

Progetto Laboratorio Basi di Dati

Autore : Georgiev David mat : 1043306

1) Progettazione Concettuale

1.1 Requisiti Iniziali

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 registratosi, 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 stellette 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, ...).

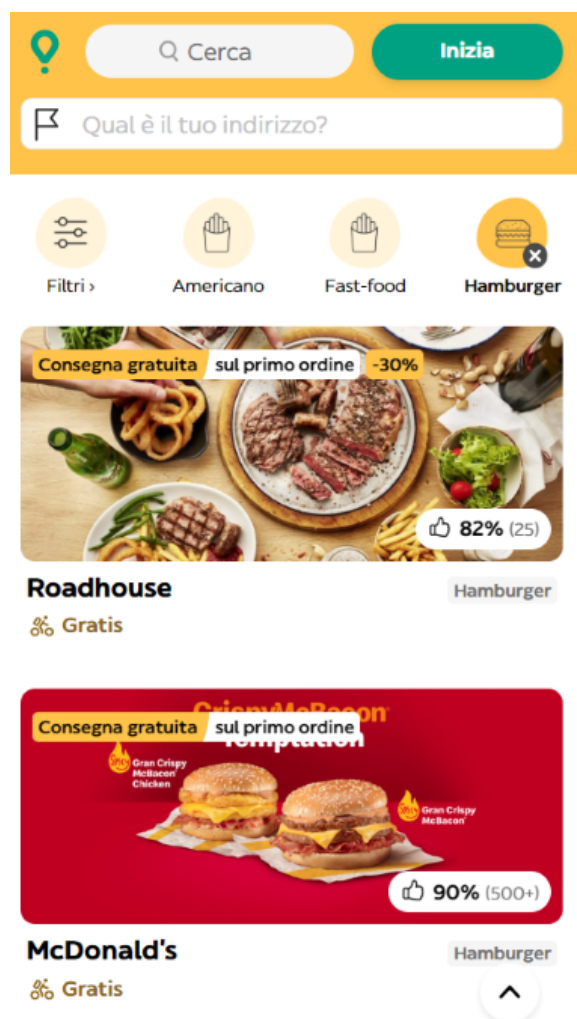
I ristoranti che dimostrano di saper garantire un ottimo servizio (almeno 20 ordini consegnati correttamente, una valutazione clienti maggiore o uguale a 4.5 stelline su cinque, una percentuale massima di ordini annullati dal ristorante dell'1.5%, una percentuale massima di ordini con reclami del 2.5%)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



1(a) Lista dei ristoranti con filtro Hamburger



1(b) I dettagli di un 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:

- 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

Termine	Descrizione	Sinonimi	Collegamenti
Utente	Colui iscritto all'app per usufruire dei servizi di	Cliente	Ordine, Mezzo di pagamento,

Termine	Descrizione	Sinonimi	Collegamenti
	consegna di cibo a domicilio. Può essere un utente premium.		Recensione
Borsellino elettronico	Saldo interno all'app che può essere ricaricato prelevando dal mezzo di pagamento scelto dal utente.		Utente
Ristorante	Il soggetto che offre la propria lista di piatti agli utenti. Categorizzati in base al tipo di piatti offerti. Può essere un Top Partner.	Ristoratore	Piatto, Recensione, Ordine
Rider	Soggetto intermediario incaricato di trasferire un ordine da un ristorante ad un cliente. Classificati in base al loro mezzo.	ciclofattorino, fattorino	Mezzo, Recensione, Consegna
Ordine	Richiesta di uno o più piatti da parte di un utente. Può essere annullato se non è stato affidato ad un rider.	Consegna, Ordinazione	Ristorante , Utente, Piatto
Piatto	Il piatto dei ristoranti soggetto agli ordini da parte degli utenti e ciò che consegnano i rider. Classificato in base al mezzo che utilizza per fare le consegne.	Pietanza, Portata, Cibo	Ristorante
Recensione	La recensione espressa dal utente verso il ristorante e il rider che ha effettuato la consegna dell'ordine. Accompagnata da un commento facoltativo		Ristorante, Utente, Rider, Ordine

Termine	Descrizione	Sinonimi	Collegamenti
Chat	Mezzo di comunicazione tra Rider e Utente oppure tra Utente e Ristorante		Utente, Rider, Ristorante, Ordine
Codice Sconto	Codice rilasciato dall'app per poter avere degli sconti sul prezzo totale del ordine effettuato da un utente		Utente

1.3 Requisiti Rivisti

a) Riscrittura dei requisiti

Si deve progettare la base di dati per Cibora (Figura 1(a)), un servizio di food delivery per gestire i dati dei ristoranti aderenti, degli utenti con i loro relativi ordini e dei **rider** che **consegnano gli ordini con il loro rispettivo mezzo**.

Per beneficiare del servizio ogni utente deve registrarsi e inserire un mezzo di pagamento (es.: carta di credito, paypal, satispay) . **Per ogni utente registrato viene rappresentato** il nome, la email, una password, il numero di telefono e l'indirizzo di recapito.

Ad ogni utente registrato viene assegnato un borsellino elettronico di cui **viene rappresentato** un saldo che viene aggiornato ad ogni **ordine**, l'utente può ricaricare il proprio borsellino in qualsiasi momento.

Gli utenti possono sottoscrivere la modalità premium, che garantisce una priorità sugli ordini. **Gli utenti possono ricevere** dei codici sconto da utilizzare al momento dell'ordine, il numero di codici sconto ricevuti **è direttamente proporzionale** al numero **totale** di ordini effettuati.

Per ogni ristorante (Figura 1(b)) viene rappresentato un nome, una descrizione, un indirizzo, il costo della spedizione, un'immagine di profilo e un numero di **stelle, fino ad un massimo di 5**. Il numero di **stelle** è aggiornato ogni lunedì **in modo direttamente proporzionale** al **numero** di recensioni positive ricevute nell'ultima settimana. Ogni ristorante appartiene a una o più categorie in base al tipo di **piatti offerti** (ad esempio: fast food, vegetariano, ...).

I ristoranti che dimostrano di saper garantire un ottimo servizio (almeno 20 ordini consegnati correttamente, una valutazione **utenti** 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 degli **utenti**. Per i Top Partner **viene rappresentata** la data in cui sono entrati a far parte della categoria.

I ristoranti propongono agli utenti una lista di piatti da ordinare. **Per ogni piatto viene rappresentato** 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 **piatti** ed effettuare l'ordine. Finché non sono affidati ad un rider per la consegna, gli ordini possono essere annullati sia dagli **utenti**, sia dai **ristoranti**. 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 e per ogni rider **viene rappresentato** il suo stato (occupato/disponibile/fuori servizio) e la sua posizione aggiornata in tempo reale tramite GPS. I riders sono classificati in base al tipo di mezzo che utilizzano (bicicletta normale, bicicletta elettrica, monopattino). Per riders che utilizzano il monopattino **viene rappresentata** la quantità di km che possono effettuare prima che si scarichi la batteria.

Quando un utente effettua un ordine, il sistema individua il rider libero che **può raggiungere il ristorante e poi la posizione del utente percorrendo la distanza totale minima**. Tuttavia, per ordini che prevedano un tragitto "*posizione corrente del rider* → ristorante → **Utente**" superiore ai 10 km, solo i rider con bici elettrica vengono **considerati dal sistema**. Per monitorare le prestazioni dei **rider viene rappresentato** il numero di **ordini consegnati** da ognuno, il momento in cui il **piatto** da consegnare viene affidato ad un rider e, per gli **ordini già consegnati**, anche l'ora in cui l'ordine è stato recapitato al utente.

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, il rider o entrambi (in due diverse recensioni)** 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 **alla consegna dell'ordine.**

Una volta al mese, vengono aggiornate le seguenti classifiche:

- Riders più veloci nel consegnare gli ordini
- **Piatti** più popolari
- Ristoranti con più recensioni positive
- **Utenti** che hanno speso di più

b) Raggruppamento dei requisiti in frasi omogenee

Frasi di carattere generale

Si deve progettare la base di dati per Cibora (Figura 1(a)), un servizio di food delivery per gestire i dati dei ristoranti aderenti, degli utenti con i loro relativi ordini e dei rider che consegnano gli ordini con il loro rispettivo mezzo.

Una volta al mese, vengono aggiornate le seguenti classifiche:

- Riders più veloci nel consegnare gli ordini
- Piatti più popolari
- Ristoranti con più recensioni positive
- Utenti che hanno speso di più

Frasi relative agli utenti

Per beneficiare del servizio ogni utente deve registrarsi e inserire un mezzo di pagamento (es.: carta di credito, paypal, satispay) . Per ogni utente registrato viene rappresentato il nome, la email, una password, il numero di telefono e l'indirizzo di recapito.

Gli utenti possono sottoscrivere la modalità premium, che garantisce una priorità sugli ordini. Gli utenti possono ricevere dei codici sconto da utilizzare al momento dell'ordine, il numero di codici sconto ricevuti è direttamente proporzionale al numero totale di ordini effettuati.

Frasi relative al borsellino elettronico

Ad ogni utente registrato viene assegnato un borsellino elettronico di cui viene rappresentato un saldo che viene aggiornato ad ogni ordine, l'utente può ricaricare il proprio borsellino in qualsiasi momento.

Frazi relative ai ristoranti

Per ogni ristorante (Figura 1(b)) viene rappresentato un nome, una descrizione, un indirizzo, il costo della spedizione, un'immagine di profilo e un numero di stelline, fino ad un massimo di 5. Il numero di stelline è aggiornato ogni lunedì in modo direttamente proporzionale al numero di recensioni positive dell'ultima settimana. Ogni ristorante appartiene a una o più categorie in base al tipo di piatti offerti (ad esempio: fast food, vegetariano, ...).

I ristoranti che dimostrano di saper garantire un ottimo servizio (almeno 20 ordini consegnati correttamente, una valutazione utenti 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 ristoranti propongono agli utenti una lista di piatti da ordinare.

Frazi relative ai tipi specifici di ristoranti

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 degli utenti. Per i Top Partner viene rappresentata la data in cui sono entrati a far parte della categoria.

Frazi relative ai rider

Il sistema gestisce un numero arbitrario di riders dove ogni rider è identificato da un codice e per ogni rider viene rappresentato il suo stato (occupato/disponibile/fuori servizio) e la sua posizione aggiornata in tempo reale tramite GPS. I riders sono classificati in base al tipo di mezzo che utilizzano (bicicletta normale, bicicletta elettrica, monopattino).

Quando un utente effettua un ordine, il sistema individua il rider libero che può raggiungere il ristorante e poi la posizione del utente percorrendo la distanza totale minima.

Per monitorare le prestazioni dei rider viene rappresentato il numero di ordini consegnati da ognuno, il momento in cui il piatto da consegnare viene affidato ad un rider e, per gli ordini già consegnati, anche l'ora in cui l'ordine è stato recapitato al utente.

Inoltre è anche presente la possibilità di dare una mancia al rider alla consegna dell'ordine.

Fraasi relative agli ordini

Finché non sono affidati ad un rider per la consegna, gli ordini possono essere annullati sia dagli utenti, sia dai ristoranti. Nel profilo dell'utente si possono ispezionare gli ordini passati ed eventualmente effettuare dei reclami inviando un messaggio al ristorante.

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.

Fraasi relative ai piatti

Per ogni piatto viene rappresentato 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 piatti ed effettuare l'ordine.

Fraasi relative ai tipi specifici di rider

Per riders che utilizzano il monopattino viene rappresentata la quantità di km che possono effettuare prima che si scarichi la batteria.

Per ordini che prevedano un tragitto "*posizione corrente del rider → ristorante → Utente*" superiore ai 10 km, solo i rider con bici elettrica vengono considerati dal sistema.

Fraasi relative alle recensioni

Quando l'ordine è consegnato l'utente può recensire il ristorante, il rider o entrambi (in due diverse recensioni). La recensione viene rappresentata da una valutazione da 1 a 5 e da un commento testuale. Il commento testuale è facoltativo.

1.4 Schema E-R

- **Annullamento** può essere vero se e soltanto se ordine non partecipa all'associazione **Trasformazione**
- Solo i **Rider** che fanno parte della generalizzazione **Bici Elettrica** possono partecipare all'associazione **Affidamento** nel caso in cui il tragitto è maggiore di 10 km.
- I **rider** che fanno parte della generalizzazione **Monopattino** non possono consegnare **ordini** con tragitto maggiore di **Km. Disponibili**.
- I ristoranti che fanno parte della generalizzazione **Top partner**:
partecipano più di 20 volte all'associazione **Riceve** e questi **Ordini** devono partecipare a loro volta all'associazione **Consegna**
hanno **stelle** ≥ 4.5
meno del 1.5 % delle partecipazioni a **Riceve** possono avere un **Ordine** con attributo **Annullamento corrispondente** ad un valore che conferma l'annullamento dell'**Ordine**
meno del 2.5 % delle partecipazioni a **Riceve** devono avere un **Ordine** che partecipa all'associazione **Riguardo**

2) Progettazione Logica

2.1) Tavola dei volumi

Concetto	Tipo	Volume
Utente	E	$15 * 10^6$
Ristorante	E	100.000
Ordine	E	$115 * 10^6$
Rider	E	200.000
Bici Elettrica	E	67.000
Bici	E	67.000
Monopattino	E	66.000
Chat Rider	E	$34 * 10^6$

Concetto	Tipo	Volume
Messaggio Rider	E	136 * 10 ⁶
Chat Ristorante	E	38 * 10 ⁶
Messaggio ristorante	E	152 * 10 ⁶
Recensione	E	30 * 10 ⁶
Recensione Ristorante	E	10 * 10 ⁶
Recensione Rider	E	20 * 10 ⁶
Categoria	E	35
Piatto	E	4 * 10 ⁶
Lista	E	30
Ingredienti	E	380.000
Top Partner	E	5000
Reclamo	E	5 * 10 ⁶
Consegna	E	103 * 10 ⁶
Codice Sconto	E	45 * 10 ⁶
Borsellino	E	15 * 10 ⁶
Consegna Completata	E	100 * 10 ⁶
Collezione	A	45 * 10 ⁶
Intestazione	A	15 * 10 ⁶
Ordina	A	115 * 10 ⁶
Riceve	A	115 * 10 ⁶
Trasformazione	A	103 * 10 ⁶
Riguardo	A	5 * 10 ⁶
ORD-CHAT	A	38 * 10 ⁶
Contenuto	A	300 * 10 ⁶
Invio ris	A	152 * 10 ⁶
Offerta	A	4 * 10 ⁶
Appartiene	A	8 * 10 ⁶
Composizione	A	24 * 10 ⁶
Categorizzazione	A	300.000
Voto	A	30 * 10 ⁶
Affidamento	A	103 * 10 ⁶

Concetto	Tipo	Volume
CHAT-CONS	A	$34 * 10^6$
Invio Rid	A	$136 * 10^6$
Rec-Rid	A	$20 * 10^6$
Rec-Ris	A	$10 * 10^6$

Motivazioni della scelta dei volumi

Analizzando alcune analisi statistiche pubbliche (si veda fine paragrafo) delle app di food delivery più conosciute nel mondo, tra cui "Just Eat", "Uber Eats", "DoorDash", "Deliveroo", "Glovo" e altre ancora, si possono dedurre alcune informazioni riguardo alla mole di dati che il database di una nuova app di food delivery possa gestire. In particolar modo si è notato come il numero di utenti delle maggiori app si aggiri sui milioni di utenti registrati e come il numero di ristoranti registrati si aggiri dai 100.000 ad 1 miliardo in totale. Da queste analisi, inoltre, si può notare come il rapporto tra numero dei riders cresca con l'aumentare del numero di ristoranti iscritti all'app, quindi si è adottato un rapporto 1 a 2 rispetto al numero dei ristoranti per determinare il numero di riders. Siccome le statistiche includono altri mezzi di trasporto, è stato deciso di suddividere equamente i rider nelle varie categorie di mezzi utilizzati da loro.

Per determinare il numero di reclami è stata considerata la percentuale di chargeback presente nel food delivery e questa corrisponde approssimativamente al 5% degli ordini totali mentre il totale degli ordini non annullati corrisponde al 97%.

In mancanza di dati precisi riguardanti le chat tra utente e rider e tra ristorante e utente si è assunto che 1/3 degli ordini venga discussa via chat e, considerando che queste chat sono molto corte solitamente, si è assunto che ogni chat contenga in media 4 messaggi.

Secondo la "Food and Agriculture Organization of the United Nations" ci sono all'incirca 380.000 ingredienti edibili nel mondo e gli allergeni alimentari previsti dai regolamenti Europei sono 14 in tutto. Si è assunto che un piatto contiene, in media, 6 ingredienti. Si è deciso anche che ogni piatto fa parte mediamente di 2 liste diverse.

Considerando delle macro categorie si è notato come i tipi di ristoranti riconosciuti si aggirano sulle 35 categorie; sempre con la stessa logica si è

potuto determinare il numero approssimativo di liste possibili in cui un piatto può ricadere. Per ogni ristorante si è assunto che esso faccia parte mediamente di 3 diverse categorie.

Le statistiche dei ristoranti e rider variano molto nelle diverse fonti quindi si è deciso di determinare una media di 100 recensioni per ristorante e rider.

Infine rispettando una media di 40 piatti a ristorante si è determinato il numero di piatti totali contenuti nel database e, per quanto riguarda i codici sconto, si è considerato una media di 3 codice per utente.

I ristoranti Top partner sono una stretta minoranza quindi si è considerato solo il 5% dei ristoranti in totale.

Fonti:

- <https://www.businessofapps.com/data/food-delivery-app-market/>
- <https://usesignhouse.com/blog/food-delivery-apps-stats/>
- <https://www.foodscience.it/elenco-allergeni-regolamento-1169-2011/>
- <https://www.restaurantbusinessonline.com/technology/order-cancellations-cost-typical-restaurant-3-digital-sales-new-data-show>
- <https://get.doordash.com/en-us/blog/food-delivery-statistics>

2.2) Tavola delle operazioni

Operazione	Descrizione	Tipo	Frequenza
1	Registrazione di un nuovo Utente	I	500 al giorno
2	Trova tutti i dati di un ristorante, i piatti che offre e le recensioni che lo riguardano	I	20.000 al giorno
3	Trova i ristoranti di una certa categoria o che offrono piatti appartenenti ad una lista particolare	I	20.000 al giorno
4	Trova tutti gli ordini in via di esecuzione di un Utente	I	9000 al giorno

Operazione	Descrizione	Tipo	Frequenza
5	Trova il Rider che ha tragitto più corto per raggiungere l'utente	B	9500 al giorno
6	Aggiornamento delle classifiche	B	1 al mese
7	Aggiornamento del numero di stelle di un ristorante	B	1 a settimana
8	Visualizzazione della posizione di un rider	I	5000 al giorno

Motivazione

Il numero di ordini completati di DoorDash nel 2023 è stato di 2,16 miliardi, ovvero quasi 6 milioni di ordini al giorno. Tenendo presente che DoorDash è stata fondata nel 2012, si è scelto di ridimensionare il numero di ordini giornalieri a solamente 10.000 al giorno.

Si è assunto che ogni utente, prima di effettuare un ordine, visiti almeno le pagine di 2 ristoranti diverse e controlli recensioni e i piatti offerti. Inoltre si è assunto che un utente applichi almeno due filtri per cercare il ristorante che offre la categoria di piatti desiderata.

Un'altra assunzione fatta è che il 90% degli utenti che hanno ordinato verifichi lo stato del proprio ordini.

Per coerenza alla tavola dei Volumi si è mantenuta la percentuale di ordini annullati quindi solo il 97% degli ordini arrivano ad essere assegnati al rider e di quelli sono una bassa percentuale di utenti controlla i dati dei rider. Inoltre si è considerato che metà degli utenti che hanno ordinato controllino la posizione del rider per avere una stima approssimativa del suo tempo d'arrivo.

Fonte: <https://www.statista.com/statistics/1374238/door-dash-total-order-volume/>

2.3) Ristrutturazione dello schema E-R

2.3.1) Analisi delle ridondanze

Ridondanze rilevate

1. L'attributo **Stelle** di **Ristorante** si può calcolare facendo la media aritmetica del attributo **Valutazione** in **Recensione**
2. L'attributo **Num. Consegne** del **Rider** si può calcolare dal conteggio del numero di partecipazioni di **Rider** all'associazione **Affidamento**.

Analisi Ridondanza 1

Operazioni che riguardano la ridondanza:

- Aggiornare il numero di stelle (n. 7)
- stampa dei dati di un ristorante (n. 2)

Scenari:

Assenza di ridondanza:

Tavola accessi operazione n. 7

L'aggiornamento del numero di stelle settimanale non è necessario in quanto è assente l'attributo in cui viene memorizzato il risultato.

Tavola accessi operazione n. 2

Per ottenere il numero di stelle di un ristorante, oltre a leggere tutti i suoi attributi si deve anche calcolare il numero di stelle in base alle recensioni.

$r \rightarrow$ recensioni per un ristorante

$$r = \frac{10 * 10^6}{10^5} = 1000$$

Per coerenza rispetto ai requisiti iniziali (ovvero il calcolo delle stelle avviene in base alle recensioni dell'ultima settimana) si assume che le recensioni siano state scritte nell'arco di 2 anni ovvero 100 settimane allora si hanno:

$$r = \frac{1000}{100} = 10(\text{recensioni a settimana})$$

Concetto	Costrutto	Accessi	Tipo
Ristorante	Entità	1	L
Rec Ris	Associazione	10	L

Concetto	Costrutto	Accessi	Tipo
Recensione Ristorante	Entità	10	L

Calcolo dei costi:

Spazio: 0 (è assente la ridondanza)

Tempo:

Operazione 7: 0 (non si deve aggiornare il numero di stelle in quanto assente)

Operazione 2: $(1 + 10 + 10) * 20.000$ (accessi al giorno) = 420.000

Totale: $0 + 420.000 = 420.000$ accessi al giorno

Presenza di ridondanza:

Tavola accessi operazione n. 7:

Sempre sulla base dei calcoli precedentemente eseguiti:

Concetto	Costrutto	Accessi	Tipo
Ristorante	Entità	1	S
Rec Ris	Associazione	10	L
Recensione Ristorante	Entità	10	L

Tavola accessi dell'operazione n.2

Il numero di stelle di un ristorante non deve essere ricalcolato

Concetto	Costrutto	Accessi	Tipo
Ristorante	Entità	1	L

Calcolo dei costi:

Spazio: Siccome il numero di stelle è compreso da 0 a 5 e si assume sia approssimato fino alla prima cifra decimale allora si hanno 51 possibili valori diversi ovvero basta 1 byte ($2^8 = 256$) per rappresentare questa cifra.

1 byte * 100.000 ristoranti = 100.000 byte → circa 100 KB in totale

Tempo: (le operazioni in scrittura costano 2 volte quelli in lettura)

Operazione 1: Per la coerenza dei calcoli si ha che un giorno sia equivalente ad un settimo di una settimana

$$\frac{2 + 10 + 10}{7} \approx 3$$

Operazione 2: $1 * 20.000$ accessi al giorno = 20.000 accessi al giorno

Totale: $3 + 20.000 = 20.003$ accessi al giorno e uno spazio utilizzato pari a 100 KB

Scelta definitiva

Senza la ridondanza si hanno circa 400.000 accessi quotidiani in più rispetto al caso in cui la ridondanza esiste, il costo spaziale di questa ridondanza è relativamente basso vista la mole di dati da trattare e considerando anche la bassa frequenza di nuovi ristoranti che si registrano sulle piattaforme. Quindi tenere la ridondanza al costo approssimativo di 100kb sembra la scelta più ragionevole.

2.3.2) Eliminazione delle Generalizzazioni

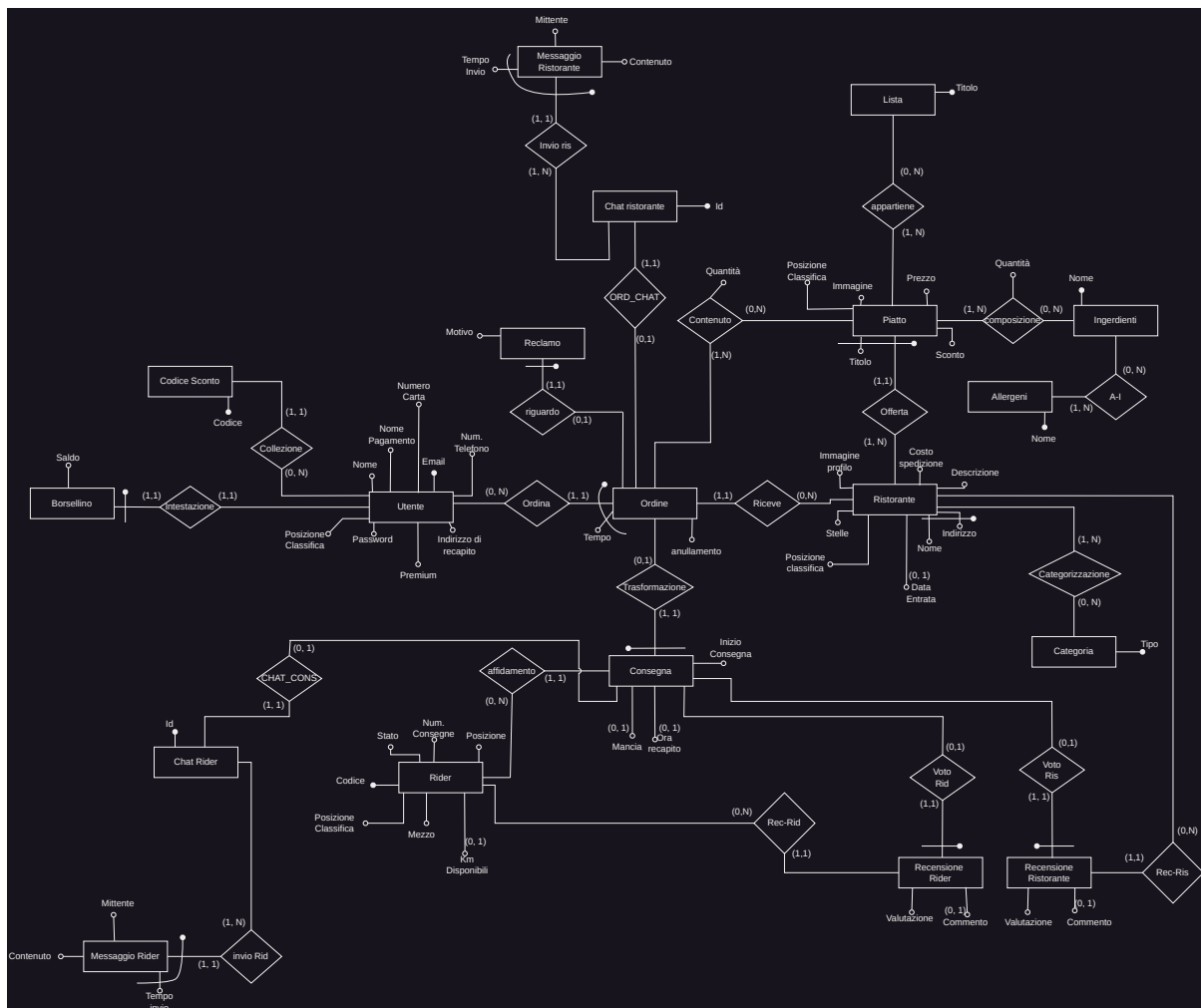
1. Accorpamento delle varie categorie di rider nell'entità padre, ovvero rider, siccome le operazioni non sono eccessivamente distinte tra le varie tipologie di rider ed esiste solo un attributo aggiuntivo per una sola entità figlia (quindi una sola regola aziendale per essa)
2. Accorpamento di Top Partner in Ristorante siccome le operazioni riguardanti Top Partner sono poco frequenti; in aggiunta siccome si tratta di una generalizzazione sotto-insiemistica basta una regola aziendale per regolare l'attributo Data Entrata. Il contro è lo spreco di memoria occupato dai valori nulli per quanto riguarda l'attributo Data Entrata.
3. Accorpamento di Consegna Completata in Consegna per non avere troppe ridondanze di dati nel database
4. Accorpamento di Recensione nelle due Entità figlie siccome le operazioni riguardano principalmente le due entità figlie e non è più necessario distinguere il soggetto della recensione

2.3.5) Scelta identificatori

Nelle entità Chat Ristorante e Chat rider sono stati introdotti degli identificatori per non avere tante ripetizioni di dati per ogni entità Messaggio, questa scelta è

stata fatta considerando che entrambe le chat sono sprovviste di attributi propri quindi ogni chat si riferisce univocamente ad un ordine.

2.4) Schema E-R ristrutturato



Business Rules

Vincoli di derivazione

- **Num. Consegne** è derivato dal conteggio con cui un **Rider** partecipa all'associazione **Consegna**
- **Stelle** è derivato dalla media aritmetica di tutte le **Valutazioni** delle **Recensioni Ristoranti** a cui il **Ristorante** è associato tramite l'associazione **Rec-Res**

Vincoli di integrità

(Per tragitto si intende la distanza da rider e ristorante sommata alla distanza da ristorante a utente ovvero: $| \text{posizione} - \text{indirizzo} | + | \text{indirizzo} - \text{indirizzo di recapito} |$ [km])

- **Inizio Consegna** < **Ora recapito**
- **Annullamento** può essere vero se e soltanto se ordine non partecipa all'associazione **Trasformazione**
- Solo i **Rider** che hanno attributo **Mezzo** uguale a **Bici Elettrica** possono partecipare all'associazione **Affidamento** nel caso in cui il tragitto è maggiore di 10 km.
- I **rider** che hanno l'attributo **Mezzo** equivalente a Monopattino non possono consegnare **ordini** con tragitto maggiore di **Km. Disponibili**.
- Ristorante ha **Data entrata** valorizzato solo se è un Top partner ovvero:
partecipa più di 20 volte all'associazione **Riceve** e questi **Ordini** devono partecipare a loro volta all'associazione **Consegna**
ha **stelle** ≥ 4.5
meno del 1.5 % delle sue partecipazioni a **Riceve** possono avere un **Ordine** con attributo **Annullamento corrispondente** ad un valore che conferma l'annullamento dell'**Ordine**
meno del 2.5 % delle partecipazioni a **Riceve** devono avere un **Ordine** che partecipa all'associazione **Riguardo**
- **Mancia** è valorizzato se e solo se **Ora recapito** è valorizzato
- **km disponibili** è valorizzato solo se l'attributo **Mezzo** ha valore equivalente a **Monopattino**
- **Consegna** può partecipare all'associazione **Recensione Rider** solo se ha attributo **ora recapito** valorizzato
- **Consegna** può partecipare all'associazione **Recensione Ristorante** solo se ha attributo **ora recapito** valorizzato

2.5) Schema Relazionale

Utente(Email, Nome, Num. Telefono, Password, Premium, Indirizzo di Recapito, Nome Pagamento, Numero Carta)

Borsellino(Utente mail, Saldo)

Codice Sconto(Codice, Utente Mail)

Ordine(Utente Mail, Tempo, Annullamento, Ristorante Nome, Ristorante Indirizzo)

Reclamo(Tempo Ordine, Utente, Motivo)

Contenuto(Tempo Ordine, Ordine Utente, Piatto Titolo, Indirizzo, Nome, Quantità)

Ristorante(Nome, Indirizzo, Costo Spedizione, Descrizione, Immagine Profilo, Stelle, Descrizione, Data Entrata*)

Categoria(Tipo)

Categorizzazione(Nome Ristorante, Indirizzo Ristorante, Tipo Categoria)

Chat Ristorante(Id, Ordine Utente, Tempo Ordine)

Messaggio Ristorante(Chat id, Tempo Invio, Contenuto, Mittente)

Piatto(Titolo, Nome Ristorante, Indirizzo Ristorante, Sconto, Prezzo, Immagine)

Composizione(Titolo Piatto, Ristorante, Luogo, Nome Ingrediente, Quantità)

Ingredienti(Nome)

Allergeni(Nome)

A-I(Nome Ingrediente, Nome Allergene)

Lista(Titolo)

Appartiene(Piatto Titolo, Ristorante, Luogo, Lista Titolo)

Consegna(Tempo Ordine, Ordine Utente, Cod Rider, Inizio Consegna, Ora Recapito*, Mancia*)

Rider(Codice, Stato, Num Consegne, Posizione, Mezzo, Km Disponibili*)

Chat Rider(Id, Inizio Ordine, Utente, Rider)

Messaggio Rider(Chat id, Tempo invio, Contenuto, Mittente)

Recensione Rider(Utente, Ordine, Valutazione, Commento*)

Recensione Ristorante(Utente, Ordine, Nome Ristorante, Indirizzo Ristorante, Valutazione, Commento*)

Vincoli

Borsellino(Utente Mail) referencia Utente(Mail)

Codice Sconto(Utente Mail) referencia Utente(Mail)

Ordine(Utente Mail) referencia Utente(Mail)

Reclamo(Tempo Ordine, Ordine Utente) referencia Ordine(Tempo, Utente Mail)

Contenuto(Tempo Ordine, Ordine Utente) referencia Ordine(Tempo, Utente Mail)

Contenuto(Piatto Titolo, Nome, Indirizzo) referencia Piatto(Titolo, Nome Ristorante, Indirizzo Ristorante)

Categorizzazione(Nome Ristorante, Indirizzo Ristorante) referencia Ristorante(Nome, Indirizzo)

Categorizzazione(Tipo Categoria) referencia Categoria(Tipo)

Chat Ristorante(Tempo Ordine, Ordine Utente) referencia Ordine(Tempo, Utente Mail)

Messaggio Ristorante(Chat id) referencia Chat Ristorante(id)

Piatto(Nome Ristorante, Indirizzo Ristorante) referencia Ristorante(Nome, Indirizzo)

Composizione(Nome Ingrediente) referencia Ingrediente(Nome)

Composizione(Titolo Piatto, Ristorante, Luogo) referencia Piatto(Titolo, Nome Ristorante, Indirizzo Ristorante)

A-I(Nome Ingrediente) referencia Ingrediente(Nome)

A-I(Nome Allergene) referencia Allergeni(Nome)

Appartiene(Lista Titolo) referencia Lista(Titolo)

Appartiene(Piatto Titolo, Ristorante, Luogo) referencia Piatto(Titolo, Nome Ristorante, Indirizzo Ristorante)

Consegna(Tempo Ordine, Ordine Utente) referencia Ordine(Tempo, Utente Mail)

Consegna(Cod Rider) referencia Rider(Codice)

Chat Rider(Inizio Ordine, Utente, Rider) referencia Consegna(Tempo Ordine, Ordine Utente, Cod Rider)

Messaggio Ristorante(Chat id) referencia Chat Rider(id)

Recensione Rider(Ordine, Utente) referencia Consegna(Tempo Ordine, Ordine Utente)

Recensione Ristorante(Ordine, Utente,) referencia Consegna(Tempo Ordine, Ordine Utente)

Recensione Ristorante(Nome Ristorante, Indirizzo Ristorante) referencia Ristorante(Nome, Indirizzo)

3) Implementazione

3.1) DDL di creazione del database

```
CREATE DOMAIN stelle_dom AS INT
CHECK (VALUE BETWEEN 0 AND 5);

CREATE DOMAIN mezzo_dom AS VARCHAR
CHECK (VALUE IN ('bici elettrica', 'bici', 'monopattino'));

CREATE DOMAIN stato_dom AS VARCHAR
CHECK (VALUE IN ('occupato', 'disponibile', 'fuori servizio'))

CREATE DOMAIN mittente_dom AS VARCHAR
CHECK (VALUE IN ('u', 'r'));

CREATE TABLE Utente (
    Email VARCHAR(255) PRIMARY KEY,
    Nome VARCHAR(255) NOT NULL,
    Num_Telefono VARCHAR(20) NOT NULL,
    Password VARCHAR(255) NOT NULL,
```

```

        Premium BOOLEAN NOT NULL,
        Indirizzo_Recapito VARCHAR(255) NOT NULL,
        Nome_Pagamento VARCHAR(255) NOT NULL,
        Numero_Carta VARCHAR(20) NOT NULL,
        Poz_classifica INT UNIQUE
    );

CREATE TABLE Borsellino (
    Utente_Mail VARCHAR(255) PRIMARY KEY,
    Saldo DECIMAL(10, 2) NOT NULL,
    FOREIGN KEY (Utente_Mail) REFERENCES Utente(Email)
);

CREATE TABLE Rider (
    Codice INT PRIMARY KEY,
    Stato stato_dom NOT NULL,
    Num_Consegne INT NOT NULL,
    Posizione VARCHAR(255) NOT NULL,
    Mezzo mezzo_dom NOT NULL,
    Km_Disponibili DECIMAL(10, 2) NOT NULL,
    Poz_classifica INT UNIQUE
);

CREATE TABLE Codice_Sconto (
    Codice VARCHAR(50) PRIMARY KEY,
    Utente_Mail VARCHAR(255) NOT NULL,
    FOREIGN KEY (Utente_Mail) REFERENCES Utente(Email)
);

CREATE TABLE Ordine (
    Utente_Mail VARCHAR(255),
    Tempo TIMESTAMP,
    Annullamento BOOLEAN NOT NULL,
    Ristorante_Nome VARCHAR(255) NOT NULL,
    Ristorante_Indirizzo VARCHAR(255) NOT NULL,
    PRIMARY KEY (Utente_Mail, Tempo),
    FOREIGN KEY (Utente_Mail) REFERENCES Utente(Email)
);

```



```

CREATE TABLE Reclamo (
    Tempo_Ordine TIMESTAMP,
    Utente VARCHAR(255),
    Motivo TEXT NOT NULL,
    PRIMARY KEY (Tempo_Ordine, Utente),
    FOREIGN KEY (Tempo_Ordine, Utente)
        REFERENCES Ordine(Tempo, Utente_Mail)
);

CREATE TABLE Ristorante (
    Nome VARCHAR(255),
    Indirizzo VARCHAR(255),
    Costo_Spedizione DECIMAL(10, 2) NOT NULL,
    Descrizione TEXT NOT NULL,
    Immagine_Profilo TEXT NOT NULL,
    Stelle stelle_dom NOT NULL,
    Data_Entrata DATE DEFAULT NULL,
    Poz_classifica INT UNIQUE,
    PRIMARY KEY (Nome, Indirizzo)
);

CREATE TABLE Piatto (
    Titolo VARCHAR(255),
    Nome_Ristorante VARCHAR(255),
    Indirizzo_Ristorante VARCHAR(255),
    Sconto DECIMAL(5, 2) NOT NULL,
    Prezzo DECIMAL(10, 2) NOT NULL,
    Immagine TEXT NOT NULL,
    Poz_classifica INT UNIQUE,
    PRIMARY KEY(Titolo, Nome_Ristorante, Indirizzo_Ristorante),
    FOREIGN KEY (Nome_Ristorante, Indirizzo_Ristorante)
        REFERENCES Ristorante(Nome, Indirizzo)
);

CREATE TABLE Contenuto (
    Tempo_Ordine TIMESTAMP,
    Ordine_Utente VARCHAR(255),

```

```

Piatto_Titolo VARCHAR(255),
Nome_Ristorante VARCHAR(255),
Indirizzo_Ristorante VARCHAR(255),
Quantità INT NOT NULL,
PRIMARY KEY (Tempo_Ordine, Ordine_Utente, Piatto_Titolo,
             Nome_Ristorante, Indirizzo_Ristorante),
FOREIGN KEY (Tempo_Ordine, Ordine_Utente)
             REFERENCES Ordine(Tempo, Utente_Mail),
FOREIGN KEY (Piatto_Titolo, Nome_Ristorante,
             Indirizzo_Ristorante)
             REFERENCES Piatto(Titolo, Nome_Ristorante,
                                Indirizzo_Ristorante)
);

CREATE TABLE Categoria (
    Tipo VARCHAR(255) PRIMARY KEY
);

CREATE TABLE Categorizzazione (
    Nome_Ristorante VARCHAR(255),
    Indirizzo_Ristorante VARCHAR(255),
    Tipo_Categoria VARCHAR(255),
    PRIMARY KEY (Nome_Ristorante, Indirizzo_Ristorante,
                 Tipo_Categoria),
    FOREIGN KEY (Nome_Ristorante, Indirizzo_Ristorante)
                 REFERENCES Ristorante(Nome, Indirizzo),
    FOREIGN KEY (Tipo_Categoria) REFERENCES Categoria(Tipo)
);

CREATE TABLE Chat_Ristorante (
    Id INT PRIMARY KEY,
    Ordine_Utente VARCHAR(255) NOT NULL,
    Tempo_Ordine TIMESTAMP NOT NULL,
    FOREIGN KEY (Tempo_Ordine, Ordine_Utente)
                 REFERENCES Ordine(Tempo, Utente_Mail)
);

CREATE TABLE Messaggio_Ristorante (

```

```

    Chat_id INT,
    Tempo_Invio TIMESTAMP,
    Contenuto TEXT NOT NULL,
    Mittente mittente_dom NOT NULL,
    PRIMARY KEY (Chat_id, Tempo_Invio),
    FOREIGN KEY (Chat_id) REFERENCES Chat_Ristorante(Id)
);

CREATE TABLE Ingredienti (
    Nome VARCHAR(255) PRIMARY KEY
);

CREATE TABLE Allergeni (
    Nome VARCHAR(255) PRIMARY KEY
);

CREATE TABLE Composizione (
    Titolo_Piatto VARCHAR(255),
    Ristorante VARCHAR(255),
    Luogo VARCHAR(255),
    Nome_Ingrediente VARCHAR(255),
    Quantità DECIMAL(10, 2) NOT NULL,
    PRIMARY KEY (Titolo_Piatto, Ristorante, Luogo,
        Nome_Ingrediente),
    FOREIGN KEY (Nome_Ingrediente)
        REFERENCES Ingredienti(Nome),
    FOREIGN KEY (Titolo_Piatto, Ristorante, Luogo)
        REFERENCES Piatto(Titolo, Nome_Ristorante,
            Indirizzo_Ristorante)
);

CREATE TABLE Ai (
    Nome_Ingrediente VARCHAR(255),
    Nome_Allergene VARCHAR(255),
    PRIMARY KEY (Nome_Ingrediente, Nome_Allergene),
    FOREIGN KEY (Nome_Ingrediente)
        REFERENCES Ingredienti(Nome),
    FOREIGN KEY (Nome_Allergene)

```

```

        REFERENCES Allergeni(Nome)
    );

CREATE TABLE Lista (
    Titolo VARCHAR(255) PRIMARY KEY
);

CREATE TABLE Appartiene (
    Piatto_Titolo VARCHAR(255),
    Ristorante VARCHAR(255),
    Luogo VARCHAR(255),
    Lista_Titolo VARCHAR(255),
    PRIMARY KEY (Piatto_Titolo, Ristorante, Luogo,
        Lista_Titolo),
    FOREIGN KEY (Lista_Titolo)
        REFERENCES Lista(Titolo),
    FOREIGN KEY (Piatto_Titolo, Ristorante, Luogo)
    REFERENCES Piatto(Titolo, Nome_Ristorante,
        Indirizzo_Ristorante)
);

CREATE TABLE Consegna (
    Tempo_Ordine TIMESTAMP,
    Ordine_Utente VARCHAR(255),
    Cod_Rider INT NOT NULL,
    Inizio_Consegna TIMESTAMP NOT NULL,
    Ora_Recapito TIMESTAMP DEFAULT NULL,
    Mancia DECIMAL(10, 2) DEFAULT NULL,
    PRIMARY KEY (Tempo_Ordine, Ordine_Utente),
    FOREIGN KEY (Tempo_Ordine, Ordine_Utente)
        REFERENCES Ordine(Tempo, Utente_Mail),
    FOREIGN KEY (Cod_Rider) REFERENCES Rider(Codice)
);

CREATE TABLE Chat_Rider (
    Id INT PRIMARY KEY,
    Inizio_Ordine TIMESTAMP,
    Utente VARCHAR(255),

```

```

    Rider INT,
    FOREIGN KEY (Inizio_Ordine, Utente)
        REFERENCES Consegna(Tempo_Ordine, Ordine_Utente)
);

CREATE TABLE Messaggio_Rider (
    Chat_id INT,
    Tempo_Invio TIMESTAMP,
    Contenuto TEXT NOT NULL,
    Mittente mittente_dom NOT NULL,
    PRIMARY KEY (Chat_id, Tempo_Invio),
    FOREIGN KEY (Chat_id) REFERENCES Chat_Rider(Id)
);

CREATE TABLE Recensione_Rider (
    Utente VARCHAR(255),
    Ordine TIMESTAMP,
    Rider INT NOT NULL,
    Valutazione INT NOT NULL,
    Commento TEXT DEFAULT NULL,
    PRIMARY KEY (Utente, Ordine),
    FOREIGN KEY (Ordine, Utente)
        REFERENCES Consegna(Tempo_Ordine, Ordine_Utente),
    FOREIGN KEY(Rider) REFERENCES Rider(Codice)
);

CREATE TABLE Recensione_Ristorante (
    Utente VARCHAR(255),
    Ordine TIMESTAMP,
    Nome_Ristorante VARCHAR(255) NOT NULL,
    Indirizzo_Ristorante VARCHAR(255) NOT NULL,
    Valutazione stelle_dom NOT NULL,
    Commento TEXT DEFAULT NULL,
    PRIMARY KEY (Utente, Ordine),
    FOREIGN KEY (Ordine, Utente)
        REFERENCES Consegna(Tempo_Ordine, Ordine_Utente),
    FOREIGN KEY (Nome_Ristorante, Indirizzo_Ristorante)

```

```
REFERENCES Ristorante(Nome, Indirizzo)
);
```

3.2 Popolamento delle tabelle del Database

```
INSERT INTO Utente (Email, Nome, Num_Telefono, Password,
Premium, Indirizzo_Recapito, Nome_Pagamento, Numero_Carta,
Poz_classifica)
VALUES
('mario.rossi@example.com', 'Mario Rossi', '3331234567',
'password1', TRUE, 'Via Roma 1, Milano', 'Mario Rossi',
'1234567890123456', 1),
('anna.bianchi@example.com', 'Anna Bianchi', '3319876543',
'password2', FALSE, 'Corso Venezia 15, Milano', 'Anna Bianchi',
'6543210987654321', 2);
```

```
INSERT INTO Borsellino (Utente_Mail, Saldo)
VALUES
('mario.rossi@example.com', 50.00),
('anna.bianchi@example.com', 30.00);
```

```
INSERT INTO Rider (Codice, Stato, Num_Consegne, Posizione,
Mezzo, Km_Disponibili, Poz_classifica)
VALUES
(1, 'disponibile', 120, 'Via Torino 5, Milano',
'bici elettrica', 50.0, 2),
(2, 'occupato', 200, 'Via Dante 20, Milano',
'monopattino', 30.0, 1);
```

```
INSERT INTO Codice_Sconto (Codice, Utente_Mail)
VALUES
('SCONTO20', 'mario.rossi@example.com'),
('PROMO10', 'anna.bianchi@example.com');
```

```
INSERT INTO Ordine (Utente_Mail, Tempo, Annullamento,
Ristorante_Nome, Ristorante_Indirizzo)
```

```

VALUES
('mario.rossi@example.com', '2023-06-25 12:30:00',
 FALSE, 'La Pizzeria', 'Via Napoli 10, Milano'),
('anna.bianchi@example.com', '2023-06-26 13:00:00',
 FALSE, 'Ristorante Italiano', 'Corso Como 25, Milano');

INSERT INTO Reclamo (Tempo_Ordine, Utente, Motivo)
VALUES
('2023-06-25 12:30:00', 'mario.rossi@example.com',
 'Ordine in ritardo'),
('2023-06-26 13:00:00', 'anna.bianchi@example.com',
 'Piatto sbagliato');

INSERT INTO Ristorante (Nome, Indirizzo, Costo_Spedizione,
 Descrizione, Immagine_Profilo, Stelle, Data_Entrata)
VALUES
('La Pizzeria', 'Via Napoli 10, Milano', 2.50,
 'La migliore pizza napoletana a Milano.',
 'immagine1.jpg', 4, '2021-05-15'),
('Ristorante Italiano', 'Corso Como 25, Milano',
 3.00, 'Cucina italiana autentica.',
 'immagine2.jpg', 5, '2019-10-20');

INSERT INTO Piatto (Titolo, Nome_Ristorante,
 Indirizzo_Ristorante, Sconto, Prezzo, Immagine,
 Poz_classifica)
VALUES
('Margherita', 'La Pizzeria', 'Via Napoli 10, Milano',
 0.10, 8.00, 'pizza_margherita.jpg', 2),
('Spaghetti alla Carbonara', 'Ristorante Italiano',
 'Corso Como 25, Milano', 0.15, 12.00, 'carbonara.jpg', 1);

INSERT INTO Contenuto (Tempo_Ordine, Ordine_Utente,
 Piatto_Titolo, Nome_Ristorante, Indirizzo_Ristorante, Quantità)
VALUES
('2023-06-25 12:30:00', 'mario.rossi@example.com',
 'Margherita', 'La Pizzeria', 'Via Napoli 10, Milano', 2),
('2023-06-26 13:00:00', 'anna.bianchi@example.com',

```

```

'Spaghetti alla Carbonara', 'Ristorante Italiano',
'Corso Como 25, Milano', 1);

INSERT INTO Categoria (Tipo)
VALUES
('Italiana'),
('Pizza'),
('Pasta');

INSERT INTO Categorizzazione (Nome_Ristorante,
                             Indirizzo_Ristorante, Tipo_Categoria)
VALUES
('La Pizzeria', 'Via Napoli 10, Milano', 'Pizza'),
('Ristorante Italiano', 'Corso Como 25, Milano', 'Pasta');

INSERT INTO Chat_Ristorante (Id, Ordine_Utente, Tempo_Ordine)
VALUES
(1, 'mario.rossi@example.com', '2023-06-25 12:30:00'),
(2, 'anna.bianchi@example.com', '2023-06-26 13:00:00');

INSERT INTO Messaggio_Ristorante (Chat_id, Tempo_Invio,
                                   Contenuto, Mittente)
VALUES
(1, '2023-06-25 12:35:00',
 'Il tuo ordine è in preparazione.', 'r'),
(1, '2023-06-25 12:40:00',
 'Grazie per l'aggiornamento.', 'u'),
(2, '2023-06-26 13:05:00',
 'Il tuo ordine è in arrivo.', 'r'),
(2, '2023-06-26 13:10:00',
 'Perfetto, grazie.', 'u');

INSERT INTO Ingredienti (Nome)
VALUES
('Pomodoro'),
('Mozzarella'),
('Pasta'),
('Uova'),

```



```

('Guanciale');

INSERT INTO Allergeni (Nome)
VALUES
('Glutine'),
('Lattosio'),
('Uova');

INSERT INTO Composizione (Titolo_Piatto, Ristorante,
                          Luogo, Nome_Ingrediente, Quantità)
VALUES
('Margherita', 'La Pizzeria', 'Via Napoli 10, Milano',
 'Pomodoro', 0.2),
('Margherita', 'La Pizzeria', 'Via Napoli 10, Milano',
 'Mozzarella', 0.15),
('Spaghetti alla Carbonara', 'Ristorante Italiano',
 'Corso Como 25, Milano', 'Pasta', 0.25),
('Spaghetti alla Carbonara', 'Ristorante Italiano',
 'Corso Como 25, Milano', 'Uova', 0.05),
('Spaghetti alla Carbonara', 'Ristorante Italiano',
 'Corso Como 25, Milano', 'Guanciale', 0.1);

INSERT INTO Ai (Nome_Ingrediente, Nome_Allergene)
VALUES
('Pasta', 'Glutine'),
('Mozzarella', 'Lattosio'),
('Uova', 'Uova');

INSERT INTO Lista (Titolo)
VALUES
('Specialità della casa'),
('Promozioni del mese');

INSERT INTO Appartiene (Piatto_Titolo, Ristorante,
                       Luogo, Lista_Titolo)
VALUES
('Margherita', 'La Pizzeria', 'Via Napoli 10, Milano',
 'Specialità della casa'),

```

```

('Spaghetti alla Carbonara', 'Ristorante Italiano',
 'Corso Como 25, Milano', 'Promozioni del mese');

INSERT INTO Consegna (Tempo_Ordine, Ordine_Utente, Cod_Rider,
 Inizio_Consegna, Ora_Recapito, Mancia)
VALUES
('2023-06-25 12:30:00', 'mario.rossi@example.com', 1,
 '2023-06-25 12:45:00', '2023-06-25 13:00:00', 2.00),
('2023-06-26 13:00:00', 'anna.bianchi@example.com', 2,
 '2023-06-26 13:15:00', '2023-06-26 13:30:00', 1.50);

INSERT INTO Chat_Rider (Id, Inizio_Ordine, Utente, Rider)
VALUES
(1, '2023-06-25 12:30:00', 'mario.rossi@example.com', 1),
(2, '2023-06-26 13:00:00', 'anna.bianchi@example.com', 2);

INSERT INTO Messaggio_Rider (Chat_id, Tempo_Invio, Contenuto,
 Mittente)
VALUES
(1, '2023-06-25 12:50:00',
 'Il tuo ordine è in consegna.', 'r'),
(1, '2023-06-25 12:55:00', 'Grazie, attendo.', 'u'),
(2, '2023-06-26 13:20:00',
 'Sto arrivando con il tuo ordine.', 'r'),
(2, '2023-06-26 13:25:00', 'Perfetto, grazie.', 'u');

INSERT INTO Recensione_Rider (Utente, Ordine, Rider,
 Valutazione, Commento)
VALUES
('mario.rossi@example.com', '2023-06-25 12:30:00',
 1, 5, 'Servizio eccellente!'),
('anna.bianchi@example.com', '2023-06-26 13:00:00',
 2, 4, 'Consegna puntuale.');
```

```

INSERT INTO Recensione_Ristorante (Utente, Ordine,
 Nome_Ristorante, Indirizzo_Ristorante, Valutazione, Commento)
VALUES
('mario.rossi@example.com', '2023-06-25 12:30:00',
```

```

        'La Pizzeria', 'Via Napoli 10, Milano', 5,
        'Pizza fantastica!'),
('anna.bianchi@example.com', '2023-06-26 13:00:00',
 'Ristorante Italiano', 'Corso Como 25, Milano', 4,
 'Ottima Carbonara.');
```

3.3 Operazioni sul database

```

-- Tutte le recensioni riguardanti un rider
SELECT rr.Utente, rr.Ordine, rr.Rider, rr.Valutazione,
       rr.Commento
FROM Recensione_Rider rr
WHERE rr.Rider = 2;

-- La chat tra un ristorante e un utente dato un ordine
SELECT mr.Tempo_Invio, mr.Contenuto, mr.Mittente
FROM Messaggio_Ristorante mr
JOIN Chat_Ristorante cr ON mr.Chat_id = cr.Id
WHERE cr.Ordine_Utente = 'mario.rossi@example.com' AND
      cr.Tempo_Ordine = '2023-06-25 12:30:00'
ORDER BY mr.Tempo_Invio;

--ricerca di tutti i reclami verso un ristorante
SELECT r.Tempo_Ordine, r.Utente, r.Motivo
FROM Reclamo r
JOIN Ordine o ON r.Tempo_Ordine = o.Tempo
              AND r.Utente = o.Utente_Mail
WHERE o.Ristorante_Nome = 'La Pizzeria' AND
      o.Ristorante_Indirizzo = 'Via Napoli 10, Milano';

--tutti gli ordini di un utente
SELECT o.Utente_Mail, o.Tempo, o.Annullamento,
       o.Ristorante_Nome, o.Ristorante_Indirizzo
FROM Ordine o
WHERE o.Utente_Mail = 'mario.rossi@example.com';
```

```

-- tutti le consegne di un rider
SELECT c.Tempo_Ordine, c.Ordine_Utente,
       c.Cod_Rider, c.Inizio_Consegna,
       c.Ora_Recapito, c.Mancia
FROM Consegna c
WHERE c.Cod_Rider = 1;

--tutti i piatti di un ristorante
SELECT p.Titolo, p.Sconto, p.Prezzo, p.Immagine
FROM Piatto p
WHERE p.Nome_Ristorante = 'La Pizzeria' AND
       p.Indirizzo_Ristorante = 'Via Napoli 10, Milano';

--tutti i piatti in un ordine
SELECT c.Piatto_Titolo, c.Nome_Ristorante,
       c.Indirizzo_Ristorante, c.Quantità
FROM Contenuto c
WHERE c.Tempo_Ordine = '2023-06-25 12:30:00' AND
       c.Ordine_Utente = 'mario.rossi@example.com';

--tutte le liste di un piatto
SELECT a.Lista_Titolo
FROM Appartiene a
WHERE a.Piatto_Titolo = 'Margherita' AND
       a.Ristorante = 'La Pizzeria' AND
       a.Luogo = 'Via Napoli 10, Milano';

-- tutti gli allergeni di un piatto
SELECT ai.Nome_Allergene
FROM Ai ai
JOIN Composizione c ON
       ai.Nome_Ingrediente = c.Nome_Ingrediente
WHERE c.Titolo_Piatto = 'Margherita' AND
       c.Ristorante = 'La Pizzeria' AND
       c.Luogo = 'Via Napoli 10, Milano';

--tutti gli ingredienti di un piatto
SELECT c.Nome_Ingrediente, c.Quantità

```

```
FROM Composizione c
WHERE c.Titolo_Piatto = 'Margherita' AND
      c.Ristorante = 'La Pizzeria' AND
      c.Luogo = 'Via Napoli 10, Milano';
```