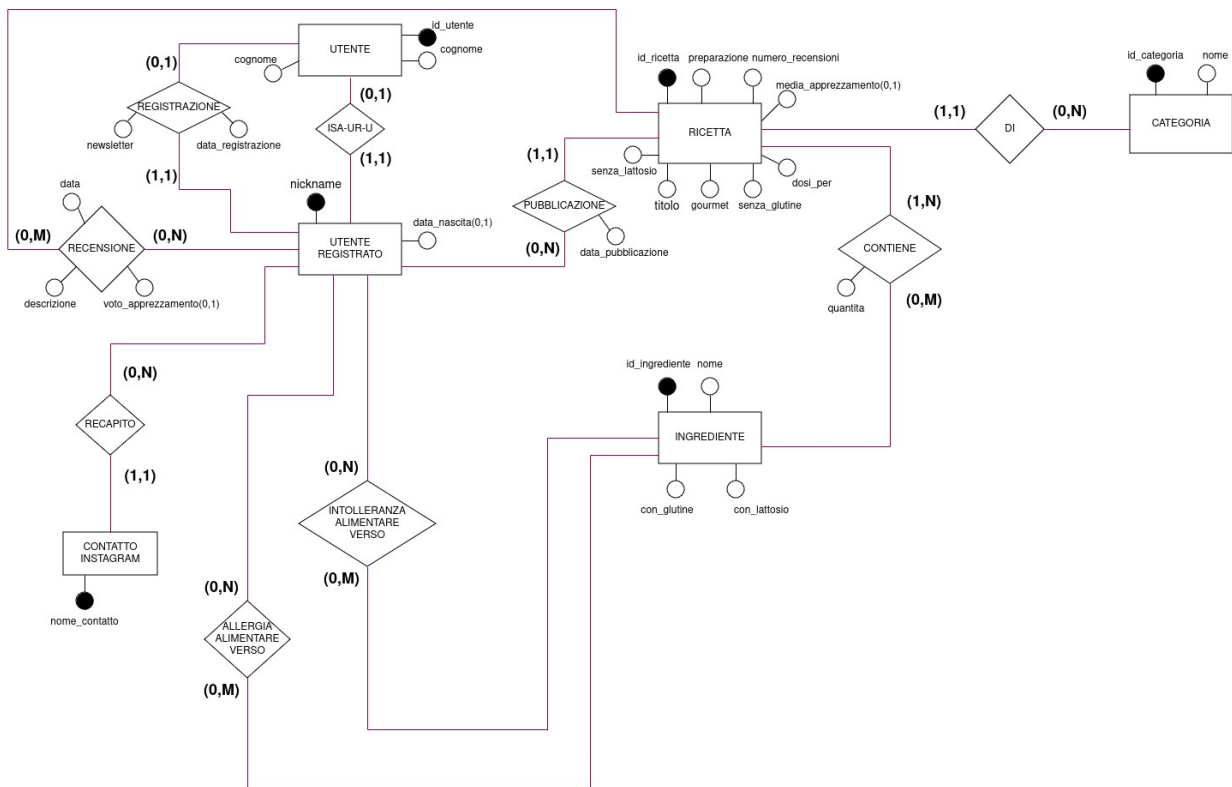
**nome\_contatto** (dell'entità contatto\_instagram)

**id\_categoria** (dell'entità categoria)

L'identificatore esterno **nickname+titolo** quindi è stato sostituito dal nuovo identificatore dell'entità "ricetta", cioè **id\_ricetta**.

Applico quindi questa ristrutturazione allo schema.

## 4.4 SCHEMA E-R RISTRUTTURATO



«Questo schema assieme al modello logico scelto (il modello relazionale in questo caso) rappresentano gli input per la seconda fase della progettazione logica, cioè la traduzione verso il modello relazionale.»

## 4.5 TRADUZIONE VERSO IL MODELLO RELAZIONALE

Per la traduzione dello schema E-R ristrutturato nello schema relazionale della nostra base di dati evidenzio gli attributi che costituiscono il vincolo di **primary key** di ogni tabella e sottolineo con la linea tratteggiata gli attributi che costituiscono il vincolo di **foreign key** tra più tabelle.

L'entità “**Categoria**” verrà tradotta in una tabella che chiameremo **CategoriaRicetta**.

La presenza dell'**asterisco** sui nomi degli attributi indica la possibilità di avere **valori nulli**.

### Schema relazionale

Ricetta(**id\_ricetta**, titolo, preparazione, gourmet, numero\_recensioni, data\_pubblicazione, senza\_glutine, senza\_lattosio, dosi\_per, nickname, id\_categoria, media\_apprezzamento\*)

Utente(**id\_utente**, nome, cognome)

UtenteRegistrato(**nickname**, data\_nascita\*, data\_registrazione, newsletter\*, id\_utente)

Contiene(id\_ricetta, id\_ingrediente, **quantita**)

Recensione(nickname, id\_ricetta, descrizione, data, voto\_apprezzamento\*)

Ingrediente(**id\_ingrediente**, nome, con\_glutine\*, con\_lattosio\*)

AllergiaAlimentareVerso(nickname, id\_ingrediente)

IntolleranzaAlimentareVerso(nickname, id\_ingrediente)

ContattoInstagram(**nome\_contatto**, nickname)

CategoriaRicetta(**id\_categoria**, nome)

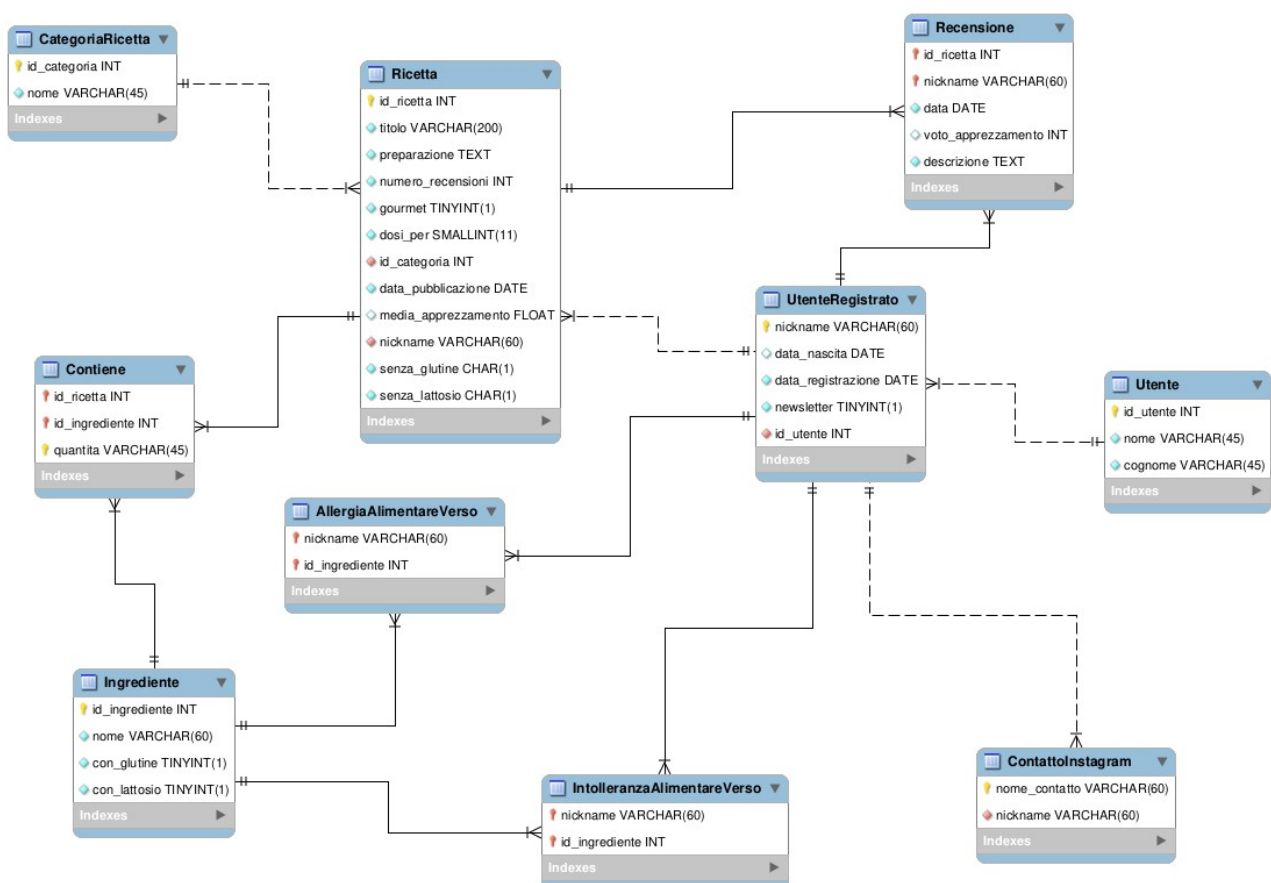
## 5 PROGETTAZIONE FISICA

Nella progettazione fisica della nostra base di dati verrà creato l'intero database e verranno implementate le operazioni previste.

- Il **DBMS relazionale** utilizzato è **MySQL**.

### 5.1 MODELLO FISICO

Grazie all'utilizzo del software MySQL Workbench è stata creata la struttura che dovrà avere lo schema:



## 5.2 DATABASE

«» Lo script **db\_ricette.sql** contiene il codice per creare il database.

«» Lo script **operazioni.sql** contiene l'implementazione in SQL delle operazioni previste.

«» Lo script **trigger.sql** contiene dei trigger.

- Di seguito è riportata una panoramica sul database che verrà popolato.

```
CREATE DATABASE 'piattaforma_ricette' DEFAULT CHARACTER SET utf8mb4;

USE 'piattaforma_ricette';

--
-- struttura della tabella 'utente'
--
CREATE TABLE utente (
  id_utente int(11) NOT NULL,
  name varchar(45) NOT NULL,
  cognome varchar(45) NOT NULL,
  PRIMARY KEY (id_utente)
)ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- struttura della tabella 'utente registrato'
--
CREATE TABLE utenteregistrato (
  nickname varchar(60) NOT NULL,
  data_nascita date DEFAULT NULL,
  data_registrazione date NOT NULL,
  newsletter tinyint(1) NOT NULL,
  id_utente int(11) NOT NULL,
  PRIMARY KEY (nickname),
  FOREIGN KEY (id_utente) REFERENCES utente (id_utente)
)ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- struttura della tabella 'ricetta'
--
CREATE TABLE ricetta (
```





```

82 -- struttura della tabella 'ingrediente'
83 --
84
85 CREATE TABLE ingrediente (
86     id_ingrediento int(11) NOT NULL,
87     nome varchar(45) NOT NULL,
88     con_glutine tinyint(1) NOT NULL,
89     con_lattosio tinyint(1) NOT NULL,
90     PRIMARY KEY (id_ingrediento)
91 )ENGINE=InnoDB DEFAULT CHARSET=utf8;
92
93
94 --
95 -- struttura della tabella 'contattoinstagram'
96 --
97
98 CREATE TABLE contattoinstagram (
99     nome_contatto varchar(60) NOT NULL,
100     nickname varchar(60) NOT NULL,
101     PRIMARY KEY (nome_contatto),
102     FOREIGN KEY (nickname) REFERENCES utenteregistrato (nickname)
103 )ENGINE=InnoDB DEFAULT CHARSET=utf8;
104

```

```

100 --
101 -- struttura della tabella 'recensione'
102 --
103
104 CREATE TABLE recensione (
105     nickname varchar(60) NOT NULL,
106     id_ricetta int(11) NOT NULL,
107     descrizione text NOT NULL,
108     data date NOT NULL,
109     voto_apprezzamento int(11) DEFAULT NULL CHECK (voto_apprezzamento >= 1 AND voto_apprezzamento <= 5),
110     PRIMARY KEY (id_ricetta, nickname),
111     FOREIGN KEY (id_ricetta) REFERENCES ricetta (id_ricetta),
112     FOREIGN KEY (nickname) REFERENCES utenteregistrato (nickname)
113 )ENGINE=InnoDB DEFAULT CHARSET=utf8;
114
115
116 --
117 -- struttura della tabella 'composizione'
118 --
119
120 CREATE TABLE contiene (
121     id_ricetta int(11) NOT NULL,
122     id_ingrediento int(11) NOT NULL,
123     quantita varchar(45) NOT NULL,
124     PRIMARY KEY (id_ricetta, id_ingrediento, quantita),
125     FOREIGN KEY (id_ricetta) REFERENCES ricetta(id_ricetta),
126     FOREIGN KEY (id_ingrediento) REFERENCES ingrediente(id_ingrediento)
127 )ENGINE=InnoDB DEFAULT CHARSET=utf8;
128

```

```
135 |
136 |
137 | CREATE TABLE intolleranza_alimentare_verso (
138 |     id_ingredienti int(11) NOT NULL,
139 |     nickname varchar(60) NOT NULL,
140 |     PRIMARY KEY (id_ingredienti, nickname),
141 |     FOREIGN KEY (id_ingredienti) REFERENCES ingredienti (id_ingredienti),
142 |     FOREIGN KEY (nickname) REFERENCES utentiregistrato (nickname)
143 | )ENGINE=InnoDB DEFAULT CHARSET=utf8;
144 |
145 | --
146 | -- struttura della tabella 'allergia_alimentare_verso'
147 | --
148 |
149 | CREATE TABLE allergia_alimentare_verso (
150 |     nickname varchar(60) NOT NULL,
151 |     id_ingredienti int(11) NOT NULL,
152 |     PRIMARY KEY (id_ingredienti, nickname),
153 |     FOREIGN KEY (id_ingredienti) REFERENCES ingredienti (id_ingredienti),
154 |     FOREIGN KEY (nickname) REFERENCES utentiregistrato (nickname)
155 | )ENGINE=InnoDB DEFAULT CHARSET=utf8;
156 |
157 |
```

## ***Riferimenti Bibliografici:***

- «Garcia-Molina, Ullman, Widrow Database Systems: The Complete Book, Prentice Hall»
- «Atzeni, Ceri, Paraboschi, Fraternali, Torlone, Basi di Dati, McGraw-Hill, quarta edizione»
- «Albano-Ghelli-Orsini, Fondamenti di basi di dati, Zanichelli»
- «Seyed M.M “Saied” Tahaghoghi and Hugh E. Williams, Learning MySQL, O'Really»

## ***Sitografia:***

- <https://dev.mysql.com/doc/refman/8.4/en/> (documentazione MySQL ufficiale).
- <https://codingnomads.com/how-to-use-the-sakila-database-in-mysql/>
- [https://it.wikipedia.org/wiki/Progettazione\\_di\\_basi\\_di\\_dati/](https://it.wikipedia.org/wiki/Progettazione_di_basi_di_dati/)