



UNIVERSITÀ
DEGLI STUDI
DI TORINO

Piattaforma di food delivery

PROGETTO BASI DI DATI - ANNO 2023/2024

ALESSANDRA RIZZA - 861889

MARCO MARIA MARCHISIO - 915317

YASSINE SADEK - 1051617

Progettazione concettuale.....	2
1.1 REQUISITI INIZIALI.....	2
1.2 GLOSSARIO DEI TERMINI.....	4
1.3 SUDDIVISIONE IN FRASI.....	5
Frase di carattere generale.....	5
Frase relative all'utente.....	5
Frase relative al borsellino elettronico.....	5
Frase relative ai ristoranti.....	5
Frase relative ai ristoranti Top Partner.....	5
Frase relative alle portate.....	5
Frase relative agli ordini.....	6
Frase relative ai riders.....	6
Frase relative alle classifiche mensili.....	6
1.4 SCHEMA E-R.....	7
Business Rules.....	8
Progettazione logica.....	9
2.1 TAVOLA DEI VOLUMI.....	9
2.2 TAVOLA DELLE OPERAZIONI.....	10
2.3 RISTRUTTURAZIONE DELLO SCHEMA E-R.....	11
Analisi delle ridondanze.....	11
Eliminazione delle generalizzazioni.....	14
2.4 SCHEMA RISTRUTTURATO.....	17
2.5 SCHEMA RELAZIONALE CON VINCOLI DI INTEGRITÀ REFERENZIALE.....	18
Implementazione.....	20
3.1 DDL (Data Definition Language).....	20
3.2 DML (Data Manipulation Language).....	25
3.3 DML (Data Manipulation Language - Modifica/Cancellazione).....	28

Progettazione concettuale

1.1 REQUISITI INIZIALI

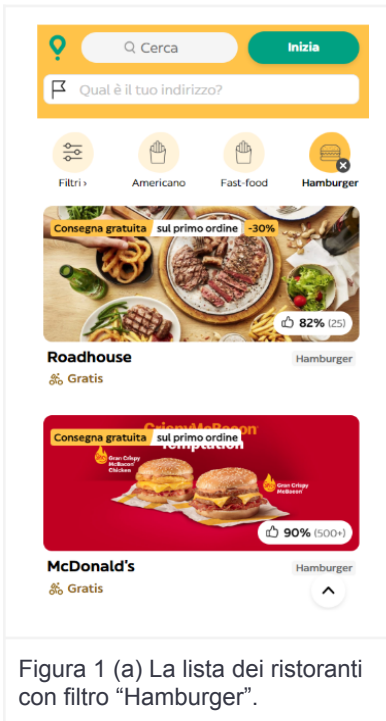


Figura 1 (a) La lista dei ristoranti con filtro "Hamburger".

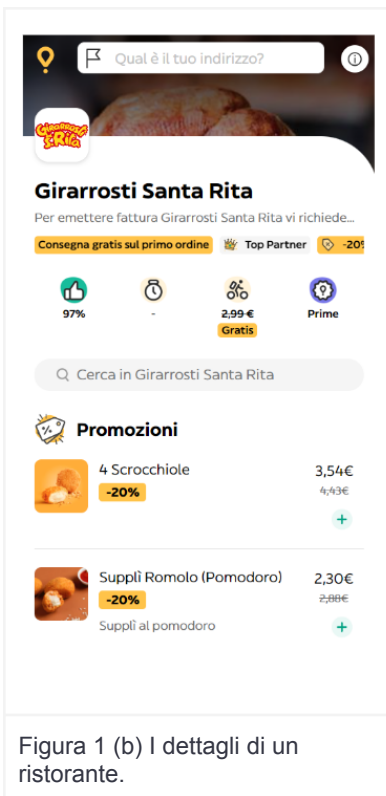


Figura 1 (b) I dettagli di un ristorante.

Si deve progettare la base di dati per Cibora (Figura 1(a)), un innovativo servizio di food delivery per gestire i dati dei ristoranti aderenti, degli utenti con i loro relativi ordini e dei fattorini che effettuano le consegne in bicicletta.

Per beneficiare del servizio, ogni utente deve registrarsi inserendo nome, email, password, numero di telefono, indirizzo di recapito.

Una volta registrato, l'utente deve inserire un mezzo di pagamento (es.: carta di credito, paypal, satispay) e ricaricare il proprio borsellino elettronico.

Il borsellino ha un saldo che viene aggiornato ad ogni ordinazione e l'utente può ricaricare il proprio borsellino in qualsiasi momento.

Inoltre, gli utenti possono sottoscrivere la modalità premium che garantisce una priorità sugli ordini. L'utente può collezionare codici di sconto da utilizzare al momento dell'ordine in base al numero di ordini effettuati in passato.

Ogni ristorante (Figura 1(b)) è rappresentato da un nome, una descrizione, un indirizzo, il costo della spedizione, un'immagine di profilo e un numero di stelline aggiornato ogni lunedì sulla base della percentuale di recensioni positive dell'ultima settimana.

Ogni ristorante appartiene a una o più categorie in base al tipo di cibo offerto (ad esempio: fast food, vegetariano, ...).

I ristoranti che dimostrano di saper garantire un ottimo servizio (almeno 20 ordini consegnati correttamente, una valutazione clienti maggiore o uguale a 4.5 stelline su cinque, una percentuale massima di ordini annullati dal ristorante dell'1.5%, una percentuale massima di ordini con reclami del 2.5%) sono considerati Top Partner.

I Top Partner compaiono in sezioni dedicate all'interno dell'app mobile Cibora e ricevono uno speciale badge che attesta il loro servizio eccellente, aiutando ad aumentare la credibilità e ottenere la fiducia dei clienti. Per i Top Partner si vuole tenere traccia della data in cui sono entrati a far parte della categoria.

I ristoranti propongono agli utenti una lista di piatti da ordinare.

Ogni portata ha un titolo, un'immagine, una lista di ingredienti, una lista di allergeni, il prezzo e un eventuale sconto. Inoltre, ogni piatto appartiene ad una o più liste (es. i più venduti, promozioni, dolci, salato, ecc.).

Ogni utente può selezionare una lista di pietanze ed effettuare l'ordine.

Finché non sono affidati ad un rider per la consegna, gli ordini possono essere annullati sia dai clienti, sia dai ristoratori.

Nel profilo dell'utente si possono ispezionare gli ordini passati ed eventualmente effettuare dei

reclami inviando un messaggio al ristorante.

Il sistema gestisce un numero arbitrario di riders dove ogni rider è identificato da un codice, dallo stato (occupato/disponibile/fuori servizio), dalla posizione aggiornata in tempo reale tramite GPS. I riders sono classificati in base al tipo di mezzo che utilizzano (bicicletta normale, bicicletta elettrica, monopattino). I riders che utilizzano il monopattino devono indicare quanti km possono effettuare prima che si scarichi la batteria.

Al momento dell'ordine, il sistema trova il rider libero con la somma minima della distanza dal ristorante più la distanza dall'utente.

Tuttavia, per ordini che prevedano un tragitto “posizione corrente del rider-> ristorante-> cliente” superiore ai 10 km, solo i rider con bici elettrica vengono interpellati.

Per monitorare le prestazioni dei ciclofattorini, si vuole tenere traccia del numero di consegne effettuate da ognuno, del momento in cui il cibo da consegnare viene affidato ad un rider e, per le consegne già completate, anche dell'ora in cui l'ordine è stato recapitato al cliente.

Dopo che l'ordine è stato effettuato l'utente ha la possibilità di chattare sia con il ristorante che con il rider in caso ci fossero dei problemi con l'ordine come mancata consegna o netto ritardo. Quando l'ordine è consegnato l'utente può recensire il ristorante e il rider con una valutazione da 1 a 5 e un commento testuale. Il commento testuale è facoltativo. Inoltre è anche presente la possibilità di dare una mancia al rider per la consegna.

Una volta al mese, vengono aggiornate le seguenti classifiche:

requisiti di
gestione del
servizio

- Riders più veloci nel consegnare gli ordini.
- Cibi più popolari.
- Ristoranti con più recensioni positive.
- Clienti che hanno speso di più.

1.2 GLOSSARIO DEI TERMINI

TERMINI	DESCRIZIONE	SINONIMI
Utente	Persona fisica che utilizza la piattaforma.	
Utente registrato	Utente che ha svolto il processo di registrazione	
Ristoranti	Rappresenta il locale presente nella piattaforma.	Ristorante aderente, Ristoratori
Ristorante “Top Partner”	Ristorante aderente che soddisfano specifici requisiti.	
Portata	Piatto che il ristorante propone.	Piatti
Lista	Raggruppamento per tipologia dei piatti proposti.	
Ordini	Richiesta di piatti da parte degli utenti	
Rider	Colui che effettua la consegna dell’ordine.	Fattorino
Borsellino	Strumento fornito dalla piattaforma per effettuare i pagamenti.	Borsellino elettronico
Classifiche	Classifica mensile	

1.3 SUDDIVISIONE IN FRASI

Requisiti rivisti e strutturati in gruppi di frasi omogenee

Frasi di carattere generale

- Si vuole progettare la base di dati per Cibora, un innovativo servizio di food delivery per gestire i dati dei ristoranti aderenti, degli utenti con i loro relativi ordini e dei fattorini che effettuano le consegne in bicicletta.

Frasi relative all'utente

- Gli utenti per utilizzare la piattaforma (per effettuare gli ordini) devono registrarsi.
- Per gli utenti memorizziamo nome, email, password, numero di telefono, indirizzo di recapito, per registrarsi.
- Gli utenti iscritti devono inserire un mezzo di pagamento (es.: carta di credito, paypal, satispay) e ricaricare il proprio borsellino elettronico.
- Gli utenti possono sottoscrivere la modalità premium che garantisce una priorità sugli ordini.
- Gli utenti possono collezionare codici di sconto da utilizzare al momento dell'ordine in base al numero di ordini effettuati in passato.
- Ogni utente può selezionare una lista di pietanze ed effettuare l'ordine.
- Nel profilo dell'utente si possono ispezionare gli ordini passati ed eventualmente effettuare dei reclami inviando un messaggio al ristorante.

Frasi relative al borsellino elettronico

- Il borsellino ha un saldo che viene aggiornato ad ogni ordinazione.
- Ogni utente può ricaricare il proprio borsellino in qualsiasi momento.

Frasi relative ai ristoranti

- Per ogni ristorante viene memorizzato un nome, una descrizione, un indirizzo, il costo della spedizione, un'immagine di profilo.
- Per ogni ristorante viene memorizzato un numero di stelline, aggiornato ogni lunedì sulla base della percentuale di recensioni positive dell'ultima settimana.
- Ogni ristorante appartiene a una o più categorie in base al tipo di cibo offerto (ad esempio: fast food, vegetariano, ...).
- Per ogni ristorante viene fornita una lista di piatti disponibili per l'ordinazione.

Frasi relative ai ristoranti Top Partner

- I ristoranti che dimostrano di saper garantire un ottimo servizio (almeno 20 ordini consegnati correttamente, una valutazione clienti maggiore o uguale a 4.5 stelline su cinque, una percentuale massima di ordini annullati dal ristorante dell'1.5%, una percentuale massima di ordini con reclami del 2.5%) sono considerati **Top Partner**.
- I Top Partner compaiono in sezioni dedicate all'interno dell'app mobile Cibora e ricevono uno speciale badge.
- Per i Top Partner si vuole tenere traccia della data in cui sono entrati a far parte della categoria.

Frasi relative alle portate

- Per ogni portata viene memorizzato un titolo, un'immagine, una lista di ingredienti, una lista di allergeni, il prezzo e un eventuale sconto.
- Per ogni portata viene associata una o più liste (es. i più venduti, promozioni, dolci, salato, ecc.)

Frase relative agli ordini

- Finché non sono affidati ad un rider per la consegna, gli ordini possono essere annullati sia dai clienti, sia dai ristoratori.
- Dopo che l'ordine è stato effettuato l'utente ha la possibilità di chattare sia con il ristorante che con il rider.
- Quando l'ordine è consegnato l'utente può recensire il ristorante e il rider con una valutazione da 1 a 5 e un commento testuale (facoltativo).
- Viene fornita la possibilità di dare una mancia al rider per la consegna.

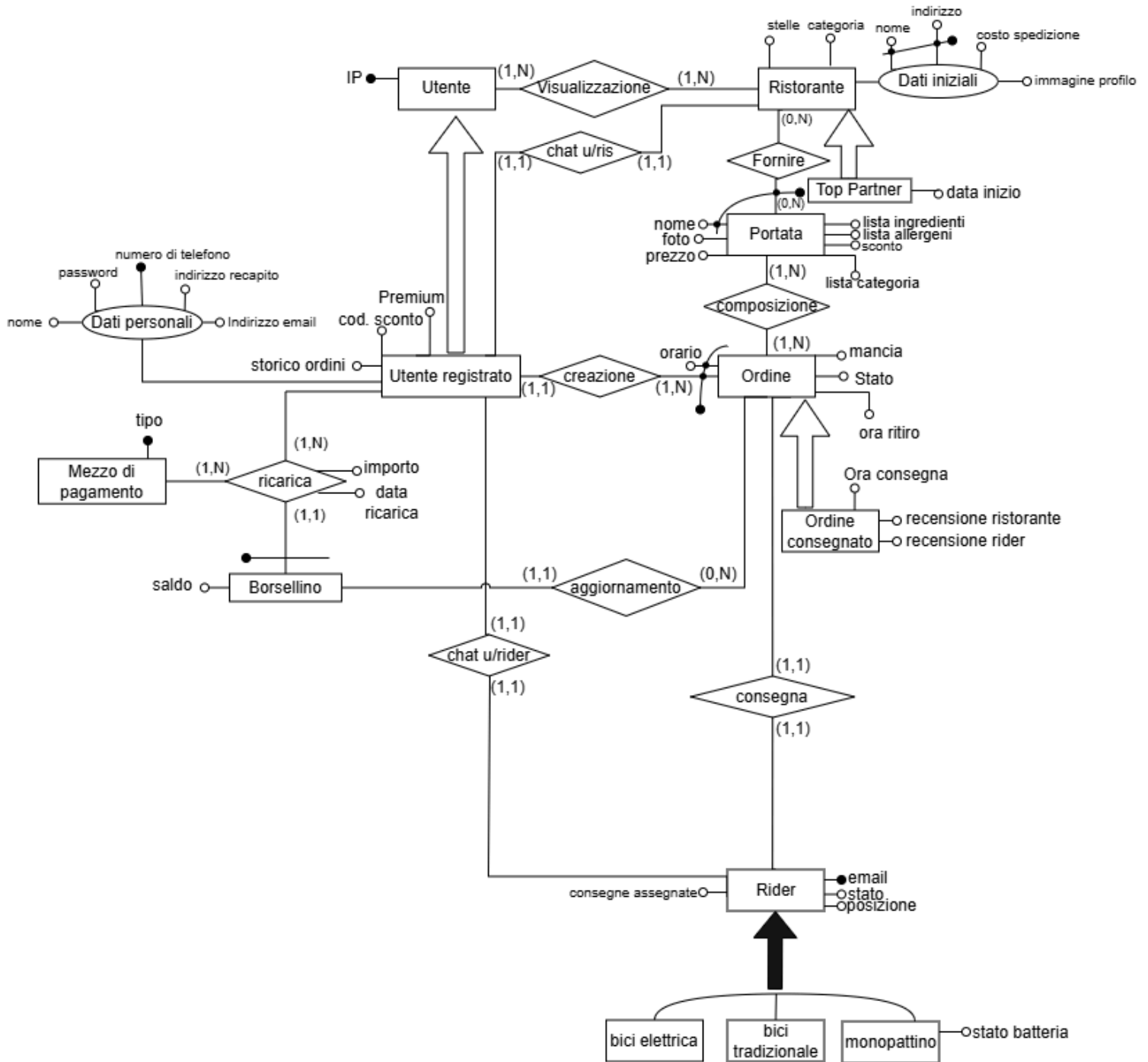
Frase relative ai riders

- Ogni rider è identificato da un codice, dallo stato (occupato/disponibile/fuori servizio) e dalla posizione aggiornata in tempo reale tramite GPS.
- I riders sono classificati in base al tipo di mezzo che utilizzano (bicycle normale, bicycle elettrica, monopattino).
- I riders che utilizzano il monopattino devono indicare quanti km possono effettuare prima che si scarichi la batteria.
- Al momento dell'ordine, il sistema trova il rider libero con la somma minima della distanza dal ristorante più la distanza dall'utente.
- Per ordini che prevedano un tragitto "posizione corrente del rider-> ristorante-> cliente" superiore ai 10 km, solo i rider con bicycle elettrica vengono interpellati.
- Per ogni fattorino si memorizza il numero di consegne effettuate, del momento in cui il cibo da consegnare viene affidato ad un rider e, per le consegne già completate, anche dell'ora in cui l'ordine è stato recapitato al cliente.

Frase relative alle classifiche mensili

- Una volta al mese, vengono aggiornate le seguenti classifiche:
 - Riders più veloci nel consegnare gli ordini.
 - Cibi più popolari.
 - Ristoranti con più recensioni positive.
 - Clienti che hanno speso di più.

1.4 SCHEMA E-R



Business Rules

- L'utente può collezionare codici di sconto da utilizzare al momento dell'ordine in base al numero di ordini effettuati in passato.
L'utente accumula un codice sconto del 5% sul prossimo ordine dopo aver effettuato con successo almeno 5 ordini superiori a 15 euro.
- Numero di stellette aggiornato ogni lunedì sulla base della percentuale di recensioni positive dell'ultima settimana.
- Un ristorante diventa Top Partner se rispetta i seguenti parametri contemporaneamente :
 - almeno 20 ordini consegnati correttamente
 - valutazione clienti maggiore o uguale a 4.5 stelline su cinque
 - percentuale massima di ordini annullati dai ristoranti dell'1.5%
 - percentuale massima di ordini con reclami del 2.5%i ristoranti che soddisfano queste condizioni compaiono in sezioni dedicate e ricevono una badge di attestazione per il servizio eccellente.
- Finché non sono affidati ad un rider per la consegna, gli ordini possono essere annullati sia dai clienti, sia dai ristoranti.
- La chat tra utente e ristorante e quella tra utente e rider possono essere avviate solo dopo la creazione di un ordine.
- Il rider destinato a consegnare l'ordine viene calcolato con la somma minima della distanza dal ristorante più la distanza dall'utente.
 - Per ordini che prevedono un tragitto "posizione corrente del rider-> ristorante-> cliente" superiore ai 10 km, solo i rider con bici elettrica vengono interpellati.

NB: per i valori nelle tavole successive abbiamo analizzato le statistiche di Glovo - Piemonte, in modo da avere dati più gestibili e una visione più controllata dell'utilizzo della piattaforma. I dati sono annuali.

Progettazione logica

analisi delle prestazioni del modello ER

2.1 TAVOLA DEI VOLUMI

Concetto	Tipo	Volume	Dettagli aggiuntivi
Utente	E	2 000 000	tutti gli utenti che consultano Cibora
Utente Registrato	E	1 500 000	utenti registrati alla piattaforma
Ristoranti	E	2 000	ristoranti registrati su Cibora
Ristoranti "Top Partner"	E	300	ristoranti con badge "Top partner"
Portate	E	30 000	numero di portate totali dei ristoranti
Ordine	E	50 000	numero di ordini totali
Ordine Consegnato	E	<50 000	al più pari agli ordini
Borsellino	E	1 500 000	un borsellino per ogni utente registrato (il borsellino può avere saldo = 0)
Rider	E	4 000	numero di riders
Rider - bici elettrica	E	1 500	numero di mezzi dei riders (1 a 1)
Rider - bici tradizionale	E	2 000	
Rider - monopattino	E	500	
Aggiornamento	R	50 000	una volta per ogni ordine
Chat U/Rider	R	<50 000	al più pari al numero di ordini
Chat U/Ris	R	<50 000	al più pari al numero di ordini
Composizione	R	>10 000	due composizioni per ogni ordine (stima a ribasso)
Consegna	R	<40 000	al più come gli ordini consegnati (l'ordine può essere annullato)
Creazione	R	50 000	pari al numero di ordini
Fornire	R	50 000	collegamento tra ristorante e portata
Ricarica	R	1 000	numero di ricariche del borsellino
Visualizzazione	R	50 000	visualizzazioni ogni giorno (non tutti ordinano)

2.2 TAVOLA DELLE OPERAZIONI

N.	Operazione	Tipo	Frequenza
1	<i>(registrazione utente)</i> Registrazione alla piattaforma.	I	2 500/mese
2	<i>(accesso utente)</i> Accesso alla piattaforma utilizzando le credenziali di accesso.	I	10 000/giorno utenti che entrano su Cibora
3	<i>(visualizzazione ristorante)</i> Accesso al ristorante per visualizzare informazioni e lista pietanze.	I	2 500/giorno tutti gli utenti che accedono vanno in qualche ristorante
4	<i>(ricarica borsellino)</i> Ricarica del saldo del borsellino.	I	500/giorno numero medio di ordini al giorno
5	<i>(creazione ordine)</i> Creazione di un nuovo ordine indicando titolo, durata e categoria.	I	500/giorno
6	<i>(controllo ordine)</i> Controllo stato dell'ordine fatto.	I	350/giorno
7	<i>(consegna ordine)</i> Chiusura dell'ordine aperto, aggiunta ora di consegna.	I	2500/giorno numero medio di spettatori
8	<i>(recensione ordine)</i> Aggiunta recensione del ristorante e dell'ordine	I	5000/giorno numero medio di interazioni
9	<i>(sottoscrizione premium)</i> Aggiornamento condizione premium.	B	3000/mese
10	<i>(creazione nuovo ristorante)</i> Aggiunta di un nuovo ristorante.	B	100/mese
11	<i>(promozione ristorante)</i> Promozione a Top Partner	B	30/mese
12	<i>(inserimento piatto)</i> Aggiunta di un nuovo piatto al menù	I	500/mese
13	<i>(visualizzazione numero consegne)</i> Visualizzare il numero di consegne fatte da un rider specifico	I	20/giorno dato che compare ad ogni assegnazione
14	<i>(gestione chat utenterider)</i> Gestione della chat tra utente e rider	I	4000/mese
15	<i>(gestione chat utente/ristorante)</i> Gestione della chat tra utente e il ristorante	I	4000/mese

2.3 RISTRUTTURAZIONE DELLO SCHEMA E-R

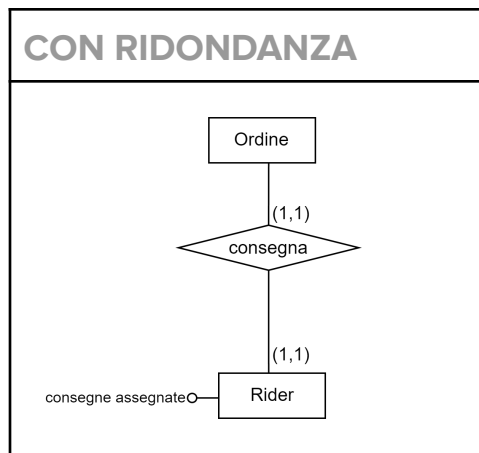
Analisi delle ridondanze

ANALISI OPERAZIONE 13: Visualizzare il numero di consegne fatte da un rider specifico.

attributo derivabile da operazioni di conteggio di occorrenze → il numero di consegne effettuate per un singolo rider può essere derivato contando le occorrenze dell'associazione "consegna" a cui tale rider partecipa.

Per effettuare questo studio possiamo dividere questa operazione in due sotto operazioni:

Operazione	note	Tipo	Frequenza
Op 1	memorizza un nuovo ordine con relativo rider	I	500/al giorno
Op 2	stampa tutti i dati di un rider, compreso il numero di consegne assegnate	I	20/al giorno



Costo in termini di spazio:

L'attributo "CONSEGNE ASSEGNATE" all'interno del DataBase sarà di tipo intero (int), quindi supponiamo 4 byte per la memorizzazione del numero di consegne assegnate ad un determinato rider.

Spazio totale occupato: $4 * 5000$ (numero di rider nella piattaforma) = 20 000 byte

Stima costo operazione Op 1:

Concetto	Costrutto	Accessi	Tipo	note
Rider	E	1	L	per cercare il rider di interesse
Rider	E	1	S	per incrementare di uno il numero di consegne di quella occorrenza
Consegna	R	1	S	per memorizzare una nuova coppia ordine-rider
Ordine	E	1	S	per memorizzare il nuovo ordine

Il tutto ripetuto per 500 volte al giorno, per un totale di:

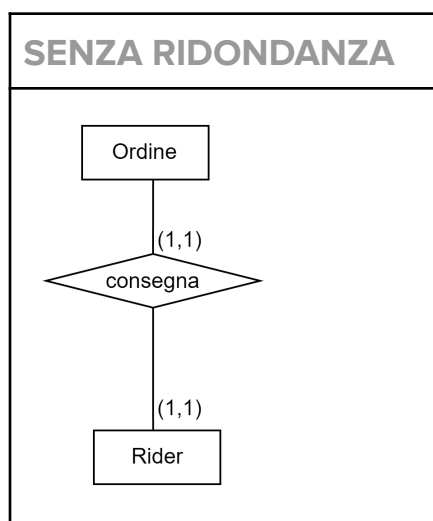
- 1500 accessi in scrittura ($500 * 3$ scritture)
- 500 accessi in lettura. ($500 * 1$ lettura)

Stima costo operazione Op 2:

Concetto	Costrutto	Accessi	Tipo	note
Rider	E	1	L	per cercare il rider di interesse

Richiede un accesso in lettura da ripetere 20 volte al giorno. ($20 * 1$ lettura)

Supponiamo che un accesso in scrittura abbia un costo doppio rispetto ad uno in lettura, abbiamo un totale di 3520 accessi in presenza di dato ridondante. [$(500 + 20$ per le letture) + $(2*1500$ per le scritture)]



Stima costo operazione Op 1:

Concetto	Costrutto	Accessi	Tipo
Ordine	E	1	S
Consegna	R	1	S

non è necessario accedere all'entità Rider per aggiornare il dato.

Il tutto ripetuto per 500 volte al giorno, per un totale di:

- 1 000 accessi in scrittura ($500 * 2$ scritture)

Stima costo operazione Op 2:

Concetto	Costrutto	Accessi	Tipo	note
Consenga	R	12 numero ordine / numero rider	L	per calcolare il numero medio di ordini assegnati ad un rider
Rider	E	1	L	per avere i dati del rider

Viene ripetuto 20 volte al giorno, per un totale di:

- 260 accessi in lettura ($20 * 13$ letture)

[$(260$ per le letture) + $(2*1000$ per le scritture)] per un totale di: 2260 accessi.

Quindi rimuovendo il dato rindondante c'è un significativo miglioramento per quanto riguarda il numero di accessi, è chiaro quindi che è meglio togliere la ridondanza risparmiando così anche in termini di spazio.

Un'altra rindondanza potrebbe creare dall'entità UTENTE REGISTRATO, in quanto possiede sia l'attributo "storico ordini" sia una associazione con l'entità ordine, potrebbe esserci così una ripetizione di dati. Duplicare queste informazioni in più entità potrebbe creare inconsistenze durante gli aggiornamenti.

Eliminazione delle generalizzazioni

1. Generalizzazione su **RIDER** → *accorpamento delle figlie della generalizzazione nel genitore*. Le entità BICI ELETTRICA, BICI TRADIZIONALE e MONOPATTINO vengono eliminate e le loro proprietà (attributi e partecipazioni ad associazioni e generalizzazioni) vengono aggiunte all'entità genitore RIDER.

A tale entità viene aggiunto un nuovo attributo (*mezzo_spostamento*) che serve a distinguere il tipo di mezzo utilizzato. questo attributo può avere valori come "bicicletta", "auto", "moto", ecc; in questo modo si potranno aggiungere facilmente nuovi tipi di mezzi in futuro senza dover ristrutturare il modello ER.

Viene aggiunto un ulteriore attributo (*stato_batteria*) questo attributo sarà popolato solo per i rider che utilizzano un mezzo che richiede una batteria (ad esempio, biciclette elettriche o scooter elettrici). Per i rider che utilizzano mezzi senza batteria, questo campo rimarrà nullo.

In questo modo, si mantiene una struttura semplice per il database e si garantisce una flessibilità per gestire le diverse tipologie di rider e le loro caratteristiche richieste.

procedura di eliminazione della generalizzazione:

1. Operazioni di lettura e modifica/inserimento

Con Generalizzazione:

- Lettura dei Dati:
 - Quando è necessario accedere alle informazioni di un rider, è obbligatorio passare attraverso la generalizzazione e successivamente accedere all'entità specifica (ad esempio, *Bici Elettrica*).
 - Per la lettura degli attributi specifici come lo *stato batteria*, bisogna accedere all'entità figlia specifica del rider.
- Modifica/Inserimento dei Dati:
 - Quando un rider cambia il tipo di mezzo di trasporto, è necessario eliminare l'istanza dall'entità figlia corrente e crearne una nuova nella corrispondente entità figlia.

Senza Generalizzazione:

- Lettura dei Dati:
 - Tutti i dati relativi al rider sono accessibili direttamente dall'entità *Rider*, inclusi attributi specifici come lo *stato batteria* e il *mezzo_spostamento*.
 - Nessuna necessità di navigare tra entità separate, riducendo il numero di accessi necessari.
- Modifica/Inserimento dei Dati:
 - L'inserimento o la modifica di dati avviene direttamente nell'entità *Rider* senza dover spostare i dati tra entità figli

2. Schemi delle Operazioni in Presenza e Assenza della Ridondanza

CON GENERALIZZAZIONE	SENZA GENERALIZZAZIONE
<p>Il rider è associato a una delle entità figlie (<i>Bici Elettrica</i>, <i>Bici Tradizionale</i>, <i>Monopattino</i>). Per accedere a informazioni specifiche come lo <i>stato batteria</i>, è necessario passare per l'entità figlia corrispondente.</p>	<p>Tutti gli attributi sono accorpati nell'entità <i>Rider</i>. Per distinguere tra i diversi mezzi di trasporto, viene aggiunto l'attributo <i>mezzo_spostamento</i>. Lo <i>stato batteria</i> è presente solo se pertinente, altrimenti rimane nullo.</p>

3. Tavole degli Accessi:

Con Generalizzazione:

Concetto	Costrutto	Accessi	Tipo Accesso
Rider	E	1	Lett/Scrit
Bici Elettrica	E	1	Lett/Scrit
Bicicletta	E	1	Lett/Scrit
Monopattino	E	1	Lett/Scrit

Senza Generalizzazione:

Concetto	Costrutto	Accessi	Tipo Accesso
Rider	E	1	Lett/Scrit

4. Spazio Occupato dalla Ridondanza

Con Generalizzazione:

- Ogni entità figlia occupa spazio separato per memorizzare gli attributi specifici e una chiave primaria che include il codice del rider.
- Considerando un sistema con 10.000 rider e un utilizzo medio di 20 byte per chiave e attributi specifici, lo spazio totale è di circa 200.000 byte.

Senza Generalizzazione:

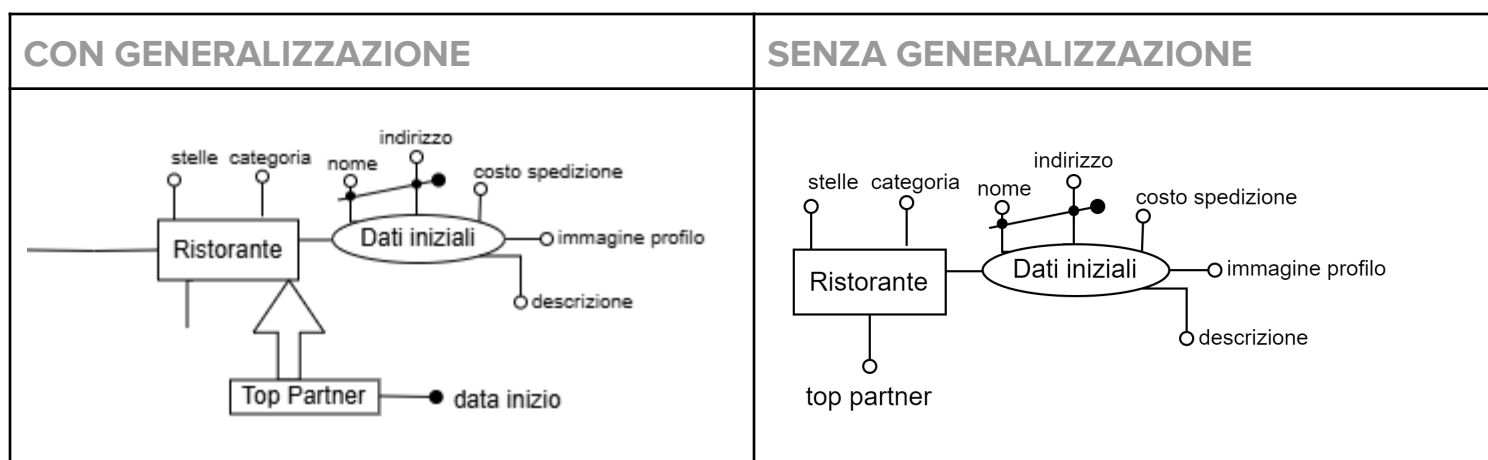
- Lo spazio è centralizzato nell'entità *Rider*, con la riduzione dello spazio di memorizzazione delle chiavi.
- Con una chiave e attributi specifici che occupano 12 byte per rider, lo spazio totale è di 120.000 byt.

5. Confronto dello Spazio e del Numero di Accessi

- Spazio:
 - Con Generalizzazione: 200.000 byte.
 - Senza Generalizzazione: 120.000 byte.
 - Risparmio di spazio: 80.000 byte.
- Numero di Accessi:
 - Con Generalizzazione: 20.000 accessi al giorno (10.000 letture e 10.000 modifiche/inserti).
 - Senza Generalizzazione: 10.000 accessi al giorno.
 - Riduzione del numero di accessi: 50%.

Generalizzazione su **RISTORANTE** → *accorpamento delle figlie della generalizzazione nel genitore.*

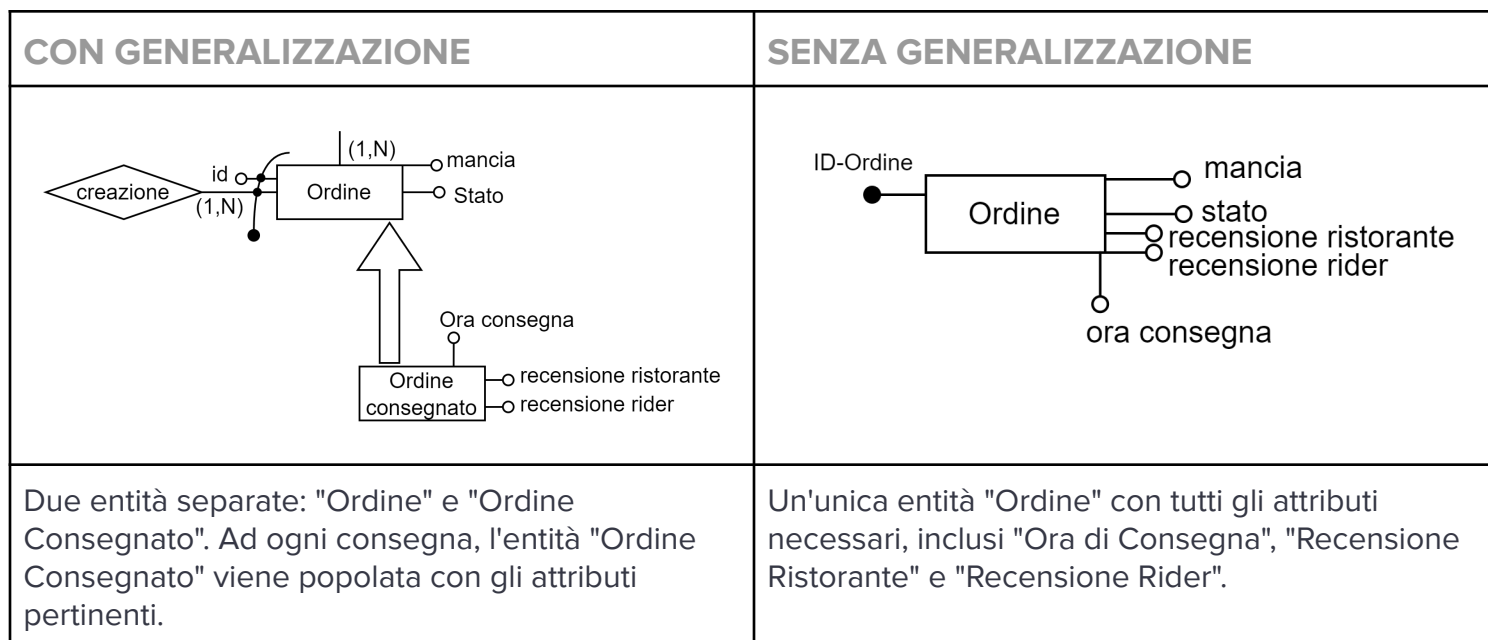
L'entità Top partner viene eliminata e le sue proprietà vengono aggiunte all'entità genitore Ristorante, alla quale viene aggiunto l'attributo "top partner"



Generalizzazione su **ORDINE** → *accorpamento delle figlie della generalizzazione nel genitore*

L'entità Ordine consegnato viene eliminata e accorpata all'entità Ordine, a quest'ultima si aggiungono gli attributi caratterizzanti dell'entità figlia. Questi attributi risultano però opzionali.

Attraverso l'attributo "stato" di ordine sarà possibile distinguere se l'ordine trattato è in uno dei seguenti casi di esempio : "In preparazione", "In consegna", "Consegnato".

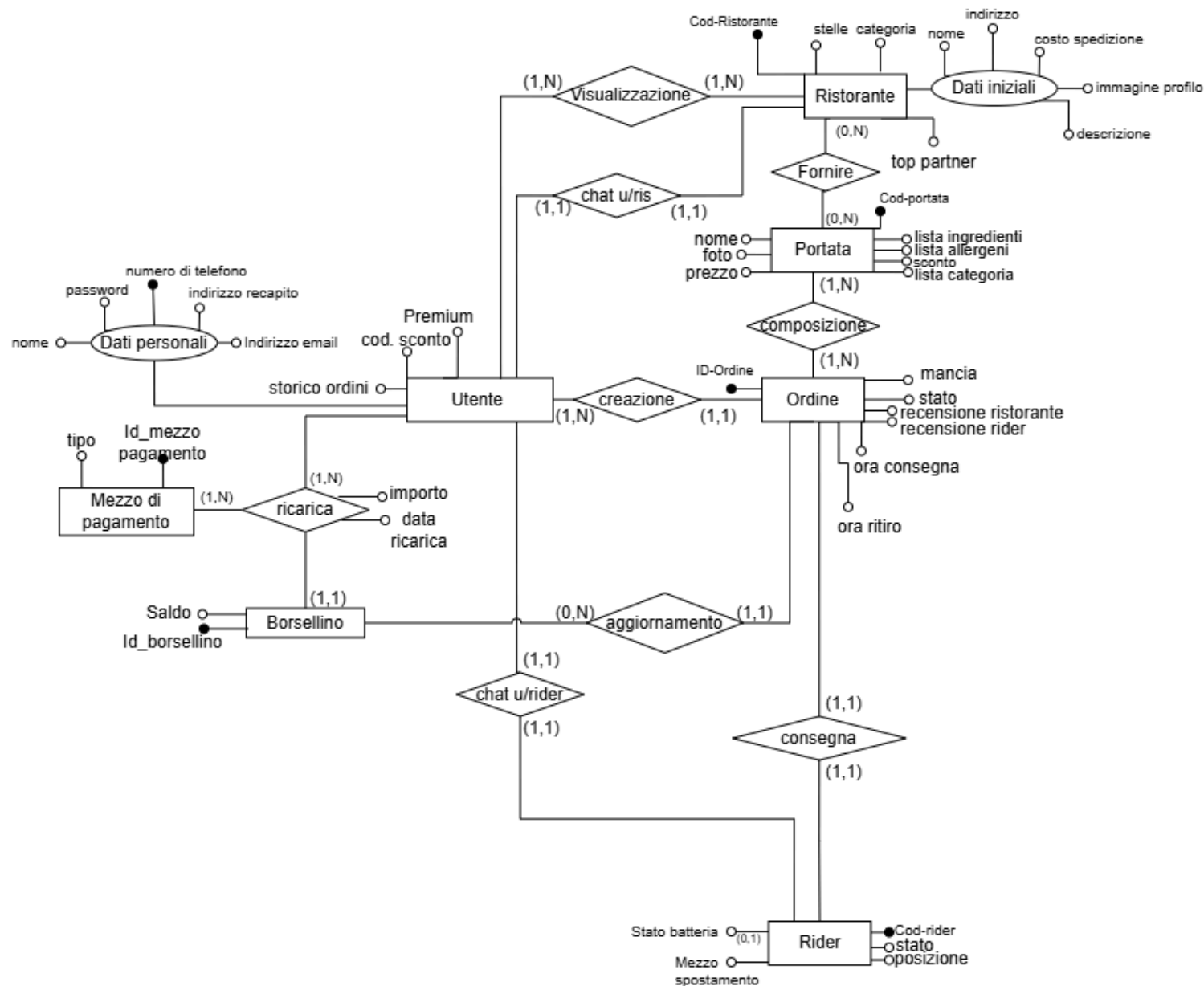


Generalizzazione su **UTENTE** → *accorpamento del genitore nella entità figlia.*

L'entità genitore UTENTE viene eliminata e, per la proprietà dell'ereditarietà, i suoi attributi, il suo identificatore e le relazioni a cui tale entità partecipava, vengono aggiunti alle entità figlia UTENTE REGISTRATO.

nb: abbiamo deciso di non considerare le visualizzazioni e le attività degli utenti anonimi (ovvero quelli non registrati) per il calcolo delle operazioni e dei costi, perché potrebbero essere molto variabili e non efficienti. (si potrebbe anche avere accessi da diversi dispositivi ma dallo stesso utente che aumentano le visualizzazioni, vengono per questo considerate solo quelle effettuate da utenti registrati).

2.4 SCHEMA RISTRUTTURATO



2.5 SCHEMA RELAZIONALE CON VINCOLI DI INTEGRITÀ REFERENZIALE

Utente (Numero_Telefono, Indirizzo_Email, Password, Nome_U, Indirizzo_Recapito, Premium, Cod_Sconto, Storico_Ordini, Borsellino.ID, Chat_U/Rist, Chat_U/Rider)

- Borsellino.ID è chiave esterna che referencia ID_Borsellino nella tabella Borsellino
- con vincolo referenziale tra l'attributo chat_U/Rist di Utente e l'attributo Cod_Ristorante di Ristorante.
- con vincolo referenziale fra l'attributo chat_U/rider di Utente e l'attributo Cod_Rider di Rider.

Borsellino (ID_borsellino, Saldo)

Mezzo_di_Pagamento (ID_Mezzo_Pagamento, Tipo, ID_Borsellino)

- ID_Borsellino è una chiave esterna che referencia ID_Borsellino nella tabella Borsellino.

Ristorante (Cod_Ristorante, Indirizzo, Nome, Costo_Spedizione, Immagine_Profilo, Descrizione, Categoria, Stelle, Top_Partner)

Portata (Cod_Portata, Nome, Foto, Prezzo, Lista_Ingredienti, Lista_Allergeni, Lista_Categoria, Sconto,

Ordine (Id_Ordine, Mancia, Stato, Recensione_Ristorante, Recensione_Rider, Ora_Consegna, Ora_Ritiro, Numero_Telefono, ID_Borsellino)

- Numero_Telefono referencia Utente
- ID_Borsellino referencia Borsellino

Rider (Cod_Rider, Stato, Posizione, Stato_Batteria, Mezzo_Spostamento, Id_Ordine)

Ricarica (Numero_Telefono, ID_Mezzo_Pagamento, Importo, Data_Ricarica)

- ID_Mezzo_Pagamento è una chiave esterna che referencia ID_Mezzo_Pagamento nella tabella Mezzo_di_Pagamento.
- Numero_Telefono è una chiave esterna che referencia Utente

Visualizzazione (Ristorante, Utente)

effettuato una ridenominazione per cui:

- il dominio dell'attributo Ristorante è un insieme di codici di ristoranti di Ristorante
 - il dominio dell'attributo Utente è un insieme di numeri_telefono
- inoltre esistono i seguenti vincoli di integrità referenziale:
- con l'attributo Cod_Ristorante della relazione Ristorante
 - con l'attributo Numero_Telefono della relazione Utente

Fornire (Ristorante, Portata)

effettuato una ridenominazione per cui:

- il dominio dell'attributo Ristorante è un insieme di codici di ristoranti di Ristorante
 - il dominio dell'attributo Portata è un insieme di Cod_Portata di Portata
- inoltre esistono i seguenti vincoli di integrità referenziale:
- con l'attributo Cod_Ristorante della relazione Ristorante
 - con l'attributo Cod_Portata della relazione Portata

Composizione (Portata, Ordine)

effettuato una ridenominazione per cui:

- il dominio dell'attributo Portata è un insieme di Cod_Portata di Portata
 - il dominio dell'attributo Ordine è un insieme di ID_Ordine di Ordine
- inoltre esistono i seguenti vincoli di integrità referenziale:
- con l'attributo Cod_Ristorante della relazione Ristorante
 - con l'attributo Cod_Portata della relazione Portata

Implementazione

3.1 DDL (Data Definition Language)

BORSELLINO

```
CREATE TABLE Borsellino (  
    ID_Borsellino INT PRIMARY KEY,  
    Saldo DECIMAL(10, 2) NOT NULL  
);
```

RISTORANTE

```
CREATE TABLE Ristorante (  
    Cod_Ristorante INT PRIMARY KEY,  
    Indirizzo VARCHAR(200) NOT NULL,  
    Nome VARCHAR(100) NOT NULL,  
    Costo_Spedizione DECIMAL(10, 2) NOT NULL,  
    Immagine_Profilo BLOB,  
    Descrizione TEXT,  
    Categoria VARCHAR(100),  
    Stelle DECIMAL(2, 1),  
    Top_Partner BOOLEAN NOT NULL  
);
```

RIDER

```
CREATE TABLE Rider (  
    Cod_Rider INT PRIMARY KEY,  
    Stato VARCHAR(50) NOT NULL,  
    Posizione VARCHAR(100),  
    Stato_Batteria DECIMAL(3, 1),  
    Mezzo_Spostamento VARCHAR(50),  
    Id_Ordine INT  
);
```

UTENTE

```
CREATE TABLE Utente (  
    Numero_Telefono VARCHAR(20) PRIMARY KEY,  
    Indirizzo_Email VARCHAR(100) NOT NULL,  
    Password VARCHAR(100) NOT NULL,  
    Nome_U VARCHAR(50) NOT NULL,  
    Indirizzo_Recapito VARCHAR(200) NOT NULL,  
    Premium BOOLEAN NOT NULL,  
    Cod_Sconto VARCHAR(50),  
    Storico_Ordini TEXT,  
    ID_Borsellino INT,  
    Chat_U_Rist INT,  
    Chat_U_Rider INT,  
    FOREIGN KEY (ID_Borsellino) REFERENCES Borsellino(ID_Borsellino),  
    FOREIGN KEY (Chat_U_Rist) REFERENCES Ristorante(Cod_Ristorante),  
    FOREIGN KEY (Chat_U_Rider) REFERENCES Rider(Cod_Rider)  
);
```

MEZZO DI PAGAMENTO

```
CREATE TABLE Mezzo_di_Pagamento (  
  ID_Mezzo_Pagamento INT PRIMARY KEY,  
  Tipo VARCHAR(50) NOT NULL,  
  ID_Borsellino INT,  
  FOREIGN KEY (ID_Borsellino) REFERENCES Borsellino(ID_Borsellino)  
);
```

PORTATA

```
CREATE TABLE Portata (  
  Cod_Portata INT PRIMARY KEY,  
  Nome VARCHAR(100) NOT NULL,  
  Foto BLOB,  
  Prezzo DECIMAL(10, 2) NOT NULL,  
  Lista_Ingredienti TEXT,  
  Lista_Allergeni TEXT,  
  Lista_Categoria TEXT,  
  Sconto DECIMAL(3, 2)  
);
```

ORDINE

```
CREATE TABLE Ordine (  
  Id_Ordine INT PRIMARY KEY,  
  Mancia DECIMAL(10, 2),  
  Stato VARCHAR(50) NOT NULL,  
  Recensione_Ristorante TEXT,  
  Recensione_Rider TEXT,  
  Ora_Consegna TIMESTAMP,  
  Ora_Ritiro TIMESTAMP,  
  Numero_Telefono VARCHAR(20),  
  ID_Borsellino INT,  
  FOREIGN KEY (Numero_Telefono) REFERENCES Utente(Numero_Telefono),  
  FOREIGN KEY (ID_Borsellino) REFERENCES Borsellino(ID_Borsellino)  
);
```

RICARICA

```
CREATE TABLE Ricarica (  
  Numero_Telefono VARCHAR(20),  
  ID_Mezzo_Pagamento INT,  
  Importo DECIMAL(10, 2) NOT NULL,  
  Data_Ricarica DATE NOT NULL,  
  PRIMARY KEY (Numero_Telefono, ID_Mezzo_Pagamento),  
  FOREIGN KEY (Numero_Telefono) REFERENCES Utente(Numero_Telefono),  
  FOREIGN KEY (ID_Mezzo_Pagamento) REFERENCES Mezzo_di_Pagamento(ID_Mezzo_Pagamento)  
);
```

VISUALIZZAZIONE

```
CREATE TABLE Visualizzazione (  
  Cod_Ristorante INT,  
  Numero_Telefono VARCHAR(20),  
  FOREIGN KEY (Cod_Ristorante) REFERENCES Ristorante(Cod_Ristorante),  
  FOREIGN KEY (Numero_Telefono) REFERENCES Utente(Numero_Telefono)
```

);

FORNIRE

```
CREATE TABLE Fornire (  
    Cod_Ristorante INT,  
    Cod_Portata INT,  
    FOREIGN KEY (Cod_Ristorante) REFERENCES Ristorante(Cod_Ristorante),  
    FOREIGN KEY (Cod_Portata) REFERENCES Portata(Cod_Portata)  
);
```

COMPOSIZIONE

```
CREATE TABLE Composizione (  
    Cod_Portata INT,  
    Id_Ordine INT,  
    FOREIGN KEY (Cod_Portata) REFERENCES Portata(Cod_Portata),  
    FOREIGN KEY (Id_Ordine) REFERENCES Ordine(Id_Ordine)  
);
```

3.2.1 DML (Data Manipulation Language)

inserimento dei dati nel database

```
INSERT INTO Borsellino (ID_Borsellino, Saldo)
VALUES (1, 100.50);
```

```
INSERT INTO Ristorante (Cod_Ristorante, Indirizzo, Nome, Costo_Spedizione, Immagine_Profilo,
Descrizione, Categoria, Stelle, Top_Partner)
VALUES (101, 'Via Roma, 10', 'Ristorante di Esempio', 5.00, 'immagine.jpg', 'Descrizione del ristorante',
'Italiana', 4.5, TRUE);
```

```
INSERT INTO Rider (Cod_Rider, Stato, Posizione, Stato_Batteria, Mezzo_Spostamento, Id_Ordine)
VALUES (201, 'Disponibile', 'Via Milano, 20', 85.0, 'Bicicletta', NULL);
```

```
INSERT INTO Utente (Numero_Telefono, Indirizzo_Email, Password, Nome_U, Indirizzo_Recapito,
Premium, Cod_Sconto, Storico_Ordini, ID_Borsellino, Chat_U_Rist, Chat_U_Rider)
VALUES ('1234567890', 'email@example.com', 'password123', 'Nome Utente', 'Via di esempio, 1', TRUE,
'SCONTO10', 'Ordini passati', 1, 101, 201);
```

```
INSERT INTO Mezzo_di_Pagamento (ID_Mezzo_Pagamento, Tipo, ID_Borsellino)
VALUES (301, 'Carta di Credito', 1);
```

```
INSERT INTO Portata (Cod_Portata, Nome, Prezzo, Lista_Ingredienti, Lista_Allergeni, Lista_Categoria,
Sconto)
VALUES (501, 'Pasta al Pomodoro', 8.50, 'Pasta, Pomodoro, Basilico', 'Glutine', 'Primi Piatti', 0.10);
```

```
INSERT INTO Ordine (Id_Ordine, Mancias, Stato, Recensione_Ristorante, Recensione_Rider,
Ora_Consegna, Ora_Ritiro, Numero_Telefono, ID_Borsellino)
VALUES (401, 2.00, 'Consegnato', 'Ottimo', 'Puntuale', '2023-08-11 13:30:00', '2023-08-11 13:00:00',
'1234567890', 1);
```

```
INSERT INTO Ricarica (Numero_Telefono, ID_Mezzo_Pagamento, Importo, Data_Ricarica)
VALUES ('1234567890', 301, 50.00, '2023-08-11');
```

```
INSERT INTO Visualizzazione (Cod_Ristorante, Numero_Telefono)
VALUES (101, '1234567890');
```

```
INSERT INTO Fornire (Cod_Ristorante, Cod_Portata)
VALUES (101, 501);
```

```
INSERT INTO Composizione (Cod_Portata, Id_Ordine)
VALUES (501, 401);
```

3.2.2 DML (Data Manipulation Language)

Modifica o eliminazione dei dati presenti

MODIFICA

-- Modifica dell'indirizzo email di un utente registrato

```
UPDATE Utente  
SET Indirizzo_Email = 'nuovoindirizzo@example.com'  
WHERE Numero_Telefono = '1234567890';
```

-- Modifica del saldo nel borsellino di un utente

```
UPDATE Borsellino  
SET Saldo = 200.00  
WHERE ID_Borsellino = 1;
```

-- Modifica del tipo di mezzo di pagamento per un utente

```
UPDATE Mezzo_di_Pagamento  
SET Tipo = 'paypal'  
WHERE ID_Mezzo_Pagamento = 1;
```

-- Modifica dell'indirizzo di recapito di un utente

```
UPDATE Utente  
SET Indirizzo_Recapito = 'Via Nuova 15, Torino'  
WHERE Numero_Telefono = '1234567890';
```

-- Modifica del nome di un ristorante

```
UPDATE Ristorante  
SET Nome = 'Nuova Pizzeria Bella'  
WHERE Cod_Ristorante = 1;
```

-- Modifica del costo di spedizione di un ristorante

```
UPDATE Ristorante  
SET Costo_Spedizione = 3.00  
WHERE Cod_Ristorante = 1;
```

-- Modifica dello stato di un rider

```
UPDATE Rider  
SET Stato = 'occupato'  
WHERE Cod_Rider = 1;
```

-- Modifica del prezzo di una portata

```
UPDATE Portata  
SET Prezzo = 7.00  
WHERE Cod_Portata = 1;
```

-- Modifica della mancia di un ordine

```
UPDATE Ordine  
SET Mancia = 2.00  
WHERE ID_Ordine = 1;
```


-- Modifica della recensione del ristorante in un ordine

UPDATE Ordine

SET Recensione_Ristorante = 'Servizio eccellente!'

WHERE ID_Ordine = 1;

-- Modifica della posizione di un rider

UPDATE Rider

SET Posizione = '45.4654, 9.1850'

WHERE Cod_Rider = 1;

-- Modifica dello stato della batteria di un rider

UPDATE Rider

SET Stato_Batteria = '80%'

WHERE Cod_Rider = 1;

-- Modifica dell'importo di una ricarica

UPDATE Ricarica

SET Importo = 60.00

WHERE Numero_Telefono = '1234567890' AND ID_Mezzo_Pagamento = 1 AND Data_Ricarica = '2024-08-13 10:00:00';

CANCELLAZIONE

-- Cancellazione di un utente registrato

DELETE FROM Utente

WHERE Numero_Telefono = '1234567890';

-- Cancellazione di un borsellino

DELETE FROM Borsellino

WHERE ID_Borsellino = 1;

-- Cancellazione di un mezzo di pagamento

DELETE FROM Mezzo_di_Pagamento

WHERE ID_Mezzo_Pagamento = 1;

-- Cancellazione di un ristorante

DELETE FROM Ristorante

WHERE Cod_Ristorante = 1;

-- Cancellazione di una portata

DELETE FROM Portata

WHERE Cod_Portata = 1;

-- Cancellazione di un ordine

DELETE FROM Ordine

WHERE ID_Ordine = 1;

-- Cancellazione di un rider

DELETE FROM Rider

WHERE Cod_Rider = 1;

-- Cancellazione di una ricarica

DELETE FROM Ricarica

WHERE Numero_Telefono = '1234567890' AND ID_Mezzo_Pagamento = 1 AND Data_Ricarica = '2024-08-13 10:00:00';

-- Cancellazione di una visualizzazione

DELETE FROM Visualizzazione

WHERE Ristorante = 1 AND Utente = '1234567890';

-- Cancellazione di una relazione fornire

DELETE FROM Fornire

WHERE Ristorante = 1 AND Portata = 1;

-- Cancellazione di una relazione composizione

DELETE FROM Composizione

WHERE Portata = 1 AND Ordine = 1;