



Basi di Dati
Progetto A.A. 2022/2023

AGENZIA DI VIAGGIO

0294136

Andrea Cantarini

Indice

1. Descrizione del Minimondo.....	2
2. Analisi dei Requisiti.....	3
3. Progettazione concettuale.....	8
4. Progettazione logica.....	12
5. Progettazione fisica.....	17
6. Appendice: implementazione.....	27

1. Descrizione del Minimondo

1	Un'agenzia di viaggio organizza viaggi organizzati della durata da un giorno ad una settimana in località italiane ed europee. I viaggi utilizzano autobus privati.
2	La compagnia intende realizzare un sistema informativo per la gestione dei viaggi e delle prenotazioni.
3	La segreteria dell'agenzia crea nuovi itinerari di viaggio. Un itinerario è caratterizzato da una data di partenza, da una data di rientro, da un insieme di pernottamenti in località potenzialmente differenti e di durata differente e da un costo.
4	Per ciascun pernottamento è di interesse tenere traccia di quale sia l'albergo scelto per ospitare i partecipanti. La scelta avviene all'interno di un insieme di alberghi mantenuto nel sistema informativo. Ciascun albergo è associato ad una capienza massima di persone, una città, un indirizzo, un insieme di recapiti (email, telefono, fax) e un referente.
5	Quando un piano di viaggio viene creato, questo è definito a partire dalle tappe che verranno coperte. Gli utenti del sistema possono prenotarsi al viaggio specificando il numero di persone per cui effettuano la prenotazione. Al termine della prenotazione viene fornito un codice che può essere utilizzato per disdire la prenotazione, fino a 20 giorni prima della partenza.
6	Quando le prenotazioni sono concluse (ovverosia, 20 giorni prima della partenza), la segreteria è a conoscenza di quante persone parteciperanno al viaggio. Assegnano quindi un insieme di autobus per consentire il trasporto di tutte le persone. Un autobus può infatti avere una capienza differente.
7	Una volta definito il numero di autobus necessari, la segreteria, per ciascuna città toccata dal viaggio, associa un albergo che abbia una disponibilità massima di posti superiore alle persone che effettuano il viaggio. Ciascun hotel ha un costo per notte per persona che è conservato nel sistema.
8	Per tutti i viaggi terminati, la segreteria può generare un report che mostri le informazioni sui partecipanti e il guadagno derivante da tale viaggio.

2. Analisi dei Requisiti

Identificazione dei termini ambigui e correzioni possibili

Linea	Termine	Nuovo termine	Motivo correzione
1	Organizzati	-	Rimozione termine superfluo
2	Compagnia	Agenzia	“Compagnia” e “Agenzia” fanno riferimento allo stesso ente
4, 7	Città	Località	“Città” e “Località” possono essere intesi allo stesso modo
5	Piano di viaggio	Itinerario	Un piano di viaggio può essere identificato come itinerario
6	Assegnano	Assegna	Correzione verbo

Specifica disambiguata

Un'agenzia di viaggio organizza viaggi della durata da un giorno ad una settimana in località italiane ed europee. I viaggi utilizzano autobus privati.

L'agenzia intende realizzare un sistema informativo per la gestione dei viaggi e delle prenotazioni.

La segreteria dell'agenzia crea nuovi itinerari di viaggio. Un itinerario è caratterizzato da una data di partenza, da una data di rientro, da un insieme di pernottamenti in località potenzialmente differenti e di durata differente e da un costo.

Per ciascun pernottamento è di interesse tenere traccia di quale sia l'albergo scelto per ospitare i partecipanti. La scelta avviene all'interno di un insieme di alberghi mantenuto nel sistema informativo. Ciascun albergo è associato ad una capienza massima di persone, una località, un indirizzo, un insieme di recapiti (email, telefono, fax) e un referente.

Quando un piano di viaggio viene creato, questo è definito a partire dalle tappe che verranno coperte. Gli utenti del sistema possono prenotarsi al viaggio specificando il numero di persone per cui effettuano la prenotazione. Al termine della prenotazione viene fornito un codice che può essere utilizzato per disdire la prenotazione, fino a 20 giorni prima della partenza.

Quando le prenotazioni sono concluse (ovverosia, 20 giorni prima della partenza), la segreteria è a conoscenza di quante persone parteciperanno al viaggio. Assegna quindi un insieme di autobus per consentire il trasporto di tutte le persone. Un autobus può infatti avere una capienza differente.

Una volta definito il numero di autobus necessari, la segreteria, per ciascuna località toccata dal viaggio, associa un albergo che abbia una disponibilità massima di posti superiore alle persone che effettuano il viaggio. Ciascun albergo ha un costo per notte per persona che è conservato nel sistema.

Per tutti i viaggi terminati, la segreteria può generare un report che mostri le informazioni sui partecipanti e il guadagno derivante da tale viaggio.

Glossario dei Termini

Termine	Descrizione	Sinonimi	Collegamenti
Viaggio	Percorso di visita a cui partecipa un numero finito di persone, definito da un itinerario		Piano di viaggio, Prenotazione
Itinerario	Successione ordinata di pernottamenti in località potenzialmente diverse che definiscono un viaggio	Piano, piano di viaggio, itinerario di viaggio	Viaggio, Segreteria, Tappa
Località	Luogo toccato dall'itinerario in una tappa	Città, Tappa	Tappa, Albergo
Pernottamento	La residenza, di durata variabile, in un albergo		Albergo, Piano di viaggio
Albergo	Struttura atta ad ospitare i partecipanti ad un viaggio per un pernottamento	Hotel	Pernottamento
Prenotazione	La riserva un numero finito di posti in un viaggio, eseguito da un cliente dell'agenzia, al fine di parteciparvi		Viaggio
Agenzia di viaggio	Committente del progetto	Agenzia	Segreteria
Segreteria	Organo dell'agenzia di viaggio atto all'organizzazione dei viaggi		Agenzia di viaggio, Piano di viaggio
Autobus	Mezzo di trasporto con una capienza massima di persone, assegnato, insieme ad altri, ad un viaggio in base al numero di prenotati		Viaggio
Report	Insiemi di dati (numero di partecipanti, prenotazioni, tappe, guadagno) relativi ad un itinerario concluso, generato dalla segreteria		Segreteria

Raggruppamento dei requisiti in insiemi omogenei

Frase relative a *Viaggio*

Un'agenzia di viaggio organizza viaggi della durata da un giorno ad una settimana in località italiane ed europee. I viaggi utilizzano autobus privati.

L'agenzia intende realizzare un sistema informativo per la gestione dei viaggi e delle prenotazioni.

La segreteria dell'agenzia crea nuovi itinerari di viaggio.

Gli utenti del sistema possono prenotarsi al viaggio specificando il numero di persone per cui effettuano la prenotazione.

Quando le prenotazioni sono concluse (ovverosia, 20 giorni prima della partenza), la segreteria è a conoscenza di quante persone parteciperanno al viaggio.

Una volta definito il numero di autobus necessari, la segreteria, per ciascuna città toccata dal viaggio, associa un albergo che abbia una disponibilità massima di posti superiore alle persone che effettuano il viaggio.

Per tutti i viaggi terminati, la segreteria può generare un report che mostri le informazioni sui partecipanti e il guadagno derivante da tale viaggio.

Frase relative a *Itinerario*

La segreteria dell'agenzia crea nuovi itinerari di viaggio. Un itinerario è caratterizzato da una data di partenza, da una data di rientro, da un insieme di pernottamenti in località potenzialmente differenti e di durata differente e da un costo.

Quando un piano di viaggio viene creato, questo è definito a partire dalle tappe che verranno coperte.

Frase relative a *Località*

Un'agenzia di viaggio organizza viaggi della durata da un giorno ad una settimana in località italiane ed europee.

Un itinerario è caratterizzato da una data di partenza, da una data di rientro, da un insieme di pernottamenti in località potenzialmente differenti e di durata differente e da un costo.

Ciascun albergo è associato ad una capienza massima di persone, una località, un indirizzo, un insieme di recapiti (email, telefono, fax) e un referente.

Una volta definito il numero di autobus necessari, la segreteria, per ciascuna località toccata dal viaggio, associa un albergo che abbia una disponibilità massima di posti superiore alle persone che effettuano il viaggio.

Frase relative a *Pernottamento*

Un itinerario è caratterizzato da una data di partenza, da una data di rientro, da un insieme di pernottamenti in località potenzialmente differenti e di durata differente e da un costo.

Per ciascun pernottamento è di interesse tenere traccia di quale sia l'albergo scelto per ospitare i partecipanti.

Frase relative a *Albergo*

Per ciascun pernottamento è di interesse tenere traccia di quale sia l'albergo scelto per ospitare i partecipanti. Ciascun albergo è associato ad una capienza massima di persone, una città, un indirizzo, un insieme di recapiti (email, telefono, fax) e un referente.

Una volta definito il numero di autobus necessari, la segreteria, per ciascuna città toccata dal viaggio, associa un albergo che abbia una disponibilità massima di posti superiore alle persone che effettuano il viaggio. Ciascun albergo ha un costo per notte per persona che è conservato nel sistema.

Frase relative a *Prenotazione*

L'agenzia intende realizzare un sistema informativo per la gestione dei viaggi e delle prenotazioni.

Gli utenti del sistema possono prenotarsi al viaggio specificando il numero di persone per cui effettuano la prenotazione. Al termine della prenotazione viene fornito un codice che può essere utilizzato per disdire la prenotazione, fino a 20 giorni prima della partenza.

Quando le prenotazioni sono concluse (ovverosia, 20 giorni prima della partenza), la segreteria è a conoscenza di quante persone parteciperanno al viaggio.

Frase relative a *Agenzia di viaggio*

Un'agenzia di viaggio organizza viaggi della durata da un giorno ad una settimana in località italiane ed europee.

L'agenzia intende realizzare un sistema informativo per la gestione dei viaggi e delle prenotazioni.

Frase relative a *Segreteria*

La segreteria dell'agenzia crea nuovi itinerari di viaggio.

Quando le prenotazioni sono concluse (ovverosia, 20 giorni prima della partenza), la segreteria è a conoscenza di quante persone parteciperanno al viaggio. Assegna quindi un insieme di autobus per consentire il trasporto di tutte le persone.

Una volta definito il numero di autobus necessari, la segreteria, per ciascuna città toccata dal viaggio, associa un albergo che abbia una disponibilità massima di posti superiore alle persone che effettuano il viaggio.

Per tutti i viaggi terminati, la segreteria può generare un report che mostri le informazioni sui partecipanti e il guadagno derivante da tale viaggi

Frase relative a *Report*

Per tutti i viaggi terminati, la segreteria può generare un report che mostri le informazioni sui partecipanti e il guadagno derivante da tale viaggio.

Frase relative a *Autobus*

I viaggi utilizzano autobus privati.

[La segreteria] Assegna quindi un insieme di autobus per consentire il trasporto di tutte le persone. Un autobus può infatti avere una capienza differente.

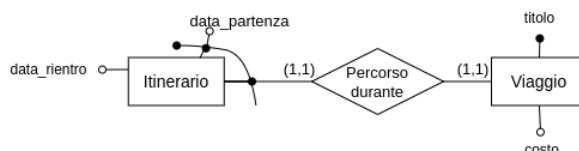
Una volta definito il numero di autobus necessari, la segreteria, per ciascuna città toccata dal viaggio, associa un albergo che abbia una disponibilità massima di posti superiore alle persone che effettuano il viaggio.

3. Progettazione concettuale

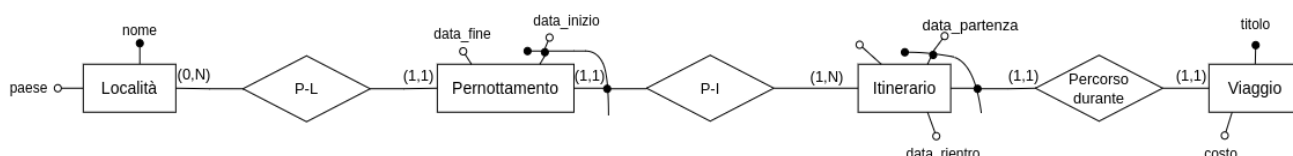
Costruzione dello schema E-R

Integrazione finale

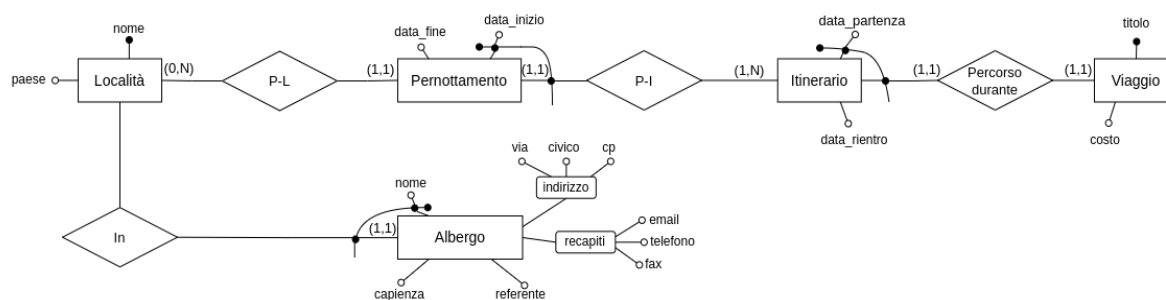
1. Ad un viaggio è associato sempre e solo un itinerario



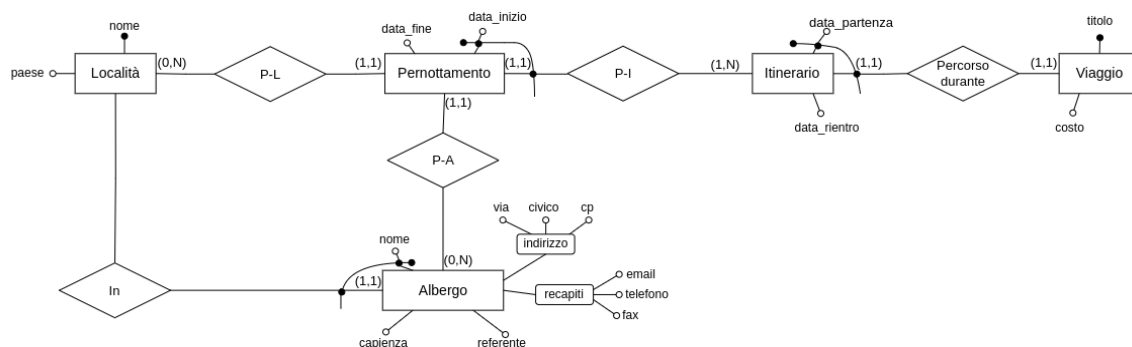
2. Un itinerario è composto da più pernottamenti, ognuno in una località potenzialmente diversa. In una località possono essere effettuati più pernottamenti, anche relativi allo stesso viaggio, ma possono esistere località nel sistema che ancora non sono state toccate da alcun viaggio. Al contrario, un itinerario deve essere composto da almeno un pernottamento.



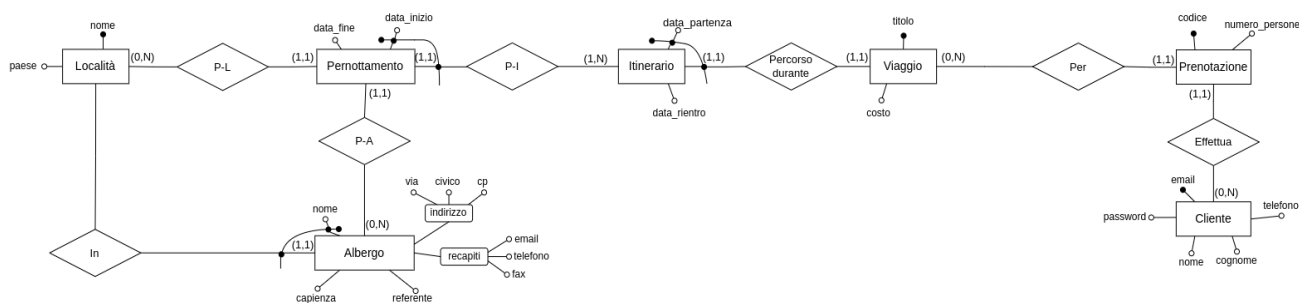
3. C'è almeno un albergo in ogni località presente nel sistema. Ogni albergo deve essere associato alla relativa località dove si trova e può essere identificato tramite il nome e la località.



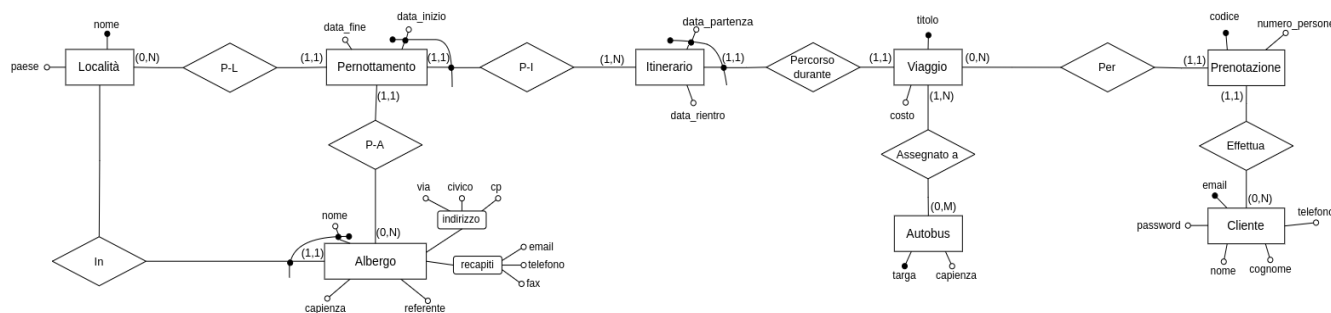
4. Un pernottamento si effettua in un albergo per ogni città toccata dall'itinerario, ma viene scelto a seguito della definizione di quest'ultimo, in base al numero dei partecipanti al viaggio. Occorre dunque mantenere sia la relazione P-A, sia la relazione In. In un albergo possono essere effettuati più pernottamenti, anche relativi allo stesso viaggio, ma possono esistere nel sistema anche alberghi in cui non è ancora stato effettuato alcun pernottamento.



5. Una prenotazione viene effettuata per un solo viaggio. Al contrario, un viaggio può avere più prenotazioni, oppure non averne affatto.



6. Uno o più autobus vengono assegnati ad un viaggio in base al numero di prenotati. Possono esistere degli autobus che non sono ancora stati assegnati ad alcun viaggio. Un autobus può essere assegnato a più viaggi, con l'accortezza che questi non abbiano date sovrapposte (vedi regole aziendali).



Regole aziendali

- (RV1) La data di partenza di un itinerario deve coincidere con la data del primo pernottamento che lo compone.
- (RV2) La data di rientro di un itinerario deve corrispondere alla data di fine dell'ultimo pernottamento che lo compone.
- (RV3) Le prenotazioni per un viaggio non devono essere effettuate nei 20 giorni antecedenti la data di partenza stabilita del relativo itinerario.

(RV4) Nei 20 giorni antecedenti la partenza, la segreteria dell'agenzia deve assegnare gli autobus al viaggio e selezionare un albergo per ogni pernottamento.

(RV5) Ogni autobus non deve essere associato a più viaggi che abbiano itinerari con date che si sovrappongono.

(RV6) Un albergo scelto per un pernottamento deve avere la disponibilità di ospitare il numero di persone che partecipano al viaggio di cui fa parte il pernottamento.

(RV7) L'insieme degli autobus associati ad un viaggio deve avere una capienza totale superiore al numero di partecipanti al viaggio.

(RV8) Un cliente non deve avere come numero di telefono il numero associato ad un altro cliente già presente nel sistema.

(RD1) Il numero di partecipanti ad un viaggio si ottiene sommando il numero di prenotati specificati in ogni prenotazione effettuata per tale viaggio.

(RD2) Il guadagno ottenuto da un viaggio si ottiene dal numero di partecipanti per il costo del viaggio.

(RD3) Il report per un viaggio si ottiene unendo le informazioni relative al viaggio (pernottamenti e relative durate), le prenotazioni e il guadagno.

(RD4) La capienza totale di un insieme di autobus si ottiene sommando la capienza dei singoli autobus nell'insieme.

Dizionario dei dati

Entità	Descrizione	Attributi	Identificatori
Viaggio	Viaggio organizzato dall'agenzia	<ul style="list-style-type: none"> costo titolo 	<ul style="list-style-type: none"> titolo
Itinerario	Itinerario percorso durante un viaggio	<ul style="list-style-type: none"> data_partenza data_rientro 	<ul style="list-style-type: none"> data_partenza Viaggio
Prenotazione	Prenotazione effettuata da un cliente per un viaggio	<ul style="list-style-type: none"> codice cliente numero_persone 	<ul style="list-style-type: none"> codice
Pernottamento	Pernottamento effettuato durante un itinerario di viaggio	<ul style="list-style-type: none"> data_inizio data_fine 	<ul style="list-style-type: none"> data_inizio itinerario
Località	Tappa di un viaggio dove viene effettuato un pernottamento	<ul style="list-style-type: none"> nome paese 	<ul style="list-style-type: none"> nome
Albergo	Struttura in cui viene effettuato un pernottamento	<ul style="list-style-type: none"> nome capienza referente indirizzo (via, civico, cp) recapiti (email, telefono, fax) località 	<ul style="list-style-type: none"> nome località

Autobus	Autobus privato utilizzato per trasportare i clienti durante i viaggi	<ul style="list-style-type: none"> • targa • capienza 	<ul style="list-style-type: none"> • targa
Cliente	Cliente dell'agenzia	<ul style="list-style-type: none"> • nome • cognome • email • telefono • password 	<ul style="list-style-type: none"> • email

Relazione	Descrizione	Entità coinvolte	Attributi
In	Descrive la presenza di un albergo in una determinata località	<ul style="list-style-type: none"> • Albergo • Località 	
Effettuata per	Associa una prenotazione ad un viaggio specifico	<ul style="list-style-type: none"> • Viaggio • Prenotazione 	
Assegnato a	Descrive l'assegnazione di un autobus ad un viaggio	<ul style="list-style-type: none"> • Viaggio • Autobus 	
P-I	Descrive l'appartenenza di un pernottamento ad un itinerario	<ul style="list-style-type: none"> • Pernottamento • Itinerario 	
P-L	Indica il luogo in cui si effettua un pernottamento	<ul style="list-style-type: none"> • Pernottamento • Località 	
P-A	Indica la struttura in cui di effettua un pernottamento	<ul style="list-style-type: none"> • Pernottamento • Albergo 	

4. Progettazione logica

Volume dei dati

I dati riportati sono valutati nell'arco di quattro anni. Inoltre, si suppone che l'agenzia organizzi in media un viaggio ogni due settimane e che ogni viaggio abbia un numero medio di 100 partecipanti. Si suppone inoltre che le prenotazioni vengano effettuate in media per 4 persone. I viaggi hanno una media di 3 pernottamenti. Per ogni località mantenuta nel sistema, si presentano in media due alberghi.

Concetto nello schema	Tipo ¹	Volume atteso
Viaggio	E	100
Itinerario	E	100
Prenotazione	E	2500
Autobus	E	25
Località	E	500
Albergo	E	1000
Cliente	E	2000
Pernottamento	E	300
Percorso durante	R	100
Per	R	2500
In	R	1000
Assegnato a	R	500
P-I	R	300
P-L	R	300
P-A	R	300

Tavola delle operazioni

Cod.	Descrizione	Frequenza attesa
V1	Crea un nuovo viaggio	2/mese
V2	Inserisci un pernottamento per un viaggio	6/mese
V3	Inserisci una nuova località	10/mese
V4	Inserisci un nuovo albergo	20/mese
E1	Elenca tutte le località	10/mese
E2	Elenca tutti gli alberghi	10/mese
P1	Prenota un viaggio	1/giorno
P2	Cancella prenotazione	3/mese
A1	Assegna degli autobus ad un viaggio	2/mese
A2	Scegli un albergo per un pernottamento	6/mese
I1	Elenca tutti i viaggi	10/giorno
I2	Elenca i dettagli di un viaggio	100/giorno
L1	Login	30/giorno
L2	Registrazione	1/giorno
R1	Crea il report	2/mese

¹ Indicare con E le entità, con R le relazioni

Costo delle operazioni

V1 – Crea un nuovo viaggio

- Viaggio: 200
- Itinerario: 200
- Percorso durante: 200

Costo totale: 600

Accessi/mese: 1200

Accorpare le entità Viaggio ed Itinerario (ed eliminando di conseguenza la relazione Percorso durante), il costo dell'operazione diminuisce a 200 e il numero di accessi al mese diviene 400.

V2 – Inserisci un pernottamento per un viaggio

- Pernottamento: 600
- Località: 500
- P-L: 600
- P-I: 600

Costo totale: 2300

Accessi/mese: 13800

V3 – Inserisci una nuova località

- Località: 1000

Costo totale: 1000

Accessi/mese: 10000

V4 – Elenca tutti gli alberghi per una località

- Albergo: 1000
- In: 1000

Costo totale: 2000

Accessi/giorno: 20000

P1 – Prenota un viaggio

- Prenotazione: 5000
- Per: 5000

Costo totale: 10000

Accessi/giorno: 10000

P2 – Cancella prenotazione

- Prenotazione: 5000
- Per: 5000

Costo totale: 10000

Accessi/mese: 30000

A1 – Assegna un autobus a un viaggio

- Autobus: 25
- Assegnato a: 1400

Costo totale: 1450

Accessi/mese: 2900

A2 – Scegli un albergo per un pernottamento

- Albergo: 1000
- Località: 500
- P-L: 300
- Pernottamento: 300
- P-A: 600

Costo totale: 2700

Accessi/mese: 16200

I1 – Elenca tutti i viaggi

- Viaggio: 100

Costo totale: 100

Accessi/giorno: 1000

I2 – Elenca i dettagli di un viaggio

- Viaggio: 100
- Itinerario: 100
- Percorso durante: 100
- Pernottamento: 300
- Località: 500
- P-L: 300
- P-I: 300

Costo totale: 1700

Accessi/giorno: 170000

Accorpare le entità viaggio ed itinerario, si ha un costo totale di 1500 e un numero di accessi/giorno pari a 150000.

L1 – Login

- Cliente: 2000

Costo totale: 2000

Accessi/giorno: 60000

L2 – Registrazione

- Cliente: 4000

Costo totale: 4000

Accessi/giorno: 4000

R1 – Crea il report

- Viaggio: 100
- Itinerario: 100
- Percorso durante: 100
- Pernottamento: 300
- P-I: 300
- Località: 500
- P-L: 300
- Albero: 1000
- P-A: 300
- In: 1000
- Prenotazione: 2500
- Per: 2500
- Autobus: 25
- Assegnato a: 700

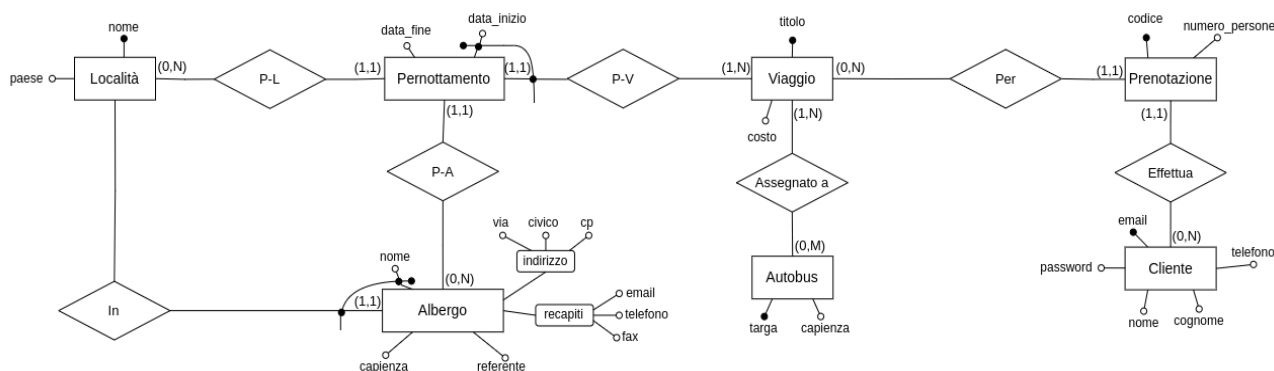
Costo totale: 9725

Accessi/mese: 19450

Accorpare le entità Viaggio ed Itinerario (eliminando la relazione Percorso durante), il costo si riduce a 9525 e gli accessi al mese a 19050.

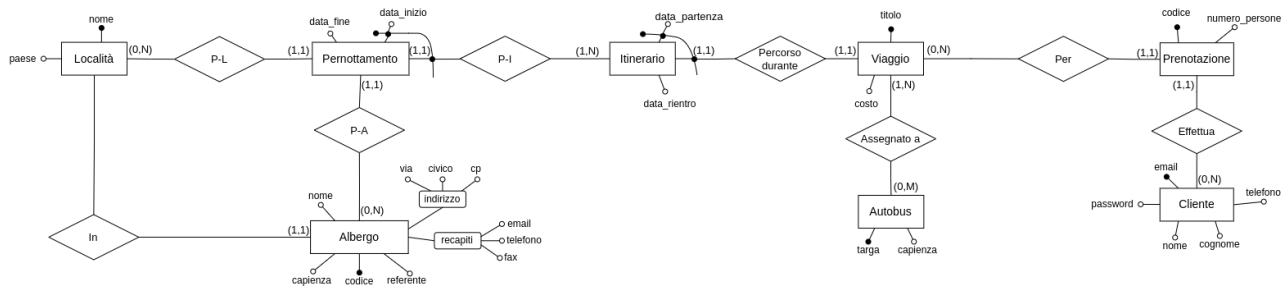
Ristrutturazione dello schema E-R

Le entità Viaggio ed Itinerario vengono accorpate in un'unica entità Viaggio, eliminando di conseguenza la relazione Percorso durante. La relazione P-I viene rinominata in P-V.

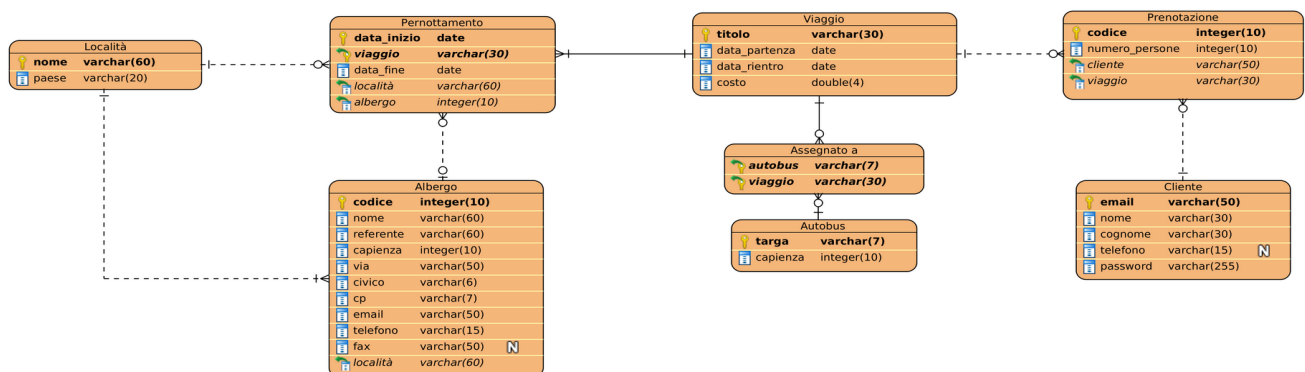


Trasformazione di attributi e identificatori

L'identificatore dell'entità Albero è composto da un identificatore esterno verso l'entità Località. Poiché non è possibile considerare il nome di un albergo univoco nel sistema, si decide di introdurre un attributo 'codice' nell'entità Albero da utilizzare come identificatore.



Traduzione di entità e associazioni



Normalizzazione del modello relazionale

Il modello non è in forma normale. Infatti, nella relazione Pernottamento, il codice dell'albergo identifica anche la località in cui si effettua il pernottamento. Tuttavia, per ragioni prestazionali, si decide di non normalizzare il modello.

5. Progettazione fisica

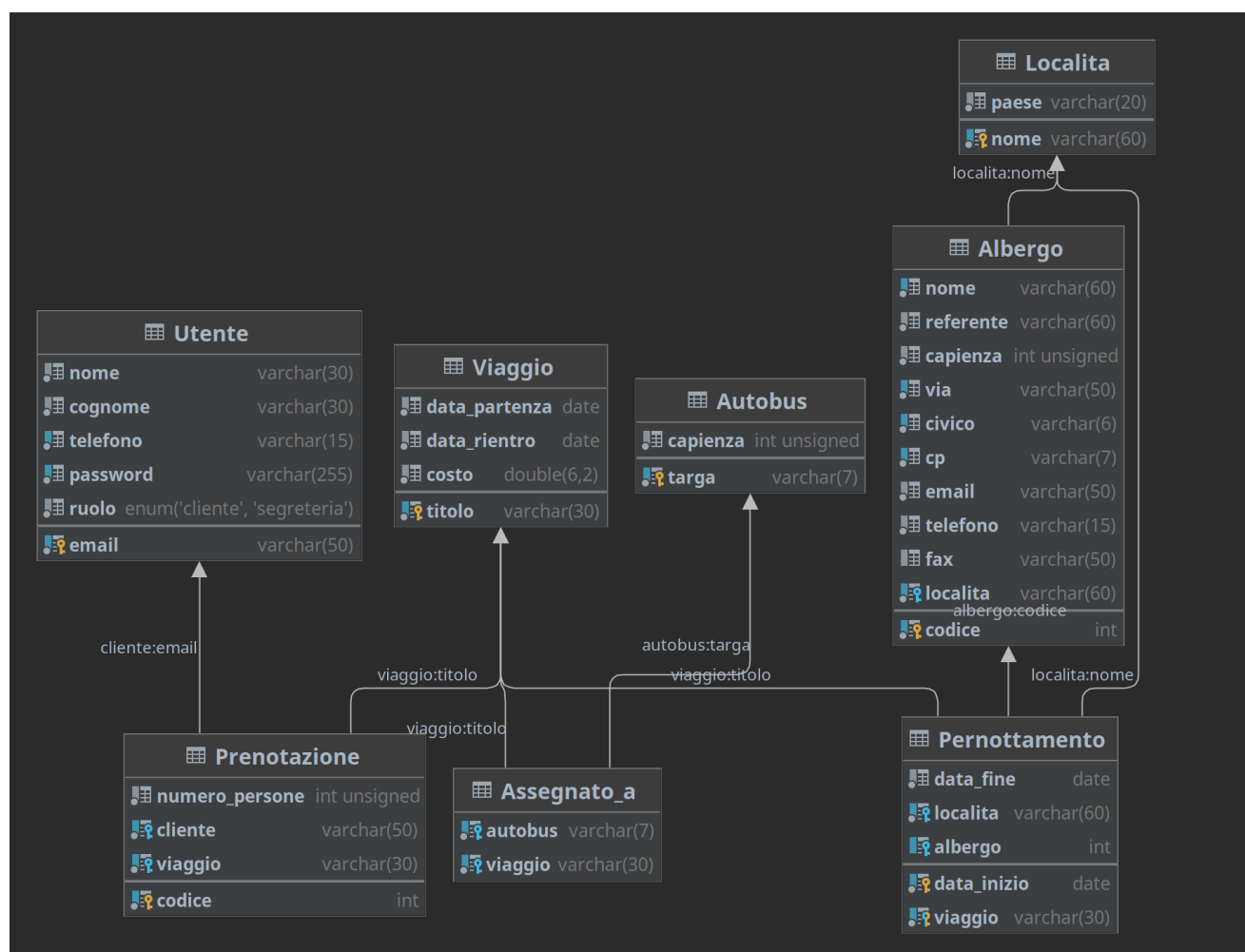
Utenti e privilegi

Descrivere, all'interno dell'applicazione, quali utenti sono stati previsti con quali privilegi di accesso su quali tabelle, giustificando le scelte progettuali.

Si decide di introdurre due tipologie di utenti per l'applicazione:

- 'Cliente'
 - Grant di esecuzione sulle operazioni P1, P2, I1, I2, L1, L2
- 'Segreteria'
 - Grant di esecuzione sulle operazioni V1, V2, V3, V4, A1, A2, R1, I1, I2, L1, L2, E1, E2, P1, P2

Per identificare gli utenti, si modifica la tabella Cliente, rinominata in Utente, a cui si aggiunge un attributo ruolo



Strutture di memorizzazione

Tabella Viaggio		
Colonna	Tipo di dato	Attributi ²
titolo	varchar(30)	PK
data_partenza	date	NN
data_rientro	date	NN
costo	double(4,2)	NN

Tabella Pernottamento		
Colonna	Tipo di dato	Attributi
data_inizio	date	PK
viaggio	varchar(30)	PK
data_fine	date	NN
località	varchar(60)	NN
albergo	int	NN

Tabella Località		
Colonna	Tipo di dato	Attributi
nome	varchar(60)	PK
paese	varchar(20)	NN

Tabella Albergo		
Colonna	Tipo di dato	Attributi
codice	int	PK, AI
nome	varchar(60)	NN
referente	varchar(60)	NN
capienza	int	UN, NN
via	varchar(50)	NN
civico	varchar(6)	NN
cp	varchar(7)	NN
email	varchar(50)	NN
telefono	varchar(15)	NN
fax	varchar(50)	
località	varchar(60)	NN

Tabella Utente		
Colonna	Tipo di dato	Attributi
email	varchar(50)	PK
nome	varchar(30)	NN
cognome	varchar(30)	NN
telefono	varchar(15)	UQ
password	varchar(255)	NN
ruolo	enum('cliente', 'segreteria')	NN

² PK = primary key, NN = not null, UQ = unique, UN = unsigned, AI = auto increment. È ovviamente possibile specificare più di un attributo per ciascuna colonna.

Tabella Prenotazione		
Colonna	Tipo di dato	Attributi
codice	int	PK, AI
numero_persones	int	UN, NN
cliente	varchar(50)	NN
viaggio	varchar(30)	NN

Tabella Autobus		
Colonna	Tipo di dato	Attributi
targa	varchar(7)	PK
capienza	int	UN

Tabella Assegnato_a		
Colonna	Tipo di dato	Attributi
autobus	varchar(7)	PK
viaggio	varchar(30)	PK

Indici

Tabella Albergo	
Indice indirizzo_univoco	Tipo ³ :
via, civico, cp, località	UQ
Indice localita_albergo	Tipo:
località, nome	IDX

Tabella Utente	
Indice login	Tipo:
email, password	IDX

Tabella Pernottamento	
Indice viaggio_pernottamento	Tipo:
data_inizio, viaggio	IDX

Tabella Prenotazione	
Indice viaggio_prenotazione	Tipo:
codice, viaggio	IDX

Trigger

Tabella Prenotazione: BEFORE DELETE

Controlla che una prenotazione per un viaggio non venga disdetta nei 20 giorni antecedenti la partenza.

```
Create trigger check_cancellation
before delete
on Prenotazione
```

³ IDX = index, UQ = unique, FT = full text, PR = primary.

```
    for each row
begin
    declare var_startDate date;
        select  Viaggio.data_partenza  from  Viaggio  join  Prenotazione  on
Viaggio.titolo=Prenotazione.viaggio into var_startDate;

    if (NOW() < var_startDate and (DATEDIFF(NOW(),var_startDate) >= 20))
        then signal sqlstate '45001' set message_text='Impossibile cancellare la
prenotazione nei venti giorni antecedenti la partenza.';
    end if;
end;
```

Tabella Prenotazione: BEFORE INSERT

Controlla che una prenotazione per un viaggio non venga effettuata nei 20 giorni antecedenti la partenza.

```
create trigger check_booking
    before delete
    on Prenotazione
    for each row
begin
    declare var_startDate date;
        select  Viaggio.data_partenza  from  Viaggio  join  Prenotazione  on
Viaggio.titolo=Prenotazione.viaggio into var_startDate;

    if (NOW() < var_startDate and DATEDIFF(NOW(),var_startDate) >= 20)
        then signal sqlstate '45001' set message_text='Impossibile effettuare una
prenotazione per questo viaggio nei venti giorni antecedenti la partenza.';
    end if;
end;
```

Tabella Assegnato_a: BEFORE INSERT

Verifica che l'autobus assegnato ad un viaggio sia disponibile per le date di quel viaggio.

```
create trigger check_assignment
    before insert
    on Assegnato_a
    for each row
begin
    declare var_startDateCur date;
    declare var_endDateCur date;
```

```
select data_partenza from Viaggio where titolo = new.viaggio into
var_startDateCur;
select data_rientro from Viaggio where titolo = new.viaggio into
var_endDateCur;

if exists(select data_partenza, data_rientro
          from Viaggio V
          join Assegnato_a Aa on V.titolo = Aa.viaggio
          where Aa.autobus = NEW.autobus
          and ((data_partenza <= var_startDateCur and data_rientro >=
var_startDateCur)
              or (data_partenza <= var_endDateCur and data_rientro >=
var_endDateCur)))
    then
        signal sqlstate '45001' set message_text = 'Questo autobus non è
disponibile per le date indicate.';
    end if;
end;
```

Tabella Viaggio: BEFORE INSERT

Controlla che la data di partenza non venga dopo quella di rientro.

```
create trigger check_dates
before insert
on Viaggio
for each row
begin
    if NEW.data_rientro < NEW.data_partenza then
        signal sqlstate '45001' set message_text = 'La data del rientro non può
precedere la data di partenza';
    end if;
end;
```

Tabella Pernottamento: BEFORE INSERT

Verifica che la data di inizio pernottamento non venga dopo la data di fine pernottamento e che la data di inizio del primo pernottamento coincida con la data di partenza del viaggio.

```
create trigger check_overnight_dates
before insert
on Pernottamento
for each row
```

```
begin

    declare var_startDate date;
    select data_partenza from Viaggio where titolo = NEW.viaggio into
var_startDate;

    if NEW.data_inizio < NEW.data_fine then
        signal sqlstate '45001' set message_text = 'La data di fine pernottamento
non può precedere la data di inizio.';
    end if;

    if not exists(select data_inizio from Pernottamento P where P.viaggio =
NEW.viaggio) then
        if not (NEW.data_inizio = var_startDate) then
            signal sqlstate '45001' set message_text =
                'La data di inizio del primo pernottamento deve coincidere
con la data di partenza del viaggio.';
        end if;
    end if;
end;
```

Tabella Prenotazione: BEFORE UPDATE

Verifica che l'albergo scelto abbia una capienza sufficiente per il numero di prenotati al viaggio.

```
create definer = root@localhost trigger check_hotel
    before update
    on Pernottamento
    for each row
begin

    declare var_bookings int unsigned;
    declare var_capacity int unsigned;

    select SUM(numero_persone) from Prenotazione P where P.viaggio = OLD.viaggio
into var_bookings;

    if (NEW.albergo is not null) then
        select capienza from Albergo where codice = NEW.albergo into
var_capacity;
        if (var_bookings > var_capacity) then
```

```

        signal sqlstate '45001' set message_text = 'Il numero di prenotati
per il viaggio supera la capienza dell'albergo.';
    end if;
end if;
end;

```

Eventi

Non sono stati utilizzati eventi.

Viste

Non sono state utilizzate viste:

Stored Procedures e transazioni

Mostrare e commentare le stored procedure che sono state realizzate per implementare la logica applicativa delle operazioni sui dati, evidenziando quando (e perché) sono state realizzate operazioni transazionali complesse.

La seguente stored procedure permette di assegnare un autobus ad un viaggio. Il livello di isolamento serve ad evitare anomalie del tipo unrepeatable read sul trigger, nel caso di aggiornamenti del viaggio.

```

create procedure assign_bus(IN _viaggio varchar(30), IN _autobus varchar(7))
begin
    declare exit handler for sqlexception
    begin
        rollback;
        resignal;
    end;

    set transaction isolation level repeatable read;
    start transaction;
    insert into Assegnato_a (autobus, viaggio) values (_autobus, _viaggio);
    commit;
end;

```

Permette di assegnare un albergo ad un pernottamento.

```

create procedure assign_hotel(IN _viaggio varchar(30), IN _data_inizio date, IN
_albergo int)
begin
    declare exit handler for sqlexception
    begin
        rollback;
        resignal;
    end;

    set transaction isolation level serializable;
    start transaction;
    update Pernottamento set albergo = _albergo where viaggio = _viaggio and
data_inizio = _data_inizio;
    commit;
end;

```

Permette di effettuare una prenotazione. Il livello di isolamento serve ad evitare anomalie del tipo unrepeatable read sul trigger, nel caso di aggiornamenti del viaggio.

```
create procedure book(IN _viaggio varchar(30), IN _cliente varchar(50), IN
_personone int unsigned,
```

```
OUT codice int)
```

```
begin
```

```
    declare exit handler for sqlexception
```

```
    begin
```

```
        rollback;
```

```
        resignal;
```

```
    end;
```

```
    set transaction isolation level repeatable read;
```

```
    start transaction;
```

```
    insert into Prenotazione(numero_personone, cliente, viaggio) values (_personone,
_clientite, _viaggio);
```

```
    commit;
```

```
    set codice = last_insert_id();
```

```
end;
```

Permette di cancellare una prenotazione. Il livello di isolamento serve ad evitare anomalie del tipo unrepeatable read sul trigger, nel caso di aggiornamenti del viaggio.

```
create procedure cancel_booking(IN _codice int)
```

```
begin
```

```
    delete from Prenotazione P where P.codice = _codice;
```

```
end;
```

```
create procedure create_trip(IN _titolo varchar(30), IN _data_partenza date,
```

```
IN _data_rientro date, IN
```

```
_costo double(4, 2))
```

```
begin
```

```
    insert into Viaggio(titolo, data_partenza, data_rientro, costo)
```

```
    values (_titolo, _data_partenza, _data_rientro, _costo);
```

```
    commit;
```

```
end;
```

Permette di inserire un pernottamento per un viaggio.

```
create procedure insert_overnight_stay(IN _viaggio varchar(30), IN _data_inizio
date,
```

```
IN _data_fine date,
```

```
IN _localita varchar(60))
```

```
begin
```

```
    insert into Pernottamento(data_inizio, viaggio, data_fine, localita)
```

```
    values (_data_inizio, _viaggio, _data_fine, _localita);
```

```
    commit;
```

```
end;
```

Elenca tutti gli hotel nel sistema.

```
create procedure list_hotels()
```

```
begin
```

```
    select * from Albergo join Localita L on Albergo.localita = L.nome;
```

```
end;
```

Elenca tutte le località nel sistema.

```
create procedure list_places()
```

```
begin
```

```
    select * from Localita;
```

```
end;
```


Elenca tutti i viaggi nel sistema. Il livello di isolamento serve ad evitare letture fantasma sul trigger.

```
create procedure list_trips(IN _data_partenza date)
begin
    declare exit handler for sqlexception
    begin
        rollback;
        resignal;
    end;

    set transaction isolation level serializable; -- Avoid dirty reads
    start transaction;
    select * from Viaggio where data_partenza >= _data_partenza;
    commit;
end;
```

Permette di eseguire il login.

```
create procedure login(IN _email varchar(50), IN _password varchar(255), OUT
 _role int)
begin
    declare var_userRole enum ('cliente','segreteria');

    select ruolo from Utente where email = _email and password = _password into
    var_userRole;

    if var_userRole = 'cliente' then
        set _role = 0;
    elseif var_userRole = 'segreteria' then
        set _role = 1;
    else
        set _role = -1;
    end if;
end;
```

Permette di inserire un nuovo albergo nel sistema.

```
create procedure new_hotel(IN _nome varchar(60), IN _referente varchar(60),
                           IN _capienza int unsigned, IN
 _via varchar(50), IN _civico varchar(6),
                           IN _cp varchar(7), IN _email
 varchar(50), IN _telefono varchar(15),
                           IN _fax varchar(50), IN
 _localita varchar(60))
begin
    insert into Albergo(nome, referente, capienza, via, civico, cp, email,
 telefono, fax, localita)
    values (_nome, _referente, _capienza, _via, _civico, _cp, _email, _telefono,
 _fax, _localita);
end;
```

Permette di inserire una nuova località nel sistema.

```
create procedure new_location(IN _nome varchar(60), IN _paese varchar(20))
begin
    insert into Localita(nome, paese) values (_nome, _paese);
    commit;
end;
```

Permette la registrazione al sistema.

```
create procedure registration(IN _email varchar(50), IN _nome varchar(30),
                             IN _cognome varchar(30), IN
 _password varchar(255),
```

```

                                IN _telefono varchar(15), IN
_ruolo enum ('cliente', 'segreteria'))
begin
    if (_ruolo is null) then
        set _ruolo = 'cliente';
    end if;

    insert into Utente (email, nome, cognome, telefono, password, ruolo)
    values (_email, _nome, _cognome, _telefono, _password, _ruolo);
end;

Genera il report.
create procedure report(IN _viaggio varchar(30))
begin
    declare var_dataRientro date;

    select V.data_rientro into var_dataRientro from Viaggio V where V.titolo =
_viaggio;
    if (NOW() < var_dataRientro) then
        signal sqlstate '45001' set message_text = 'Il report non può essere
generato prima delle fine del viaggio';
    end if;

    select V.titolo,
           SUM(P.numero_persone) as partecipanti,
           (V.costo * SUM(P.numero_persone)) as guadagno
    from Viaggio V
         join Prenotazione P on V.titolo = P.viaggio
    where V.titolo = _viaggio
    group by V.titolo;

    select P.data_inizio, P.data_fine, A.nome, P.localita, L.paese
    from Pernottamento P
         join Localita L on L.nome = P.localita
         join Albergo A on P.albergo = A.codice
    where P.viaggio = _viaggio
    order by data_inizio;
end;

Elenca i dettagli di un viaggio.
create procedure trip_details(IN _viaggio varchar(30))
begin
    select P.data_inizio, P.data_fine, A.nome as albergo, L.nome as localita,
L.paese
    from Pernottamento P
         join Albergo A on A.codice = P.albergo
         join Localita L on P.localita = L.nome
    where P.viaggio = _viaggio;
    commit;
end;

```

6. Appendice: implementazione

Il codice sorgente del client è disponibile su Github:

https://github.com/AndrewKant-ML/agenzia_di_viaggi.git