

Laboratorio Basi di Dati 2021/22

Progetto di piattaforma di home booking

Progetto svolto da:

Samuele Perrotta

Matricola 943460

Anita Scanu

Matricola 951947

Stefano Pittavino

Matricola 948864

Schema guida

1. [Progettazione concettuale](#)

1.1 [Requisiti iniziali](#)

1.2 [Glossario dei termini](#)

1.3 [Requisiti rivisti e strutturati in gruppi di frasi omogenee](#)

1.3.1 [Ristrutturazione dei requisiti](#)

1.3.2 [Requisiti ristrutturati](#)

1.3.3 [Requisiti strutturati in gruppi di frasi omogenee](#)

1.3.4 [Specifiche sulle operazioni](#)

1.4 [Schema E-R + business rules](#)

1.4.1 [Schema E-R](#)

1.4.2 [Dizionario dei dati \(entità\)](#)

1.4.3 [Dizionario dei dati \(relazioni\)](#)

1.4.4 [Vincoli d'integrità](#)

1.4.5 [Derivazioni](#)

2. [Progettazione logica](#)

2.1 [Tavola dei volumi](#)

2.2 [Tavola delle operazioni](#)

2.3 [Ristrutturazione dello schema E-R](#)

2.3.1 [Analisi delle ridondanze](#)

2.3.2 [Eliminazione delle generalizzazioni](#)

2.3.3 [Eventuale partizionamento/accorpamento di entità e associazioni](#)

2.3.4 [Eventuale scelta degli identificatori principali](#)

2.4 [Schema E-R ristrutturato + business rules](#)

2.4.1 [Schema E-R ristrutturato](#)

2.4.2 [Dizionario dei dati \(entità\)](#)

2.4.3 [Dizionario dei dati \(relazioni\)](#)

2.4.4 [Vincoli d'integrità](#)

2.4.5 [Derivazioni](#)

2.5 [Schema relazionale + vincoli di integrità referenziale](#)

2.5.1 [Schema relazionale](#)

2.5.2 [Vincoli di integrità referenziale](#)

3. [Implementazione](#)

3.1 [DDL di creazione del database](#)

3.2 [DML di popolamento di tutte le tabelle del database](#)

3.3 [Qualche operazione di cancellazione e modifica per verificare i vincoli e gli effetti causati da operazioni su chiavi esterne](#)

3.3.1 [Operazioni di cancellazione](#)

3.3.2 [Operazioni di modifica](#)

1. Progettazione concettuale

1.1 Requisiti iniziali

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. 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**.

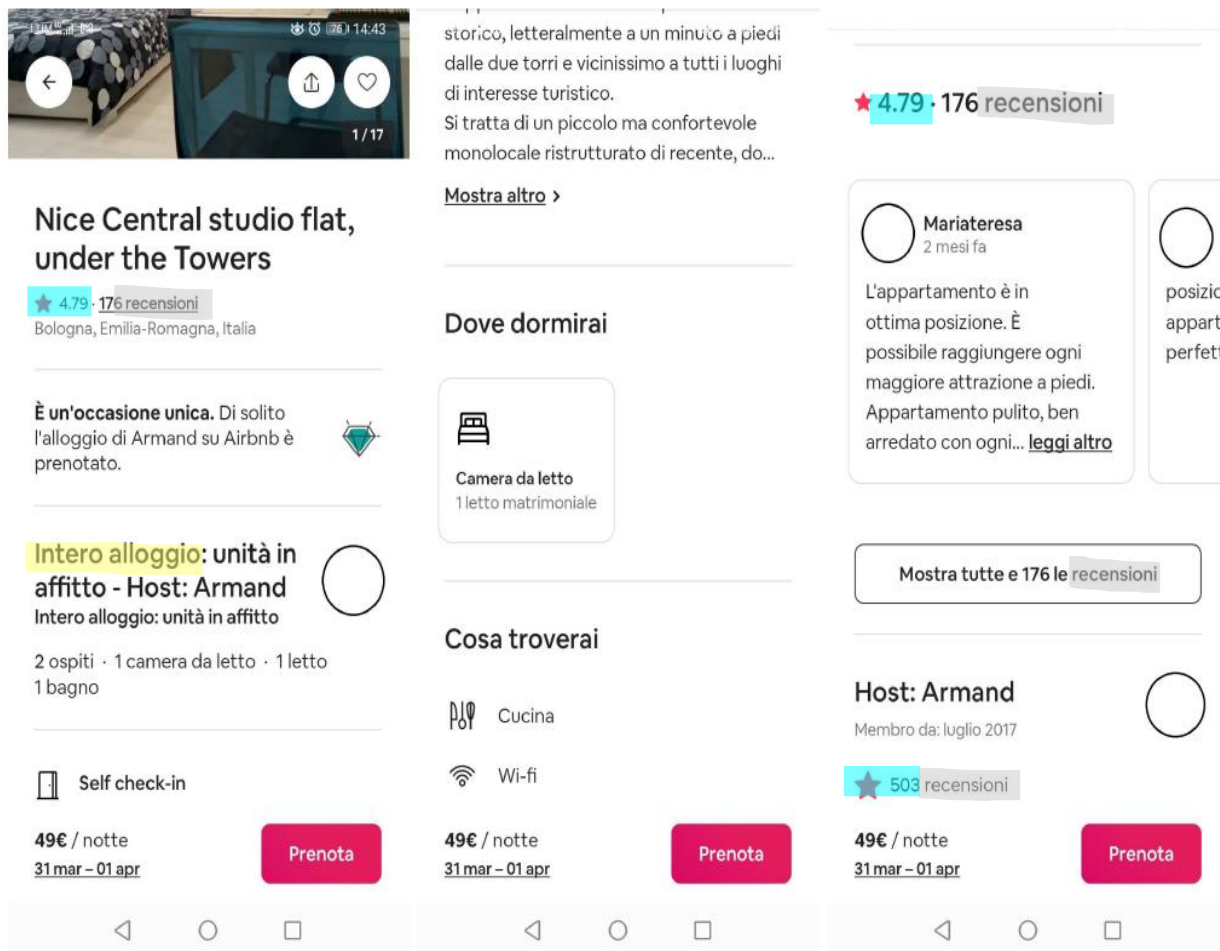


Figura 1: Dettagli di un **appartamento**

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

| Termine | Descrizione | Sinonimi | Collegamenti |
|----------------------------|--|----------------------|--|
| Alloggio | Posto in cui soggiornare (interi appartamenti, stanze private o condivise) | Casa, appartamento | Prenotazione, soggiorno, recensione, utente, intero appartamento, stanza privata, stanza condivisa |
| Intero appartamento | <u>Tipo particolare di alloggio</u> Camera e spazio privati | Alloggio | Alloggio |
| Stanza privata | <u>Tipo particolare di alloggio</u> Camera privata e spazio condiviso | Alloggio | Alloggio |
| Stanza condivisa | <u>Tipo particolare di alloggio</u> Camera e spazio condivisi | Alloggio | Alloggio |
| Utente | Individuo registrato al servizio | Persona | Ospite, host, superhost, alloggio, recensione, valutazione, prenotazione, soggiorno |
| Ospite | <u>Caso particolare di utente</u> Individuo che viene ospitato | Utente | Utente, recensione, valutazione, prenotazione, soggiorno, alloggio |
| Host | <u>Caso particolare di utente</u> Individuo che ospita | Utente | Utente, recensione, alloggio |
| Superhost | <u>Caso particolare di host</u> | Utente, host, utente | Utente, recensione, |

| | | | |
|--------------|--|-------------------|---|
| | Host con determinati requisiti | superhost | alloggio |
| Recensione | Commento testuale inteso a giudicare un alloggio o un utente | Review | Utente, ospite, host, superhost, alloggio |
| Valutazione | Serie di punteggi su una scala da 1 a 5 riguardanti alcune dimensioni | Rating, punteggio | Utente, ospite, host, alloggio |
| Prenotazione | Procedura per riservare un determinato alloggio per effettuarci un soggiorno | | Utente, ospite, alloggio |
| Soggiorno | Permanenza temporanea in un luogo | | Utente, ospite, alloggio |

1.3 Requisiti rivisti e strutturati in gruppi di frasi omogenee

1.3.1 Ristrutturazione dei requisiti

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).

Fasi dell'analisi dei requisiti

- Individuare sinonimi/omonimi e unificare i termini
- Rendere esplicito il riferimento tra termini
- Scegliere il corretto livello di astrazione
- Standardizzare la struttura delle frasi

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

- Devono aver completato almeno 10 soggiorni, per un totale di almeno 100 notti.
- Devono aver conservato un tasso una percentuale 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 **in tutti i suoi alloggi**.

Gli utenti superhost **I superhost** ricevono un badge sul loro profilo.

Per gli alloggi sono descritti indicando **rappresentiamo** un il nome, l'indirizzo (visibile all'ospite solo quando **nel momento in cui** la prenotazione è confermata, altrimenti è visibile solo il comune **dell'alloggio**), una descrizione, il prezzo per notte per persona **ospite** 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 **l'host** a cui appartiene **l'alloggio**, il rating medio **la valutazione media** e il numero di recensioni.

Per gli utenti **si possono rappresentare** ~~possono aggiungere alcune case~~ **alcuni alloggi** tra i preferiti. Gli utenti ~~possono avere diverse~~ **e alcune** liste, ad esempio in base al viaggio che vogliono compiere.

~~Quando~~ Gli utenti possono prenotare degli alloggi di qualsiasi tipo. **Per le prenotazioni gli utenti devono indicare** ~~indicando~~ un intervallo di date per il soggiorno e il numero degli ospiti **per cui si vuole prenotare**, e se gli ospiti **per cui si vuole prenotare** sono a loro volta utenti del servizio, **si possono rappresentare** ~~se ne possono indicare i nominativi~~ i loro **nomi e cognomi**. La prenotazione deve essere confermata o rifiutata dall'host. **Per le prenotazioni rappresentiamo** ~~il ha un~~ costo totale. ~~e se~~ **Quando la prenotazione viene** confermata **dall'host** viene eseguito il pagamento **da parte dell'ospite**. ~~Inoltre,~~ La prenotazione può essere cancellata sia dall'ospite che dall'host.

Al termine del soggiorno, gli ospiti e gli host si possono valutare a vicenda. La recensione fatta dagli ospiti ~~comprende~~ **è composta da** due testi (uno per l'appartamento **l'alloggio** e uno per l'host) e **da** una serie di punteggi **valutazioni** 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 **fatte dagli** ~~degli~~ host ~~comprendono~~ **sono composte da** solo un commento testuale. Le recensioni possono essere visibili o non visibili. **Le recensioni** diventano visibili quando ~~entrambi~~ **sia l'host che l'ospite** 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~~ **recensioni** in cui sono coinvolti, creando ~~un thread di discussione~~ **delle discussioni**. 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~~ **la percentuale** 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.

1.3.2 Requisiti ristrutturati

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).

Per gli utenti che si registrano al servizio rappresentiamo indirizzo email, password, nome, cognome, numero o numeri di telefono. Gli utenti vengono riconosciuti come verificati quando forniscono la foto della carta d'identità. Gli utenti devono indicare un metodo di pagamento per poter prenotare l'alloggio. Gli utenti possono essere ospiti o "host". Gli "host" possono ospitare altri utenti del servizio in uno o più alloggi di loro proprietà. Gli "host" vengono riconosciuti come "superhost" se soddisfano i seguenti requisiti:

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

I superhost ricevono un badge sul loro profilo.

Per gli alloggi rappresentiamo il nome, l'indirizzo (visibile all'ospite nel momento in cui la prenotazione è confermata, altrimenti è visibile solo il comune dell'alloggio), una descrizione, il prezzo per notte per ospite e i costi di pulizia, delle foto, i servizi (ad esempio, cucina, wi-fi, lavatrice, ecc.), numero di letti, orario di check-in e check-out, l'host a cui appartiene l'alloggio, la valutazione media e il numero di recensioni.

Per gli utenti si possono rappresentare alcuni alloggi preferiti e alcune liste, ad esempio in base al viaggio che vogliono compiere.

Gli utenti possono prenotare degli alloggi di qualsiasi tipo. Per le prenotazioni gli utenti devono indicare un intervallo di date per il soggiorno e il numero degli ospiti per cui si vuole prenotare, e se gli ospiti per cui si vuole prenotare sono a loro volta utenti del servizio si possono rappresentare i loro nomi e cognomi. La prenotazione deve essere confermata o rifiutata dall'host. Per le prenotazioni rappresentiamo il costo totale. Quando la prenotazione viene confermata dall'host viene eseguito il pagamento da parte dell'ospite. La prenotazione può essere cancellata sia dall'ospite che dall'host.

Al termine del soggiorno, gli ospiti e gli host si possono valutare a vicenda. La recensione fatta dagli ospiti è composta da due testi (uno per l'alloggio e uno per l'host) e da una serie di valutazioni 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 fatte dagli host sono composte da solo un commento testuale. Le recensioni possono essere visibili o non visibili. Le recensioni diventano visibili quando sia l'host che l'ospite 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 recensioni in cui sono coinvolti, creando delle discussioni. 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 la percentuale 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.

1.3.3 Requisiti strutturati in gruppi di frasi omogenee

| | |
|------------------------------------|---|
| Frasi di carattere generale | <ul style="list-style-type: none">• 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). |
|------------------------------------|---|

| | |
|--|--|
| Frase relative agli alloggi | <ul style="list-style-type: none">• Per gli alloggi rappresentiamo il nome, l'indirizzo (visibile all'ospite nel momento in cui la prenotazione è confermata, altrimenti è visibile solo il comune dell'alloggio), una descrizione, il prezzo per notte per ospite e i costi di pulizia, delle foto, i servizi (ad esempio, cucina, wi-fi, lavatrice, ecc.), numero di letti, orario di check-in e check-out, l'host a cui appartiene l'alloggio, la valutazione media e il numero di recensioni. |
| Frase relative agli utenti | <ul style="list-style-type: none">• Per gli utenti che si registrano al servizio rappresentiamo indirizzo email, password, nome, cognome, numero o numeri di telefono.• Gli utenti vengono riconosciuti come verificati quando forniscono la foto della carta d'identità.• Gli utenti devono indicare un metodo di pagamento per poter prenotare l'alloggio.• Gli utenti possono essere ospiti o "host".• Per gli utenti si possono rappresentare alcuni alloggi preferiti e alcune liste, ad esempio in base al viaggio che vogliono compiere.• Gli utenti possono prenotare degli alloggi di qualsiasi tipo.• Gli host e gli ospiti possono commentare più volte le recensioni in cui sono coinvolti, creando delle discussioni. |
| Frase relative a tipi specifici di utenti | <ul style="list-style-type: none">• Gli "host" possono ospitare altri utenti del servizio in uno o più alloggi di loro proprietà.• Gli "host" vengono riconosciuti come "superhost" se soddisfano i seguenti requisiti:<ul style="list-style-type: none">○ Devono aver completato almeno 10 |

| | |
|---|--|
| | <p>soggiorni, per un totale di almeno 100 notti.</p> <ul style="list-style-type: none">○ Devono aver conservato una percentuale di cancellazione dell'1% (una cancellazione ogni 100 prenotazioni) massimo.○ Devono aver mantenuto una valutazione complessiva di 4,8 considerando tutti i soggiorni in tutti i suoi alloggi. <ul style="list-style-type: none">• I superhost ricevono un badge sul loro profilo.• Al termine del soggiorno, gli ospiti e gli host si possono valutare a vicenda. |
| Fraasi relative alle recensioni | <ul style="list-style-type: none">• La recensione fatta dagli ospiti è composta da due testi (uno per l'alloggio e uno per l'host) e da una serie di valutazioni in una scala da 1 a 5 su dimensioni come pulizia, comunicazione, posizione, qualità/prezzo.• Le recensioni fatte dagli host sono composte da solo un commento testuale.• Le recensioni possono essere visibili o non visibili.• Le recensioni diventano visibili quando sia l'host che l'ospite 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.• Le recensioni sono visibili sui profili degli utenti suddivise in base a quelle ricevute come ospite e come host. |
| Fraasi relative alle valutazioni | <ul style="list-style-type: none">• La valutazione complessiva del soggiorno è una media delle valutazioni ricevute sulle singole dimensioni. |

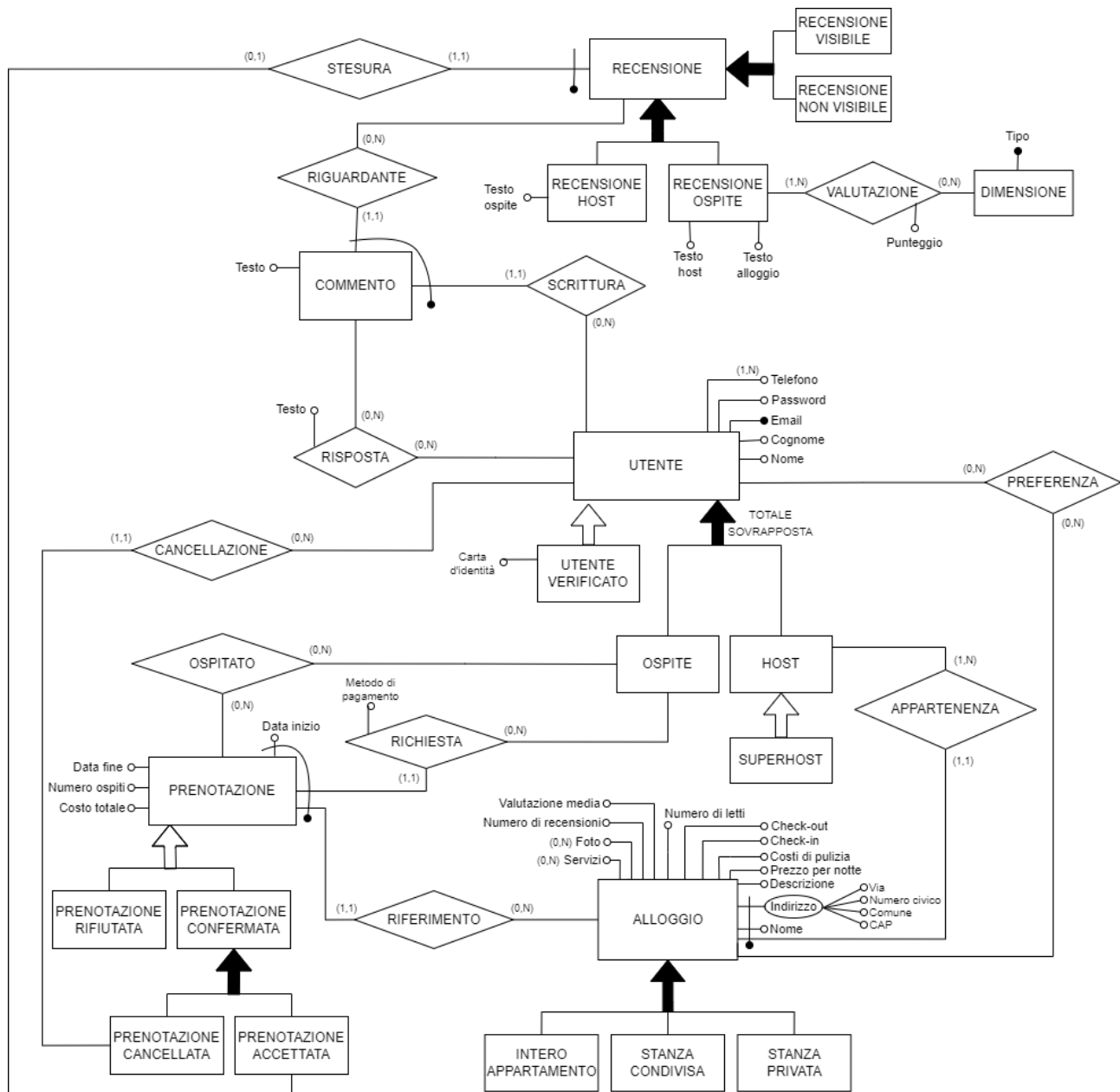
| | |
|---|---|
| Frase relative alle prenotazioni | <ul style="list-style-type: none"> • Per le prenotazioni gli utenti devono indicare un intervallo di date per il soggiorno e il numero degli ospiti per cui si vuole prenotare, e se gli ospiti per cui si vuole prenotare sono a loro volta utenti del servizio si possono rappresentare i loro nomi e cognomi. • La prenotazione deve essere confermata o rifiutata dall'host. • Quando la prenotazione viene confermata dall'host viene eseguito il pagamento da parte dell'ospite. • La prenotazione può essere cancellata sia dall'ospite che dall'host. |
| Frase relative alle operazioni | <ul style="list-style-type: none"> • La base di dati deve supportare le seguenti operazioni: <ul style="list-style-type: none"> ○ Una volta a settimana viene effettuato un calcolo per aggiornare la percentuale 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. |

1.3.4 Specifiche sulle operazioni

| Operazione | Descrizione | Frequenza |
|--------------|--|---------------------|
| Operazione 1 | Effettuare un calcolo per aggiornare la percentuale di cancellazione di ciascun host. | 1 volta / settimana |
| Operazione 2 | Controllare le condizioni per la qualifica di superhost e aggiornare lo status degli host. | 1 volta / giorno |
| Operazione 3 | Calcolare la classifica degli alloggi più graditi | 1 volta / mese |

1.4 Schema E-R + business rules

1.4.1 Schema E-R



1.4.2 Dizionario dei dati (entità)

| Entità | Descrizione | Attributi | Identificatore |
|-------------------|---|--|----------------|
| Utente | Utente del servizio | Email, password, nome, cognome, telefono (1,N) | Email |
| Utente verificato | <u>Entità figlia di utente</u> Utente che ha caricato la | Carta d'identità | Email |

| | | | |
|-------------------------|---|--|-------------------------------|
| | carta d'identità | | |
| Ospite | <u>Entità figlia di utente</u> Utente che soggiorna nell'alloggio | | Email |
| Host | <u>Entità figlia di utente</u> Utente proprietario dell'alloggio | | Email |
| Superhost | <u>Entità figlia di host</u> Host con un badge identificativo | | Email |
| Alloggio | Dimora del soggiorno | Indirizzo (via, numero civico, comune, CAP), nome, servizi (0,N), descrizione, prezzo per notte, foto (0,N), numero di letti, check-in, check-out, valutazione media, numero di recensioni, costi di pulizia | Indirizzo, nome, host |
| Intero appartamento | <u>Entità figlia di alloggio</u> Intero alloggio privato (stanza e spazi privati) | | |
| Stanza privata | <u>Entità figlia di alloggio</u> Stanza d'alloggio privata e spazi condivisi | | |
| Stanza condivisa | <u>Entità figlia di alloggio</u> Stanza d'alloggio condivisa e spazi condivisi | | |
| Prenotazione | Riserva un alloggio ad un ospite | Data inizio, data fine, numero ospiti, costo totale | Data inizio, ospite, alloggio |
| Prenotazione confermata | <u>Entità figlia di prenotazione</u> Prenotazione confermata dall'host | | Data inizio, ospite, alloggio |
| Prenotazione rifiutata | <u>Entità figlia di prenotazione</u> Prenotazione rifiutata dall'host | | Data inizio, ospite, alloggio |
| Prenotazione accettata | <u>Entità figlia di prenotazione confermata</u> Prenotazione accettata dall'ospite, che comporta | | Data inizio, ospite, alloggio |

| | | | |
|-------------------------|---|----------------------------|-------------------------------|
| | un soggiorno | | |
| Prenotazione cancellata | <u>Entità figlia di prenotazione confermata</u> Prenotazione cancellata dall'ospite o dall'host | | Data inizio, ospite, alloggio |
| Recensione | Recensione di host e ospiti | | Prenotazione accettata |
| Recensione host | <u>Entità figlia di recensione</u> Recensione fatta dall'host relativa all'ospite | Testo ospite | Prenotazione accettata |
| Recensione ospite | <u>Entità figlia di recensione</u> Recensione fatta dall'ospite relativa all'host e all'alloggio | Testo host, testo alloggio | Prenotazione accettata |
| Recensione visibile | <u>Entità figlia di recensione</u> Recensione visibile a tutti | | Prenotazione accettata |
| Recensione non visibile | <u>Entità figlia di recensione</u> Recensione non visibile a nessuno (se non a chi l'ha fatta) | | Prenotazione accettata |
| Dimensione | Caratteristica del soggiorno (pulizia, precisione, comunicazione, posizione, check-in e qualità/prezzo) | Tipo | Tipo |
| Commento | Commento ad una recensione | Testo | Recensione, utente |

1.4.3 Dizionario dei dati (relazioni)

| Relazione | Descrizione | Componenti | Attributi |
|---------------|---|------------------------------------|---------------------|
| Appartenenza | Appartenenza di un alloggio ad un host | Host (1,N), alloggio (1,1) | |
| Preferenza | Preferenza di alloggi | Utente (0,N), alloggio (0,N) | |
| Richiesta | Richiesta di prenotazione | Ospite (0,N), prenotazione (1,1) | Metodo di pagamento |
| Ospitato | Ospite ospitato in un soggiorno | Ospite (0,N), prenotazione (0,N) | |
| Riferimento | Riferimento della prenotazione ad un alloggio | Prenotazione (1,1), alloggio (0,N) | |
| Cancellazione | Cancellazione della prenotazione | Utente (0,N), prenotazione | |

| | | | |
|-------------|--|--|-----------|
| | | cancellata (1,1) | |
| Stesura | Stesura di una recensione | Recensione (1,1), prenotazione accettata (0,1) | |
| Valutazione | Valutazione di una dimensione | Recensione ospite (1,N), dimensione (0,N) | Punteggio |
| Scrittura | Scrittura di un commento a una recensione | Commento (1,1), utente (0,N) | |
| Risposta | Risposta ad un commento | Commento (0,N), utente (0,N) | Testo |
| Riguardante | Commento riguardante una recensione | Recensione (0,N), commento (1,1) | |

1.4.4 Vincoli d'integrità

1. Il punteggio delle valutazioni **deve** essere compreso tra 1 e 5.
2. Il superhost **deve** aver completato almeno 10 soggiorni, per un totale di almeno 100 notti.
3. Il superhost **deve** aver conservato una percentuale di cancellazione dell'1% (una cancellazione ogni 100 prenotazioni) massimo.
4. Il superhost **deve** aver mantenuto una valutazione complessiva di 4,8 considerando tutti i soggiorni in tutte le case di sua proprietà.
5. La recensione **deve** diventare visibile quando sia host che ospite hanno fatto la recensione.
6. Se uno tra host o ospite non ha fatto la recensione e l'altro l'ha fatta, quest'ultima **deve** diventare visibile dopo 7 giorni dalla fine del soggiorno.
7. L'utente che cancella una prenotazione **deve** essere l'host a cui appartiene l'alloggio prenotato o l'ospite che ha effettuato la prenotazione.
8. La data d'inizio **deve** essere antecedente la data di fine.

1.4.5 Derivazioni

1. La valutazione media **si ottiene** facendo la media tra le medie delle valutazioni su tutte le dimensioni date da tutte le persone che hanno soggiornato nell'alloggio.
2. Il numero di recensioni dell'alloggio **si ottiene** conteggiando le recensioni riguardanti le prenotazioni accettate che fanno riferimento a quel determinato alloggio.
3. Il costo totale **si ottiene** sommando i costi di pulizia con il prezzo per notte moltiplicato per data fine meno data inizio, moltiplicato per il numero degli ospiti.
4. La percentuale di cancellazione **si ottiene** dal rapporto del numero di prenotazioni cancellate, dall'host o dall'ospite, con il numero di prenotazioni confermate dall'host.

NOTA: per ottenere la percentuale di cancellazione abbiamo considerato sia le cancellazioni effettuate dagli host che dagli ospiti.

5. L'autore di una conferma o di un rifiuto di una prenotazione **si ottiene** risalendo all'host a cui appartiene l'alloggio.

2. Progettazione logica

2.1 Tavola dei volumi

| Concetto | Tipo | Volume |
|-------------------------|------|---------------|
| Utente | E | 30 000 000 |
| Utente verificato | E | 20 000 000 |
| Ospite | E | 27 500 000 |
| Host | E | 4 000 000 |
| Superhost | E | 2 000 000 |
| Alloggio | E | 9 000 000 |
| Intero appartamento | E | 5 000 000 |
| Stanza privata | E | 3 000 000 |
| Stanza condivisa | E | 1 000 000 |
| Prenotazione | E | 1 500 000 000 |
| Prenotazione confermata | E | 1 100 000 000 |
| Prenotazione rifiutata | E | 350 000 000 |
| Prenotazione accettata | E | 1 000 000 000 |
| Prenotazione cancellata | E | 100 000 000 |
| Recensione | E | 800 000 000 |
| Recensione host | E | 600 000 000 |
| Recensione ospite | E | 200 000 000 |
| Recensione visibile | E | 550 000 000 |
| Recensione non visibile | E | 250 000 000 |
| Dimensione | E | 6 |
| Commento | E | 1 300 000 000 |
| Appartenenza | R | 9 000 000 |
| Preferenza | R | 120 000 000 |

| | | |
|---------------|---|---------------|
| Richiesta | R | 1 500 000 000 |
| Ospitato | R | 55 000 000 |
| Riferimento | R | 1 500 000 000 |
| Cancellazione | R | 100 000 000 |
| Stesura | R | 800 000 000 |
| Valutazione | R | 1 200 000 000 |
| Scrittura | R | 1 300 000 000 |
| Risposta | R | 2 600 000 000 |
| Riguardante | R | 1 300 000 000 |

Motivazioni:

- Abbiamo pensato che un valore verosimile per il numeri di utenti del servizio potesse essere 30 milioni.
- Abbiamo supposto che 2/3 degli utenti siano verificati (e quindi che abbiano caricato la carta d'identità), quindi 20 milioni.
- Abbiamo supposto che quasi tutti gli utenti siano ospiti, solo una piccola percentuale di utenti è solo host.
- Abbiamo supposto che gli host siano 4 milioni (seguendo le stime del sito airbnb).
- Abbiamo supposto che i superhost siano la metà degli host, ovvero 2 milioni.
- Abbiamo pensato che un valore verosimile per il numero di alloggi potesse essere 9 milioni, di cui 5 milioni di interi appartamenti, 3 milioni di stanze private e 1 milione di stanze condivise.
- Abbiamo ipotizzato che ci siano 1 miliardo e mezzo di prenotazioni, di cui la maggioranza (1 miliardo e 100 milioni) confermate dall'host. La maggioranza (1 miliardo) delle prenotazioni confermate viene anche accettata.
- Abbiamo ipotizzato 800 milioni di recensioni poichè una parte degli utenti una volta finito il soggiorno non scrivono la recensione.
- Per l'entità dimensione abbiamo supposto che le dimensioni siano 6: pulizia, precisione, comunicazione, posizione, check-in e qualità/prezzo.
- Molti utenti scrivono più commenti, per cui abbiamo pensato che 1 miliardo e 300 milioni fosse una scelta verosimile.

- Appartenenza ha volume 9 milioni proprio come alloggio, questo perchè ogni alloggio appartiene a uno e un solo host.
- Per la relazione preferenza abbiamo supposto che ogni utente avesse in media 4 preferenze e quindi abbiamo fatto: $30\,000\,000$ (utente) $\times 4 = 120\,000\,000$ (preferenza).
- Richiesta ha volume 1 miliardo e mezzo proprio come prenotazione, questo perchè una prenotazione viene effettuata da un solo host.
- Per la relazione ospitato abbiamo supposto che ogni ospite includa nella sua prenotazione mediamente altri 2 ospiti registrati sul servizio e quindi abbiamo fatto: $27\,500\,000$ (ospite) $\times 2 = 55\,000\,000$ (ospitato).
- Ogni prenotazione si riferisce a uno e un solo alloggio, per cui riferimento avrà lo stesso volume di prenotazione (1 milione e mezzo).
- Cancellazione avrà lo stesso volume di prenotazione cancellata.
- Stesura avrà lo stesso volume di prenotazione accettata.
- Per la valutazione abbiamo supposto che ogni recensione ospite debba valutare tutte e 6 le dimensioni e quindi abbiamo fatto: $200\,000\,000$ (recensione ospite) $\times 6$ (dimensione) = $1\,200\,000\,000$ (valutazione).
- Riguardante e scrittura avranno tutte lo stesso volume di commento.
- Per quanto riguarda la relazione risposta abbiamo supposto che ogni commento riceva in media 2 risposte e quindi abbiamo fatto: $1\,300\,000\,000$ (commento) $\times 2 = 2\,600\,000\,000$.

2.2 Tavola delle operazioni

| N° | Operazione | Tipo | Frequenza |
|----|--|------|-----------------|
| 1 | Effettua un calcolo per aggiornare la percentuale di cancellazione di ciascun host. | B | 1 / settimana |
| 2 | Controlla le condizioni per la qualifica di superhost e aggiorna lo status degli host. | B | 1 / giorno |
| 3 | Calcola la classifica degli alloggi più graditi. | B | 1 / mese |
| 4 | Aggiorna lo status di utente ad utente verificato se soddisfa il requisito. | I | 1 000 / giorno |
| 5 | Inserimento nuova prenotazione. | I | 10 000 / giorno |
| 6 | Avviene la scrittura di una recensione. | I | 2 000 / giorno |
| 7 | Inserimento nuovo utente. | I | 1 500 / giorno |

| | | | |
|----|---|---|-----------------|
| 8 | Elenco degli alloggi prenotabili in un certo periodo di tempo. | 1 | 40 000 / giorno |
| 9 | Aggiunta di un alloggio tra i preferiti da parte di un utente. | 1 | 5 000 / giorno |
| 10 | Stampa di tutti i dati relativi ad una prenotazione, incluso il costo totale. | 1 | 12 000 / giorno |

Motivazioni:

- Le prime tre operazioni le troviamo nel testo dei requisiti con le relative frequenze (1, 2, 3).
- Supponiamo che una piccola percentuale di utenti ogni giorno carichi la carta d'identità per diventare utente verificato (4).
- Un'operazione rilevante nella nostra base di dati è sicuramente l'inserimento di una nuova prenotazione. Supponiamo che vengano fatte 10 000 prenotazioni al giorno (5).
- Supponiamo che il numero di recensioni giornaliere scritte sia 2 000 (6).
- Un'altra operazione che secondo noi è rilevante è l'inserimento di un nuovo utente. Supponiamo che ogni giorno si registrino al nostro servizio mediamente 1 500 utenti (7).
- Quando un utente utilizza il nostro servizio l'operazione principale di suo interesse sarà la ricerca di alloggi, perciò abbiamo aggiunto l'operazione di elenco degli alloggi prenotabili in un certo periodo di tempo (8).
- Un'altra funzionalità permessa all'utente è quella di mettere un alloggio tra i preferiti (9).
- Una volta che un utente effettua una prenotazione può visualizzare tutti i dati della prenotazione (10).

2.3 Ristrutturazione dello schema E-R

2.3.1 Analisi delle ridondanze

Le ridondanze che abbiamo individuato sono:

- **Costo totale** (PRENOTAZIONE): attributo derivabile da altra entità.
- **Valutazione media** (ALLOGGIO): attributo derivabile da altra entità.
- **Numero recensioni** (ALLOGGIO): attributo derivabile da un conteggio.
- **Numero ospiti registrati** (PRENOTAZIONE): attributo derivabile da un conteggio.

NOTA: attributo non presente nello schema E-R, ma aggiunto.

Svolgiamo l'analisi delle ridondanze per la ridondanza "costo totale".

Riportiamo di seguito due operazioni che riguardano l'entità PRENOTAZIONE (entità che ha come attributo l'attributo costo totale).

| Operazione | Frequenza |
|---|-----------------|
| Inserimento nuova prenotazione. | 10 000 / giorno |
| Stampa di tutti i dati relativi ad una prenotazione, incluso il costo totale. | 12 000 / giorno |

Eseguiamo l'analisi delle ridondanze per entrambe le operazioni.

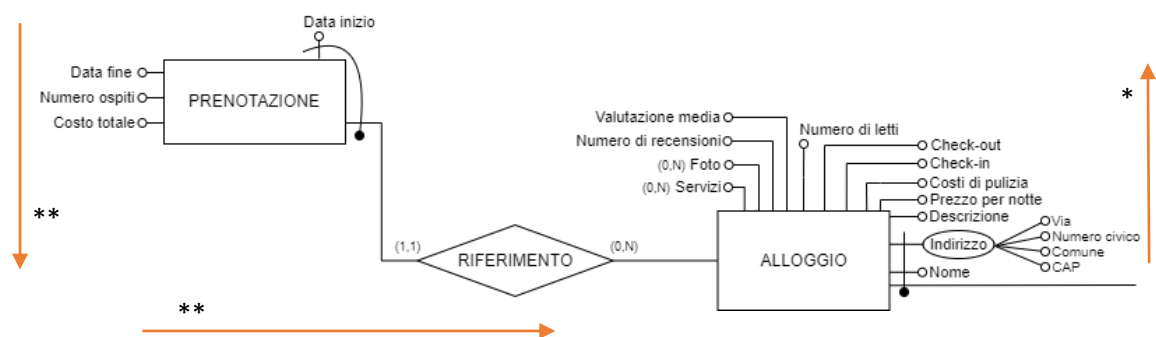
NOTA: non abbiamo considerato le relazioni OSPITATO e RICHIESTA e l'entità OSPITE negli schemi di navigazione, perché per l'effettiva analisi delle ridondanze sono superflue, in quanto sia con ridondanza che senza, gli accessi su queste relazioni ed entità rimarrebbero costanti.

Operazione 1: inserimento nuova prenotazione.

- **Analisi con ridondanza**

1) Tavola dei volumi

| Concetto | Tipo | Volume |
|--------------|------|---------------|
| Prenotazione | E | 1 500 000 000 |
| Riferimento | R | 1 500 000 000 |
| Alloggio | E | 9 000 000 |



* Navigazione da ALLOGGIO a PRENOTAZIONE

** Navigazione da PRENOTAZIONE a RIFERIMENTO

2) Schema di navigazione

3) Tavola degli accessi

| Concetto | Costrutto | Accessi | Tipo |
|--------------|-----------|---------|------|
| Prenotazione | Entità | 1 | S |

| | | | |
|-------------|--------------|---|---|
| Riferimento | Associazione | 1 | S |
| Alloggio | Entità | 1 | L |

1. Accesso in lettura ad alloggio
2. Accesso in scrittura a prenotazione
3. Accesso in scrittura a riferimento

Totale accessi operazione: 2 accessi in scrittura, 1 in lettura

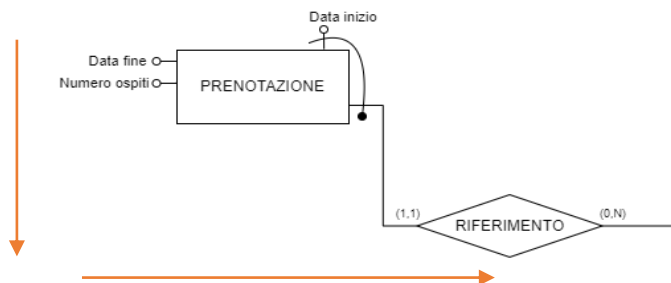
Totale accessi per giorno: $2 * 10\,000 = 20\,000$ in scrittura, $1 * 10\,000 = 10\,000$ in lettura

- **Analisi senza ridondanza**

1) Tavola dei volumi

| Concetto | Tipo | Volume |
|--------------|------|---------------|
| Prenotazione | E | 1 500 000 000 |
| Riferimento | R | 1 500 000 000 |

2) Schema di navigazione



3) Tavola degli accessi

| Concetto | Costrutto | Accessi | Tipo |
|--------------|--------------|---------|------|
| Prenotazione | Entità | 1 | S |
| Riferimento | Associazione | 1 | S |

1. Accesso in scrittura in prenotazione
2. Accesso in scrittura in riferimento

Totale accessi operazione: 2 accessi in scrittura

Totale accessi per giorno: $2 * 10\,000 = 20\,000$ in scrittura

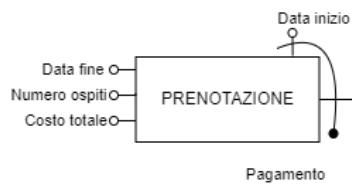
Operazione 2: stampa di tutti i dati relativi ad una prenotazione, incluso il costo totale.

- **Analisi con ridondanza**

1) Tavola dei volumi

| Concetto | Tipo | Volume |
|--------------|------|---------------|
| Prenotazione | E | 1 500 000 000 |

2) Schema di navigazione



3) Tavola degli accessi

| Concetto | Costrutto | Accessi | Tipo |
|--------------|-----------|---------|------|
| Prenotazione | Entità | 1 | L |

1. Accesso in lettura in prenotazione per leggere i dati della prenotazione incluso il costo totale

Totale accessi operazione: 1 accesso in lettura

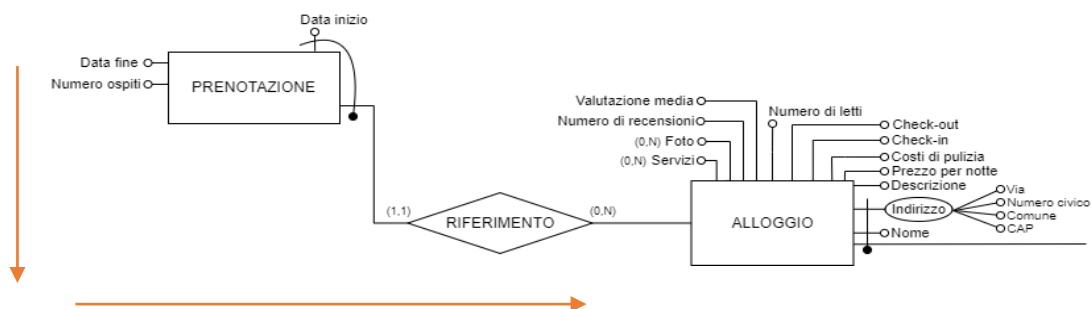
Totale accessi per giorno: $1 * 12\ 000 = 12\ 000$ in lettura

- **Analisi senza ridondanza**

1) Tavola dei volumi

| Concetto | Tipo | Volume |
|--------------|------|---------------|
| Prenotazione | E | 1 500 000 000 |
| Riferimento | R | 1 500 000 000 |
| Alloggio | E | 9 000 000 |

2) Schema di navigazione



3) Tavola degli accessi

| Concetto | Costrutto | Accessi | Tipo |
|--------------|-----------|---------|------|
| Prenotazione | Entità | 1 | L |

| | | | |
|-------------|--------------|---|---|
| Riferimento | Associazione | 1 | L |
| Alloggio | Entità | 1 | L |

1. Accesso in lettura a prenotazione

2. Accesso in lettura a riferimento

3. Accesso in lettura ad alloggio

Totale accessi operazione: 3 accessi in lettura

Totale accessi per giorno: $3 * 12\ 000 = 36\ 000$ in lettura

Analisi tempo

- Con ridondanza:
 - Operazione 1:
20 000 accessi in scrittura (valgono come 40 000 accessi) e 10 000 accessi in lettura.
 - Operazione 2:
12 000 accessi in lettura.

Totale di 62 000 accessi al giorno.

- Senza ridondanza:
 - Operazione 1:
20 000 accessi in scrittura (valgono come 40 000 accessi).
 - Operazione 2:
36 000 accessi in lettura.

Totale di 76 000 accessi al giorno.

Analisi spazio

- Con ridondanza:
Ipotesi: si utilizzano 4 byte per memorizzare l'attributo costo totale.
Spazio totale necessario: $4 * 1\ 500\ 000\ 000$ (Prenotazione) = 6 000 000 000 byte (≈ 6 GigaByte).
- Senza ridondanza:
Non memorizziamo l'attributo costo totale, per cui non necessita di spazio aggiuntivo.

| Con ridondanza | Senza ridondanza |
|--------------------------|--------------------------|
| 62 000 accessi al giorno | 76 000 accessi al giorno |

| | |
|---------------------------------|---------------------------------|
| 6 GigaByte di spazio aggiuntivo | 0 GigaByte di spazio aggiuntivo |
|---------------------------------|---------------------------------|

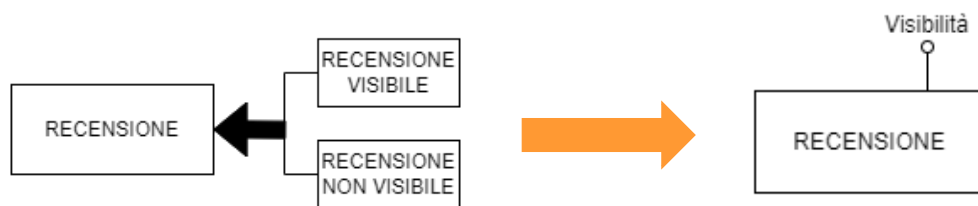
In questo caso scegliamo di tenere la ridondanza perché effettua 14 000 accessi in meno e nonostante utilizza 6 GigaByte di spazio contro gli zero Gigabyte della versione senza ridondanza è una dimensione di dati esigua rispetto all'intera base dati del servizio.

2.3.2 Eliminazione delle generalizzazioni

- Entità padre: RECENSIONE

Entità figlie: RECENSIONE VISIBILE, RECENSIONE NON VISIBILE

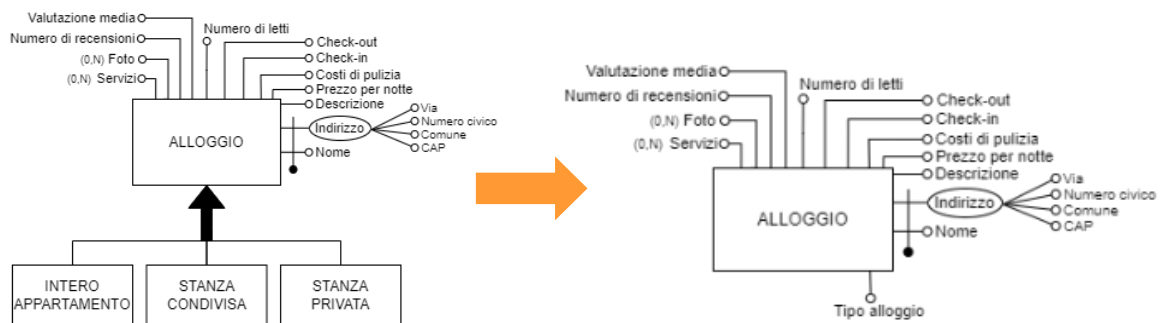
Primo tipo: accorpamento delle entità figlie della generalizzazione nell'entità genitore.



- Entità padre: ALLOGGIO

Entità figlie: INTERO APPARTAMENTO, STANZA PRIVATA, STANZA CONDIVISA

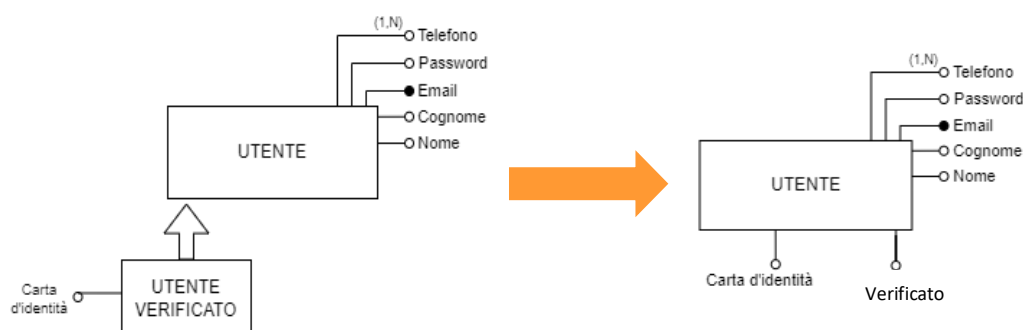
Primo tipo: accorpamento delle entità figlie della generalizzazione nell'entità genitore.



- Entità padre: UTENTE

Entità figlia: UTENTE VERIFICATO

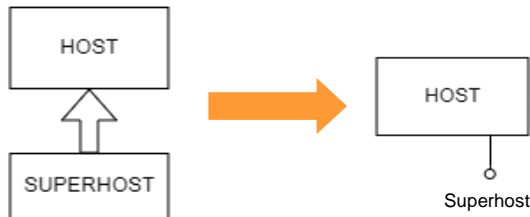
Primo tipo: accorpamento delle entità figlie della generalizzazione nell'entità genitore.



- Entità padre: HOST

Entità figlia: SUPERHOST

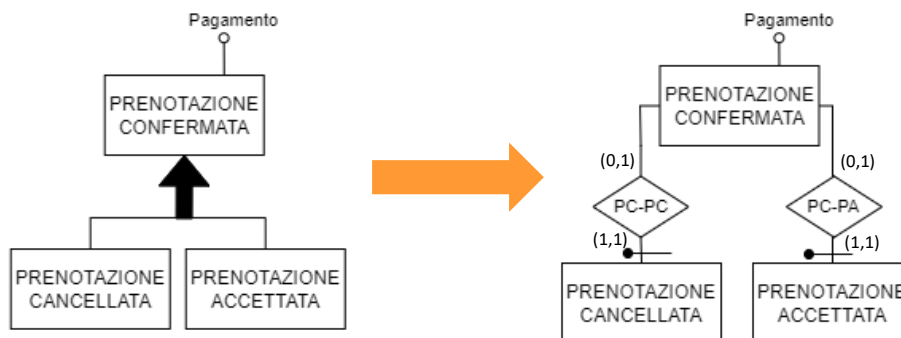
Primo tipo: accorpamento delle entità figlie della generalizzazione nell'entità genitore.



- Entità padre: PRENOTAZIONE CONFERMATA

Entità figlie: PRENOTAZIONE CANCELLATA, PRENOTAZIONE ACCETTATA

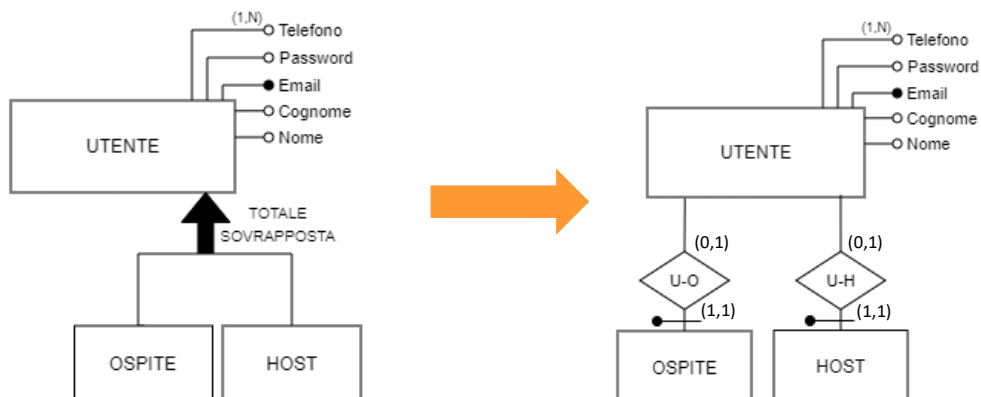
Secondo tipo: accorpamento dell'entità genitore della generalizzazione nelle entità figlie.



- Entità padre: UTENTE

Entità figlie: OSPITE, HOST

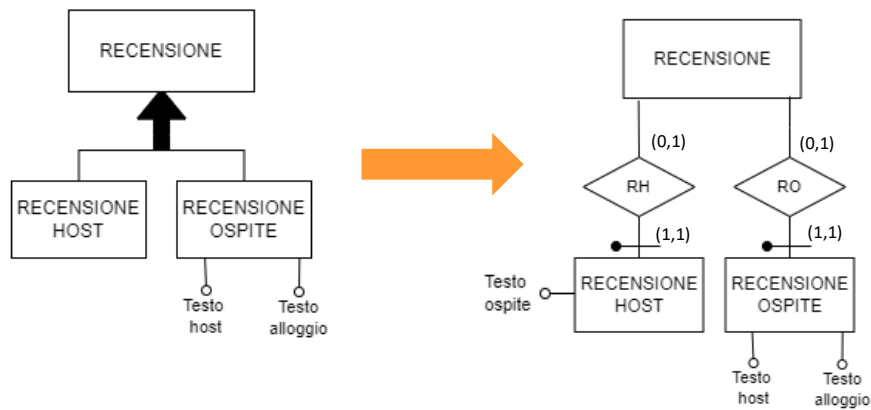
Terzo tipo: sostituzione della generalizzazione con relazioni.



- Entità padre: RECENSIONE

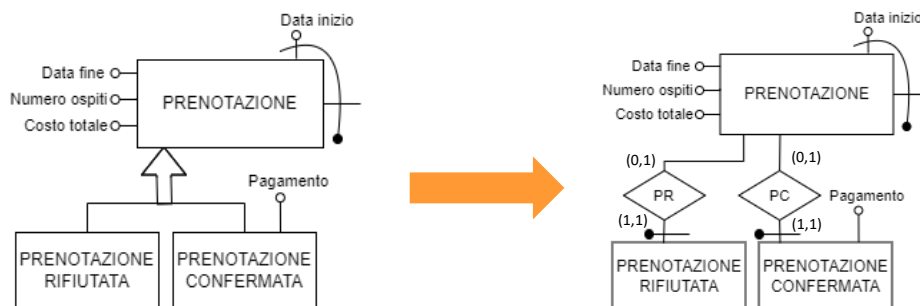
Entità figlie: RECENSIONE HOST, RECENSIONE OSPITE

Terzo tipo: sostituzione della generalizzazione con relazioni.



- Entità padre: PRENOTAZIONE

Entità figlie: PRENOTAZIONE RIFIUTATA, PRENOTAZIONE CONFERMATA

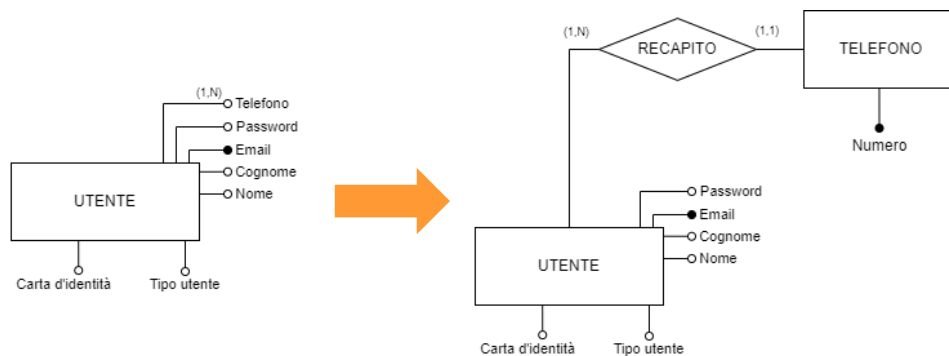


Terzo tipo: sostituzione della generalizzazione con relazioni.

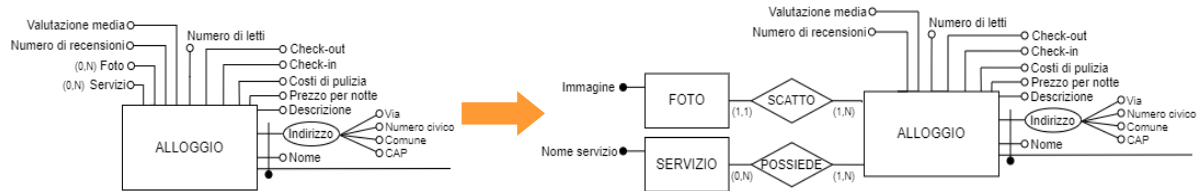
2.3.3 Eventuale partizionamento/accorpamento di entità e associazioni

Eliminazione degli attributi multivalore:

- Telefono in UTENTE

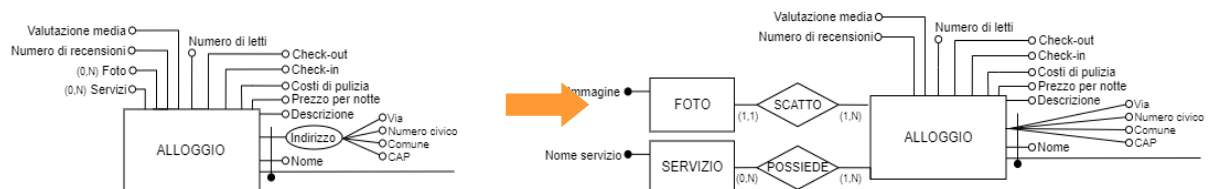


- Foto e servizi in ALLOGGIO



Eliminazione degli attributi composti:

- Indirizzo in ALLOGGIO

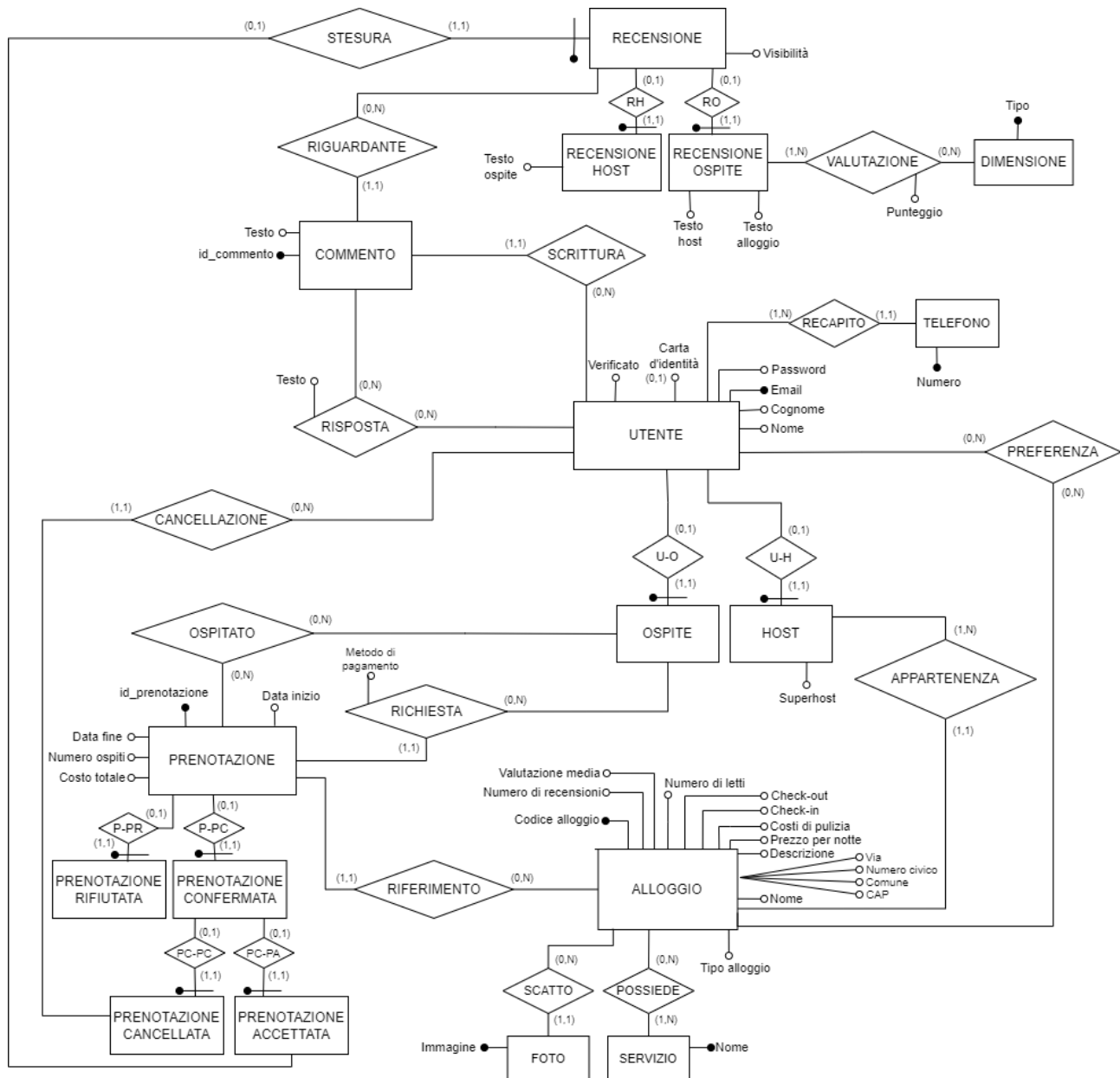


2.3.4 Eventuale scelta degli identificatori principali

- In UTENTE l'identificatore è e-mail. Supponiamo quindi che ogni utente debba avere un e-mail diversa per registrarsi al servizio.
- In ALLOGGIO introduciamo l'identificatore codice alloggio per semplicità e per rendere le operazioni più semplici. Codice è il codice identificativo dell'alloggio stesso.
- In PRENOTAZIONE introduciamo l'identificatore id prenotazione per semplicità.
- In RECENSIONE l'identificatore rimane esterno poichè soddisfa i criteri di semplicità, assenza di opzionalità e utilizzo nelle operazioni più frequenti.
- In DIMENSIONE l'identificatore è tipo e rimane così perchè ogni dimensione ha un tipo differente.
- In COMMENTO introduciamo l'identificatore id commento per semplicità.
- In FOTO l'identificatore è immagine.
- In SERVIZIO l'identificatore è nome servizio.
- In TELEFONO l'identificatore è numero.
- In PRENOTAZIONE CONFERMATA e PRENOTAZIONE RIFIUTATA l'identificatore è esterno con l'entità padre PRENOTAZIONE.
- In PRENOTAZIONE CANCELLATA e PRENOTAZIONE ACCETTATA l'identificatore è esterno con PRENOTAZIONE.
- In RECENSIONE HOST e RECENSIONE OSPITE l'identificatore è esterno con l'entità padre RECENSIONE.
- In OSPITE e HOST l'identificatore è esterno con l'entità padre UTENTE.

2.4 Schema E-R ristrutturato + business rules

2.4.1 Schema E-R ristrutturato



2.4.2 Dizionario dei dati (entità)

| Entità | Descrizione | Attributi | Identificatore |
|--------|------------------------------------|--|----------------|
| Utente | Utente del servizio | Email, password, nome, cognome, carta d'identità, verificato | Email |
| Ospite | Utente che soggiorna nell'alloggio | | Utente |
| Host | Utente proprietario | Tipo host | Utente |

| | dell'alloggio | | |
|-------------------------|---|---|-------------------------------|
| Alloggio | Dimora del soggiorno | Via, numero civico, comune, CAP, nome, descrizione, prezzo per notte, numero di letti, check-in, check-out, valutazione media, numero di recensioni, costi di pulizia, tipo alloggio, codice alloggio | Codice alloggio |
| Prenotazione | Riserva un alloggio ad un ospite | Data inizio, data fine, numero ospiti, costo totale | Id_prenotazione |
| Prenotazione confermata | Prenotazione confermata dall'host | Pagamento | Data inizio, ospite, alloggio |
| Prenotazione rifiutata | Prenotazione rifiutata dall'host | | Prenotazione |
| Prenotazione accettata | Prenotazione accettata dall'ospite, che comporta un soggiorno | Pagamento | Prenotazione |
| Prenotazione cancellata | Prenotazione cancellata dall'ospite o dall'host | Pagamento | Prenotazione |
| Recensione | Recensione di host e ospiti | Visibilità | Prenotazione accettata |
| Recensione host | Recensione fatta dall'host relativa all'ospite | Testo ospite | Recensione |
| Recensione ospite | Recensione fatta dall'ospite relativa all'host e all'alloggio | Testo host, testo alloggio | Recensione |
| Dimensione | Caratteristica del soggiorno | Tipo | Tipo |
| Commento | Commento ad una recensione | Testo | Id |
| Foto | Foto dell'alloggio | Immagine | Immagine |
| Servizio | Servizio dell'alloggio | Nome | Nome |
| Telefono | Telefono dell'utente | Numero | Numero |

2.4.3 Dizionario dei dati (relazioni)

| Relazione | Descrizione | Componenti | Attributi |
|--------------|-----------------------------|----------------------|-----------|
| Appartenenza | Appartenenza di un alloggio | Host (1,N), alloggio | |

| | | | |
|---------------|---|---|---------------------|
| | ad un host | (1,1) | |
| Preferenza | Preferenza di alloggi | Utente (0,N), alloggio (0,N) | |
| Richiesta | Richiesta di prenotazione | Ospite (0,N), prenotazione (1,1) | Metodo di pagamento |
| Ospitato | Ospite ospitato in un soggiorno | Ospite (0,N), prenotazione (0,N) | |
| Riferimento | Riferimento della prenotazione ad un alloggio | Prenotazione (1,1), alloggio (0,N) | |
| Cancellazione | Cancellazione della prenotazione | Utente (0,N), prenotazione cancellata (1,1) | |
| Stesura | Stesura di una recensione | Recensione (1,1), prenotazione accettata (0,1) | |
| Valutazione | Valutazione di una dimensione | Recensione ospite (1,N), dimensione (1,1) | Punteggio |
| Scrittura | Scrittura di un commento a una recensione | Commento (1,1), utente (0,N) | |
| Risposta | Risposta ad un commento | Commento (0,N), utente (0,N) | Testo |
| Riguardante | Commento riguardante una recensione | Recensione (0,N), commento (1,1) | |
| U-O | Utente - Ospite | Ospite (1,1), utente (0,1) | |
| U-H | Utente - Host | Host (1,1), utente (0,1) | |
| RH | Recensione - Recensione host | Recensione host (1,1), recensione (0,1) | |
| RO | Recensione - Recensione ospite | Recensione ospite (1,1), recensione (0,1) | |
| P-PR | Prenotazione - prenotazione rifiutata | Prenotazione rifiutata (1,1), prenotazione (0,1) | |
| P-PC | Prenotazione - prenotazione confermata | Prenotazione confermata (1,1), prenotazione (0,1) | |
| PC-PC | Prenotazione confermata – prenotazione cancellata | Prenotazione confermata (0,1), prenotazione | |

| | | | |
|----------|--|---|--|
| | | cancellata (1,1) | |
| PC-PA | Prenotazione confermata – prenotazione accettata | Prenotazione confermata (0,1), prenotazione accettata (1,1) | |
| Recapito | Recapito telefonico di un utente | Telefono (1,1), Utente (1,N) | |
| Scatto | Scatto delle foto di un alloggio | Foto (1,1), Alloggio (0,N) | |
| Possiede | Servizio posseduto da un alloggio | Servizio (0,N), Alloggio (0,N) | |

2.4.4 Vincoli d'integrità

1. Il punteggio delle valutazioni **deve** essere compreso tra 1 e 5.
2. Il superhost, tipo specifico di host, **deve** aver completato almeno 10 soggiorni, per un totale di almeno 100 notti.
3. Il superhost, tipo specifico di host, **deve** aver conservato una percentuale di cancellazione dell'1% (una cancellazione ogni 100 prenotazioni) massimo.
4. Il superhost, tipo specifico di host, **deve** aver mantenuto una valutazione complessiva di 4,8 considerando tutti i soggiorni in tutte le case di sua proprietà.
5. La recensione **deve** diventare di tipo visibile quando sia host che ospite hanno fatto la recensione.
6. Se uno tra host o ospite non ha fatto la recensione e l'altro l'ha fatta, quest'ultima **deve** diventare di tipo visibile dopo 7 giorni dalla fine del soggiorno.
7. L'utente che cancella una prenotazione **deve** essere o l'host a cui appartiene l'alloggio prenotato o l'ospite che ha effettuato la prenotazione.
8. La data d'inizio **deve** essere antecedente la data di fine.
9. L'attributo verificato **deve** essere TRUE se viene caricata la carta d'identità.
10. L'attributo superhost **deve** essere TRUE se l'host soddisfa tutti i requisiti.

2.4.5 Derivazioni

1. La valutazione media **si ottiene** facendo la media tra le medie delle valutazioni su tutte le dimensioni date da tutte le persone che hanno soggiornato nell'alloggio.
2. Il numero di recensioni dell'alloggio **si ottiene** conteggiando le recensioni riguardanti le prenotazioni accettate che fanno riferimento a quel determinato alloggio.

3. Il costo totale **si ottiene** sommando i costi di pulizia con il prezzo per notte moltiplicato per data fine meno data inizio, moltiplicato per il numero degli ospiti.
4. La percentuale di cancellazione **si ottiene** dal rapporto del numero di prenotazioni cancellate, dall'host o dall'ospite, con il numero di prenotazioni confermate dall'host.
NOTA: per ottenere la percentuale di cancellazione abbiamo considerato sia le cancellazioni effettuate dagli host che dagli ospiti.
5. L'autore di una conferma o di un rifiuto di una prenotazione **si ottiene** risalendo all'host a cui appartiene l'alloggio.

2.5 Schema relazionale + vincoli di integrità referenziale

2.5.1 Schema relazionale

UTENTE (Email, password, cognome, nome, verificato, carta d'identità*)

OSPITE (Utente)

HOST (Utente, superhost)

ALLOGGIO (Codice alloggio, nome, via, numero civico, comune, cap, descrizione, prezzo per notte, costi di pulizia, check-in, check-out, numero di letti, valutazione media, numero di recensioni, tipo alloggio, host)

FOTO (Immagine, alloggio)

SERVIZIO (Nome)

TELEFONO (Numero, utente)

PRENOTAZIONE (id prenotazione, data inizio, data fine, numero ospiti, costo totale, metodo di pagamento, ospite, alloggio)

PRENOTAZIONE RIFIUTATA (Prenotazione)

PRENOTAZIONE CONFERMATA (Prenotazione)

PRENOTAZIONE CANCELLATA (Prenotazione confermata, utente)

PRENOTAZIONE ACCETTATA (Prenotazione confermata)

RECENSIONE (Prenotazione accettata, visibilità)

RECENSIONE HOST (Recensione, testo ospite)

RECENSIONE OSPITE (Recensione, testo host, testo alloggio)

DIMENSIONE (Tipo)

COMMENTO (id commento, testo, recensione, utente)

PREFERENZA (Utente, alloggio)

POSSIEDE (Servizio, alloggio)

OSPITATO (Prenotazione, ospite)

VALUTAZIONE (Recensione ospite, dimensione, punteggio)

RISPOSTA (Commento, utente, testo)

2.5.2 Vincoli di integrità referenziale

Con vincoli d'integrità referenziale tra:

- L'attributo utente in OSPITE e la chiave di UTENTE.
- L'attributo utente in HOST e la chiave di UTENTE.
- L'attributo host in ALLOGGIO e la chiave di HOST.
- L'attributo utente in TELEFONO e la chiave di UTENTE.
- L'attributo utente in PREFERENZA e la chiave di UTENTE.
- L'attributo alloggio in PREFERENZA e la chiave di ALLOGGIO.
- L'attributo alloggio in FOTO e la chiave di ALLOGGIO.
- L'attributo servizio in POSSIEDE e la chiave di SERVIZIO.
- L'attributo alloggio in POSSIEDE e la chiave di ALLOGGIO.
- L'attributo prenotazione in OSPITATO e la chiave di PRENOTAZIONE.
- L'attributo ospite in OSPITATO e la chiave di OSPITE.
- L'attributo ospite in PRENOTAZIONE e la chiave di OSPITE.
- L'attributo alloggio in PRENOTAZIONE e la chiave di ALLOGGIO.
- L'attributo utente in PRENOTAZIONE CANCELLATA e la chiave di UTENTE.
- L'attributo recensione in COMMENTO e la chiave di RECENSIONE.
- L'attributo utente in COMMENTO e la chiave di UTENTE.
- L'attributo recensione ospite in VALUTAZIONE e la chiave di RECENSIONE OSPITE.

- L'attributo dimensione in VALUTAZIONE e la chiave di DIMENSIONE.
- L'attributo commento in RISPOSTA e la chiave di COMMENTO.
- L'attributo utente in RISPOSTA e la chiave di UTENTE.
- L'attributo prenotazione in PRENOTAZIONE RIFUTATA e la chiave di PRENOTAZIONE.
- L'attributo prenotazione in PRENOTAZIONE CONFERMATA e la chiave di PRENOTAZIONE.
- L'attributo prenotazione confermata in PRENOTAZIONE CANCELLATA e la chiave di PRENOTAZIONE CONFERMATA.
- L'attributo prenotazione confermata in PRENOTAZIONE ACCETTATA e la chiave di PRENOTAZIONE CONFERMATA.
- L'attributo recensione in RECENSIONE HOST e la chiave di RECENSIONE.
- L'attributo recensione in RECENSIONE OSPITE e la chiave di RECENSIONE.
- L'attributo prenotazione accettata in RECENSIONE e la chiave di PRENOTAZIONE ACCETTATA.
- L'attributo prenotazione accettata in RECENSIONE e la chiave di PRENOTAZIONE ACCETTATA.

3. Implementazione

3.1 DDL di creazione del database

```
create table Utente (  
    email varchar(50) check (email like '%@%.%'),  
    password varchar(25) not null,  
    nome varchar(20) not null,  
    cognome varchar(20) not null,  
    carta_d_identità bytea,  
    verificato boolean not null,  
    constraint Utente_PK primary key(email)  
);  
  
create table Ospite (  
    utente varchar(50),  
    constraint Ospite_PK primary key(utente),  
    constraint Ospite_FK_Utente foreign key(utente) references Utente(email)  
        on update cascade on delete cascade  
);
```

```
create table Host (  
    utente varchar(50),  
    superhost boolean not null,  
    constraint Host_PK primary key(utente),  
    constraint Host_FK_Utente foreign key(utente) references Utente(email)  
        on update cascade on delete cascade  
);
```

```
create table Alloggio (  
    codice_alloggio varchar(15),  
    via varchar(30) not null,  
    numero_civico smallint not null check (numero_civico > 0),  
    comune varchar(25) not null,  
    cap integer not null check (cap >= 00000 and cap <= 99999),  
    nome varchar(30) not null,  
    descrizione text not null,  
    prezzo_per_notte float(2) not null check (prezzo_per_notte >= 0),  
    numero_di_letti smallint default 1 not null check (numero_di_letti > 0),  
    check_in time not null,  
    check_out time not null,  
    valutazione_media float (1) default 0 not null  
        check (valutazione_media >= 0 and valutazione_media <= 5),  
    numero_di_recensioni integer default 0 not null  
        check (numero_di_recensioni >= 0),  
    costi_di_pulizia float(2) not null check (costi_di_pulizia >= 0),  
    tipo_alloggio varchar(19) not null  
        check (tipo_alloggio like 'intero appartamento' or  
            tipo_alloggio like 'stanza singola' or  
            tipo_alloggio like 'stanza condivisa'),  
    host varchar(50),  
    constraint Alloggio_PK primary key(codice_alloggio),  
    constraint Host_FK_Utente foreign key(host) references Host(utente)  
        on update cascade on delete cascade  
);
```

```
create table Foto (  
    immagine bytea,  
    alloggio varchar(15),  
    constraint Foto_PK primary key(immagine),  
    constraint Foto_FK_Alloggio foreign key(alloggio) references Alloggio(codice_alloggio)  
        on update cascade on delete cascade  
);
```

```
create table Servizio (  
    nome varchar(30),  
    constraint Servizio_PK primary key(nome)  
);
```

```
create table Telefono (
    numero varchar(15),
    utente varchar(50),
    constraint Telefono_PK primary key(numero),
    constraint Telefono_FK_Utente foreign key(utente) references Utente(email)
        on update cascade on delete cascade
);

create table Prenotazione (
    id_prenotazione varchar(10),
    data_inizio date not null check (data_inizio < data_fine),
    data_fine date not null check (data_fine > data_inizio),
    numero_ospiti smallint default 0 not null check (numero_ospiti >= 0),
    costo_totale float(2) not null check (costo_totale >= 0),
    metodo_di_pagamento varchar(16) not null
    check (metodo_di_pagamento like 'carta di credito' or
        metodo_di_pagamento like 'paypal' or
        metodo_di_pagamento like 'bancomat' or
        metodo_di_pagamento like 'satispay'),
    ospite varchar(50),
    alloggio varchar(15),
    constraint Prenotazione_PK primary key(id_prenotazione),
    constraint Prenotazione_FK_Ospite foreign key(ospite) references Ospite(utente)
        on update cascade on delete cascade,
    constraint Prenotazione_FK_Alloggio foreign key(alloggio) references Alloggio(codice_alloggio)
        on update cascade on delete cascade
);

create table PrenotazioneRifiutata (
    prenotazione varchar(10),
    constraint PrenotazioneRifiutata_PK primary key(prenotazione),
    constraint PrenotazioneRifiutata_FK_Prenotazione foreign key(prenotazione)
        references Prenotazione(id_prenotazione) on update cascade on delete cascade
);

create table PrenotazioneConfermata (
    prenotazione varchar(10),
    constraint PrenotazioneConfermata_PK primary key(prenotazione),
    constraint PrenotazioneConfermata_FK_Prenotazione foreign key(prenotazione)
        references Prenotazione(id_prenotazione) on update cascade on delete cascade
);

create table PrenotazioneCancellata (
    prenotazione_confermata varchar(10),
    utente varchar(50),
    constraint PrenotazioneCancellata_PK primary key(prenotazione_confermata),
    constraint PrenotazioneCancellata_FK_PrenotazioneConfermata
        foreign key(prenotazione_confermata) references PrenotazioneConfermata(prenotazione)
        on update cascade on delete cascade,
    constraint PrenotazioneCancellata_FK_Utente
        foreign key(utente) references Utente(email)
        on update cascade on delete cascade
);
```

```
create table PrenotazioneAccettata (  
    prenotazione_confermata varchar(10),  
    constraint PrenotazioneAccettata_PK primary key(prenotazione_confermata),  
    constraint PrenotazioneAccettata_FK_PrenotazioneConfermata  
        foreign key(prenotazione_confermata) references PrenotazioneConfermata(prenotazione)  
        on update cascade on delete cascade  
);  
  
create table Recensione (  
    prenotazione_accettata varchar(10),  
    visibilità boolean not null,  
    constraint Recensione_PK primary key(prenotazione_accettata),  
    constraint Recensione_FK_PrenotazioneAccettata  
        foreign key(prenotazione_accettata) references PrenotazioneAccettata(prenotazione_confermata)  
        on update cascade on delete cascade  
);  
  
create table RecensioneHost (  
    recensione varchar(10),  
    testo_ospite text not null,  
    constraint RecensioneHost_PK primary key(recensione),  
    constraint RecensioneHost_FK_Recensione  
        foreign key(recensione) references Recensione(prenotazione_accettata)  
        on update cascade on delete cascade  
);  
  
create table RecensioneOspite (  
    recensione varchar(10),  
    testo_host text not null,  
    testo_alloggio text not null,  
    constraint RecensioneOspite_PK primary key(recensione),  
    constraint RecensioneOspite_FK_Recensione  
        foreign key(recensione) references Recensione(prenotazione_accettata)  
        on update cascade on delete cascade  
);  
  
create table Dimensione (  
    tipo varchar(20),  
    constraint Dimensione_PK primary key(tipo)  
);  
  
create table Preferenza (  
    utente varchar(50),  
    alloggio varchar(15),  
    constraint Preferenza_PK primary key(utente, alloggio),  
    constraint Preferenza_FK_Utente  
        foreign key(utente) references Utente(email)  
        on update cascade on delete cascade,  
    constraint Preferenza_FK_Alloggio  
        foreign key(alloggio) references Alloggio(codice_alloggio)  
        on update cascade on delete cascade  
);
```

```
create table Possiede (  
    servizio varchar(30),  
    alloggio varchar(15),  
    constraint Possiede_PK primary key(servizio, alloggio),  
    constraint Possiede_FK_Servizio foreign key(servizio) references Servizio(nome)  
        on update cascade on delete cascade,  
    constraint Possiede_FK_Alloggio foreign key(alloggio) references Alloggio(codice_alloggio)  
        on update cascade on delete cascade  
);
```

```
create table Ospitato (  
    prenotazione varchar(10),  
    ospite varchar(50),  
    constraint Ospitato_PK primary key(prenotazione, ospite),  
    constraint Ospitato_FK_Prenotazione  
        foreign key(prenotazione) references Prenotazione(id_prenotazione)  
        on update cascade on delete cascade,  
    constraint Ospitato_FK_Ospite  
        foreign key(ospite) references Ospite(utente)  
        on update cascade on delete cascade  
);
```

```
create table Valutazione (  
    recensione_ospite varchar(10),  
    dimensione varchar(20),  
    punteggio smallint default 1 not null check (punteggio >= 1 and punteggio <= 5),  
    constraint Valutazione_PK primary key(recensione_ospite, dimensione),  
    constraint Valutazione_FK_RecensioneOspite  
        foreign key(recensione_ospite) references RecensioneOspite(recensione)  
        on update cascade on delete cascade,  
    constraint Valutazione_FK_Dimensione  
        foreign key(dimensione) references Dimensione(tipo)  
        on update cascade on delete cascade  
);
```

```
create table Commento (  
    id_commento varchar(12),  
    testo text not null,  
    recensione varchar(10),  
    utente varchar(50),  
    constraint Commento_PK primary key(id_commento),  
    constraint Commento_FK_Recensione  
        foreign key(recensione) references Recensione(prenotazione_accettata)  
        on update cascade on delete cascade,  
    constraint Commento_FK_Utente  
        foreign key(utente) references Utente(email)  
        on update cascade on delete cascade  
);
```



```
create table Risposta (  
    commento varchar(12),  
    utente varchar(50),  
    testo text not null,  
    constraint Risposta_PK primary key(commento, utente),  
    constraint Risposta_FK_Commento  
        foreign key(commento) references Commento(id_commento)  
        on update cascade on delete cascade,  
    constraint Risposta_FK_Utente  
        foreign key(utente) references Utente(email)  
        on update cascade on delete cascade  
);
```

3.2 DML di popolamento di tutte le tabelle del database

```
insert into Utente values ('samuele.perrotta@gmail.com', 'abc34de12gb34', 'Samuele', 'Perrotta',  
    'file.SamuelePerrotta.jpg', TRUE);  
insert into Utente values ('anita.scanu@gmail.com', 'dfgby6', 'Anita', 'Scanu',  
    'file.AnitaScanu.jpg', TRUE);  
insert into Utente values ('stefano.pittavino@libero.com', 'fhgb567', 'Stefano', 'Pittavino',  
    'file.StefanoPittavino.jpg', TRUE);  
insert into Utente values ('mario.porri@gmail.com', 'abcdethb123664', 'Mario', 'Porri',  
    null, FALSE);  
insert into Utente values ('federica.marani@yahoo.com', 'afbcdetsh128934', 'Federica', 'Marani',  
    null, FALSE);  
  
insert into Ospite values ('samuele.perrotta@gmail.com');  
insert into Ospite values ('anita.scanu@gmail.com');  
insert into Ospite values ('stefano.pittavino@libero.com');  
insert into Ospite values ('federica.marani@yahoo.com');  
  
insert into Host values ('samuele.perrotta@gmail.com', TRUE);  
insert into Host values ('anita.scanu@gmail.com', FALSE);  
insert into Host values ('federica.marani@yahoo.com', TRUE);  
insert into Host values ('mario.porri@gmail.com', FALSE);  
  
insert into Telefono values ('392 0875692', 'samuele.perrotta@gmail.com');  
insert into Telefono values ('390 4568265', 'samuele.perrotta@gmail.com');  
insert into Telefono values ('392 0025874', 'anita.scanu@gmail.com');  
insert into Telefono values ('347 7893215', 'stefano.pittavino@libero.com');  
insert into Telefono values ('333 1754787', 'stefano.pittavino@libero.com');  
insert into Telefono values ('392 7412589', 'mario.porri@gmail.com');  
insert into Telefono values ('391 4565633', 'mario.porri@gmail.com');  
insert into Telefono values ('333 4883210', 'mario.porri@gmail.com');  
insert into Telefono values ('333 4787744', 'federica.marani@yahoo.com');  
  
insert into Dimensione values ('pulizia');  
insert into Dimensione values ('precisione');  
insert into Dimensione values ('comunicazione');  
insert into Dimensione values ('posizione');  
insert into Dimensione values ('check-in');  
insert into Dimensione values ('qualità/prezzo');
```

```
insert into Alloggio values ('as120801', 'via Italia', 45, 'Biella', 13900,
                             'Casa blu', 'Piccola casa in centro Biella', 64,
                             3, '10:00:00', '11:00:00', 4.34, 56, 6, 'intero appartamento',
                             'samuele.perrotta@gmail.com');
insert into Alloggio values ('ad231024', 'corso Bernardino Telesio', 26, 'Torino', 10094,
                             'Stanza Aurora', 'Ampia stanza con balcone', 28,
                             1, '9:30:00', '10:00:00', 4.67, 223, 10, 'stanza singola',
                             'anita.scanu@gmail.com');
insert into Alloggio values ('gt212234', 'via dei Mille', 77, 'Bologna', 10121,
                             'Appartamento Garibaldi', 'Appartamento luminoso al 5° piano', 53,
                             2, '10:00:00', '12:00:00', 4.77, 128, 12, 'intero appartamento',
                             'federica.marani@yahoo.com');
insert into Alloggio values ('ui124311', 'Ladbroke Grove', 123, 'Londra', 23111,
                             'Stanza rosa', 'Letto in dormitorio femminile', 40,
                             2, '9:30:00', '18:30:00', 4.23, 28, 8, 'stanza condivisa',
                             'mario.porri@gmail.com');
insert into Alloggio values ('er977473', 'via Umberto II', 298, 'Milano', 20068,
                             'Stanza condivisa', 'Stanza con letto a una piazza e mezza', 72,
                             1, '11:00:00', '11:00:00', 4.33, 112, 6, 'stanza singola',
                             'anita.scanu@gmail.com');

insert into Servizio values ('Wi-fi');
insert into Servizio values ('Aria condizionata');
insert into Servizio values ('Cucina');
insert into Servizio values ('Lavatrice');
insert into Servizio values ('Parcheggio');
insert into Servizio values ('Riscaldamento');
insert into Servizio values ('TV');
insert into Servizio values ('Piscina');

insert into Foto values ('file.fotocucina123.jpeg', 'as120801');
insert into Foto values ('file.fotocameraletto123.jpeg', 'as120801');
insert into Foto values ('file.fotosoggiorno123.jpeg', 'as120801');
insert into Foto values ('file.foto_stanza.png', 'ad231024');
insert into Foto values ('file.immagine_cameraletto.png', 'gt212234');
insert into Foto values ('file.immagine_soggiorno.png', 'gt212234');
insert into Foto values ('file.photo_cameraletto.png', 'ui124311');
insert into Foto values ('file.stanza_condivisa.png', 'er977473');

insert into Prenotazione values ('dyd65847', '2019\05\21', '2019\05\31', 3, 654.41,
                                 'paypal', 'samuele.perrotta@gmail.com', 'as120801');
insert into Prenotazione values ('rrt65489', '2020\08\10', '2020\09\15', 0, 522,
                                 'carta di credito', 'stefano.pittavino@libero.com', 'gt212234');
insert into Prenotazione values ('hhh45982', '2017\12\03', '2017\12\27', 1, 456.23,
                                 'paypal', 'anita.scanu@gmail.com', 'er977473');
insert into Prenotazione values ('ejk45656', '2022\06\15', '2022\06\30', 5, 1223.78,
                                 'bancomat', 'stefano.pittavino@libero.com', 'ad231024');
insert into Prenotazione values ('wih01789', '2020\01\01', '2020\01\31', 1, 458,
                                 'satispay', 'federica.marani@yahoo.com', 'ad231024');

insert into PrenotazioneRifiutata values ('hhh45982');
```

```
insert into PrenotazioneConfermata values ('dyd65847');
insert into PrenotazioneConfermata values ('rrt65489');
insert into PrenotazioneConfermata values ('ejk45656');
insert into PrenotazioneConfermata values ('wih01789');

insert into PrenotazioneCancellata values ('dyd65847', 'samuele.perrotta@gmail.com');

insert into PrenotazioneAccettata values ('rrt65489');
insert into PrenotazioneAccettata values ('ejk45656');
insert into PrenotazioneAccettata values ('wih01789');

insert into Possiede values ('Wi-fi','as120801');
insert into Possiede values ('Riscaldamento','as120801');
insert into Possiede values ('TV','as120801');
insert into Possiede values ('Lavatrice','as120801');
insert into Possiede values ('Cucina','as120801');
insert into Possiede values ('Riscaldamento','ad231024');
insert into Possiede values ('TV','ad231024');
insert into Possiede values ('Wi-fi','ad231024');
insert into Possiede values ('Cucina','ad231024');
insert into Possiede values ('Aria condizionata','gt212234');
insert into Possiede values ('Parcheggio','gt212234');
insert into Possiede values ('TV','gt212234');
insert into Possiede values ('Wi-fi','gt212234');
insert into Possiede values ('Aria condizionata','ui124311');
insert into Possiede values ('Riscaldamento','ui124311');
insert into Possiede values ('Cucina','er977473');
insert into Possiede values ('Wi-fi','er977473');
insert into Possiede values ('Aria condizionata','er977473');
insert into Possiede values ('Piscina','er977473');

insert into Preferenza values ('anita.scanu@gmail.com','as120801');
insert into Preferenza values ('anita.scanu@gmail.com','ui124311');
insert into Preferenza values ('samuele.perrotta@gmail.com','ad231024');
insert into Preferenza values ('samuele.perrotta@gmail.com','ui124311');
insert into Preferenza values ('stefano.pittavino@libero.com','as120801');
insert into Preferenza values ('stefano.pittavino@libero.com','ui124311');
insert into Preferenza values ('federica.marani@yahoo.com','er977473');
insert into Preferenza values ('federica.marani@yahoo.com','as120801');

insert into Ospitato values ('dyd65847','anita.scanu@gmail.com');
insert into Ospitato values ('dyd65847','stefano.pittavino@libero.com');
insert into Ospitato values ('rrt65489','federica.marani@yahoo.com');

insert into Recensione values ('rrt65489', FALSE);
insert into Recensione values ('ejk45656', FALSE);
insert into Recensione values ('wih01789', TRUE);
```

```
insert into RecensioneHost values ('rrt65489', 'Ospite molto educato e gentile.');
```

```
insert into RecensioneHost values ('ejk45656', 'Ospiti scortesi e insolenti!');
```

```
insert into RecensioneHost values ('wih01789', 'Ospite gentile e di buone maniere.');
```

```
insert into RecensioneOspite values ('rrt65489', 'Host molto gentile e disponibile.',  
                                     'Struttura molto bella e qualità/prezzo ottimo, la consiglio!');
```

```
insert into RecensioneOspite values ('ejk45656', 'Scortesia e maleducazione da parte della proprietaria',  
                                     'Alloggio tenuto male e trascurato, a mai più.');
```

```
insert into RecensioneOspite values ('wih01789', 'Gentilezza della proprietaria top.',  
                                     'Ottima posizione, ci riverrò sicuramente. La stanza era molto calda  
                                     e faceva caldo, ma peccato che non si possa controllare la  
                                     temperatura della caldaia (forse non sono riuscita a trovarla)');
```

```
insert into Valutazione values ('rrt65489', 'pulizia', 4);
```

```
insert into Valutazione values ('rrt65489', 'precisione', 5);
```

```
insert into Valutazione values ('rrt65489', 'comunicazione', 5);
```

```
insert into Valutazione values ('rrt65489', 'posizione', 4);
```

```
insert into Valutazione values ('rrt65489', 'check-in', 5);
```

```
insert into Valutazione values ('rrt65489', 'qualità/prezzo', 5);
```

```
insert into Valutazione values ('ejk45656', 'pulizia', 1);
```

```
insert into Valutazione values ('ejk45656', 'precisione', 1);
```

```
insert into Valutazione values ('ejk45656', 'comunicazione', 1);
```

```
insert into Valutazione values ('ejk45656', 'posizione', 2);
```

```
insert into Valutazione values ('ejk45656', 'check-in', 2);
```

```
insert into Valutazione values ('ejk45656', 'qualità/prezzo', 1);
```

```
insert into Valutazione values ('wih01789', 'pulizia', 3);
```

```
insert into Valutazione values ('wih01789', 'precisione', 5);
```

```
insert into Valutazione values ('wih01789', 'comunicazione', 5);
```

```
insert into Valutazione values ('wih01789', 'posizione', 5);
```

```
insert into Valutazione values ('wih01789', 'check-in', 4);
```

```
insert into Valutazione values ('wih01789', 'qualità/prezzo', 4);
```

```
insert into Commento values ('sdc3213', 'Grazie per il tuo soggiorno',  
                             'rrt65489', 'federica.marani@yahoo.com');
```

```
insert into Commento values ('kfj3210', 'Ciao. Ti avrei informato subito  
se avessi avuto una richiesta, ma mi dispiace  
che tu possa essere stato a disagio. Quando  
ci visiterai la prossima volta, se hai domande,  
ti preghiamo di comunicarcelo e ti risponderemo  
gentilmente.', 'wih01789', 'anita.scanu@gmail.com');
```

```
insert into Commento values ('flr8512', 'Anche io ho riscontrato lo stesso  
problema', 'wih01789', 'samuele.perrotta@gmail.com');
```

```
insert into Risposta values ('kfj3210', 'federica.marani@yahoo.com',  
                             'Perfetto, grazie mille, la prossima volta le farò sapere!!');
```

```
insert into Risposta values ('flr8512', 'anita.scanu@gmail.com',  
                             'Mi spiace del disagio, la prossima volta non esiti a farci  
                             domande');
```

3.3 Qualche operazione di cancellazione e modifica

3.3.1 Operazioni di cancellazione

- Cancellazione degli alloggi il cui comune è Bologna:

```
delete from Alloggio where comune like 'Bologna';
```

Cancellando la tupla nella tabella Alloggio, che ha come valore dell'attributo comune 'Bologna', vengono cancellate di conseguenza tutte le tuple nel database ad esso collegate con vincoli d'integrità referenziale perchè abbiamo aggiunto la clausola on delete cascade a tutti i vincoli di chiave esterna.

- Cancellazione di una foto di un determinato alloggio:

```
delete from Foto where alloggio like 'as120801';
```

Cancellando la tupla nella tabella Foto, che ha come valore dell'attributo alloggio 'as120801', non vengono cancellate altre tuple nel database perchè nessun attributo ha un vincolo d'integrità referenziale con la tabella Foto.

3.3.2 Operazioni di modifica

- Modifica dell'attributo email nella tabella Utente:

```
update Utente
set email = 'm.por@libero.com'
where email like 'mario.porri@gmail.com';
```

Aggiornando l'attributo email nella tabella Utente, vengono aggiornate di conseguenza tutte le email delle tabelle esterne che hanno un vincolo d'integrità referenziale con l'attributo email in Utente per via della clausola on update cascade.

- Modifica dell'attributo punteggio nella tabella Valutazione:

```
update Valutazione
set punteggio = 5
where recensione_ospite like 'rrt65489' and dimensione like 'pulizia'
and punteggio = 4;
```

Aggiornando l'attributo punteggio nella tabella Valutazione, non vengono aggiornati altri valori perchè nessun'altra tabella ha un vincolo d'integrità referenziale con un attributo di Valutazione.