

Corso di Laurea Triennale in Informatica Dipartimento di
Informatica
Università di Torino

Basi di Dati
Progetto di Laboratorio

***Oliviu Gratii** [matricola 944575]*

oliviu.gratii@edu.unito.it

***Matteo Barone** [matricola 951558]*

matteo.barone415@edu.unito.it

Anno Accademico: 2021/2022

1. Progettazione concettuale

1.1. Requisiti iniziali

House booking

Si vuole realizzare una base di dati per un servizio che permette di affittare e prenotare alloggi di vario tipo ad esempio interi appartamenti, stanze private (camera privata e spazi comuni) e stanze condivise (spazio in comune e camera condivisa).^[1]

Gli utenti si registrano al servizio fornendo indirizzo email, password, nome, cognome, numero o numeri di telefono. Se l'utente fornisce la foto della carta d'identità, viene riconosciuto come verificato. Inoltre, l'utente deve indicare un metodo di pagamento per poter prenotare. Gli utenti possono essere ospiti o "host" ovvero possono a loro volta ospitare altri utenti del servizio in uno o più alloggi di loro proprietà. Inoltre gli "host" possono diventare "superhost" se soddisfano i seguenti requisiti:

- Devono aver completato almeno 10 soggiorni, per un totale di almeno 100 notti.
- Devono aver conservato un tasso di cancellazione dell'1% (una cancellazione ogni 100 prenotazioni) massimo.
- Devono aver mantenuto una valutazione complessiva di 4,8 considerando tutti i soggiorni in tutte le case di sua proprietà.

Gli utenti superhost ricevono un badge sul loro profilo.

Gli alloggi sono descritti indicando un nome, l'indirizzo (visibile all'ospite solo quando la prenotazione è confermata, altrimenti è visibile solo il comune), una descrizione, il prezzo per notte per persona e i costi di pulizia, delle foto, i servizi (ad esempio, cucina, wi-fi, lavatrice, ecc.), numero di letti e orario di check-in e check-out oltre all'host a cui appartiene, il rating medio e il numero di recensioni (vedere Fig. 1).

Gli utenti possono aggiungere alcune case tra i preferiti. Gli utenti possono avere diverse liste, ad esempio in base al viaggio che vogliono compiere.

Gli utenti possono prenotare degli alloggi di qualsiasi tipo indicando un intervallo di date per il soggiorno e il numero degli ospiti. Se gli ospiti sono a loro volta utenti del servizio, se ne possono indicare i nominativi. La prenotazione deve essere confermata o rifiutata dall'host. La prenotazione ha un costo totale e se confermata viene eseguito il pagamento. Inoltre, la prenotazione può essere cancellata sia dall'ospite che dall'host.

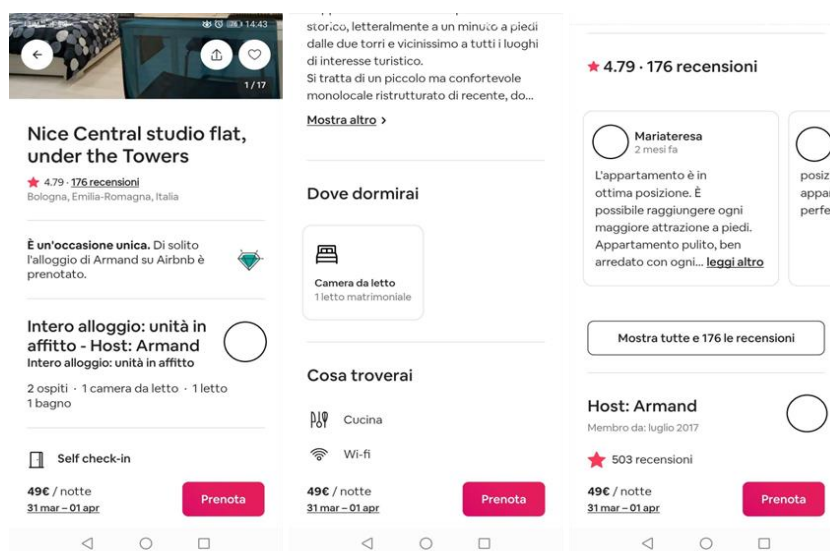


Fig. 1: Dettagli di un appartamento.

Al termine del soggiorno, gli ospiti e gli host si possono valutare a vicenda. La recensione fatta dagli ospiti comprende due testi (uno per l'appartamento e uno per l'host) e una serie di punteggi in una scala da 1 a 5 su dimensioni come pulizia, comunicazione, posizione, qualità/prezzo. La valutazione complessiva del soggiorno è una media delle valutazioni ricevute sulle singole dimensioni. Le recensioni degli host comprendono solo un commento testuale. Le recensioni possono essere visibili o non visibili. Diventano visibili quando entrambi hanno fatto la recensione oppure se uno dei due non ha fatto la recensione, l'altra diventa visibile dopo 7 giorni dalla fine del soggiorno. Gli host e gli ospiti possono commentare più volte le review in cui sono coinvolti, creando un thread di discussione.

Le recensioni sono visibili sui profili degli utenti suddivise in base a quelle ricevute come ospite e come host.

La base di dati deve supportare le seguenti operazioni:

- Una volta a settimana viene effettuato un calcolo per aggiornare il tasso di cancellazione di ciascun host.
- Una volta al giorno si controllano le condizioni per la qualifica di superhost e viene aggiornato lo status degli host.
- Una volta al mese viene calcolata la classifica degli alloggi più graditi.

1.2. Glossario dei termini

Si vuole realizzare una base di dati per un servizio che permette di affittare e prenotare alloggi di vario tipo ad esempio interi appartamenti, stanze private (camera privata e spazi comuni) e stanze condivise (spazio in comune e camera condivisa).^[2]

Gli utenti si registrano al servizio fornendo indirizzo email, password, nome, cognome, numero o numeri di telefono. Se l'utente fornisce la foto della carta d'identità, viene riconosciuto come verificato. Inoltre, l'utente deve indicare un metodo di pagamento per poter prenotare. Gli utenti possono essere ospiti o "host" ovvero possono a loro volta ospitare altri utenti del servizio in uno o più alloggi di loro proprietà. Inoltre gli "host" possono diventare "superhost" se soddisfano i seguenti requisiti:

- Devono aver completato almeno 10 soggiorni, per un totale di almeno 100 notti.
- Devono aver conservato un tasso di cancellazione dell'1% (una cancellazione ogni 100 prenotazioni) massimo.
- Devono aver mantenuto una valutazione complessiva di 4,8 considerando tutti i soggiorni in tutte le case di sua proprietà.

Gli utenti superhost ricevono un badge sul loro profilo.

Gli alloggi sono descritti indicando un nome, l'indirizzo (visibile all'ospite solo quando la prenotazione è confermata, altrimenti è visibile solo il comune), una descrizione, il prezzo per notte per persona e i costi di pulizia, delle foto, i servizi (ad esempio, cucina, wi-fi, lavatrice, ecc.), numero di letti e orario di check-in e check-out oltre all'host a cui appartiene, il rating medio e il numero di recensioni (vedere Fig. 1).

Gli utenti possono aggiungere alcune case tra i preferiti. Gli utenti possono avere diverse liste, ad esempio in base al viaggio che vogliono compiere.

Gli utenti possono prenotare degli alloggi di qualsiasi tipo indicando un intervallo di date per il soggiorno e il numero degli ospiti. Se gli ospiti sono a loro volta utenti del servizio, se ne possono indicare i nominativi. La prenotazione deve essere confermata o rifiutata dall'host. La prenotazione ha un costo totale e se confermata viene eseguito il pagamento. Inoltre, la prenotazione può essere cancellata sia dall'ospite che dall'host.

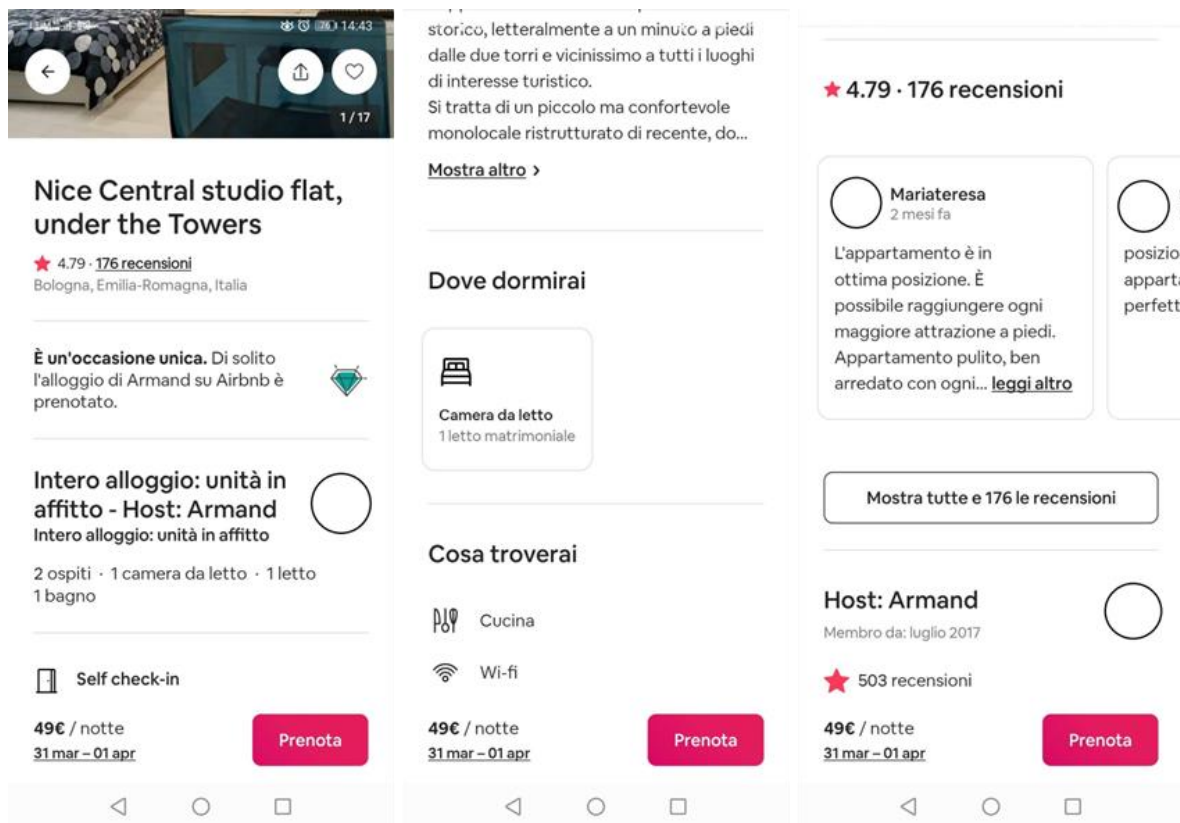


Fig. 1: Dettagli di un appartamento.

Al termine del **soggiorno**, gli **ospiti** e gli **host** si possono **valutare** a vicenda. La **recensione** fatta dagli **ospiti** comprende due testi (uno per l'**appartamento** e uno per l'**host**) e una serie di punteggi in una scala da 1 a 5 su dimensioni come pulizia, comunicazione, posizione, qualità/prezzo. La **valutazione** complessiva del **soggiorno** è una media delle valutazioni ricevute sulle singole dimensioni. Le **recensioni** degli **host** comprendono solo un **commento** testuale. Le **recensioni** possono essere visibili o non visibili. Diventano visibili quando entrambi hanno fatto la **recensione** oppure se uno dei due non ha fatto la **recensione**, l'altra diventa visibile dopo 7 giorni dalla fine del **soggiorno**. Gli **host** e gli **ospiti** possono commentare più volte le **review** in cui sono coinvolti, creando un thread di discussione.

Le **recensioni** sono visibili sui profili degli **utenti** suddivise in base a quelle ricevute come **ospite** e come **host**. La base di dati deve supportare le seguenti operazioni:

- Una volta a settimana viene effettuato un calcolo per aggiornare il tasso di cancellazione di ciascun **host**.
- Una volta al giorno si controllano le condizioni per la qualifica di **superhost** e viene aggiornato lo status degli **host**.
- Una volta al mese viene calcolata la classifica degli **alloggi** più graditi.

[1] Il servizio descritto è liberamente ispirato a Airbnb (<https://www.airbnb.it/>) a cui è possibile fare riferimento per completare e disambiguare i requisiti.

Termine	Descrizione	Sinonimo	Collegamenti
utenti	Persona che si registra al sito	persona	Liste
ospiti	Utente che affitta un alloggio		Prenotazione, Commenti, Recensione Ospite, Recensione Soggiorno
host	Utente che da in affitto uno dei suoi alloggi		Alloggio, Commenti ,Recensione Soggiorno
alloggi	Posto fisico posseduto dall'host , affittabile agli ospiti	Case, appartamenti, stanze condivise, stanze private	Prenotazione, Host, Recensione Soggiorno
recensione	Esame critico, in forma di articolo più o meno esteso	Valutazione, review	Host, Ospite, Alloggio
Commenti	Serie di note illustrative o di giudizi critici apposite a un testo		Recensione Host, Recensione Soggiorno ,Host, Ospite
Super host	Host che mantiene un certo tasso di rendimento		Host
Liste	Liste di alloggi preferiti di un utente (anche in base al viaggio)		Utente ,Alloggio

Prenotazioni	Pattuizione preliminare ed accessoria con cui ci si garantisce la stipulazione di un contratto finale	Soggiorno	Utente, Alloggio, Recensione
Pagamento	Mezzo di pagamento con cui viene effettuata la prenotazione		Prenotazione, Utente
Recensione Utente	Recensione fatta dall'host sull'utente		Host, Ospite
Recensione Soggiorno	Recensione fatta dall'utente nei confronti dell'alloggio e dell'host		Host, Ospite, Alloggio

1.3. Requisiti rivisti

text: testo inizialmente presente

~~text~~-testo presente inizialmente ma successivamente rimosso da me

text-testo aggiunto da me in base ad un'analisi delle foto nei requisiti iniziali e facendo riferimento al modello reale di Airbnb

Si vuole realizzare una base di dati per un servizio che permette di affittare e prenotare alloggi di vario tipo ad esempio interi appartamenti, stanze private (camera privata e spazi comuni) e stanze condivise (spazio in comune e camera condivisa).

Gli utenti si registrano al servizio fornendo indirizzo email, password, nome, cognome, numero o numeri di telefono, e data di registrazione. Se l'utente fornisce la foto della carta d'identità, viene riconosciuto come verificato. Inoltre, l'utente deve indicare un metodo di pagamento per poter prenotare, composto dai semplici dati di una carta di credito più un soprannome assegnatogli dall'utente che si registra.

Gli utenti possono essere ospiti o "host" ovvero possono a loro volta ospitare altri utenti del servizio in uno o più alloggi di loro proprietà.

Inoltre, gli "host" possono diventare "superhost" se soddisfano i seguenti requisiti:

- Devono aver completato almeno 10 soggiorni, per un totale di almeno 100 notti.
- Devono aver conservato un tasso di cancellazione dell'1% (una cancellazione ogni 100 prenotazioni) massimo sul totale delle prenotazioni ricevute.
- Devono aver mantenuto una valutazione complessiva di 4,8 considerando tutti i soggiorni in tutte le case gli alloggi di sua proprietà.

Gli utenti superhost ricevono un badge sul loro profilo.

Gli alloggi sono descritti indicando un nome, l'indirizzo (visibile all'ospite solo quando la prenotazione è confermata, altrimenti è visibile solo il comune) (composto da : via,nr ,città, regione e paese) , una descrizione, il prezzo per notte per persona e i costi di pulizia, delle foto, i servizi (ad esempio, cucina, wi-fi, lavatrice, ecc.), numero di letti e orario di check-in e check-out oltre all'host a cui appartiene, il rating medio, il numero di recensioni, nr servizi e tipo. Inoltre, rappresentiamo la possibilità di indicare il tipo dell'alloggio, offrire servizi e il tipo di camere presenti in un alloggio. Per camera rappresentiamo il nome, il nr dei posti letto e la disposizione dei letti. Per i servizi rappresentiamo solo il nome.

L'identificatore dell'alloggio è : CIR – Codice identificativo di riferimento, è il codice identificativo regionale della tua struttura, attraverso il quale un proprietario o gestore di case vacanza, b&b o qualsiasi altro affitto breve comunica ufficialmente l'inizio dell'attività al Municipio di competenza, adempiendo a tutti gli obblighi previsti.

Gli utenti possono aggiungere alcune case tra i preferiti. Gli utenti possono avere diverse liste, ad esempio in base al viaggio che vogliono compiere.

Per lista di preferiti rappresentiamo il nome e la possibilità di contenere alloggi.

Gli utenti possono prenotare degli alloggi di qualsiasi tipo indicando un intervallo di date per il soggiorno la data inizio, data fine del soggiorno e il numero degli ospiti. Se gli ospiti sono a loro volta utenti del servizio, se ne

possono indicare i nominativi. La prenotazione deve essere confermata o rifiutata dall'host. **La prenotazione ha un costo totale e se confermata viene eseguito il pagamento.** Inoltre, la prenotazione può essere cancellata sia dall'ospite che dall'host. **Mentre si attende una conferma da parte dell'host la prenotazione è da considerarsi pendente. Quando è giunta la data di fine soggiorno la prenotazione è da considerarsi conclusa.**

Al termine del soggiorno, gli ospiti e gli host si possono valutare a vicenda.

La recensione fatta dagli ospiti comprende due testi (uno per l'appartamento e uno per l'host) e una serie di punteggi in una scala da 1 a 5 su dimensioni come pulizia, comunicazione, posizione, qualità/prezzo. La valutazione complessiva del soggiorno è una media delle valutazioni ricevute sulle singole dimensioni.

Le recensioni degli host comprendono solo un commento testuale.

Le recensioni possono essere visibili o non visibili. Diventano visibili quando entrambi hanno fatto la recensione oppure se uno dei due non ha fatto la recensione, l'altra diventa visibile dopo 7 giorni dalla fine del soggiorno.

Gli host e gli ospiti possono commentare più volte le review in cui sono coinvolti, creando un thread di discussione.

Le recensioni sono visibili sui profili degli utenti suddivise in base a quelle ricevute come ospite e come host.

Per commento rappresentiamo timestamp e testo.

La base di dati deve supportare le seguenti operazioni:

- Una volta a settimana viene effettuato un calcolo per aggiornare il tasso di cancellazione di ciascun host.
- Una volta al giorno si controllano le condizioni per la qualifica di superhost e viene aggiornato lo status degli host.
- Una volta al mese viene calcolata la classifica degli alloggi più graditi.

Frase di carattere Generale

Si vuole realizzare una base di dati per un servizio che permette di affittare e prenotare alloggi

Frase relativi agli Utenti

Gli utenti si registrano al servizio fornendo indirizzo email, password, nome, cognome, numero o numeri di telefono, e data di registrazione. Se l'utente fornisce la foto della carta d'identità, viene riconosciuto come verificato. Inoltre, l'utente deve indicare un metodo di pagamento per poter prenotare, composto dai semplici dati di una carta di credito più un soprannome assegnatogli dall'utente che si registra.

Frase relative alle liste

Gli utenti possono avere diverse liste. Per lista di preferiti rappresentiamo il nome e la possibilità di contenere alloggi.

Frase relative agli Host

Gli utenti possono essere ospiti o "host".

Inoltre gli "host" possono diventare "superhost" se soddisfano i seguenti requisiti:

- Devono aver completato almeno 10 soggiorni, per un totale di almeno 100 notti.
- Devono aver conservato un tasso di cancellazione dell'1% (una cancellazione ogni 100 prenotazioni) massimo.
- Devono aver mantenuto una valutazione complessiva di 4,8 considerando tutti i soggiorni in tutte gli alloggi di sua proprietà.

Frasi relative agli alloggi

Gli alloggi sono descritti indicando un nome, l'indirizzo (composto da : via, nr , città, regione e paese) , una descrizione, il prezzo per notte per persona e i costi di pulizia, delle foto, i servizi (ad esempio, cucina, wi-fi, lavatrice, ecc.), numero di letti e orario di check-in e check-out oltre all'host a cui appartiene, il rating medio, il numero di recensioni, nr servizi e tipo. Inoltre rappresentiamo la possibilità di indicare il tipo dell'alloggio, offrire servizi e il tipo di camere presenti in un alloggio.

Per camera rappresentiamo il nome, il nr dei posti letto e la disposizione dei letti.
rappresentiamo solo il nome.

Per i servizi

L'identificatore dell'alloggio è : CIR – Codice identificativo di riferimento, è il codice identificativo regionale della tua struttura, attraverso il quale un proprietario o gestore di case vacanza, b&b o qualsiasi altro affitto breve comunica ufficialmente l'inizio dell'attività al Municipio di competenza, adempiendo a tutti gli obblighi previsti.

Frasi relative alla prenotazione

Gli utenti possono prenotare degli alloggi di qualsiasi tipo indicando la data inizio, data fine del soggiorno e il numero degli ospiti. Se gli ospiti sono a loro volta utenti del servizio, se ne possono indicare i nominativi. La prenotazione deve essere confermata o rifiutata dall'host. Inoltre, la prenotazione può essere cancellata sia dall'ospite che dall'host. Mentre si attende una conferma da parte dell'host la prenotazione è da considerarsi pendente. Quando è giunta la data di fine soggiorno la prenotazione è da considerarsi conclusa.

Frasi relative alle recensioni ospiti

La recensione fatta dagli ospiti comprende due testi (uno per l'appartamento e uno per l'host) e una serie di punteggi in una scala da 1 a 5 su dimensioni come pulizia, comunicazione, posizione, qualità/prezzo. La valutazione complessiva del soggiorno è una media delle valutazioni ricevute sulle singole dimensioni.

Frase relative alle recensioni Host

Le recensioni degli host comprendono solo un commento testuale.

Frase relative al metodo di pagamento

L'utente deve indicare un metodo di pagamento per poter prenotare, composto dai semplici dati di una carta di credito più un soprannome assegnatogli dall'utente che si registra.

Frase relative ai commenti

Gli host e gli ospiti possono commentare più volte le review in cui sono coinvolti, creando un thread di discussione.
Per commento rappresentiamo timestamp e testo.

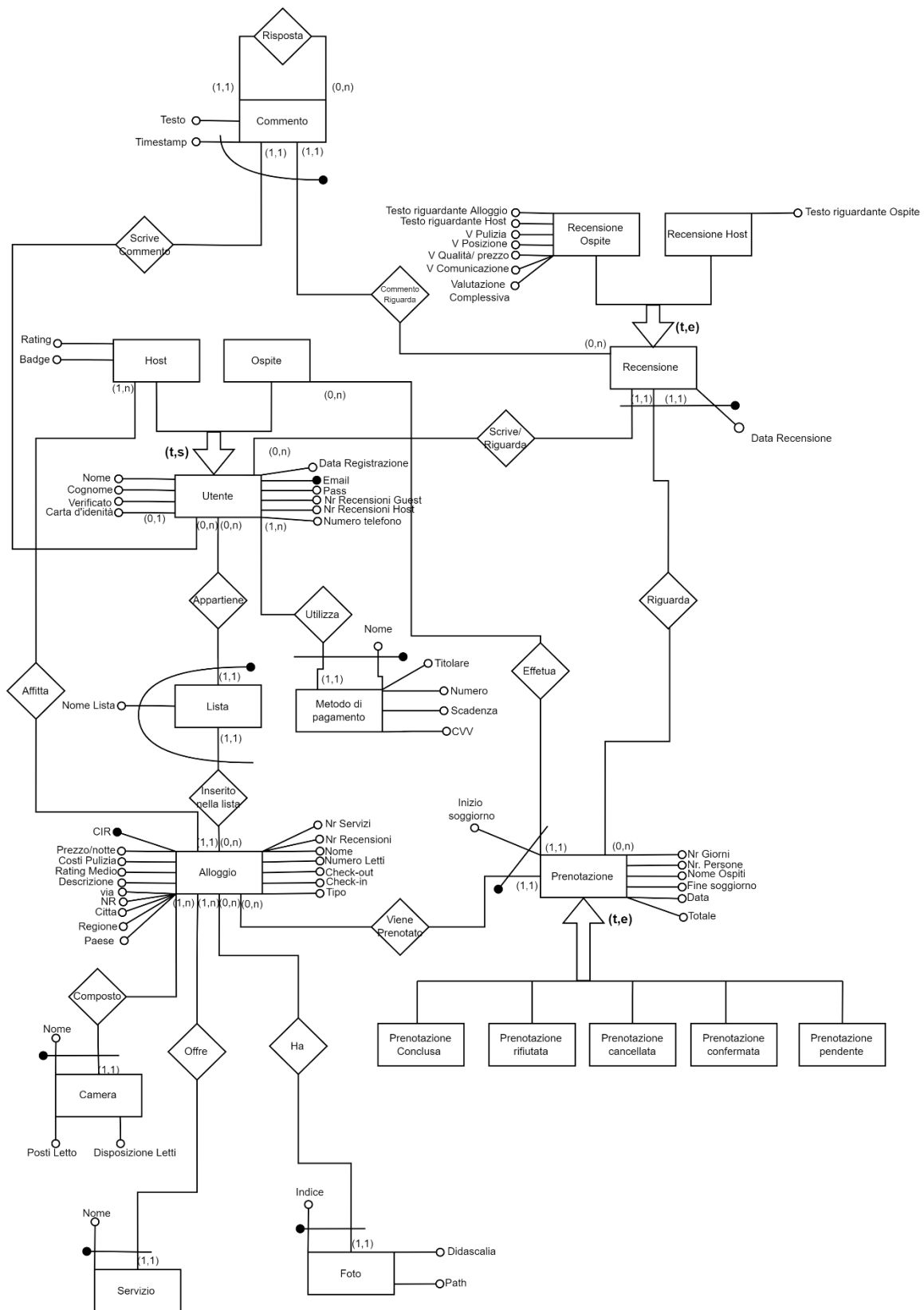
Frase relative ai servizi

Per i servizi rappresentiamo solo il nome.

Frase relative alle camere

Per camera rappresentiamo il nome, il nr dei posti letto e la disposizione dei letti.

1.4 Schema E-R + business rules



Business Rules

- In una prenotazione, per essere ritenuta valida, la data di inizio soggiorno deve essere strettamente minore alla data di fine soggiorno
- Gli utenti possono valutarsi a vicenda solo dopo la fine del soggiorno, quindi la data di fine soggiorno deve essere strettamente minore alla data di pubblicazione della recensione
- La risposta ad un commento non è d'obbligo.
- I voti si possono esprimere tra 1-5.
- Il numero di persone della prenotazione deve essere minore o uguale al numero massimo di persone che l'alloggio può ospitare
- Se una delle parti non scrive una contro recensione, quella della controparte diventa visibile decorsi 7 giorni.
- Il badge indica il tipo dell'host true = superhost false = normal host
- Se carta d'identità è null, verificato è false e viceversa
- I tipi degli alloggi sono : interi appartamenti, stanze private e posti letto in stanza comune.

Regole di derivazione

- Per diventare Superhost un host deve avere il rating(Hosts) maggiore o uguale a 4.8. Ed una percentuale cancellazione inferiore all'1% del totale di prenotazione andata a buon fine
- L'importo totale della prenotazione viene calcolato moltiplicando nr persone(prenotazione) per prezzo a persona(alloggio) e sommando a ciò i costi pulizia(alloggi).
- Il rating medio dell'alloggio viene calcolato facendo una media dei voti ricevuti dagli ospiti(Recensioni).
- Valutazione complessiva e la media dei voti ricevuti
- Il tasso di cancellazione viene calcolato facendo il rapporto tra il numero delle cancellazioni e delle prenotazioni concluse nel periodo degli ultimi 100 giorni da quello attuale.
- Il rating viene calcolato facendo la media dei voti di tutte le prenotazioni concluse in tutti i suoi alloggi.
- Il Nr giorni viene calcolato facendo la differenza tra la data di fine soggiorno e la data di inizio soggiorno.
- La media dei voti viene calcolata facendo la media aritmetica dei vari voti.
- **Utente.Nr recensioni** viene calcolato il numero effettivo di recensioni che lo riguardano
- **Alloggio.Nr recensioni** viene calcolato il numero effettivo di recensioni che lo riguardano
- Il Nr letti e la somma dei Posti letto di ogni camera di un alloggio
- Il totale prenotazione è calcolato facendo la moltiplicazione del prezzo/notte +costi pulizia.

Dizionario dei dati (entità)

Entità	Descrizione	Attributi	Identificatore
Utente	Entità genitore di : Host, Ospite	<ol style="list-style-type: none"> 1. Email 2. Password 3. Nome 4. Cognome 5. Numero di telefono 6. Verificato 7. Carta d'identità 8. Data registrazione 9. Nr Recensioni Guest 10. Nr Recensioni Host 	Email
Host	Entità figlia di Utente	<ol style="list-style-type: none"> 1. Email 2. Password 3. Nome 4. Cognome 5. Numero di telefono 6. Verificato 7. Carta d'identità 8. Data registrazione 9. Nr Recensioni Guest 10. Nr Recensioni Host 11. Rating 12. Badge 	Email
Ospite	Entità figlia di Utente	<ol style="list-style-type: none"> 1. Email 2. Password 3. Nome 4. Cognome 5. Numero di telefono 6. Verificato 7. Carta d'identità 8. Data registrazione 9. Nr Recensioni Guest 10. Nr Recensioni Host 	Email
Metodo di pagamento	Metodo si pagamento di un utente	<ol style="list-style-type: none"> 1. Nome 2. Titolare 3. Numero 4. Scadenza 5. CVV 	

Alloggio	Possedimento di un utente affittabile in qualità di host agli ospiti	<ol style="list-style-type: none"> 1. CIR 2. Nome 3. via 4. nr 5. città 6. regione 7. paese 8. Descrizione 9. Numero letti 10. Prezzo/notte 11. Check-in 12. Check-out 13. Costi pulizia 14. Rating medio 15. Nr Recensioni 16. Foto 17. Nr Servizi 	CIR
Servizio	Servizi offerti da un alloggio	<ol style="list-style-type: none"> 1. Nome 	Nome +(Alloggio.CIR)
Camera	Tipi di camera disponibili in un alloggio	<ol style="list-style-type: none"> 1. Nome 2. Posti Letto 3. Disposizione Letti 	Nome+(Alloggio.CIR)
Foto	Foto di un alloggio	<ol style="list-style-type: none"> 1. Indice 2. Didascalia 3. Path 	Indice + (Alloggio.CIR)
Lista	Entità che rappresenta le liste degli alloggi preferiti dagli utenti	<ol style="list-style-type: none"> 1. Nome Lista 	Nome Lista+ Alloggio.CIR+ Utente.Email
Prenotazioni	Entità genitore di :Prenotazione pendente, prenotazione confermata, prenotazione cancellata , prenotazione rifiutata, prenotazione pendente	<ol style="list-style-type: none"> 1. Inizio soggiorno 2. Nr Giorni 3. Nr persone 4. Nome Ospiti 5. Fine Soggiorno 6. Data 7. Totale 	Inizio soggiorno +Alloggio.CIR+ Ospite.Email

Prenotazione Pendente		<ol style="list-style-type: none"> 1. Inizio soggiorno 2. Nr Giorni 3. Nr persone 4. Nome Ospiti 5. Fine Soggiorno 6. Data 7. Totale 	Inizio soggiorno +Alloggio.CIR+ Ospite.Email
Prenotazione Confermata	Entità figlia di Prenotazione	<ol style="list-style-type: none"> 1. Inizio soggiorno 2. Nr Giorni 3. Nr persone 4. Nome Ospiti 5. Fine Soggiorno 6. Data 7. Totale 	Inizio soggiorno +Alloggio.CIR+ Ospite.Email
Prenotazione Rifiutata	Entità figlia di Prenotazione	<ol style="list-style-type: none"> 1. Inizio soggiorno 2. Nr Giorni 3. Nr persone 4. Nome Ospiti 5. Fine Soggiorno 6. Data 7. Totale 	Inizio soggiorno +Alloggio.CIR+ Ospite.Email
Prenotazione cancellata	Entità figlia di Prenotazione	<ol style="list-style-type: none"> 1. Inizio soggiorno 2. Nr Giorni 3. Nr persone 4. Nome Ospiti 5. Fine Soggiorno 6. Data 7. Totale 	Inizio soggiorno+ Alloggio.CIR+ Ospite.Email
Prenotazione conclusa	Entità figlia di Prenotazione. Che si conclude con il decorrere dei giorni di prenotazione	<ol style="list-style-type: none"> 8. Inizio soggiorno 9. Nr Giorni 10. Nr persone 11. Nome Ospiti 12. Fine Soggiorno 13. Data 14. Totale 	Inizio soggiorno + Alloggio.CIR+ Ospite.Email
Recensione Host	Recensione fatta dall'Ospite	<ol style="list-style-type: none"> 1. Data Recensione 2. Testo Riguardante Host 3. Testo Riguardante Alloggio 4. V Pulizia 5. V Posizione 6. V Qualità/Prezzo 7. V Comunicazione 8. Valutazione Complessiva 	Data Recensione + (Prenotazione.Inizio soggiorno + Prenotazione.CIR+ Prenotazione.Email Ospite)+Utente.Email

Recensione Ospite	Recensione fatta dall'Host	1. Data Recensione 2. Testo Riguardante Ospite	Data Recensione + (Prenotazione.Inizio soggiorno + Prenotazione.CIR+ Prenotazione.Email Ospite)+Utente.Email
Commento	Entità che accoglie il thread di commenti tra ospite e host .	1. Timestamp 2. Testo	TimeStamp+(Recensione.Data + (Recensione.Inizio soggiorno + Recensione.CIR+ Recensione.Email Ospite)+ Recensione.Email) +Utente.Email

Dizionario dei dati (relazioni)

Relazioni	Descrizioni	Componenti	Attributi
Affitta	Host affitta alloggio	Host(1,n) Alloggio(1,1)	Non presente.
Appartiene	Liste appartengono a utenti	Utente(0,n) Lista(1,1)	Non presente.
Inserito nella lista	Alloggi sono inseriti in una lista	Alloggio(0,n) Lista(1,1)	Non presente.
Viene prenotato	Alloggi vengono prenotati	Alloggio(0,n) Prenotazione(1,1)	Non presente.
Effettua	Ospite effettua prenotazione	Ospite(0,n) Prenotazione(1,1)	Non presente.
Scrive/riguarda	Un utente scrive delle recensioni/Le recensioni riguardano l'utente	Utente(0,n) Recensione(1,1)	Non presente.
Riguarda	Una recensione riguarda una prenotazione	Recensione(1,1) Prenotazione(0,n)	Non presente.
Commento riguarda	Un commento riguarda una recensione	Recensione(0,n) Commento(1,1)	Non presente.
Scrive Commento	Utente scrive commento	Utente(0,n) Commento(1,1)	Non presente.
Offre	Un alloggio offre dei servizi	Alloggio(1,n) Servizio(1,1)	Non presente.
Composto	Un alloggio è composto di camere	Alloggio(1,n) Camera(1,1)	Non presente.
Ha	Un alloggio ha delle foto	Alloggio(0,n) Foto(1,1)	Non presente.
Utilizza	Un utente utilizza dei metodi di pagamento	Utente(1,n) Metodo di pagamento(1,1)	Non presente.

2.1 Tavola dei volumi

SRC: www.thezebra.com

Concetto	Tipo	Volume (milioni)	Motivazione
Utente	E	150	il report aziendale ha evidenziato che il servizio accoglie 150 milioni di utenti unici.
Ospite	E	150	Dato che gli utenti possono essere contemporaneamente sia ospiti che host.
Host	E	4	il report aziendale ha evidenziato che il servizio accoglie 4 milioni di alloggio.
Metodo di pagamento	E	270	Ipotizzo che la media di metodi di pagamento per utente sia 1.8.
Liste	E	2250	Ipotizzo che $\frac{3}{4}$ degli utenti ha delle liste e la media di lista/utente è 3 liste con 5 alloggi per lista. $150 \cdot 3 \cdot 5 = 2250$.
Alloggio	E	5.6	il report aziendale ha evidenziato che il servizio accoglie 5.6 milioni di alloggio.
Camera	E	14	Ipotizzo che la media camera/alloggio è 2,5.
Servizio	E	28	Ogni alloggio ha una lista di servizi, ipotizzo che la media di servizi/alloggio è 5.
Foto	E	44.8	Ogni alloggio ha delle foto, ipotizzo che la media delle foto per alloggio sia 8.
Prenotazione	E	140	il report aziendale ha evidenziato che il servizio accoglie 140 milioni di prenotazioni.
Prenotazione rifiutata	E	1,4	Ipotizzando che 1/100 prenotazioni non vengano confermate.
Prenotazione Confermata	E	138,6	Ipotizzando che 99/100 prenotazioni vengano confermate
Prenotazione cancellata	E	13,86	Ipotizzando che 10/100 prenotazioni vengano cancellate. Dato che le prenotazioni per essere cancellate prima devono essere confermate ho fatto il 10% su 138.6.
Prenotazione Conclusa	E	124.74	Le prenotazioni rimanenti sono da definirsi concluse. $140 - 1.4 - 13.86 = 124.74$
Prenotazione pendente	E	140	Ipotizzo che tutte le prenotazioni prima di essere accettate o rifiutate sono state pendenti.
Recensione	E	174.6	Ipotizzando che nel totale delle prenotazioni concluse il 70/100 degli utenti lascia una recensione. $124.74 \cdot 70\% \cdot 2 = 174.636$

Recensione Ospite	E	87,3	Ipotizzando che nel totale delle prenotazioni concluse il 70/100 degli ospiti lascia una recensione.
Recensione Host	E	87,3	Ipotizzando che nel totale delle prenotazioni concluse il 70/100 degli host lascia una recensione.
Commento	E	873	Ipotizzando che per una recensione commentata in media si scrivono 10 commenti, inoltre ipotizzo che solamente la meta delle recensioni hanno commenti. $174.6 * 10 * 0.5 = 873$
Affitta	A	4	Dato che ci sono 4 milioni alloggi.
Inserito nella lista	A	2250	Ipotizzo che $\frac{3}{4}$ degli utenti ha delle liste e la media di lista/utente è 3 liste con 5 alloggi per lista. $150 * 3 * 5 = 2250$.
Appartiene	A	2250	Ipotizzo che $\frac{3}{4}$ degli utenti ha delle liste e la media di lista/utente è 3 liste con 5 alloggi per lista. $150 * 3 * 5 = 2250$.
Composto	A	14	Ipotizzo che la media camera/alloggio è 2,5.
Offre	A	28	Ogni alloggio ha una lista di servizi , ipotizzo che la media di servizi/alloggio è 5.
Ha	A	44.8	Ogni alloggio ha delle foto, ipotizzo che la media delle foto per alloggio sia 8.
Viene prenotato	A	140	Conoscendo il numero di prenotazioni.
Effettua prenotazione	A	140	Conoscendo il numero di prenotazioni.
Scriva/ Riguarda	A	174.6	Ipotizzando che nel totale delle prenotazioni concluse il 70/100 degli utenti lascia una recensione. $124 * 70\% * 2 = 174.636$
Riguarda	A	174.6	Ipotizzando che nel totale delle prenotazioni concluse il 70/100 degli utenti lascia una recensione. $124 * 70\% * 2 = 174.636$
Scriva Commento	A	873	Ipotizzando che per una recensione commentata in media si scrivono 10 commenti, inoltre ipotizzo che solamente la meta delle recensioni hanno commenti. $174.6 * 10 * 0.5 = 873$
Commento riguarda	A	873	Ipotizzando che per una recensione commentata in media si scrivono 10 commenti, inoltre ipotizzo che solamente la meta delle recensioni hanno commenti. $174.6 * 10 * 0.5 = 873$

2.2 Tavola delle operazioni

Operazione	Tipo	Frequenza	Motivazione
1.Iscrizione nuovi utenti	I	20000/giorno	Operazione principale e di importanza primaria.
2.Prenotazione alloggi	I	400000/giorno	Operazione principale e di importanza primaria.
3.Recensione	I	238000/giorno	Operazione di media importanza, infatti ipotizzo che nelle 340000 prenotazioni che si consumano giornalmente il 30% degli utenti non lasciano una recensione.
4.Aggiornare il tasso di cancellazione di ciascun host	B	1 /settimana	Operazione importante definita dai requisiti iniziali
5.Controllo delle condizioni per la qualifica di superhost e viene aggiornato lo status degli host	B	1 /giorno	Operazione importante definita dai requisiti iniziali
6.Calcolo della classifica degli alloggi più graditi	B	1 /mese	Operazione importante definita dai requisiti iniziali
7.Aggiornamento Status Prenotazione	B	1/ora	Operazione importante per mantenere lo stato delle prenotazioni aggiornato
8.Aggiornare il numero di servizi di un alloggio	B	1/ora	Operazione necessaria

2.3 Ristrutturazione dello schema E-R

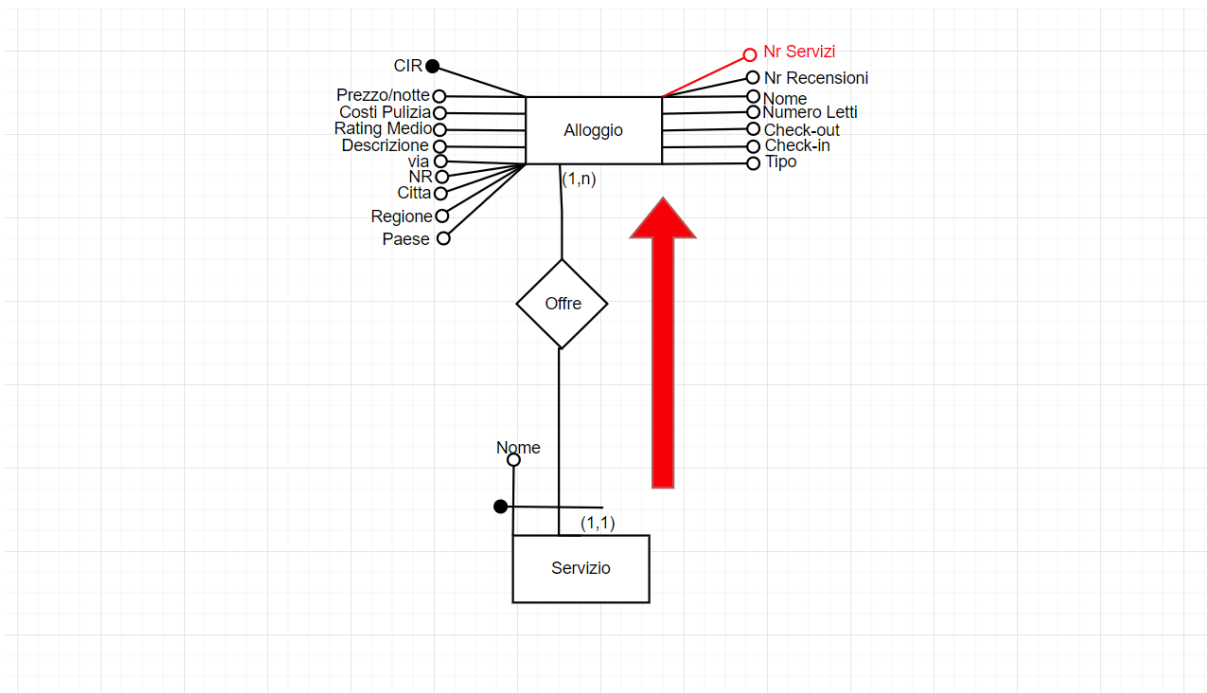
2.3.1 Analisi delle ridondanze

Ridondanze identificate nello schema E-R:

- Numero Servizi
 - Attributo presente in alloggio
Possibile calcolarlo:
-contando in numero delle occorrenze in servizi
- Nr Recensioni Alloggio
 - Presente in alloggio
Possibile calcolarlo:
-contando le occorrenze in recensioni
- Nr Recensioni Utenti
 - Attributo presente in utente
Possibile calcolarlo:
-contando le occorrenze in recensioni
- Nr Giorni
 - Attributo presente in Prenotazione
Possibile calcolarlo:
Data Fine Soggiorno - Data Inizio Soggiorno
- Valutazione Complessiva
 - Attributo presente in Recensione Host
Possibile calcolarlo:
media aritmetica dei voti
- Rating
 - Attributo presente in Utente
Possibile calcolarlo:
media aritmetica delle occorrenze delle valutazioni complessive

La ridondanza analizzata riguarda il numero dei servizi in alloggio

Schema di navigazione con ridondanza



Schema di navigazione senza Ridondanza

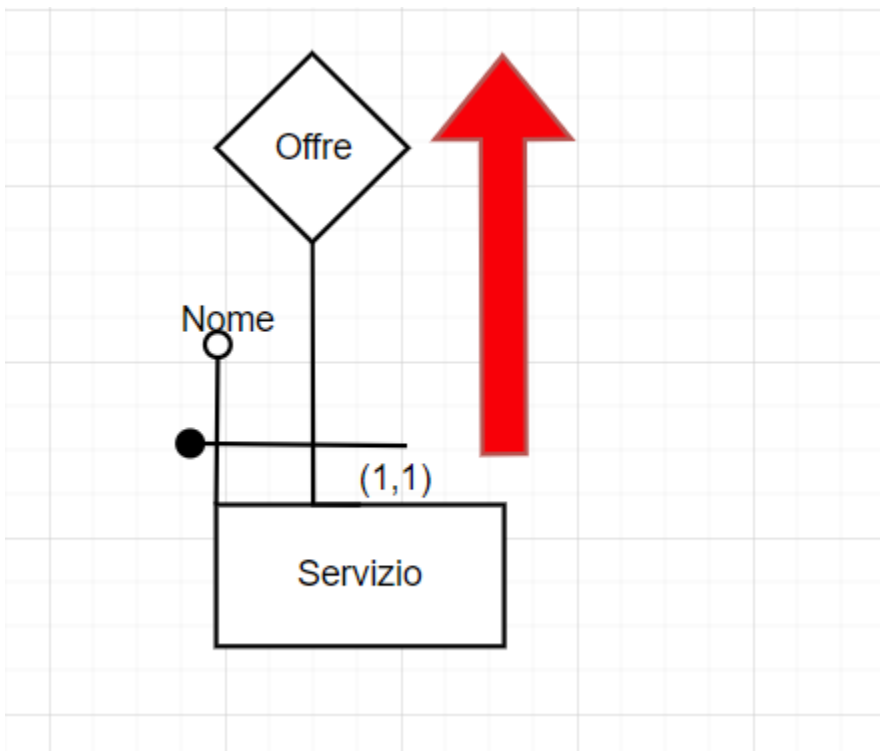


Tavola degli accessi

Contare il numero servizi per alloggio

CON RIDONDANZA

Concetto	Costrutto	Nr Accessi	Tipo
Alloggio	E	1	L

Operazione: conoscere il numero servizi per alloggio

SENZA RIDONDANZA

Concetto	Costrutto	Nr Accessi	Tipo
Servizio	E	1	S
Offre	A	1	S
Alloggio	E	1	L
Alloggio	E	1	S

CONFRONTO IN SPAZIO E TEMPO

Analisi spazio e tempo con ridondanza:

- Frequenza Operazione: 24 al giorno
- 1 accesso in lettura
- spazio attributo "ridondante": 2 byte (stima)

$24 \times 1 = 24$ accessi al giorno

Considerando di avere 5.600.000 alloggi, abbiamo in totale uno spazio occupato di: $5.600.000 \times 2 = 11.200.000$ byte cioè all'incirca 11.2 Megabyte

Analisi spazio e tempo **senza** ridondanza:

- Frequenza Operazione: 24 al giorno

- 1 accesso in lettura

- 3 accessi in scrittura

$24 \times 1 = 24$ accessi in lettura al giorno

$24 \times 3 = 72$ accessi in scrittura al giorno

Considerando gli accessi in scrittura il doppio degli accessi in lettura abbiamo:

$72 \times 2 = 144$ accessi in scrittura al giorno

per un totale di: $144 + 24 = 168$ accessi al giorno

Conclusione:

Eliminando la ridondanza risparmieremmo 11.2 megabyte ma avremmo circa 140 accessi al giorno in più al database.

Quindi possiamo concludere che, mantenere aggiornato il dato ridondante costa molto meno che ricalcolarlo.

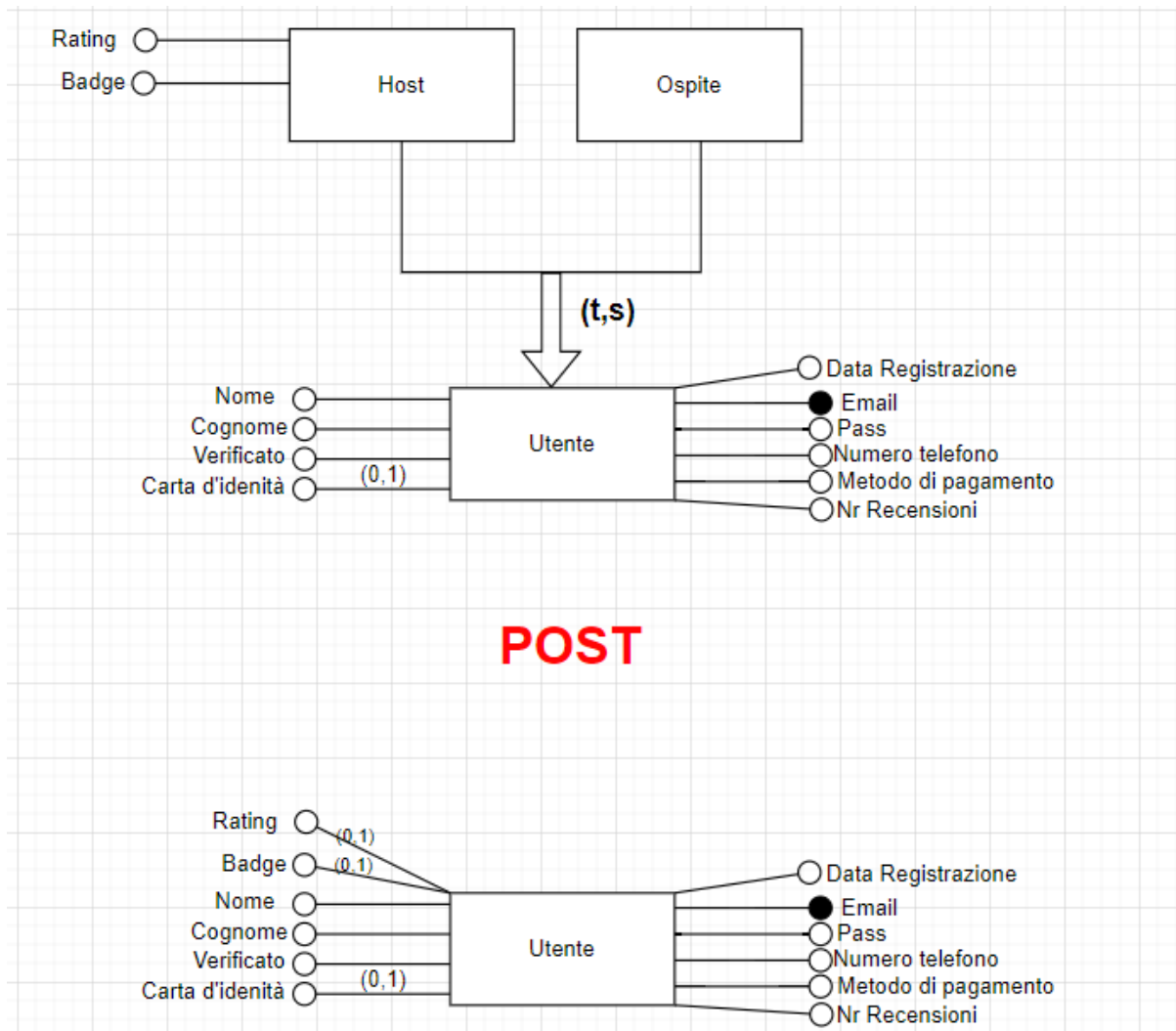
2.3.2 Eliminazione delle generalizzazioni

1) Generalizzazione :

Generalizzazione **totale e sovrapposta**.

Totale - non ci sono altri tipi di istanze per ogni utente oltre al host e ospite

Sovrapposta - un utente può essere sia ospite che host



Eliminazione della generalizzazione

Accorpamento delle entità figlie da parte del genitore

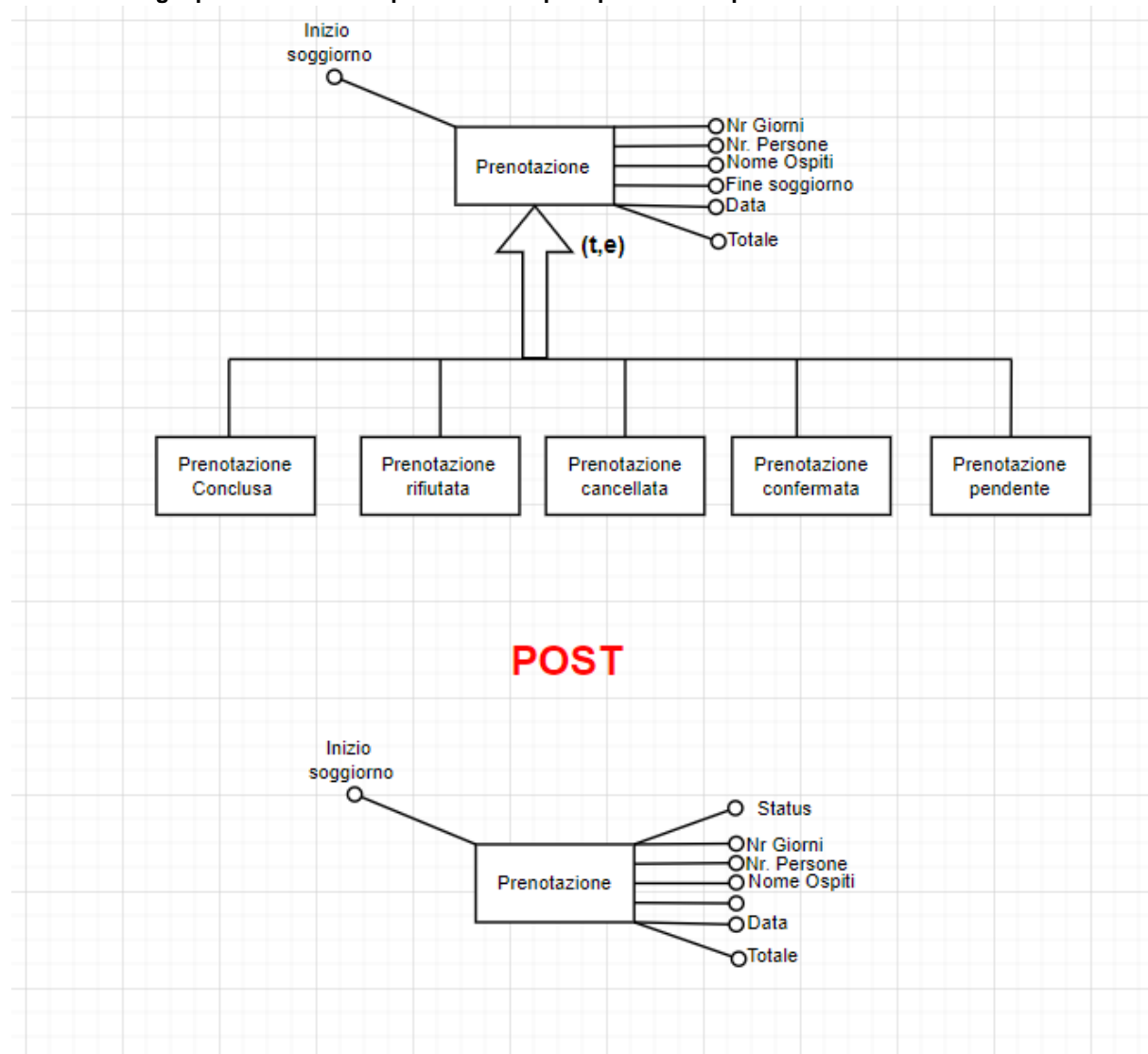
- Nell'entità genitore vengono definiti gli attributi: Rating e Badge
- La cardinalità presente nell'attributo Rating e Badge dichiara la presenza o meno del valore, di conseguenza si tratta dell'host, nella loro assenza l'opposto cioè l'ospite.
- Di conseguenza l'entità genitore acquisisce le relazioni delle figlie.

2) Generalizzazione :

Generalizzazione **totale ed esclusiva**.

Totale – non ci sono altri tipi di prenotazioni

Esclusiva – ogni prenotazione non può essere di più tipi in contemporanea



Eliminazione della generalizzazione

Accorpamento delle entità figlie nel genitore

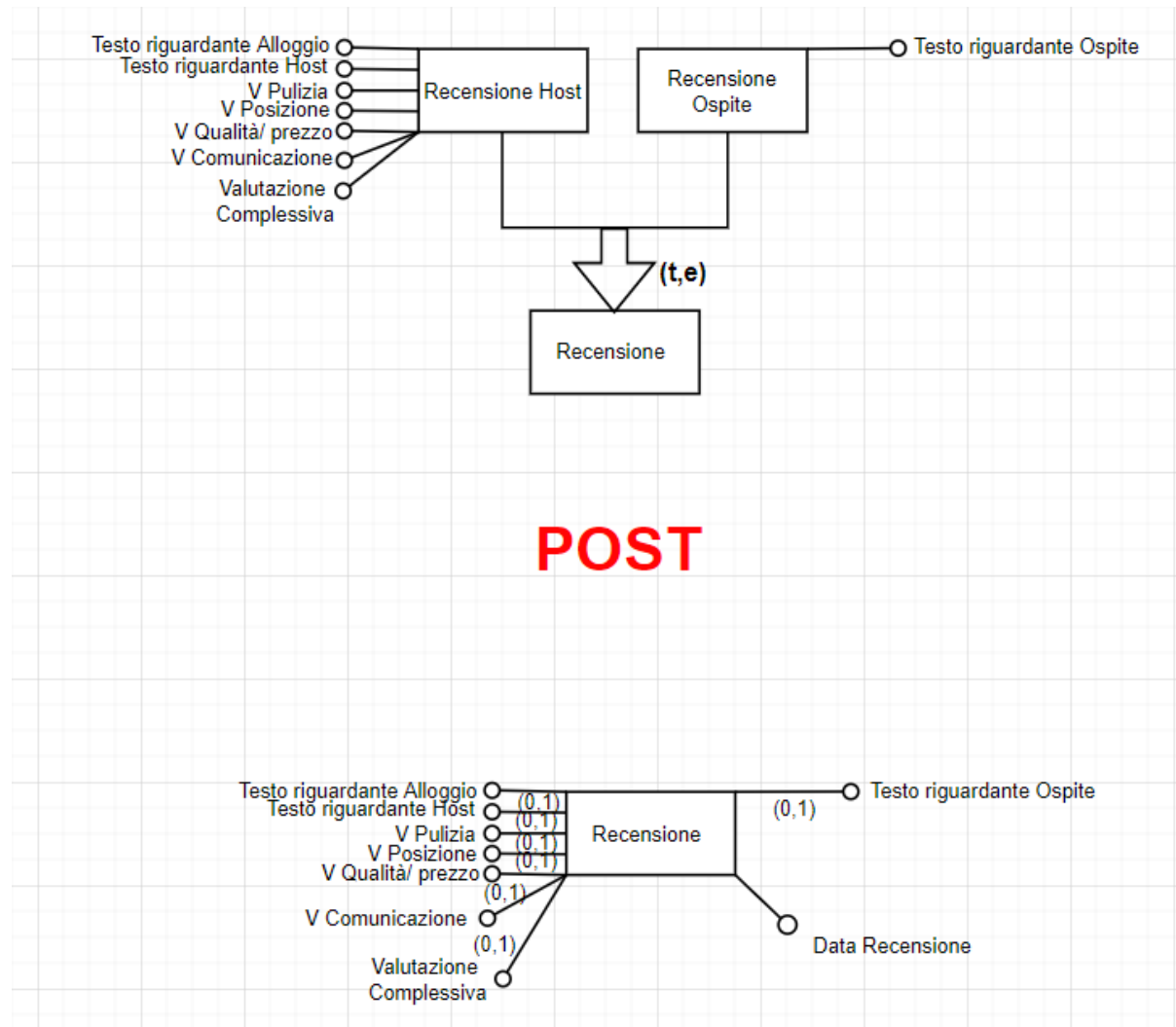
- Le entità figlie vengono eliminate.
- Nell'entità genitore viene definito un nuovo attributo: Status ,che dovrà obbligatoriamente essere NOT NULL
- Di conseguenza l'entità genitore acquisisce le relazioni delle figlie.

3) Generalizzazione :

Generalizzazione **totale ed esclusiva**.

Totale – non ci sono altri tipi di recensione

Esclusiva – ogni recensione non può essere di più tipi in contemporanea

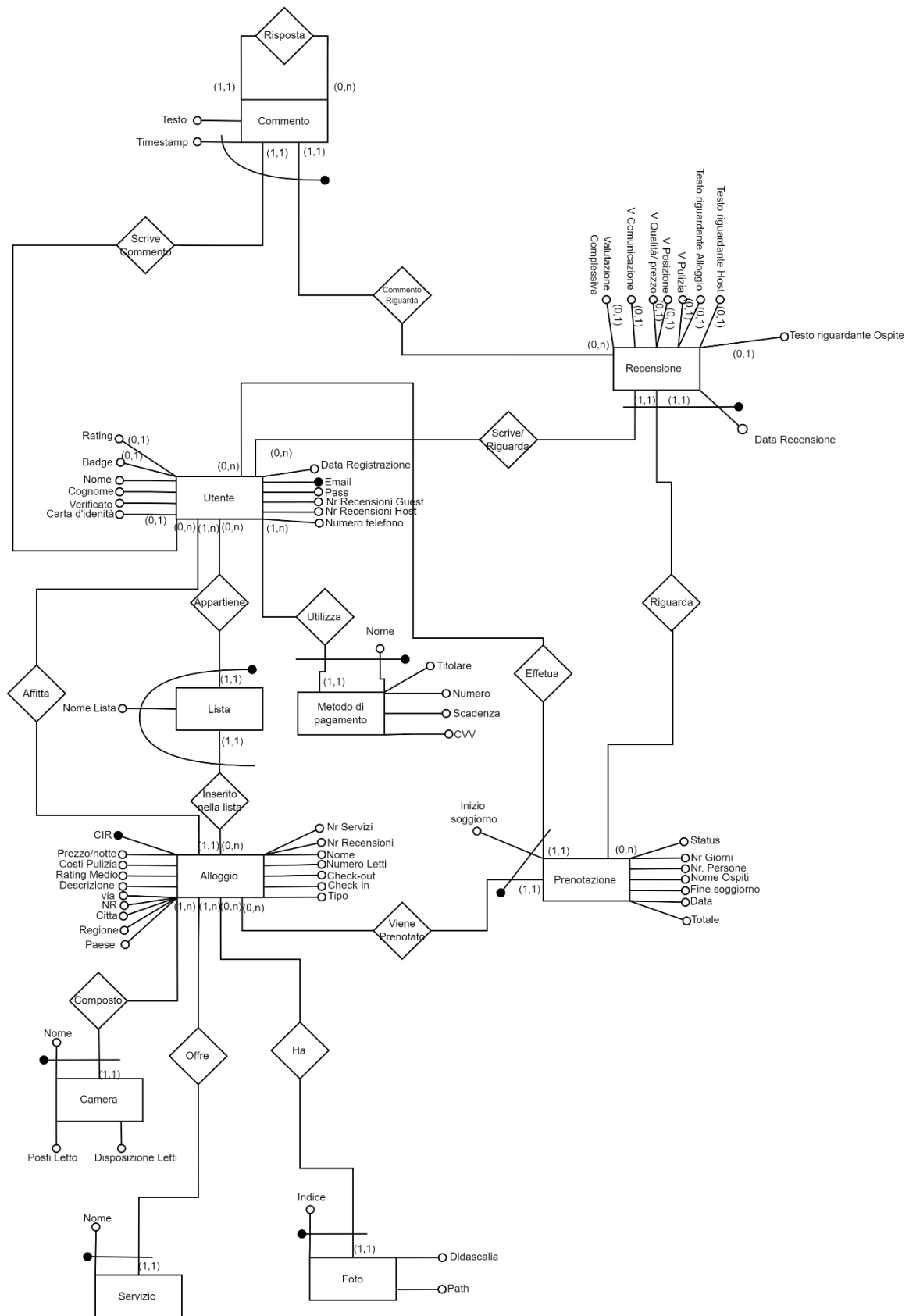


Eliminazione della generalizzazione

Accorpamento delle entità figlie nel genitore

- Le entità figlie vengono eliminate.
- Nell'entità genitore vengono definiti nuovi attributi: Testo Riguardante Host, Testo Riguardante Alloggio, V Pulizia, V Posizione, V Qualità/Prezzo, V Comunicazione, Valutazione Complessiva e Testo riguardante Ospite.
- La cardinalità presente nei nuovi attributi dichiara la presenza o meno del valore, di conseguenza si tratta della Recensione Ospite, quando l'attributo Testo riguardante Ospite non è NULL, nel caso opposto è un'istanza di tipo recensione host, cioè quando il set di attributi: Testo Riguardante Host, Testo Riguardante Alloggio, V Pulizia, V Posizione, V Qualità/Prezzo, V Comunicazione, Valutazione Complessiva non sono NULL.
- Di conseguenza l'entità genitore acquisisce le relazioni delle figlie.

2.4. Schema E-R ristrutturato + business rules



Business Rules

- In una prenotazione, per essere ritenuta valida, la data di inizio soggiorno deve essere strettamente minore alla data di fine soggiorno
- Gli utenti possono valutarsi a vicenda solo dopo la fine del soggiorno, quindi la data di fine soggiorno deve essere strettamente minore alla data di pubblicazione della recensione
- La risposta ad un commento non è d'obbligo.
- I voti si possono esprimere tra 1-5.
- Il numero di persone della prenotazione deve essere minore o uguale al numero massimo di persone che l'alloggio può ospitare
- Se una delle parti non scrive una contro recensione, quella della controparte diventa visibile decorsi 7 giorni.
- Il badge indica il tipo dell'host true = superhost false = normal host
- Se carta d'identità è null, verificato è false e viceversa
- I tipi degli alloggi sono : interi appartamenti, stanze private e posti letto in stanza comune.
- La cardinalità presente nei nuovi attributi dell'entità Recensione dichiara la presenza o meno del valore, di conseguenza si tratta della Recensione Ospite ,quando l'attributo Testo riguardante Ospite non è NULL, nel caso opposto è un'istanza di tipo recensione host, cioè quando il set di attributi: Testo Riguardante Host, Testo Riguardante Alloggio, V Pulizia, V Posizione, V Qualità/Prezzo, V Comunicazione, Valutazione Complessiva non sono NULL.
- I tipi degli alloggi sono : interi appartamenti, stanze private e posti letto in stanza comune.
- La cardinalità presente nell'attributo Rating e Badge dell'entità Utente dichiara la presenza o meno del valore, di conseguenza si tratta dell'host, nella loro assenza l'opposto cioè l'ospite.
- Nell'entità Prenotazione viene definito un nuovo attributo: Status ,che dovrà obbligatoriamente essere NOT NULL, e sono : Cancellata, Rifiutata, Conclusa, Confermata e Pendente.

Regole di derivazione

- Per diventare Superhost un host deve avere il rating(Hosts) maggiore o uguale a 4.8. Ed una percentuale cancellazione inferiore all'1% del totale di prenotazione andata a buon fine
- L'importo totale della prenotazione viene calcolato moltiplicando nr persone(prenotazione) per prezzo a persona(alloggio) e sommando a ciò i costi pulizia(alloggi).
- Il rating medio dell'alloggio viene calcolato facendo una media dei voti ricevuti dagli ospiti(Recensioni).
- Valutazione complessiva e la media dei voti ricevuti
- Il tasso di cancellazione viene calcolato facendo il rapporto tra il numero delle cancellazioni e delle prenotazioni concluse nel periodo degli ultimi 100 giorni da quello attuale.
- Il rating viene calcolato facendo la media dei voti di tutte le prenotazioni concluse in tutti i suoi alloggi.
- Il Nr giorni viene calcolato facendo la differenza tra la data di fine soggiorno e la data di inizio soggiorno.
- La media dei voti viene calcolata facendo la media aritmetica dei vari voti.
- **Utente.Nr recensioni** viene calcolato il numero effettivo di recensioni che lo riguardano
- **Alloggio.Nr recensioni** viene calcolato il numero effettivo di recensioni che lo riguardano
- Il Nr letti e la somma dei Posti letto di ogni camera di un alloggio
- Il totale prenotazione è calcolato facendo la moltiplicazione del prezzo/notte +costi pulizia.

2.5 Schema relazionale+vincoli di integrità referenziale

2.5.1 Schema relazionale

UTENTE(Email, Pass, Nome, Cognome, Numero telefono, Verificato, Carta d'identità*, Rating*, Badge*)

METODO DI PAGAMENTO(Nome, Email, Titolare, Numero, Scadenza, CVV)

ALLOGGIO(CIR, Via, NR, Città, Regione, Paese, Descrizione, Rating Media, Prezzo/notte, Costi pulizia, Nr Recensioni, Nome, Numero Letti, Check-in, Check-out, Email(AS Host), Tipo, Nr Servizi)

CAMERA(Nome, CIR, Posti Letto, Disposizione Letti)

SERVIZIO(NOME, CIR)

FOTO(Indice, CIR, Didascalia, Path)

LISTA(Nome Lista, Email, CIR)

PRENOTAZIONE(Inizio Soggiorno, CIR, Email(AS Email Ospite), Nr Giorni, Nr Persone, Fine Soggiorno, Status, Host, Nome Ospiti)

RECENSIONE(Data Recensione, Prenotazione.Inizio Soggiorno, Prenotazione CIR, Prenotazione.Email Ospite, UTENTE.Email(AS Email Autore Recensione), Prenotazione.Host, Testo riguardante Alloggio*, Testo riguardante Alloggio*, Voto Pulizia*, Voto Qualità/prezzo*, Voto Posizione*, Voto Comunicazione *, Valutazione Complessiva *)

COMMENTO(Timestamp, Recensione.Data Recensione, Recensione.Inizio Soggiorno, Recensione.CIR, Recensione.Email Ospite, Recensione.Autore Recensione, Testo)

2.5.2 Vincoli di integrità referenziale

Alloggio.Host **REFERENZIA** Utente.Email

Camera.CIR **REFERENZIA** Alloggio.CIR

Servizio.CIR **REFERENZIA** Alloggio.CIR

Lista.CIR **REFERENZIA** Alloggio.CIR

Lista.Email **REFERENZIA** Utente.Email

Prenotazione.Email Ospite **REFERENZIA** Utente.Email

Prenotazione.CIR **REFERENZIA** Alloggio.CIR

Prenotazione.Host **REFERENZIA** Alloggio.Host

Recensione.Email Autore Recensione **REFERENZIA** Utente.Email

Recensione.Inizio Soggiorno **REFERENZIA** Prenotazione.Inizio Soggiorno

Recensione.CIR **REFERENZIA** Prenotazione.CIR

Recensione.Email Ospite **REFERENZIA** Prenotazione.Email Ospite

Recensione.Host **REFERENZIA** Prenotazione.Host

Recensione.Status **REFERENZIA** Prenotazione.Status

Commento.Email Autore Commento **REFERENZIA** Utente.Email

Commento. Autore **REFERENZIA** Recensione.Email Autore Recensione

Commento. Inizio Soggiorno **REFERENZIA** Recensione.Inizio Soggiorno

Commento. CIR **REFERENZIA** Recensione.CIR

Commento. Email Ospite **REFERENZIA** Recensione.Email Ospite

Commento.Data Recensione **REFERENZIA** Recensione.Data Recensione

Foto.CIR **REFERENZIA** Alloggio.CIR

Metodo_di_Pagamento.Email **REFERENZIA** Utente.Email

3.1. DDL di creazione del database.

```
BEGIN TRANSACTION;
```

```
drop table if exists utente cascade;  
drop table if exists alloggio cascade;  
drop table if exists camera cascade;  
drop table if exists servizio cascade;  
drop table if exists lista cascade;  
drop table if exists prenotazione cascade;  
drop table if exists recensione cascade;  
drop table if exists commento cascade;  
drop table if exists metodo_di_pagamento cascade;  
drop table if exists foto cascade;
```

```
CREATE TABLE utente (  
    email varchar NOT NULL UNIQUE,  
    pass varchar NOT NULL,  
    nome varchar NOT NULL,  
    cognome varchar NOT NULL,  
    numero_telefono varchar NOT NULL,  
    verificato boolean NOT NULL DEFAULT false,  
    carta_didentita varchar,  
    rating float(1) ,  
    badge boolean ,
```

```
    CONSTRAINT utente_pk PRIMARY KEY (email)  
);
```

```
CREATE TABLE metodo_di_pagamento(  
    nome varchar NOT NULL,  
    Email varchar NOT NULL,  
    Titolare varchar not null,  
    numero varchar not null,  
    scadenza varchar not null,  
    cvv varchar not null,
```

```
    CONSTRAINT mp_pk PRIMARY KEY (nome,Email),  
    CONSTRAINT mpfk1 FOREIGN KEY (Email)  
    REFERENCES utente(email) ON UPDATE CASCADE ON DELETE CASCADE  
  
);
```

```
CREATE TABLE alloggio (  
    CIR varchar(14) NOT NULL UNIQUE ,  
    Nome varchar NOT NULL,  
    descrizione varchar NOT NULL,  
    host varchar NOT NULL REFERENCES utente(email) ON UPDATE CASCADE ON DELETE
```

```

CASCADE,
    via varchar NOT NULL,
    nr int NOT NULL,
    citta varchar NOT NULL,
    regione varchar NOT NULL,
    paese varchar NOT NULL,
    prezzo_notte int NOT NULL,
    Costi_pulizia int NOT NULL,
    numero_letti int NOT NULL,
    check_in time NOT NULL,
    check_out time NOT NULL,

    nr_servizi int not null,
    nr_recensioni int NOT NULL DEFAULT 0,
    Rating_media float(1) NOT NULL DEFAULT 0,

```

```

CONSTRAINT alloggi_pk PRIMARY KEY (cir),

```

```

CONSTRAINT alloggio_ref UNIQUE (cir,host)

```

```

);

```

```

CREATE TABLE camera(
    nome varchar NOT NULL,
    CIR varchar NOT NULL,
    Posti_Letto int NOT NULL,
    disposizione_letti varchar NOT NULL,

    CONSTRAINT camera_pk PRIMARY KEY (nome,CIR),
    CONSTRAINT camera_fk FOREIGN KEY (CIR)
    REFERENCES alloggio(CIR) ON UPDATE CASCADE ON DELETE CASCADE

```

```

);

```

```

CREATE TABLE servizio(
    nome varchar NOT NULL,
    CIR varchar NOT NULL,

    CONSTRAINT servizio_pk PRIMARY KEY (nome,CIR),
    CONSTRAINT servizio_fk FOREIGN KEY (CIR)
    REFERENCES alloggio(CIR) ON UPDATE CASCADE ON DELETE CASCADE

```

```

);

```

```

CREATE TABLE foto(
    Indice varchar not null,
    CIR varchar not null,
    didascalia varchar not null,

```

path varchar not null,

CONSTRAINT foto_pl PRIMARY KEY (Indice,CIR),
CONSTRAINT foto1 FOREIGN KEY (CIR)
REFERENCES alloggio(CIR) ON UPDATE CASCADE ON DELETE CASCADE

);

CREATE TABLE lista (

Nome_Lista varchar NOT NULL ,
proprietario varchar NOT NULL,
alloggio varchar NOT NULL,

CONSTRAINT liste_pk PRIMARY KEY (Nome_Lista,proprietario,alloggio),
CONSTRAINT liste_fk1 FOREIGN KEY (proprietario)
REFERENCES utente(email) ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT liste_fk2 FOREIGN KEY (alloggio)
REFERENCES alloggio(cir) ON UPDATE CASCADE ON DELETE CASCADE

);

CREATE TABLE prenotazione (

inizio_soggiorno date NOT NULL ,
fine_soggiorno date NOT NULL,
CIR varchar NOT NULL ,
Email_ospite varchar NOT NULL REFERENCES utente(email) ON UPDATE CASCADE ON DELETE

CASCADE,

host varchar NOT NULL ,
nr_persone int NOT NULL,
nr_giorni int NOT NULL,

status varchar NOT NULL ,

data_p date not null,
totale varchar not null,
nomi_ospiti varchar ,

CONSTRAINT pren_pk PRIMARY KEY(inizio_soggiorno,CIR,Email_ospite),

CONSTRAINT pren_fk FOREIGN KEY(CIR,host)
REFERENCES alloggio(cir,host) ON UPDATE CASCADE ON DELETE CASCADE,
CONSTRAINT pren_fk1 FOREIGN KEY(Email_ospite)
REFERENCES utente(email) ON UPDATE CASCADE ON DELETE CASCADE,

CONSTRAINT ref_recensione UNIQUE(inizio_soggiorno,CIR,Email_ospite,host,status)

);

```

CREATE TABLE recensione (
    data_recensione date NOT NULL,
    inizio_soggiorno date NOT NULL ,
    CIR varchar NOT NULL,
    Email_ospite varchar NOT NULL ,
    host varchar NOT NULL,
    status varchar NOT NULL,
    email_autore_recensione varchar NOT NULL,
    testo_riguardante_ospite varchar ,
    testo_riguardante_host varchar ,
    testo_riguardante_alloggio varchar ,
    voto_pulizia int,
    voto_qualita_prezzo int,
    voto_posizione int,

    CONSTRAINT rec_pk PRIMARY KEY
(data_recensione,inizio_soggiorno,CIR,Email_ospite,email_autore_recensione),
    CONSTRAINT rec_fk FOREIGN KEY (inizio_soggiorno,CIR,Email_ospite,host,status)
REFERENCES prenotazione(inizio_soggiorno,CIR,Email_ospite,host,status) ON UPDATE CASCADE
ON DELETE CASCADE,
    CONSTRAINT rec_fk1 FOREIGN KEY(email_autore_recensione)
REFERENCES utente(email) ON UPDATE CASCADE ON DELETE CASCADE,

    CONSTRAINT rec_ref UNIQUE
(data_recensione,inizio_soggiorno,CIR,Email_ospite,email_autore_recensione)

);

```

```

CREATE TABLE commento (
    timestamp_timestamp NOT NULL,
    data_recensione date NOT NULL,
    inizio_soggiorno date NOT NULL,
    CIR varchar NOT NULL,
    Email_Ospite varchar NOT NULL,
    email_autore_recensione varchar NOT NULL,

    email_autore_commento varchar NOT NULL,
    Testo varchar NOT NULL,

    timestamp_f timestamp ,
    data_recensione_f date ,
    inizio_soggiorno_f date ,
    CIR_f varchar ,
    Email_Ospite_f varchar ,
    email_autore_recensione_F varchar,

    CONSTRAINT comm_pk PRIMARY KEY
(timestamp_,data_recensione,inizio_soggiorno,CIR,Email_ospite,email_autore_recensione),

```

```
        CONSTRAINT comm_fk FOREIGN KEY
(data_recensione,inizio_soggiorno,CIR,Email_ospite,email_autore_recensione)
REFERENCES
recensione(data_recensione,inizio_soggiorno,CIR,Email_ospite,email_autore_recensione)
ON UPDATE CASCADE ON DELETE CASCADE,
        CONSTRAINT comm_father_reference FOREIGN KEY
(timestamp_f,inizio_soggiorno_f,CIR_f,Email_ospite_f,email_autore_recensione_f,data_recensione_f)
REFERENCES
commento(timestamp_,inizio_soggiorno,CIR,Email_ospite,email_autore_recensione,data_recensione)
ON UPDATE CASCADE ON DELETE CASCADE,
```

```
        CONSTRAINT comm1_fk FOREIGN KEY (email_autore_commento)
REFERENCES utente(email) ON UPDATE CASCADE ON DELETE CASCADE,
        CONSTRAINT comm_father_ref UNIQUE
(timestamp_,inizio_soggiorno,CIR,Email_ospite,email_autore_recensione,data_recensione_f)
```

```
);
```

```
END TRANSACTION;
```

3.2. DML di popolamento di tutte le tabelle del database.

BEGIN TRANSACTION;

```
insert into utente
values('olly@mail.com','Alfa1234','oliviù','gratii','3477021536','true','c:/img/user/olly_mail_com.jpeg',0,false);
insert into utente
values('kevin@mail.com','Beta1234','kevin','doga','33333333','false','c:/img/user/kevin_mail_com.jpeg',0,false);
insert into utente
values('ale@mail.com','Alex1234','alex','rider','3334445555','true','c:/img/user/ale_mail_com.jpeg');
insert into utente values('jack@mail.com','Jack1234','jack','daniels','3334456555','false');
insert into utente
values('jhony@mail.com','Jhony1234','jhony','deep','3337745555','true','c:/img/user/jhony_mail_com.jpeg');
insert into utente values('alan@mail.com','Alan1234','alan','turing','333333545','false');
```

```
insert into metodo_di_pagamento values('visa1','olly@mail.com','Oliviu G.','444433322221111','22/10',123);
```

```
insert into alloggio values ('aaaaaaa000000','Villa delle
rose','xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx','olly@mail.com','delle rose',1,'porto
cervo','sardegna','Italia',30,50,7,'14:00','11:00',5);
insert into alloggio values ('aaaaaaa000001','Villa delle
palme','xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx','olly@mail.com','delle palme',2,'porto
cervo','sardegna','Italia',40,100,10,'14:00','11:00',5);
insert into alloggio values ('aaaaaaa000002','blue bay
apartment','xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx','olly@mail.com','Smeraldina',3,'porto
cervo','sardegna','Italia',30,20,3,'14:00','11:00',5);
insert into alloggio values ('aaaaaaa000003','Diamond Hostel
','xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx','kevin@mail.com','Diamante ',4,'porto
cervo','sardegna','Italia',15,5,6,'14:00','11:00',5);
insert into alloggio values ('aaaaaaa000004','Sweet
Loft','xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx','kevin@mail.com','Moscova',5,'porto
cervo','sardegna','Italia',100,50,2,'14:00','11:00',5);
```

```
insert into camera values('camera 1','aaaaaaa000000',2,'un letto matrimoniale');
insert into camera values('camera 2','aaaaaaa000000',2,'un letto matrimoniale');
insert into camera values('camera 3','aaaaaaa000000',3,'un letto matrimoniale+ divano letto');
```

```
insert into camera values('camera 1','aaaaaaa000001',3,'un letto matrimoniale+ divano letto');
insert into camera values('camera 2','aaaaaaa000001',3,'un letto matrimoniale+ divano letto');
insert into camera values('camera 3','aaaaaaa000001',4,'due letti matrimoniali');
```

```
insert into camera values('camera 1','aaaaaaa000002',3,'un letto matrimoniale+ divano letto');
```

```
insert into camera values('camera 1','aaaaaaa000003',3,'un letto matrimoniale+ divano letto');
insert into camera values('camera 2','aaaaaaa000003',3,'un letto matrimoniale+ divano letto');
```

```
insert into camera values('camera 1','aaaaaaa000004',3,'un letto matrimoniale+ divano letto');
```

/*5 servizi per alloggio*/

```
insert into servizio values ('Wifi','aaaaaaa000000');
insert into servizio values ('AC','aaaaaaa000000');
insert into servizio values ('Cucina','aaaaaaa000000');
insert into servizio values ('Parcheggio','aaaaaaa000000');
```

```

insert into servizio values ('Animali domestici ammessi','aaaaaaa000000');

insert into servizio values ('Wifi','aaaaaaa000001');
insert into servizio values ('AC','aaaaaaa000001');
insert into servizio values ('Cucina','aaaaaaa000001');
insert into servizio values ('Parcheggio','aaaaaaa000001');
insert into servizio values ('Animali domestici ammessi','aaaaaaa000001');

insert into servizio values ('Wifi','aaaaaaa000002');
insert into servizio values ('AC','aaaaaaa000002');
insert into servizio values ('Cucina','aaaaaaa000002');
insert into servizio values ('Parcheggio','aaaaaaa000002');
insert into servizio values ('Animali domestici ammessi','aaaaaaa000002');

insert into servizio values ('Wifi','aaaaaaa000003');
insert into servizio values ('AC','aaaaaaa000003');
insert into servizio values ('Cucina','aaaaaaa000003');
insert into servizio values ('Parcheggio','aaaaaaa000003');
insert into servizio values ('Animali domestici ammessi','aaaaaaa000003');

insert into servizio values ('Wifi','aaaaaaa000004');
insert into servizio values ('AC','aaaaaaa000004');
insert into servizio values ('Cucina','aaaaaaa000004');
insert into servizio values ('Parcheggio','aaaaaaa000004');
insert into servizio values ('Animali domestici ammessi','aaaaaaa000004');


insert into lista values ('vacanze','olly@mail.com','aaaaaaa000004');
insert into lista values ('famiglia','olly@mail.com','aaaaaaa000001');
insert into lista values ('lavoro','olly@mail.com','aaaaaaa000002');
insert into lista values ('piacere','olly@mail.com','aaaaaaa000000');

insert into lista values ('piacere','kevin@mail.com','aaaaaaa000003');
insert into lista values ('famiglia','kevin@mail.com','aaaaaaa000002');
insert into lista values ('piacere','kevin@mail.com','aaaaaaa000000');
insert into lista values ('famiglia','kevin@mail.com','aaaaaaa000001');

insert into lista values ('piacere','ale@mail.com','aaaaaaa000004');
insert into lista values ('vacanze','ale@mail.com','aaaaaaa000001');
insert into lista values ('vacanze','ale@mail.com','aaaaaaa000003');
insert into lista values ('famiglia','ale@mail.com','aaaaaaa000000');

insert into lista values ('lavoro','jack@mail.com','aaaaaaa000000');
insert into lista values ('lavoro','jack@mail.com','aaaaaaa000002');
insert into lista values ('lavoro','jack@mail.com','aaaaaaa000004');
insert into lista values ('piacere','jack@mail.com','aaaaaaa000003');

insert into lista values ('piacere','jhony@mail.com','aaaaaaa000004');
insert into lista values ('famiglia','jhony@mail.com','aaaaaaa000002');
insert into lista values ('piacere','jhony@mail.com','aaaaaaa000001');
insert into lista values ('famiglia','jhony@mail.com','aaaaaaa000003');

insert into lista values ('piacere','alan@mail.com','aaaaaaa000004');
insert into lista values ('vacanze','alan@mail.com','aaaaaaa000003');
insert into lista values ('piacere','alan@mail.com','aaaaaaa000001');
insert into lista values ('vacanze','alan@mail.com','aaaaaaa000000');

```

```
insert into prenotazione values('2021-10-01','2021-10-06','aaaaaaa000000','alan@mail.com','olly@mail.com',7,5,'conclusa','2021-09-09','280');
insert into prenotazione values('2021-10-09','2021-10-24','aaaaaaa000000','jhony@mail.com','olly@mail.com',6,15,'conclusa','2021-09-09','280');
insert into prenotazione values('2021-10-26','2021-10-30','aaaaaaa000000','jack@mail.com','olly@mail.com',7,4,'confermata','2021-09-09','280');
```

```
insert into prenotazione values('2021-11-01','2021-11-06','aaaaaaa000001','alan@mail.com','olly@mail.com',10,5,'conclusa','2021-09-09','280');
insert into prenotazione values('2021-11-09','2021-11-24','aaaaaaa000001','ale@mail.com','olly@mail.com',8,15,'confermata','2021-09-09','280');
insert into prenotazione values('2021-11-26','2021-10-30','aaaaaaa000001','jack@mail.com','olly@mail.com',9,4,'conclusa','2021-09-09','280');
```

```
insert into prenotazione values('2021-12-01','2021-12-06','aaaaaaa000002','alan@mail.com','olly@mail.com',3,5,'conclusa','2021-09-09','280');
insert into prenotazione values('2021-12-09','2021-12-24','aaaaaaa000002','kevin@mail.com','olly@mail.com',1,15,'cancellata','2021-09-09','280');
insert into prenotazione values('2021-12-26','2021-12-30','aaaaaaa000002','ale@mail.com','olly@mail.com',2,4,'conclusa','2021-09-09','280');
```

```
insert into prenotazione values('2022-01-01','2022-01-06','aaaaaaa000003','jack@mail.com','kevin@mail.com',6,5,'conclusa','2021-09-09','280');
insert into prenotazione values('2022-01-09','2022-01-24','aaaaaaa000003','jhony@mail.com','kevin@mail.com',3,15,'rifiutata','2021-09-09','280');
insert into prenotazione values('2022-01-26','2022-01-30','aaaaaaa000003','ale@mail.com','kevin@mail.com',5,4,'conclusa','2021-09-09','280');
```

```
insert into prenotazione values('2022-02-01','2022-02-06','aaaaaaa000004','olly@mail.com','kevin@mail.com',2,5,'rifiutata','2021-09-09','280');
insert into prenotazione values('2022-02-09','2022-02-24','aaaaaaa000004','alan@mail.com','kevin@mail.com',2,15,'conclusa','2021-09-09','280');
insert into prenotazione values('2022-02-26','2022-03-02','aaaaaaa000004','ale@mail.com','kevin@mail.com',2,4,'conclusa','2021-09-09','280');
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_ospite)
values('2021-10-07','2021-10-01','aaaaaaa000000','alan@mail.com','olly@mail.com','conclusa','olly@mail.com','ospite gentile');
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_ospite)
values('2021-10-25','2021-10-09','aaaaaaa000000','jhony@mail.com','olly@mail.com','conclusa','olly@mail.com','ospite gentile');
```



```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_ospite)
values('2021-12-07','2021-12-
01','aaaaaaa000002','alan@mail.com','olly@mail.com','conclusa','olly@mail.com','ospite gentile');
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_ospite)
values('2022-01-07','2022-01-
01','aaaaaaa000003','jack@mail.com','kevin@mail.com','conclusa','kevin@mail.com','ospite gentile');
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_ospite)
values('2022-02-01','2022-01-
26','aaaaaaa000003','ale@mail.com','kevin@mail.com','conclusa','kevin@mail.com','ospite gentile');
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_ospite)
values('2022-02-25','2022-02-
09','aaaaaaa000004','alan@mail.com','kevin@mail.com','conclusa','kevin@mail.com','ospite gentile');
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_ospite)
values('2022-03-23','2022-02-
26','aaaaaaa000004','ale@mail.com','kevin@mail.com','conclusa','kevin@mail.com','ospite gentile');
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_host,testo_riguardante_alloggio,voto_pulizia,voto_qualita_prezzo,voto_posizione)
values('2021-10-07','2021-10-
01','aaaaaaa000000','alan@mail.com','olly@mail.com','conclusa','alan@mail.com','host gentile','alloggio
accogliente',4,5,4);
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_host,testo_riguardante_alloggio,voto_pulizia,voto_qualita_prezzo,voto_posizione)
values('2021-10-25','2021-10-
09','aaaaaaa000000','jhony@mail.com','olly@mail.com','conclusa','jhony@mail.com','host gentile','alloggio
accogliente',4,5,4);
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_host,testo_riguardante_alloggio,voto_pulizia,voto_qualita_prezzo,voto_posizione)
values('2021-12-07','2021-12-
01','aaaaaaa000002','alan@mail.com','olly@mail.com','conclusa','alan@mail.com','host gentile','alloggio
accogliente',4,5,4);
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_host,testo_riguardante_alloggio,voto_pulizia,voto_qualita_prezzo,voto_posizione)
```

```
values('2022-01-07','2022-01-01','aaaaaaa000003','jack@mail.com','kevin@mail.com','conclusa','jack@mail.com','host gentile','alloggio accogliente',4,5,4);
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_host,testo_riguardante_alloggio,voto_pulizia,voto_qualita_prezzo,voto_posizione)
values('2022-02-01','2022-01-26','aaaaaaa000003','ale@mail.com','kevin@mail.com','conclusa','ale@mail.com','host gentile','alloggio accogliente',4,5,4);
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_host,testo_riguardante_alloggio,voto_pulizia,voto_qualita_prezzo,voto_posizione)
values('2022-02-25','2022-02-09','aaaaaaa000004','alan@mail.com','kevin@mail.com','conclusa','alan@mail.com','host gentile','alloggio accogliente',4,5,4);
```

```
insert into
recensione(data_recensione,inizio_soggiorno,cir,email_ospite,host,status,email_autore_recensione,testo_riguard
ante_host,testo_riguardante_alloggio,voto_pulizia,voto_qualita_prezzo,voto_posizione)
values('2022-03-23','2022-02-26','aaaaaaa000004','ale@mail.com','kevin@mail.com','conclusa','ale@mail.com','host gentile','alloggio accogliente',4,5,4);
```

```
insert into commento
values('2021-10-07 10:14:07','2021-10-07','2021-10-01','aaaaaaa000000','alan@mail.com','olly@mail.com','olly@mail.com','graziemille');
insert into commento
values('2021-10-07 11:14:07','2021-10-07','2021-10-01','aaaaaaa000000','alan@mail.com','olly@mail.com','alan@mail.com','prego','2021-10-07 10:14:07','2021-10-07','2021-10-01','aaaaaaa000000','alan@mail.com','olly@mail.com');
insert into commento
values('2021-10-07 12:14:07','2021-10-07','2021-10-01','aaaaaaa000000','alan@mail.com','olly@mail.com','olly@mail.com','La riaspettiamo nella nostra struttura','2021-10-07 11:14:07','2021-10-07','2021-10-01','aaaaaaa000000','alan@mail.com','olly@mail.com');
```

```
END TRANSACTION;
```

3.3. Qualche operazione di cancellazione e modifica per verificare i vincoli e gli effetti causati da operazioni su chiavi esterne.

```
UPDATE utente
SET email = 'oliviu@gmail.com'
WHERE email='olly@mail.com';
```

```
DELETE FROM utente
WHERE email='oliviu@gmail.com';
```

