

INGEGNERIA DEL SOFTWARE

CarPoolingTV

Realizzato da: Paris, Riggi, Gorini



Sommario

INTRODUZIONE	3
GLOSSARIO	5
DEFINIZIONE DEI REQUISITI UTENTE	6
Requisiti utente testuali	6
Diagramma use case utenti	8
Requisiti utente tabelle	9
Diagramma use case Operatore	18
Diagramma use case Amministratore di sistema	20
SPECIFICHE REQUISITI DI SISTEMA.....	24
Requisiti non funzionali di sistema:	28
Test sui requisiti funzionali	29
Requisiti di dominio:	32
FASE ARCHITETTURA DI SISTEMA.....	33
Activity Diagram	33
Sequence Diagram	38
Class Diagram.....	42
APPENDICE	44
DESIGN PATTERNS.....	45
Observer pattern	45
Adapter.....	46

INTRODUZIONE

Il progetto presentato ha l'obiettivo di sviluppare un sistema software dedicato al servizio di car-pooling per la comunità studentesca dell'Università di Tor Vergata. Il car-pooling rappresenta una soluzione sostenibile per ridurre le emissioni inquinanti e diminuire la congestione del traffico nei percorsi casa-università, offrendo agli studenti un'opportunità per risparmiare sui costi di trasporto e favorire nuove interazioni sociali tra colleghi.

L'obiettivo principale del sistema è facilitare la condivisione di viaggi tra gli utenti, i quali possono ricoprire il ruolo di passeggero nel caso in cui non posseggano un proprio mezzo da poter utilizzare per gli spostamenti casa-università, oppure il ruolo di guidatore nel caso in cui posseggano un mezzo con cui effettuare gli spostamenti e vogliano mettere quest'ultimo a disposizione di tutti, col vantaggio di abbattere i costi dividendo la spesa del viaggio con i passeggeri.

Il sistema permette agli utenti non registrati di registrarsi specificando se intendono essere guidatori, e in tal caso chiede di fornire tutta la documentazione necessaria, tra cui l'inserimento della patente e delle informazioni relative al veicolo che si intende utilizzare. Tutti gli utenti registrati, una volta effettuato il login, potranno visualizzare la lista dei viaggi disponibili (aiutandosi con l'utilizzo di alcuni filtri in caso di necessità) e potranno prenotare i viaggi a cui sono più interessati. Potranno inoltre gestire una personale lista delle tratte preferite impostando un luogo di partenza e di arrivo per cui intendono ricevere notifiche in caso ci fosse un viaggio che copre tale tratta.

Per rendere l'esperienza il migliore possibile gli utenti possono inoltre scrivere o consultare valutazioni riguardanti altri utenti, con lo scopo di incoraggiare gli utenti della piattaforma a condividere viaggi con coloro che hanno valutazioni elevate rispetto a chi ha ricevuto valutazioni basse.

Per i guidatori, il sistema offre la possibilità di creare nuovi viaggi ai quali poi possono unirsi i passeggeri, più passeggeri si aggiungono più la spesa del viaggio diminuisce in quanto uno dei fattori su cui il sistema basa il calcolo del costo dei viaggi è il numero di passeggeri. Il sistema di calcolo dei costi del viaggio è implementato per garantire una tariffa equa basata, oltre che sul numero di passeggeri, anche sulla distanza della tratta da percorrere e il costo del carburante.

Per garantire supporto agli utenti, un team di operatori si occupa della gestione di tutte le segnalazioni inviate da questi ultimi. Gli operatori hanno anche i compiti di confermare o rigettare registrazioni/modifiche effettuate dagli utenti in base alla verifica dei documenti inseriti, e di gestire le valutazioni che hanno subito segnalazioni.

Gli operatori del sistema hanno il compito di confermare o rigettare le registrazioni e le modifiche dei dati, nonché di gestire le valutazioni segnalate e le richieste di supporto degli utenti.

L'app fa uso di un database su cui vengono memorizzati tutti i dati ad essa necessari. Gli amministratori di sistema svolgono operazioni di backup/recovery del database, si occupano degli aspetti di manutenzione quali l'aggiornamento del software e la gestione dei server e si impegnano a garantire la sicurezza all'interno del sistema.

Questi elementi costituiscono le linee guida del progetto, mirate a creare una piattaforma efficiente, affidabile e user-friendly per promuovere il car-pooling all'interno della comunità universitaria di Tor Vergata. Il progetto contribuisce alla riduzione dell'impatto ambientale e alla promozione di una cultura della condivisione e della sostenibilità.

GLOSSARIO

<u>Termine</u>	<u>Descrizione</u>
Car pooling	Il car pooling indica una modalità di trasporto che consiste nella condivisione di automobili private tra un gruppo di persone, con il fine principale di ridurre i costi del trasporto e ridurre l'impatto ambientale
Tratta	Per il termine tratta si intende la coppia luogo di partenza-luogo di destinazione
Viaggio	Con il termine viaggio si intende un evento creato da un utente guidatore contenente: tratta (partenza-destinazione), orario di partenza.
Viaggi attivi	Con il termine viaggio attivo si intende un viaggio prenotabile che ancora deve svolgersi, per il quale mancano più di trenta minuti alla partenza.
Viaggio cancellabile	Con il termine viaggio cancellabile si intende un viaggio, al quale manca più di un'ora alla partenza.
Valutazione	Con il termine valutazione si intende una recensione, inserita da un utente nei confronti di un altro utente, composta da valore numerico che va da 1 a 5 ed un eventuale giudizio di tipo testuale.
Valutazione personale	Il termine "valutazione personale" si riferisce sia alle valutazioni che un utente fornisce ad altri utenti, sia alle valutazioni che riceve dagli altri utenti.
DB	Abbreviazione della parola "DataBase".
Notifica	Invio di una mail di notifica da parte operatore nei confronti dell'utente, utile a segnalare l'esito con cui si è conclusa la pratica svolta dall'operatore

DEFINIZIONE DEI REQUISITI UTENTE

In questa sezione saranno descritti i requisiti utente del sistema attraverso l'utilizzo di use case UML, descrizioni testuali e tabelle di supporto in modo da chiarire i requisiti utente, gli attori del sistema e le interazioni con il sistema.

Requisiti utente testuali

Utente non registrato

1. L'utente non registrato deve avere la possibilità di registrarsi specificando se è un utente guidatore o meno.
2. Se un utente non registrato vuole registrarsi come guidatore deve fornire i dati del veicolo che mette a disposizione e la patente.
3. L'utente non registrato deve avere la possibilità di consultare una guida introduttiva.

Utente registrato

4. L'utente registrato deve avere la possibilità di consultare una guida introduttiva.
5. L'utente registrato deve avere la possibilità di effettuare il login e il logout dall'applicazione
6. L'utente registrato deve avere la possibilità di visualizzare la lista dei viaggi attivi utilizzando dei filtri utili a visualizzare i viaggi a cui è più interessato
7. L'utente registrato deve avere la possibilità di prenotare un viaggio.
8. L'utente registrato durante la prenotazione di un viaggio deve visualizzare i dettagli riguardanti la prenotazione
9. Il sistema deve gestire i pagamenti al momento delle prenotazioni
10. L'utente registrato deve avere la possibilità di annullare la prenotazione per un viaggio con adeguato anticipo rispetto alla partenza.
11. L'utente registrato deve avere la possibilità di modificare i dati inseriti in fase di registrazione
12. L'utente registrato deve avere la possibilità di inserire una valutazione numerica, eventualmente accompagnata da un testo, riguardo un altro utente.
13. Gli utenti registrati devono poter consultare le valutazioni riguardanti gli altri utenti dell'applicazione
14. L'utente registrato deve avere la possibilità di visualizzare le valutazioni da lui scritte e ricevute
15. L'utente registrato può segnalare valutazioni che ha ricevuto da altri utenti se le ritiene non adatte o offensive
16. L'utente registrato deve avere la possibilità di inserire delle proprie tratte preferite
17. L'utente registrato deve avere la possibilità di visualizzare le proprie tratte preferite
18. L'utente registrato deve poter rimuovere una tratta precedentemente inserita come preferita

19. L'utente registrato deve avere la possibilità di visualizzare le informazioni delle proprie prenotazioni.

Utente registrato come guidatore

20. L'utente registrato come guidatore deve poter effettuare tutte le operazioni che effettua l'utente registrato come non guidatore
21. L'utente registrato come guidatore deve poter creare un nuovo viaggio.
22. L'utente registrato come guidatore deve poter visualizzare i viaggi che ha creato
23. L'utente registrato come guidatore deve poter eliminare un viaggio che ha creato.
24. Nel caso di utente guidatore l'utente può modificare i dati del veicolo.

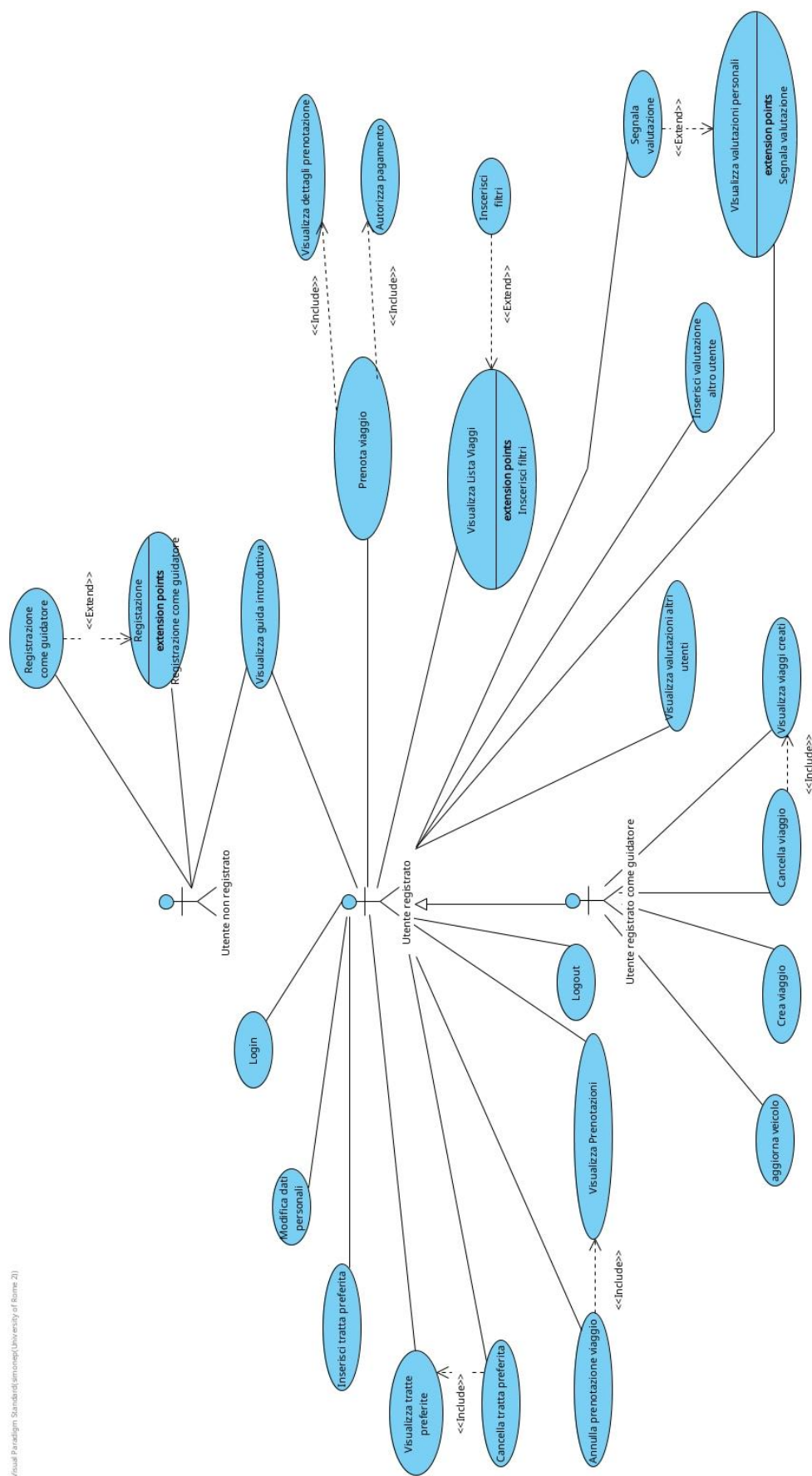
Operatore

25. L'operatore deve effettuare controlli sui dati inseriti dagli utenti in fase di registrazione.
26. L'operatore deve poter confermare o rigettare le modifiche ogni volta che gli utenti modificano i propri dati personali.
27. L'operatore deve poter confermare o rigettare la modifica dei dati relativi al veicolo.
28. L'operatore deve poter gestire le valutazioni segnalate.
29. L'operatore deve inviare una notifica tramite mail ad ogni utente una volta gestita la sua richiesta di registrazione, modifica dati o segnalazione valutazione.

Amministratore di sistema

30. L'amministratore di sistema deve occuparsi degli aggiornamenti software.
31. L'amministratore di sistema deve occuparsi della gestione server.
32. L'amministratore di sistema deve occuparsi della gestione del database.
33. L'amministratore di sistema deve occuparsi della gestione della sicurezza.

Diagramma use case utenti



Requisiti utente tabelle

Use case	Registrazione	
Descrizione	Passo	Azione
	1	L'utente non registrato richiede di potersi registrare
	2	L'utente non registrato può scegliere di registrarsi come utente
	2	L'utente non registrato può scegliere di registrarsi come utente guidatore
	3	L'utente non registrato che decide di registrarsi come utente inserisce i dati personali necessari alla registrazione
	3	L'utente non registrato che decide di registrarsi come utente guidatore inserisce i dati personali necessari alla registrazione come guidatore
	4	L'utente manda richiesta di registrazione al sistema
Attori	Utente non registrato	
Precondizioni	L'utente deve aver aperto l'applicazione e aver acconsentito all'utilizzo dei propri dati personali	
Scenario principale	L'utente non registrato può registrarsi e scegliere se diventare utente oppure utente guidatore dell'applicazione.	
Scenari alternativi	L'utente non può registrarsi se non acconsente al trattamento dei dati oppure se non vengono rispettati i canoni di correttezza e veridicità dei dati inseriti durante la registrazione.	
Post-condizioni	<p>L'operatore verifica i dati della registrazione e decide se accettarla o rigettarla.</p> <p>Se l'operatore accetta la registrazione l'utente registrato potrà effettuare il login con le sue credenziali di accesso.</p> <p>Se l'operazione non va a buon fine l'utente deve ripetere l'operazione</p>	

Use case	Visualizza guida introduttiva	
Descrizione	Passo	Azione
	1	L'utente richiede di visualizzare la guida introduttiva
	2	Il sistema mostra la guida introduttiva all'utente
Attori	Utente non registrato, utente registrato	
Precondizioni	Accedere all'app per prendere visione della guida introduttiva	
Scenario principale	L'utente prende visione della guida introduttiva per avere maggiore conoscenza dell'applicazione	
Scenari alternativi	L'utente non prende visione della guida introduttiva	

Post-condizioni	L'utente tramite la visione della guida introduttiva acquisisce maggiori informazioni
------------------------	---

Use case	Login	
Descrizione	Passo	Azione
	1	L'utente registrato richiede di effettuare il login
	2	L'utente inserisce e-mail e password
	3	Le credenziali vengono inviate al sistema una volta che l'utente fa submit
	4	Il sistema controlla se le credenziali inserite sono corrispondono a quelle inserite in fase di registrazione
	5	Le credenziali sono valide e l'utente registrato effettua il login
	5	Le credenziali non sono valide e viene mostrato un messaggio di errore all'utente
Attori	Utente registrato	
Precondizioni	L'utente deve aver aperto l'applicazione e aver acconsentito all'utilizzo dei propri dati personali	
Scenario principale	L'utente effettua l'operazione di login	
Scenari alternativi	L'utente non effettua l'operazione di login	
Post-condizioni	<p>Se l'operazione va a buon fine l'utente una volta effettuato il login può accedere ai servizi del sistema.</p> <p>Se l'operazione non va a buon fine l'utente deve ripetere l'operazione, e viene mostrato un messaggio di errore.</p>	

Use case	Modifica dati personali	
Descrizione	Passo	Azione
	1	L'utente seleziona l'operazione di modifica dei dati personali
	2	L'utente sceglie uno o più campi da modificare
	3	L'utente inserisce i dati modificati
	4	L'utente invia la modifica
Attori	Utente registrato	
Precondizioni	L'utente registrato deve aver effettuato il login	
Scenario principale	L'utente richiede la modifica dei propri dati personali	
Scenari alternativi	L'utente non richiede alcuna modifica dei dati personali	

Post-condizioni	Inviata richiesta di verifica dei dati aggiornati ad un operatore il provvederà ad accettarli se validi o rigettarli se non validi.
------------------------	---

Use case	Visualizza tratte preferite	
Descrizione	Passo	Azione
	1	L'utente richiede visualizzazione delle proprie tratte preferite
	2	Il sistema mostra all'utente la lista delle sue tratte preferite
Attori	Utente registrato	
Precondizioni	L'utente registrato deve aver effettuato il login	
Scenario principale	L'utente richiede la visualizzazione delle tratte preferite	
Scenari alternativi	L'utente non richiede la visualizzazione delle tratte preferite	
Post-condizioni	Se visualizzazione andata a buon fine l'utente può consultare le tratte preferite che ha impostato. Se la visualizzazione non andata a buon fine l'utente deve ripetere l'operazione.	

Use case	Inserisci tratte preferite	
Descrizione	Passo	Azione
	1	L'utente inserisce una nuova tratta preferita
	2	Il sistema acquisisce la nuova tratta preferita inserita
Attori	Utente registrato	
Precondizioni	L'utente registrato deve aver effettuato il login	
Scenario principale	L'utente inserisce una nuova tratta preferita	
Scenari alternativi	L'utente non inserisce una nuova tratta preferita	
Post-condizioni	Se l'inserimento andato a buon fine il sistema ha registrato la nuova tratta preferita per l'utente che verrà poi visualizzata in visualizza tratte preferite. Se l'inserimento non andato a buon fine la nuova tratta preferita non viene registrata.	

Use case	Cancella tratta preferita	
Descrizione	Passo	Azione

	1	L'utente richiede la cancellazione di una delle sue tratte preferite.
	2	Il sistema elimina la tratta selezionata dalle tratte preferite dell'utente
Attori	Utente registrato	
Precondizioni	L'utente registrato deve aver effettuato il login	
Scenario principale	L'utente richiede la cancellazione di una delle sue tratte preferite	
Scenari alternativi	L'utente non richiede la cancellazione di una tratta preferita	
Post-condizioni	<p>Se cancellazione andata a buon fine la tratta cancellata non può più far parte delle tratte preferite dell'utente.</p> <p>Se la cancellazione non andata a buon fine l'utente deve ripetere l'operazione.</p>	

Use case	Logout	
Descrizione	Passo	Azione
	1	L'utente seleziona l'operazione di logout
	2	L'utente viene reindirizzato alla pagina di accesso
Attori	Utente registrato	
Precondizioni	L'utente registrato deve aver effettuato il login	
Scenario principale	L'utente seleziona l'operazione di logout	
Scenari alternativi	L'utente non seleziona l'operazione di logout e resta loggato nel sistema.	
Post-condizioni	<p>Se logout andato a buon fine l'utente dell'applicazione deve di nuovo effettuare il login per accedere alle principali funzionalità dell'app.</p> <p>Se logout non andato a buon fine l'utente deve ripetere l'operazione.</p>	

Use case	Inserisci valutazione altro utente	
Descrizione	Passo	Azione
	1	L'utente registrato richiede di inserire una nuova valutazione nei confronti di un altro utente
	2	L'utente inserisce la valutazione con valore numerico da 1 a 5, accompagnata opzionalmente da messaggio testuale
	3	L'utente invia la valutazione al sistema
Attori	Utente registrato	
Precondizioni	L'utente registrato deve aver effettuato il login	

Scenario principale	L'utente inserisce una valutazione nei confronti di un altro utente
Scenari alternativi	L'utente non inserisce una valutazione
Post-condizioni	Se operazione andata a buon fine la valutazione è inserita all'interno del sistema. Se l'operazione non andata a buon fine l'utente deve ripetere l'operazione.

Use case	Visualizza valutazioni altri utenti	
Descrizione	Passo	Azione
	1	L'utente richiede di visualizzare le valutazioni di un altro utente
	2	Il sistema fornisce all'utente la lista delle valutazioni relative all'utente a cui è interessato
	3	L'utente consulta la lista delle valutazioni che ha richiesto
Attori	utente registrato	
Precondizioni	L'utente deve aver effettuato il login	
Scenario principale	L'utente visualizza le valutazioni di un altro utente	
Scenari alternativi	L'utente non ha interesse nel visualizzare le recensioni di un altro utente	
Post condizioni	Se l'operazione va a buon fine l'utente acquisisce informazioni riguardo un altro specifico utente. Se non va a buon fine l'utente deve ripetere l'operazione.	

Use case	Visualizza valutazioni personali	
Descrizione	Passo	Azione
	1	L'utente richiede la visualizzazione delle valutazioni personali
	2	Il sistema fornisce le valutazioni personali dell'utente
	3	L'utente visualizza le valutazioni personali
Attori	Utente registrato	
Precondizioni	L'utente registrato deve aver effettuato il login	
Scenario principale	L'utente visualizza le valutazioni personali	
Scenari alternativi	L'utente non seleziona l'operazione di visualizzazione	

Post-condizioni	Se andato a buon fine l'utente viene a conoscenza delle valutazioni che gli altri hanno inserito su di esso e visualizza le valutazioni che ha inserito nel corso del tempo riguardo altri utenti. Se non andato a buon fine l'utente deve ripetere l'operazione.
------------------------	--

Use case	Segnala valutazione	
Descrizione	Passo	Azione
	1	L'utente segnala una valutazione tra quelle personali ricevute
	2	Il sistema si occupa di gestire la segnalazione
Attori	Utente registrato	
Precondizioni	L'utente registrato deve aver effettuato il login	
Scenario principale	L'utente seleziona una valutazione personale per lui non conforme alle linee guida	
Scenari alternativi	L'utente non trova nessuna valutazione da segnalare.	
Post-condizioni	Se segnalazione andata a buon fine viene assegnata automaticamente ad un operatore che si occuperà di gestire la segnalazione. Se non andato a buon fine l'utente deve ripetere l'operazione.	

Use case	Visualizza lista viaggi	
Descrizione	Passo	Azione
	1	L'utente richiede al sistema di visualizzare la lista dei viaggi attivi
	1	L'utente richiede al sistema di visualizzare la lista dei viaggi attivi specificando dei filtri
	2	Il sistema fornisce all'utente la lista dei viaggi attivi richiesta
	3	L'utente visualizza la lista dei viaggi
Attori	Utente registrato	
Precondizioni	L'utente registrato deve aver effettuato il login	
Scenario principale	L'utente visualizza la lista dei viaggi attivi selezionando o meno dei filtri	
Scenari alternativi	L'utente non visualizza la lista dei viaggi disponibili	
Post-condizioni	L'operazione va a buon fine e l'utente consulta i viaggi attivi ed eventualmente procede a sceglierne uno da prenotare. L'operazione non va a buon fine e l'utente deve ripetere l'operazione.	

Use case	Prenota viaggio	
Descrizione	Passo	Descrizione

	1	L'utente seleziona un viaggio per cui vuole prenotarsi
	2	L'utente visualizza i dettagli della prenotazione che sta effettuando
	3	L'utente autorizza il pagamento
Attori	Utente registrato	
Precondizioni	L'utente deve essere registrato e deve aver eseguito l'accesso	
Scenario principale	L'utente effettua la prenotazione di un viaggio visualizzando i dettagli di quest'ultima e autorizzando il pagamento	
Scenari alternativi	L'utente non effettua la prenotazione L'utente non autorizza il pagamento	
Post-condizioni	Se operazione andata a buon fine l'utente ha prenotato un posto per quel viaggio. Se operazione non andata a buon fine l'utente deve ripetere l'operazione.	

Use case	Visualizza prenotazioni	
Descrizione	Passo	Azione
	1	L'utente richiede di visualizzare le informazioni delle prenotazioni che ha effettuato
	2	Il sistema fornisce all'utente una lista dei viaggi da lui prenotati con le relative informazioni
	3	L'utente visualizza le sue prenotazioni
Attori	Utente registrato	
Precondizioni	L'utente registrato deve aver eseguito l'accesso	
Scenario principale	L'utente visualizza la lista delle prenotazioni da lui effettuate con relative informazioni	
Scenari alternativi	L'utente non visualizza le informazioni.	
Post condizioni	Se l'operazione va a buon fine l'utente acquisisce informazioni riguardo le sue prenotazioni. Se l'operazione non va a buon fine l'utente deve riprovare.	

Use case	Annulla prenotazione viaggio	
Descrizione	Passo	Azione
	1	L'utente visualizza la lista delle prenotazioni
	2	L'utente registrato seleziona l'operazione di annullamento prenotazione viaggio.
	3	Il sistema rimuove l'utente come passeggero del viaggio che aveva prenotato

Attori	Utente registrato
Precondizioni	L'utente registrato deve aver eseguito l'accesso, inoltre il viaggio deve essere attivo
Scenario principale	L'utente procede ad annullare la prenotazione di un viaggio attivo a cui si era precedentemente prenotato.
Scenari alternativi	L'utente non ha nessuna prenotazione viaggio attiva da poter cancellare
Post condizioni	Se il processo va a buon fine la prenotazione viene cancellata. Se il processo non va a buon fine l'utente deve riprovare l'operazione.

Use case	Crea viaggio	
Descrizione	Passo	Azione
	1	L'utente registrato come guidatore seleziona l'operazione di creazione viaggio
	2	L'utente inserisce i dati utili alla creazione del viaggio
	3	Il sistema memorizza il nuovo viaggio
Attori	Utente registrato come guidatore	
Precondizioni	L'utente deve essere registrato come guidatore e deve aver eseguito l'accesso	
Scenario principale	L'utente guidatore crea un nuovo viaggio inserendo i dati necessari	
Scenari alternativi	L'utente non procede alla creazione di un nuovo viaggio	
Post condizioni	Se l'operazione va a buon fine il nuovo viaggio viene memorizzato nel sistema e viene trovato dagli utenti durante l'operazione di visualizzazione lista viaggi. Se l'operazione non va a buon fine all'utente viene mostrato un errore e deve ripetere l'operazione.	

Use case	Cancella viaggio	
Descrizione	Passo	Azione
	1	L'utente registrato come guidatore visualizza la lista di viaggi da lui creati
	2	L'utente seleziona un viaggio che vuole cancellare
	3	Il viaggio viene rimosso dal sistema
Attori	Utente registrato come guidatore	
Precondizioni	L'utente deve essere registrato come guidatore e deve aver eseguito l'accesso, inoltre il viaggio che decide di cancellare deve essere un viaggio cancellabile che lui stesso ha creato in precedenza.	

Scenario principale	L'utente procede alla cancellazione di un viaggio cancellabile che ha precedentemente creato
Scenari alternativi	L'utente non elimina nessun viaggio
Post condizioni	Se l'operazione è andata a buon fine il viaggio viene eliminato dal sistema e non viene più trovato dagli utenti durante l'operazione di visualizzazione lista viaggi. Se l'operazione non è andata a buon fine l'utente deve ripetere l'operazione.

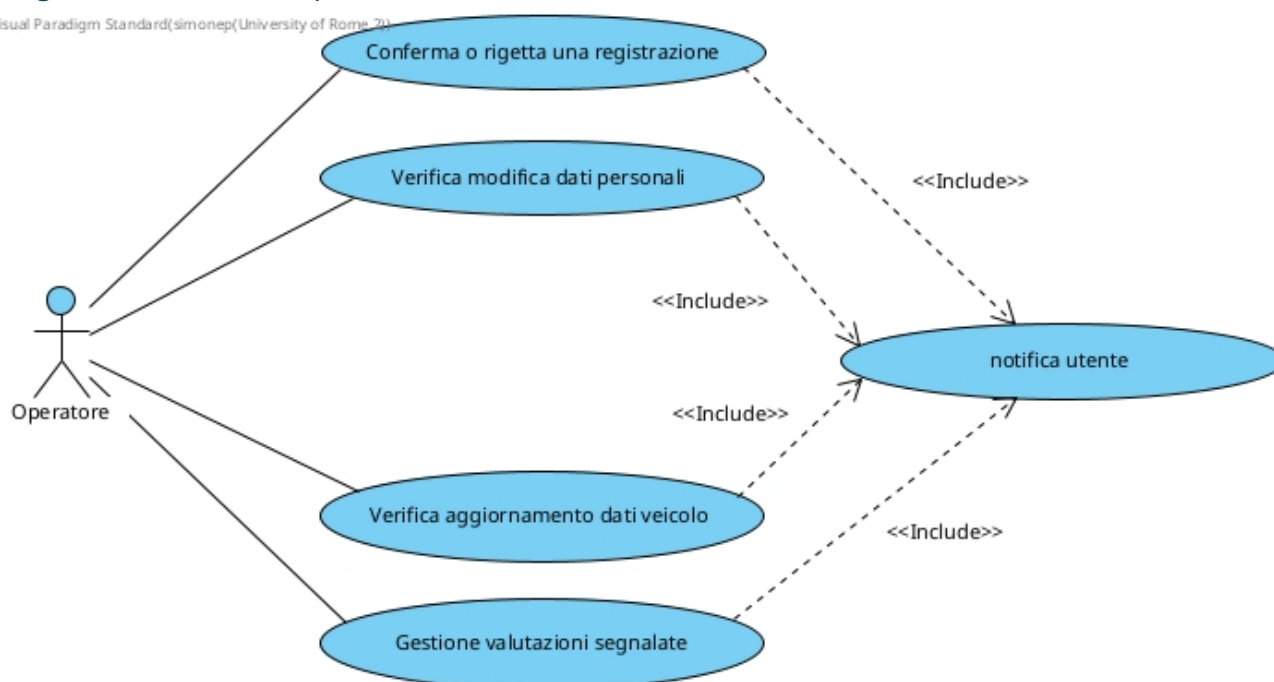
Use case	Aggiorna veicolo	
Descrizione	Passo	Azione
	1	L'utente seleziona l'operazione di aggiornamento dati veicolo
	2	L'utente inserisce i nuovi dati riguardanti il veicolo
Attori	Utente registrato come guidatore	
Precondizioni	L'utente deve essere correttamente registrato come guidatore e deve aver effettuato il login	
Scenario principale	L'utente aggiorna i dati del veicolo	
Scenari alternativi	L'utente non effettua l'operazione di aggiornamento dati veicolo	
Post condizioni	Se l'operazione è andata a buon fine viene inviata all'operatore la segnalazione di verifica della nuova documentazione. Nel caso in cui l'operazione non è andata a buon fine l'utente deve ripetere l'operazione.	

Use case	Visualizza viaggi creati	
Descrizione	Passo	Azione
	1	L'utente richiede di visualizzare la lista di viaggi che lui ha creato
	2	Il sistema fornisce all'utente una lista dei viaggi da lui creati
	3	L'utente visualizza la lista di viaggi da lui creati
Attori	Utente registrato come guidatore	
Precondizioni	L'utente registrato deve aver eseguito l'accesso	
Scenario principale	L'utente visualizza la lista dei viaggi da lui creati	
Scenari alternativi	L'utente non ha nessun viaggio da visualizzare non avendone creato alcuno	
Post condizioni	L'utente visualizza i viaggi che ha creato e può procedere alla cancellazione di un viaggio cancellabile.	

Nel caso in cui l'operazione non è andata a buon fine l'utente deve ripetere l'operazione.

Diagramma use case Operatore

Visual Paradigm Standard(simonep(University of Rome 2))



Use case	Conferma o rigetta una registrazione	
Descrizione	Passo	Azione
	1	Operatore visualizza richiesta di registrazione in arrivo
	2	Operatore verifica i dati inseriti dall'utente in fase di registrazione
	3	Se i dati sono coerenti accetta la registrazione, altrimenti la rigetta.
	4	L'operatore invia una notifica tramite mail sull'esito dell'operazione all'utente
Attori	Operatore	
Precondizioni	L'utente ha effettuato una richiesta di registrazione	
Scenario principale	L'operatore gestisce le richieste di registrazione	
Scenari alternativi	L'operatore mette temporaneamente in attesa la richiesta di registrazione	

Post-condizioni	<p>Se l'operatore accetta la registrazione l'utente ora può accedere all'applicazione.</p> <p>Se l'operatore rigetta la registrazione l'utente deve inviarne una nuova.</p>
------------------------	---

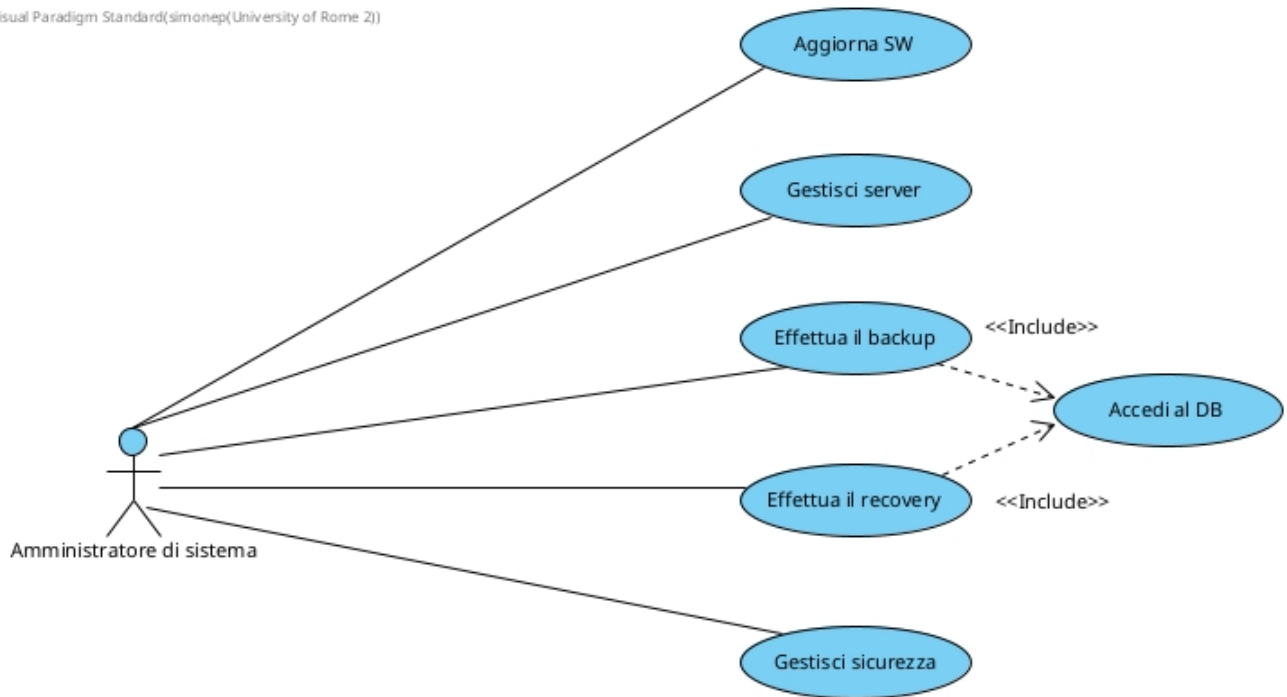
Use case	Verifica modifica dati personali	
Descrizione	Passo	Azione
	1	L'Operatore visualizza i nuovi dati inseriti dall'utente durante la modifica dei dati personali
	2	L'operatore accetta o rigetta la modifica
	3	L'operatore invia una notifica tramite mail sull'esito dell'operazione all'utente
Attori	Operatore	
Precondizioni	L'utente ha effettuato una operazione di modifica dati e fornito nuovi dati che richiedono la verifica dell'operatore	
Scenario principale	L'operatore valuta la modifica effettuata dall'utente	
Scenari alternativi	L'operatore mette temporaneamente in attesa la richiesta.	
Post-condizioni	<p>Se l'operatore ha accettato la modifica i dati inseriti dell'utente risultano aggiornati.</p> <p>Se l'operatore ha rigettato la modifica l'utente deve eseguire nuovamente l'operazione di modifica.</p>	

Use case	Verifica aggiornamento dati veicolo	
Descrizione	Passo	Azione
	1	L'operatore visualizza la modifica dei documenti relativi al veicolo
	2	L'operatore accetta o rigetta la modifica dei dati
	3	L'operatore invia una notifica tramite mail sull'esito dell'operazione all'utente
Attori	Operatore	
Precondizioni	Un utente guidatore ha effettuato una operazione di aggiornamento dati relativi al veicolo	
Scenario principale	L'operatore valuta la modifica effettuata dall'utente registrato come guidatore sui dati relativi al veicolo	
Scenari alternativi	L'operatore mette temporaneamente in attesa la richiesta.	
Post-condizioni	Se l'operatore ha accettato la modifica dei dati allora il sistema viene aggiornato.	

	Se l'operatore ha rigettato la modifica l'utente deve eseguire nuovamente l'operazione di modifica.
--	---

Use case	Gestione valutazioni segnalate	
Descrizione	Passo	Azione
	1	L'operatore verifica se la valutazione testuale contiene testo non idoneo
	2	L'operatore elimina la valutazione se questa risulta non idonea
	2	L'operatore ritiene che la valutazione sia appropriata, quindi decide di non eliminarla
	3	L'operatore invia una notifica tramite mail sull'esito dell'operazione all'utente
Attori	Operatore	
Precondizioni	Un utente ha effettuato una operazione di segnalazione valutazione	
Scenario principale	L'operatore gestisce una valutazione testuale che ha ricevuto una segnalazione, nel caso contenga testo non idoneo le elimina altrimenti le mantiene e chiude la segnalazione	
Scenari alternativi	L'operatore mette temporaneamente in attesa la segnalazione	
Post-condizioni	la segnalazione è stata gestita e chiusa	

Diagramma use case Amministratore di sistema



Use case	Aggiorna SW	
Descrizione	Passo	Azione
	1	L'amministratore di sistema effettua l'aggiornamento software
Attori	Amministratore di sistema	
Precondizioni	L'attività di manutenzione richiede aggiornamento software	
Scenario principale	L'amministratore di sistema si occupa dell'aggiornamento software	
Scenari alternativi	Il sistema risulta già aggiornato	
Post-condizioni	Il sistema è stato aggiornato a seguito della manutenzione. Se operazione non andata a buon fine l'amministratore di sistema deve ripeterla.	

Use case	Gestisci server	
Descrizione	Passo	Azione
	1	L'amministratore di sistema verifica lo stato dei server
	2	L'amministratore di sistema procede all'upgrade e all'ottimizzazione dei server

	2	L'amministratore di sistema procede studiando e riparando eventuali errori software e/o hardware
Attori	Amministratore di sistema	
Precondizioni	L'attività di manutenzione richiede che vengano gestiti i server	
Scenario principale	L'amministratore di sistema si occupa della gestione dei server	
Scenari alternativi	I server non necessitano di gestione	
Post-condizioni	È stata eseguita manutenzione sui server. Se operazione non andata a buon fine l'amministratore di sistema deve ripeterla.	

Use case	Effettua il backup	
Descrizione	Passo	Azione
	1	L'amministratore di sistema accede al database
	2	L'amministratore di sistema esegue l'operazione di backup sul database.
Attori	Amministratore di sistema	
Precondizioni	È necessario effettuare un backup	
Scenario principale	L'amministratore di sistema effettua il backup del database per proteggere i dati da eventuali guasti	
Scenari alternativi	Ultimo backup recente, non c'è bisogno di effettuarne uno nuovo.	
Post-condizioni	È stata eseguito il backup dei dati con successo. Se operazione non andata a buon fine l'amministratore di sistema deve ripeterla.	

Use case	Effettua il recovery	
Descrizione	Passo	Azione
	1	L'amministratore di sistema accede al database
	2	L'amministratore di sistema esegue l'operazione di ripristino sul database, ripristinando l'ultimo backup fatto.
Attori	Amministratore di sistema	

Precondizioni	Si è verificato un guasto nel database per il quale è necessaria l'operazione di ripristino
Scenario principale	L'amministratore di sistema effettua il ripristino del database all'ultimo backup
Scenari alternativi	Non è possibile utilizzare l'ultimo backup, ne viene utilizzato uno precedente.
Post-condizioni	Vengono ripristinati i dati dal backup utilizzato. Se operazione non andata a buon fine l'amministratore di sistema deve ripeterla.

Use case	Gestisci sicurezza	
Descrizione	Passo	Azione
	1	L'amministratore di sistema verifica se ci sono minacce o problemi di sicurezza nell'applicazione
	2	Se si riscontrano criticità le gestisce
Attori	Amministratore di sistema	
Precondizioni	L'applicazione richiede operazioni di controllo e gestione della sicurezza	
Scenario principale	L'amministratore di sistema si occupa della gestione di eventuali minacce che minano la sicurezza dell'applicazione, con lo scopo di mantenerla sicura e affidabile	
Scenari alternativi	A seguito di attenta analisi non risultano problemi di sicurezza dell'applicazione	
Post-condizioni	A seguito della gestione sulla sicurezza dell'applicazione e della gestione di eventuali minacce da parte dell'amministratore l'applicazione è sicura	

SPECIFICHE REQUISITI DI SISTEMA

Requisiti funzionali di sistema:

Attore	Id	Descrizione
Utente non registrato	1	L'utente non registrato deve avere la possibilità di registrarsi specificando se è un utente guidatore o meno. Tutti gli utenti devono inserire nome, cognome, e-mail, username, password, documento di riconoscimento, numero di telefono e un metodo di pagamento. Nel caso in cui la registrazione avvenga come utente guidatore il documento di riconoscimento deve essere obbligatoriamente la patente di guida, inoltre deve inserire i dati riguardanti il veicolo che intende utilizzare tra cui: carta di circolazione, assicurazione e targa.
	2	L'utente non registrato deve avere la possibilità di consultare la guida introduttiva che contiene una descrizione generale sulle funzionalità dell'applicazione.
Utente registrato	3	Ogni utente registrato deve accettare termini e condizioni del sistema
	4	L'utente registrato deve avere la possibilità di consultare la guida introduttiva che contiene una descrizione generale sulle funzionalità dell'applicazione.
	5	L'utente registrato deve avere la possibilità di effettuare il login attraverso e-mail e password
	6	L'utente registrato per poter accedere alle funzionalità di sistema deve aver effettuato il login
	7	L'utente registrato deve avere la possibilità di visualizzare la lista dei viaggi attivi utilizzando dei filtri tra cui: tratta del viaggio, orario di partenza, ranking del guidatore e costo previsto attuale
	8	L'utente registrato deve avere la possibilità di prenotare un viaggio dopo averlo selezionato dalla lista dei viaggi attivi, l'utente può prenotare un viaggio attivo fino ad un massimo di 30 minuti della partenza. Il sistema, quindi, deve cambiare lo stato del viaggio da attivo a non attivo 30 minuti prima della partenza.
	9	Il prezzo della prenotazione effettuata da utente registrato è calcolato dinamicamente al momento della sua prenotazione in base al numero di passeggeri attualmente

		prenotati per lo stesso viaggio, alla distanza tratta e al costo carburante.
	10	Il prezzo della prenotazione relativa ad un viaggio effettuata da un utente può variare per tutto il tempo in cui la prenotazione rimane attiva, questo accade nel caso in cui altri utenti si prenotino o annullino la prenotazione per lo stesso viaggio.
	11	L'utente registrato durante la prenotazione di un viaggio deve visualizzare i dettagli riguardanti la prenotazione, i quali includono: tratta, orario, data, guidatore che ha creato il viaggio, passeggeri prenotati fino a quel momento e prezzo previsto
	12	L'utente registrato dopo aver effettuato la prenotazione per un viaggio deve autorizzare il pagamento, successivamente, una volta che la prenotazione non è più attiva il sistema detrae il costo del viaggio dal metodo di pagamento dell'utente.
	13	Una volta chiusa la prenotazione relativa ad un viaggio, se il sistema è impossibilitato a effettuare la detrazione dell'importo dovuto da uno dei passeggeri, il sistema elimina la prenotazione di quest'ultimo ridistribuendo l'importo dovuto sul costo che devono pagare gli altri passeggeri.
	14	Il sistema trasferisce l'importo detratto dagli utenti passeggeri al metodo di pagamento inserito dal guidatore.
	15	L'utente registrato deve avere la possibilità di annullare la prenotazione per un viaggio fino a massimo trenta minuti dalla partenza
	16	L'utente registrato deve avere la possibilità di modificare il metodo di pagamento inserito, il documento di riconoscimento, l'e-mail, l'username, il numero di telefono e la password
	17	L'utente registrato deve avere la possibilità di effettuare il logout, a seguito del quale deve essere reindirizzato alla pagina di accesso

	18	L'utente registrato deve avere la possibilità di inserire una valutazione riguardo un altro utente specificandone l'username, una valutazione comprende una valutazione da 1 a 5 e opzionalmente un contenuto testuale
	19	Gli utenti registrati devono poter consultare le valutazioni riguardanti gli altri utenti dell'applicazione specificando il loro username
	20	L'utente registrato deve avere la possibilità di visualizzare le proprie valutazioni personali, che comprendono sia le valutazioni che ha ricevuto da altri utenti sia le valutazioni che lui ha personalmente scritto
	21	L'utente registrato una volta visualizzate le valutazioni personali che ha ricevuto da altri utenti se le ritiene non adatte o offensive può segnalarle. Una volta effettuata la segnalazione quest'ultima viene affidata automaticamente a un operatore.
	22	L'utente registrato deve avere la possibilità di inserire delle proprie tratte preferite inserendo luogo di partenza e di destinazione della tratta
	23	L'utente registrato deve avere la possibilità di visualizzare le proprie tratte preferite
	24	L'utente registrato dopo aver visualizzato le proprie tratte preferite deve avere la possibilità di cancellare le tratte che desidera rimuovere dalle preferite
	25	L'utente registrato deve avere la possibilità di visualizzare le informazioni delle proprie prenotazioni, tra cui: tratta, orario di partenza, data, guidatore che ha creato il viaggio, passeggeri prenotati fino a quel momento e prezzo previsto
Utente registrato come guidatore	26	L'utente registrato come guidatore deve poter effettuare tutte le operazioni che effettua l'utente registrato come non guidatore
	27	L'utente registrato come guidatore deve poter creare un nuovo viaggio inserendo: luogo di partenza e di destinazione con la data e l'orario di partenza

	28	Quando un utente registrato come guidatore crea un viaggio, il sistema genera automaticamente una prenotazione per il guidatore relativa al viaggio appena creato.
	29	L'utente registrato come guidatore deve poter visualizzare i viaggi che ha creato
	30	L'utente registrato come guidatore deve poter eliminare un viaggio che ha creato, al massimo un'ora prima della partenza
	31	Nel caso di utente guidatore l'utente può modificare i dati del veicolo, tra cui targa, carta di circolazione, assicurazione
Operatore	32	L'operatore conferma o rigetta una registrazione in base ai dati inseriti. Nel caso di utente guidatore, oltre a verificare i dati inseriti come per tutti gli utenti, l'operatore si occupa di verificare che il documento di riconoscimento inserito sia patente di guida e si occupa anche di verificare i dati del veicolo
	33	L'operatore deve poter confermare o rigettare le modifiche ogni volta che gli utenti modificano i propri dati personali
	34	L'operatore deve poter confermare o rigettare la modifica dei dati relativi al veicolo, tra cui carta di circolazione, targa e assicurazione
	35	L'operatore deve poter gestire le valutazioni segnalate, rimuovendole nel caso contengano contenuto offensivo oppure non idoneo alle linee guida
	36	L'operatore invia una notifica tramite mail all'utente una volta che la sua registrazione viene accettata o rifiutata, descrivendo il risultato del controllo effettuato e le motivazioni dell'esito
	37	L'operatore invia una notifica tramite mail all'utente una volta che la sua modifica dei dati personali viene accettata o rifiutata, descrivendo il risultato del controllo effettuato e le motivazioni dell'esito
	38	L'operatore invia una notifica tramite mail all'utente una volta che la sua modifica dei dati del veicolo viene accettata o rifiutata, descrivendo il risultato del controllo effettuato e le motivazioni dell'esito
	39	L'operatore invia una notifica tramite mail all'utente una volta che una sua segnalazione valutazione viene verificata, descrivendo il risultato del controllo effettuato e le motivazioni dell'esito

Amministratore di sistema	40	L'amministratore di sistema deve occuparsi degli aggiornamenti software
	41	L'amministratore di sistema deve occuparsi della gestione server
	42	L'amministratore di sistema deve poter accedere al database contenenti tutti i dati memorizzati dall'applicazione
	43	L'amministratore di sistema dopo aver effettuato l'accesso al database deve poterne effettuare il backup
	44	L'amministratore di sistema dopo aver effettuato l'accesso al database deve poter effettuare il ripristino dei dati.
	45	L'amministratore di sistema deve occuparsi della gestione della sicurezza.

Requisiti non funzionali di sistema:

Campo	Id	Descrizione
Usabilità	1	1.1 Il sistema deve essere user-friendly e facilmente navigabile. 1.2 Deve essere presente una guida introduttiva consultabile sia dagli utenti registrati che non registrati.
Sicurezza	2	2.1 Tutti i dati sensibili degli utenti (password, documenti di riconoscimento, dettagli di pagamento) devono essere crittografati. 2.2 Il sistema deve implementare controlli di accesso rigorosi per garantire che solo utenti autorizzati possano accedere a determinate funzionalità e dati. 2.3 Il sistema fa uso dell'autenticazione a due fattori
Efficienza	3	3.1 Il sistema deve garantire tempi di attesa non elevati per ogni richiesta effettuata dagli utenti
Manutenibilità	4	4.1 Il codice del sistema deve essere ben documentato e strutturato per facilitare la manutenzione e l'aggiornamento. 4.2 Devono essere presenti strumenti per monitorare e diagnosticare problemi nel sistema.
Portabilità	5	5.1 Minima versione Android richiesta per poter supportare tutte le funzionalità dell'app
Scalabilità	6	6.1 Devono essere possibili aggiornamenti e espansioni del sistema senza interrompere il servizio. 6.2 Il sistema deve essere scalabile per supportare un numero crescente di utenti e viaggi senza degrado delle prestazioni.

Affidabilità	7	<p>7.1 Il sistema deve garantire una gestione efficiente dei dati per evitare perdite di informazioni.</p> <p>7.2 Devono essere implementate misure per il backup e il ripristino dei dati.</p>
--------------	---	---

Test sui requisiti funzionali

ID N.F	Come è testato?
1 - Usabilità	<p>Test di usabilità: valuta il sistema per uso umano e verifica se è idoneo all'uso. Attraverso il feedback degli utenti, è possibile migliorare l'usabilità. I test effettuati includono:</p> <p>Test dell'interfaccia utente: per valutare la GUI.</p> <p>Fornire a un piccolo gruppo di persone scelte casualmente per il test, anziché a professionisti esperti.</p>
2 - Sicurezza	<p>Security Testing: coinvolge diverse attività per garantire la sicurezza del sistema informatico. Queste includono:</p> <p>Riesame dei requisiti funzionali e delle specifiche tecniche: analisi approfondita della documentazione disponibile per comprendere e verificare le funzionalità e le specifiche tecniche del sistema.</p> <p>Penetration Testing: simulazione di attacchi informatici per valutare la sicurezza del sistema.</p> <p>Code Review: analisi del codice sorgente e delle applicazioni per identificare potenziali problemi di sicurezza e migliorare la qualità del codice, specialmente in vista di future implementazioni.</p> <p>Verifica della qualità del codice: controllo sistematico della qualità del codice per assicurarsi che sia sicuro, efficiente e conforme agli standard di programmazione.</p>
3 - Efficienza	<p>Performance Testing: serve a verificare il funzionamento e l'efficienza dei vari componenti del software. Questi test sono cruciali per identificare e correggere eventuali errori di progettazione e per valutare le prestazioni complessive dell'architettura software. I principali obiettivi del performance testing includono:</p> <p>Valutare la velocità del sistema: misurare i tempi di risposta del software sotto diverse condizioni di carico.</p>

	<p>Utilizzo delle risorse: monitorare come il software utilizza le risorse di sistema, come CPU, memoria e larghezza di banda.</p>
4 - Manutenibilità	<p>Test di Manutenzione: Dopo che un'applicazione o un prodotto è stato lanciato, possono emergere problemi nell'ambiente di produzione oppure il cliente potrebbe richiedere miglioramenti o nuove funzionalità. In questi casi, il team che si occupa della manutenzione interviene per testare e gestire tali scenari. Anche dopo il rilascio, l'applicazione richiede manutenzione continua per garantire il suo corretto funzionamento e per implementare eventuali aggiornamenti richiesti. Il team che si occupa della manutenzione è responsabile di:</p> <p>Identificare e risolvere problemi: diagnosticare e correggere eventuali bug o malfunzionamenti che si verificano nell'ambiente live.</p> <p>Testare miglioramenti: verificare che le nuove funzionalità o gli aggiornamenti richiesti dal cliente siano integrati senza introdurre nuovi problemi.</p> <p>Assicurarsi che le modifiche o correzioni apportate non abbiano effetti negativi sulle funzionalità esistenti.</p> <p>Aggiornare documentazione: mantenere aggiornata la documentazione tecnica e dei test per riflettere le modifiche apportate.</p>
5 - Portabilità	<p>Test di Portabilità: Il test di portabilità verifica se un software o un'applicazione funziona correttamente quando installato su diversi sistemi o piattaforme. L'obiettivo è assicurarsi che nessuna funzionalità venga compromessa a causa di un cambiamento nell'ambiente. Durante questo processo, vengono effettuate verifiche su vari aspetti, tra cui:</p> <p>Compatibilità con diverse piattaforme: viene eseguito per verificare se, nel caso in cui un software / applicazione sia installato su un sistema diverso o su una piattaforma diversa, funzioni correttamente.</p> <p>Configurazioni di rete diverse: verificare il funzionamento del software in differenti configurazioni di rete, inclusi vari tipi di connessione e condizioni di larghezza di banda.</p>
6 - Scalabilità	<p>Il test di scalabilità verifica se l'applicazione può gestire l'aumento di traffico, transazioni e dati, mantenendo le prestazioni. Valuta la capacità di gestire più utenti e transazioni, e l'efficienza con volumi di dati maggiori. Esamina anche la risposta a risorse hardware aggiuntive e la scalabilità dinamica. È essenziale per garantire crescita e prestazioni ottimali sotto carichi variabili.</p>
7 - Affidabilità	<p>Il test di affidabilità assicura che l'applicazione funzioni costantemente senza errori nel tempo. Monitora il sistema per guasti, valutando la</p>

	capacità di recupero, stabilità e gestione dei picchi di carico. Verifica anche la coerenza dei dati. È cruciale per garantire un servizio continuo e affidabile, minimizzando i tempi di inattività.
--	---

Requisiti di dominio:

1. Rispetto definizioni gazzetta ufficiale:

<https://www.gazzettaufficiale.it/atto/regioni/caricaArticolo?art.versione=1&art.idGruppo=1&art.codiceRedazionale=21R00221&art.idArticolo=2&art.idSottoArticolo=1&art.dataPubblicazioneGazzetta=2021-08-28&art.progressivo=0>

2. Misure per incentivare il car sharing e il car pooling:

<https://www.gazzettaufficiale.it/atto/regioni/caricaArticolo?art.versione=1&art.idGruppo=1&art.codiceRedazionale=17R00404&art.idArticolo=9&art.idSottoArticolo=1&art.dataPubblicazioneGazzetta=2018-01-20&art.progressivo=0>

3. Normativa europea per il trattamento dei dati personali GDPR (General Data Protection Regulation) 2016/679:

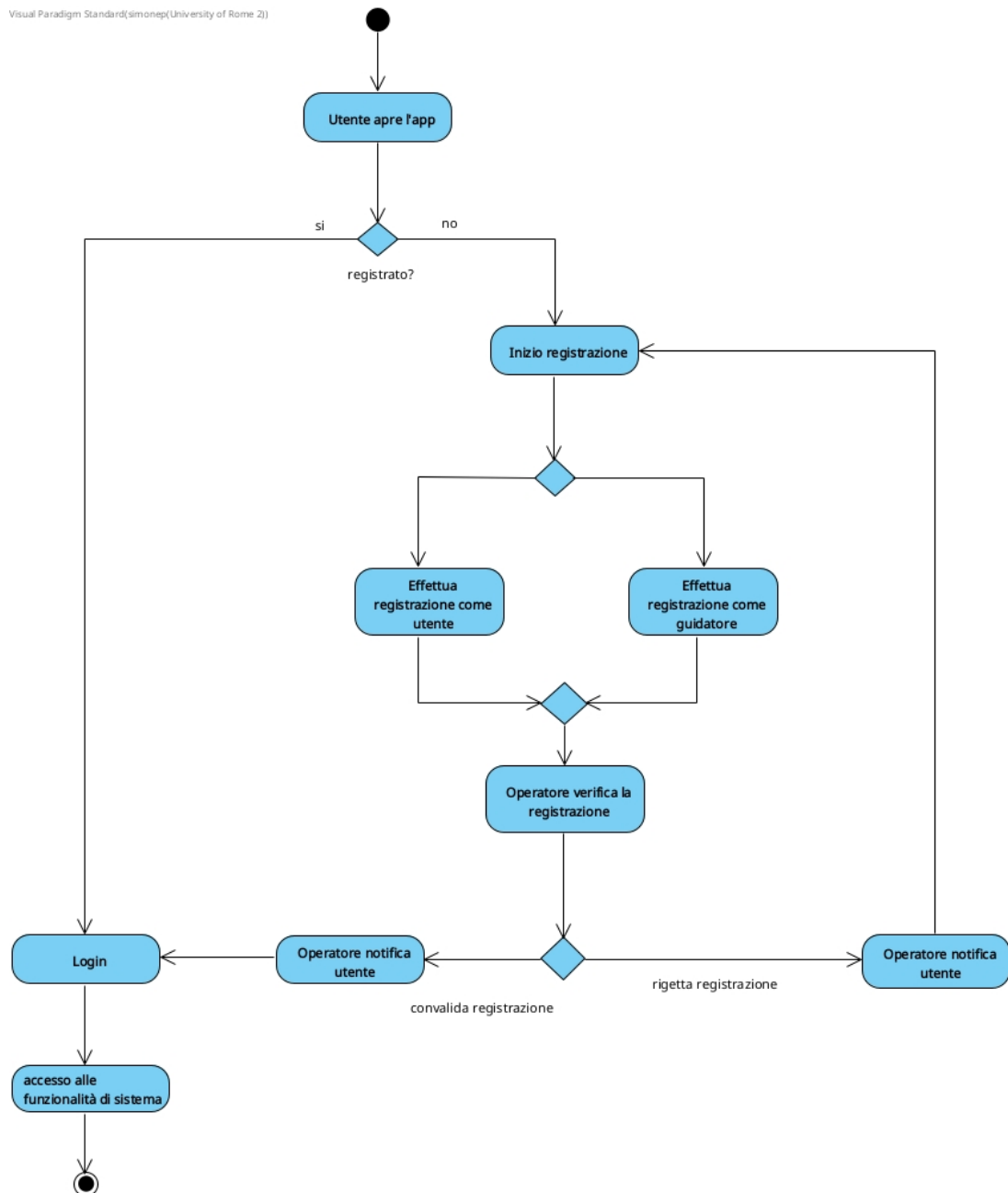
<https://eur-lex.europa.eu/legal-content/IT/TXT/PDF/?uri=CELEX:32016R0679&qid=1717231555698>

4. Regolamento (UE) 2016/679 del Parlamento europeo e del Consiglio, del 27 aprile 2016, relativo alla protezione delle persone fisiche con riguardo al trattamento dei dati personali, nonché alla libera circolazione di tali dati e che abroga la direttiva 95/46/CE (regolamento generale sulla protezione dei dati) (16CE1278)

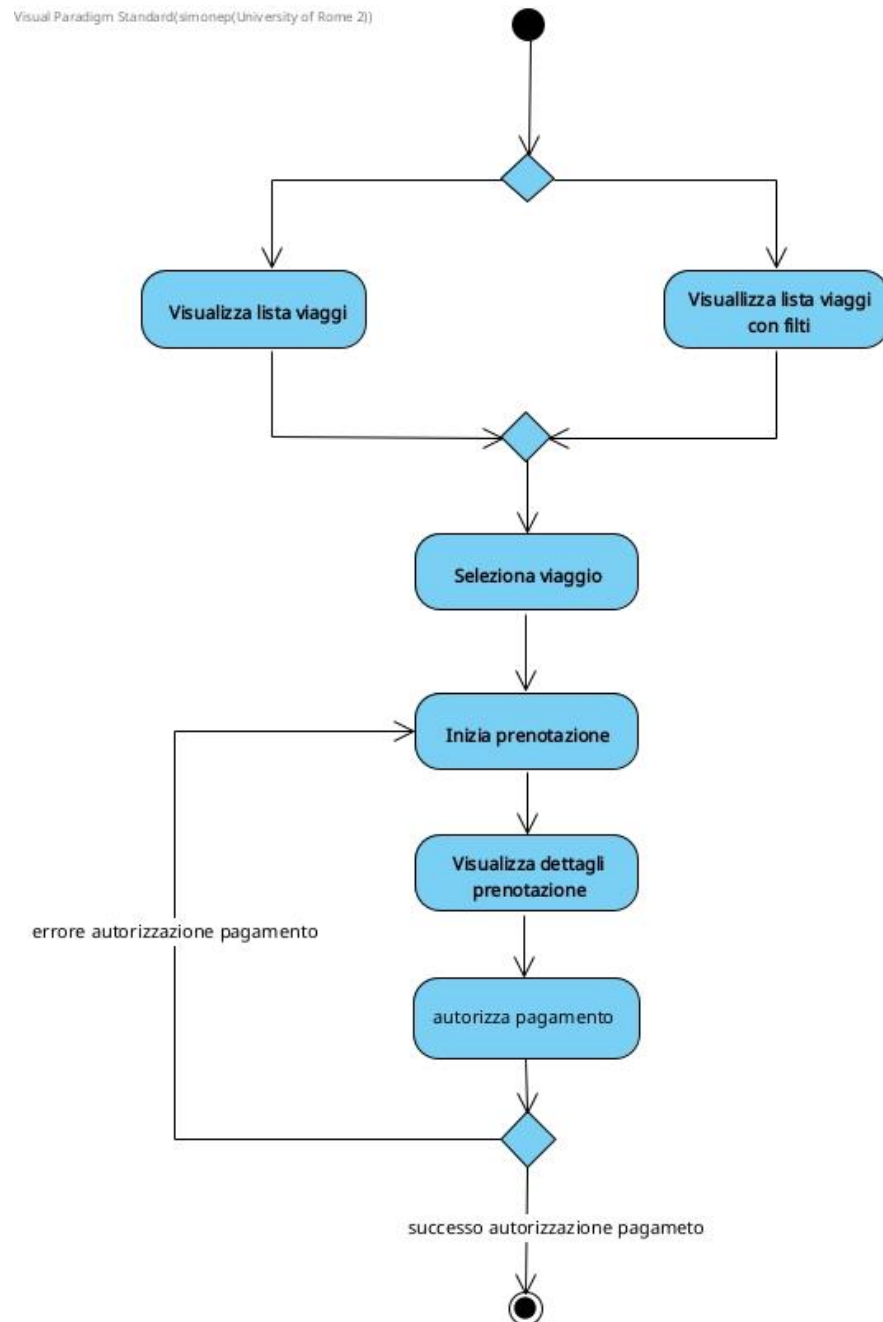
FASE ARCHITETTURA DI SISTEMA

Activity Diagram

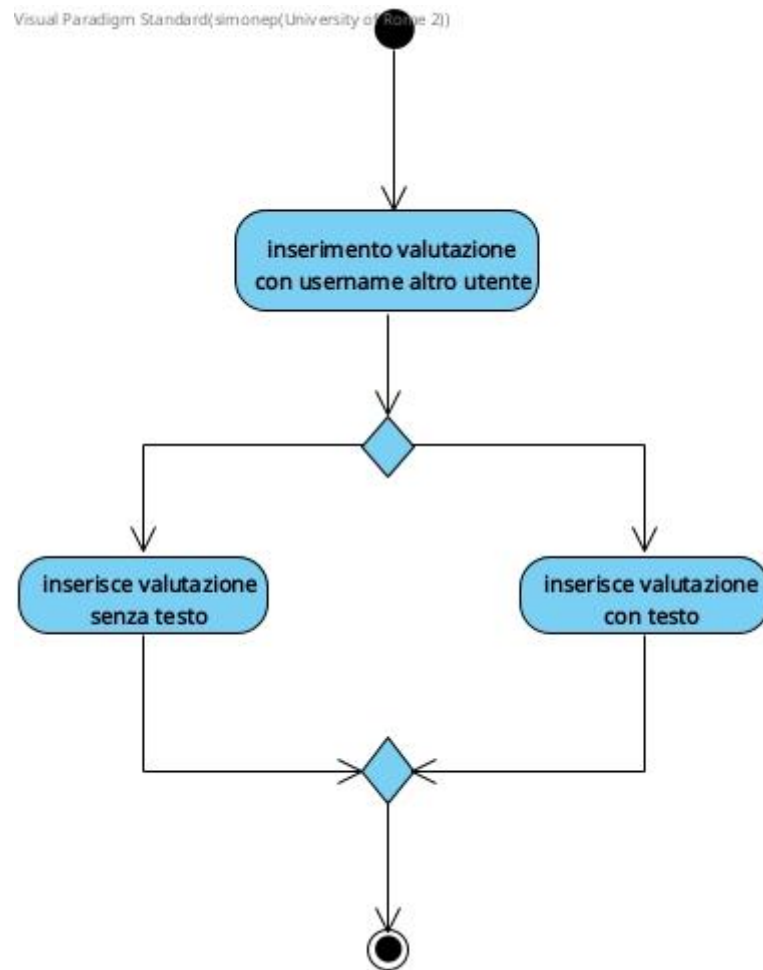
Activity diagram dell'accesso all'app da parte degli utenti:



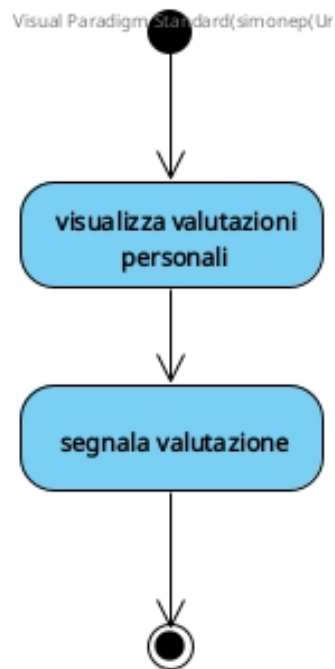
Activity diagram della prenotazione viaggio di utente registrato:



Activity diagram dell'inserimento valutazione altro utente svolto da utente registrato:



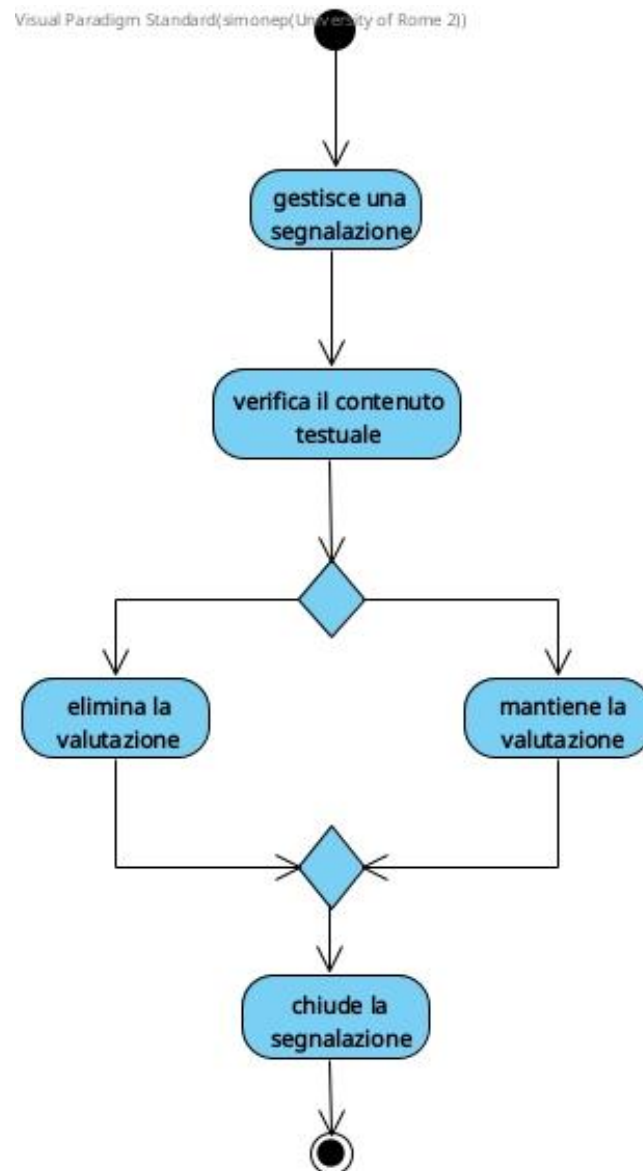
Activity diagram della segnalazione di una valutazione personale da parte di utente registrato:



Activity diagram di cancellazione viaggio da parte di utente registrato come guidatore:

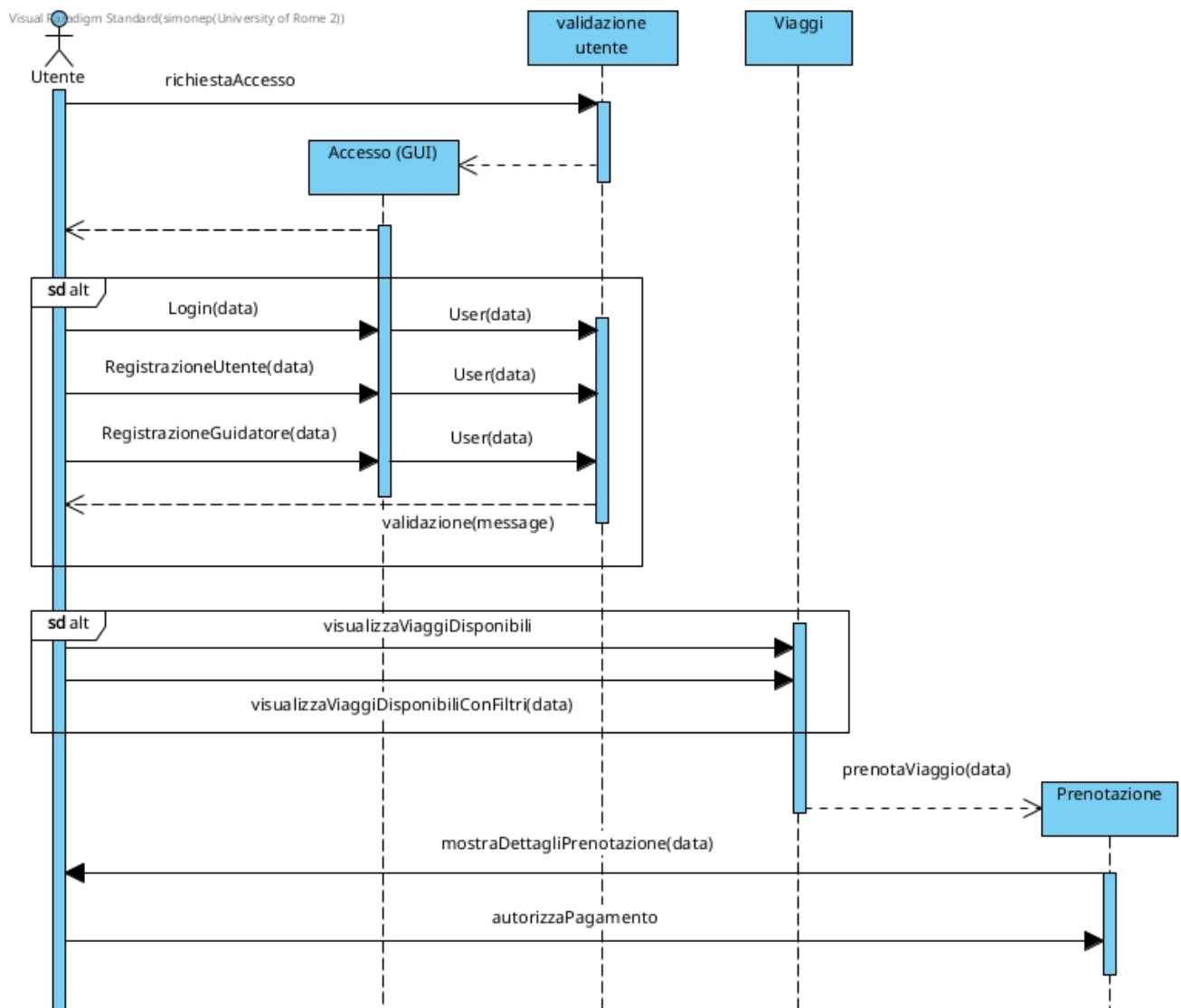


Activity diagram della gestione valutazioni da parte di un operatore:

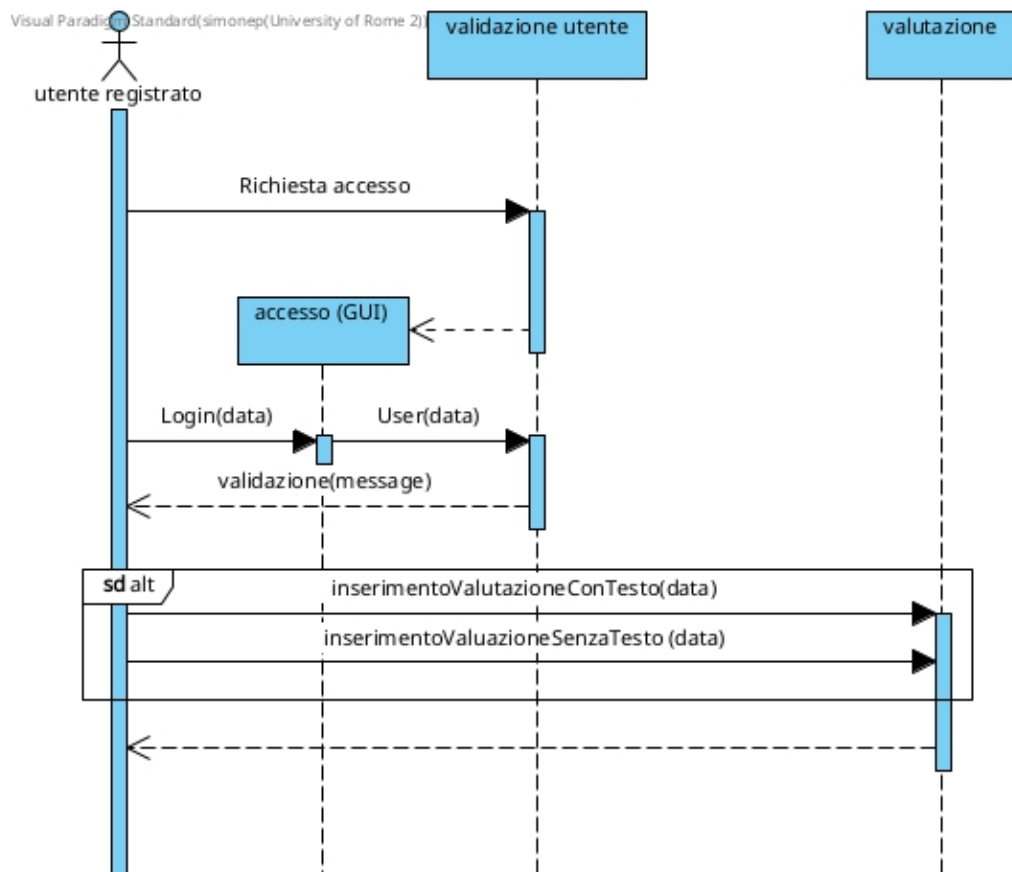


Sequence Diagram

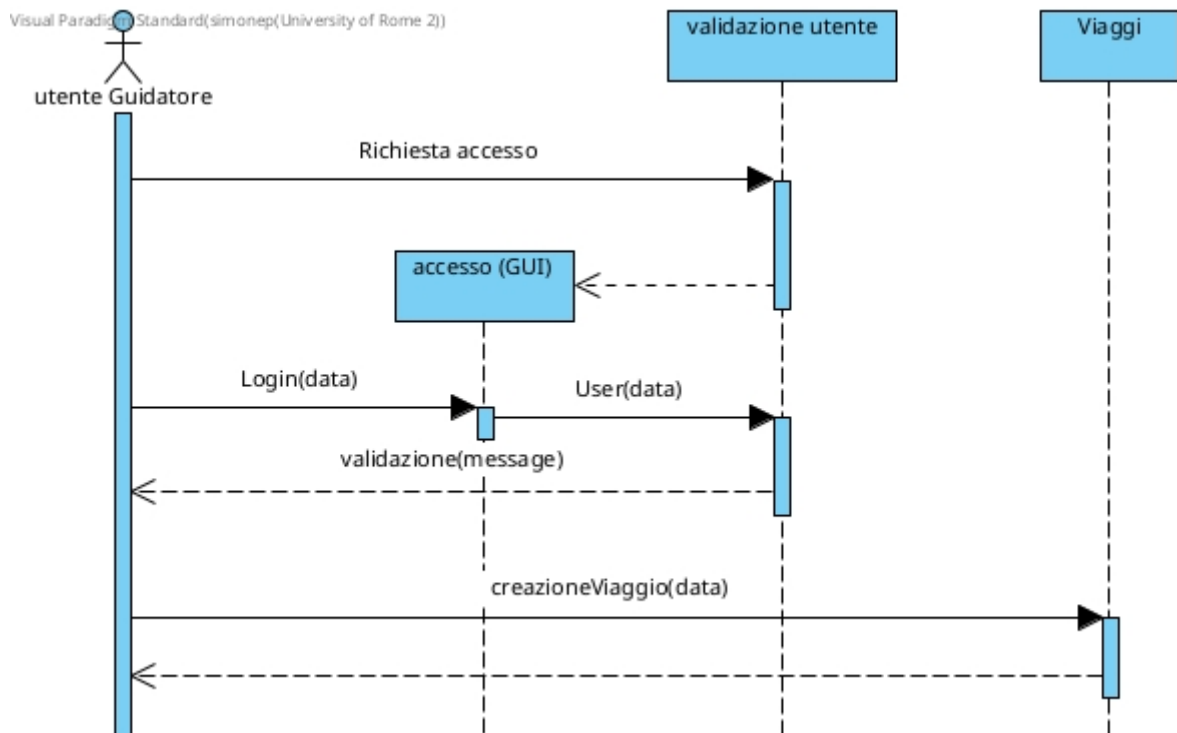
Sequence diagram del caso d'uso prenotazione viaggio:



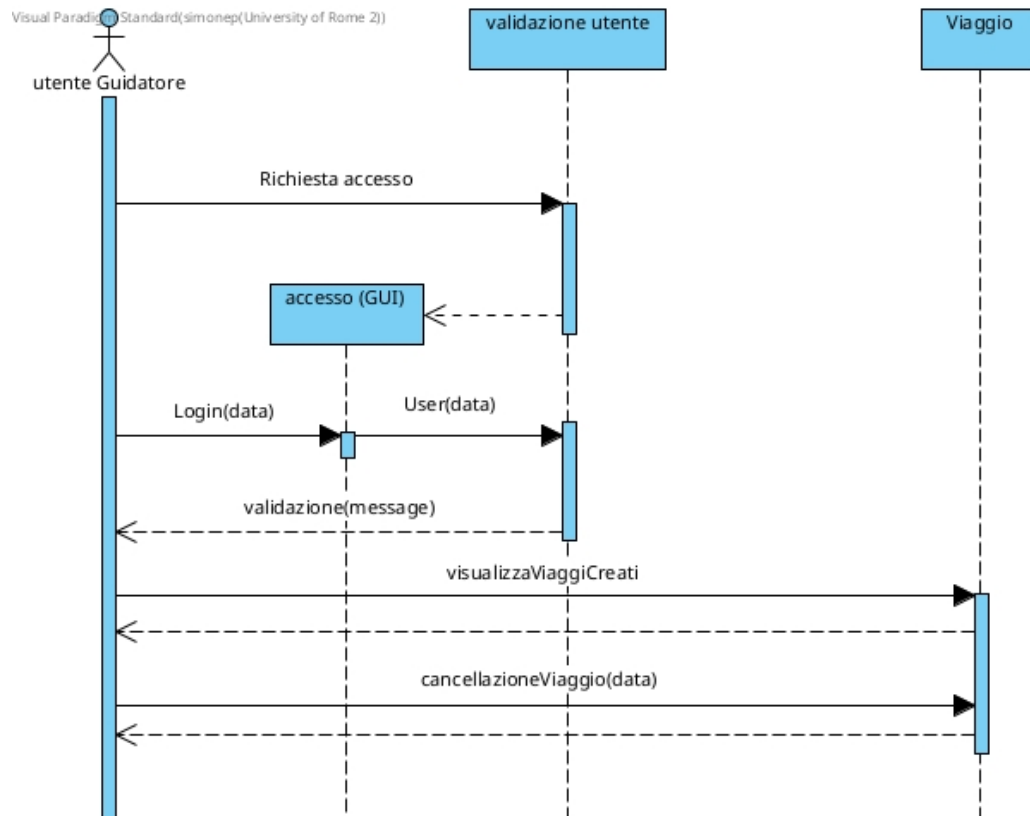
Sequence diagram del caso d'uso inserisci valutazione altro utente:



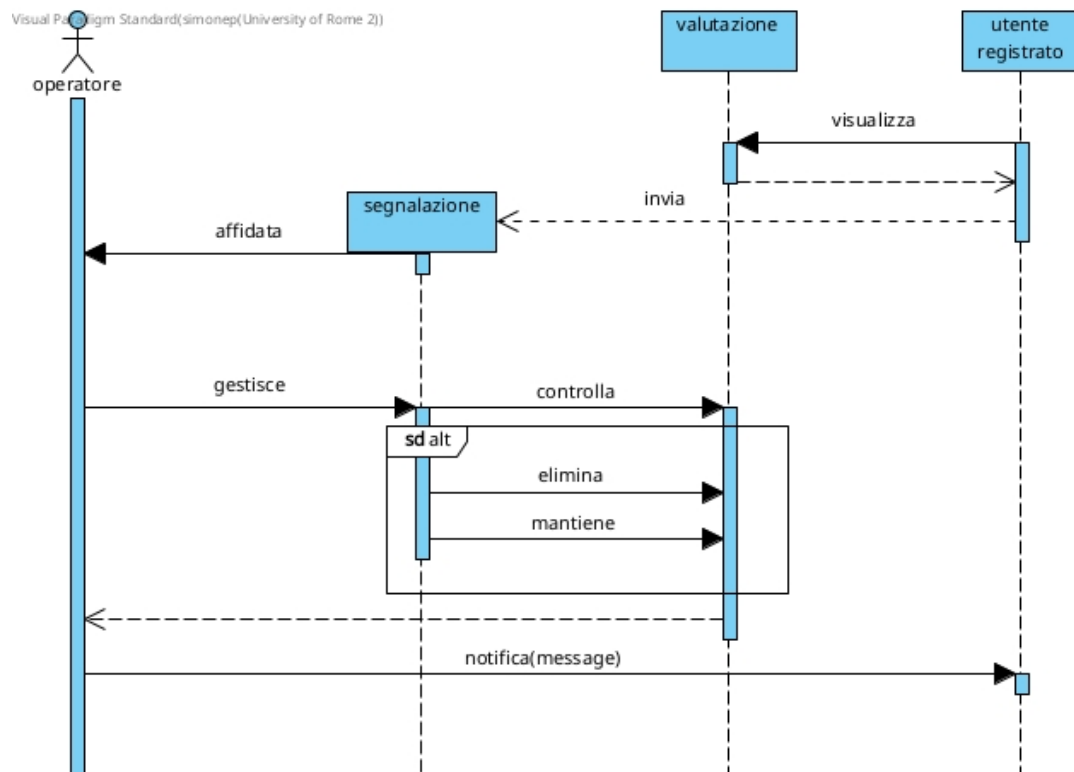
Sequence diagram del caso d'uso creazione viaggio:



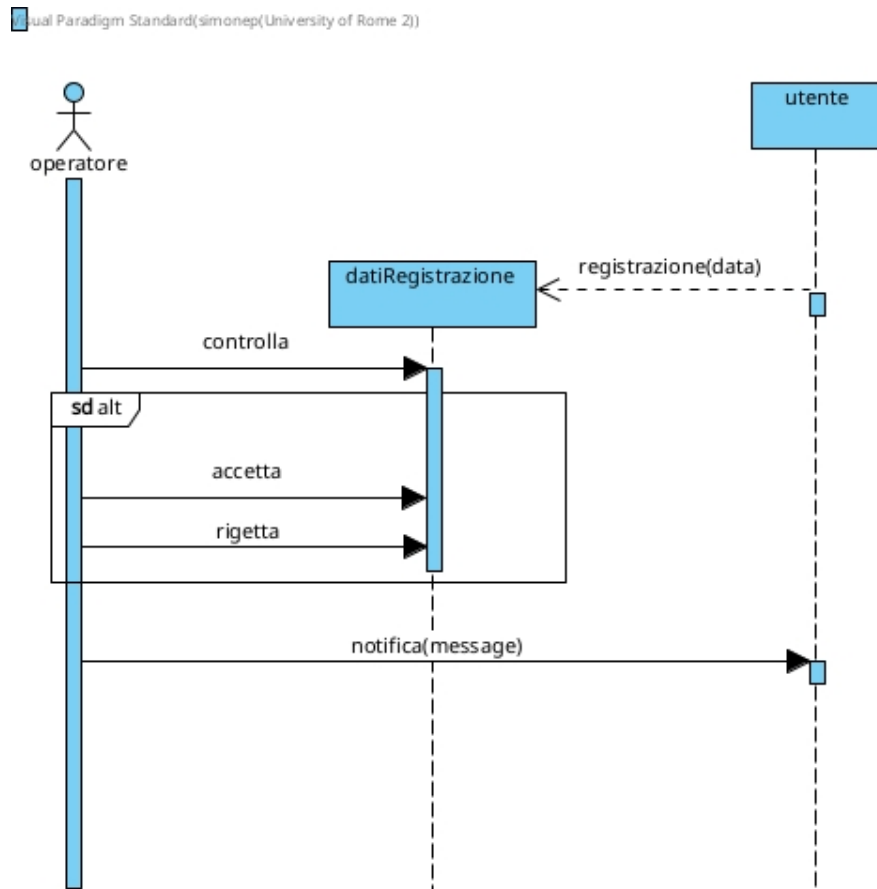
Sequence diagram del caso d'uso cancellazione viaggio:



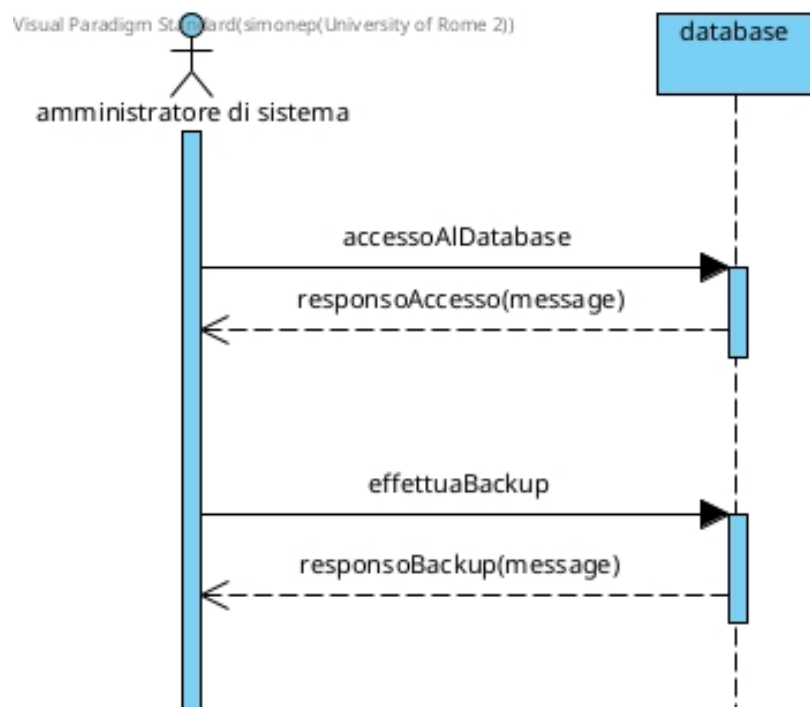
Sequence diagram del caso d'uso gestione valutazioni segnalate:



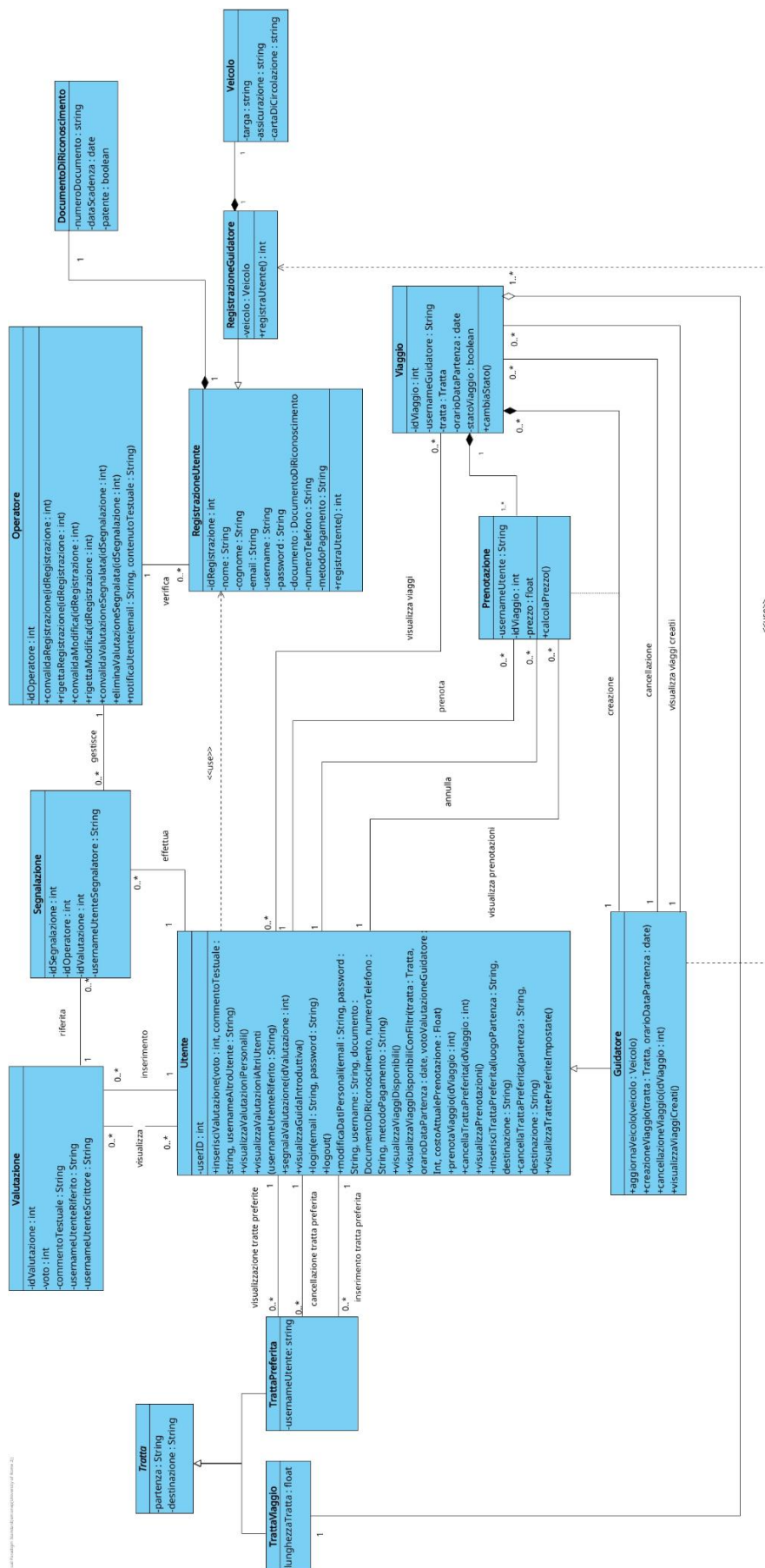
Sequence diagram del caso d'uso conferma o rigetta una registrazione:



Sequence diagram del caso d'uso effettua il backup:



Class Diagram



Vincoli:

- Se la registrazione di un utente avviene come guidatore, e quindi i dati sono inseriti all'interno della classe RegistrazioneGuidatore, l'attributo patente all'interno dell'istanza di DocumentoDiRiconoscimento associata alla classe RegistrazioneGuidatore deve essere impostato a true
- Se l'attributo orarioDataPartenza all'interno di un'istanza di viaggio indica un orario distante meno di un'ora, l'utente guidatore non deve avere la possibilità di eliminare il viaggio
- L'operazione inserisciValutazione può avere il parametro commentoTestuale NULL, questo avviene a livello di implementazione grazie all'utilizzo di classi wrapper, se il commentoTestuale è NULL viene inserita una valutazione senza commentoTestuale
- L'operazione modificaDatiPersonali può avere parametri NULL, questo avviene a livello di implementazione grazie all'utilizzo di classi wrapper, per i parametri impostati a NULL non viene effettuata la modifica dei dati
- L'operazione visualizzaViaggiDisponibiliConFiltri può avere parametri NULL, questo avviene a livello di implementazione grazie all'utilizzo di classi wrapper, i parametri impostati a NULL non vengono utilizzati per l'ordinamento della lista viaggi disponibili
- Una nuova istanza della classe Utente viene creata quando l'operatore chiama il metodo registraUtente() utilizzando i dati di registrazione forniti. Durante questo processo, il valore dell'attributo idRegistrazione, che si riferisce ai dati inseriti al momento della registrazione, viene assegnato all'attributo userID dell'istanza Utente. In altre parole, l'operatore accetta i dati di registrazione e crea una nuova istanza di Utente basata su quei dati, collegandoli tramite lo stesso ID (idRegistrazione coincide con userID). A questo punto, l'utente che aveva richiesto la registrazione diventa un utente effettivo dell'applicazione, potendo così effettuare il login e accedere alle funzionalità disponibili.

APPENDICE

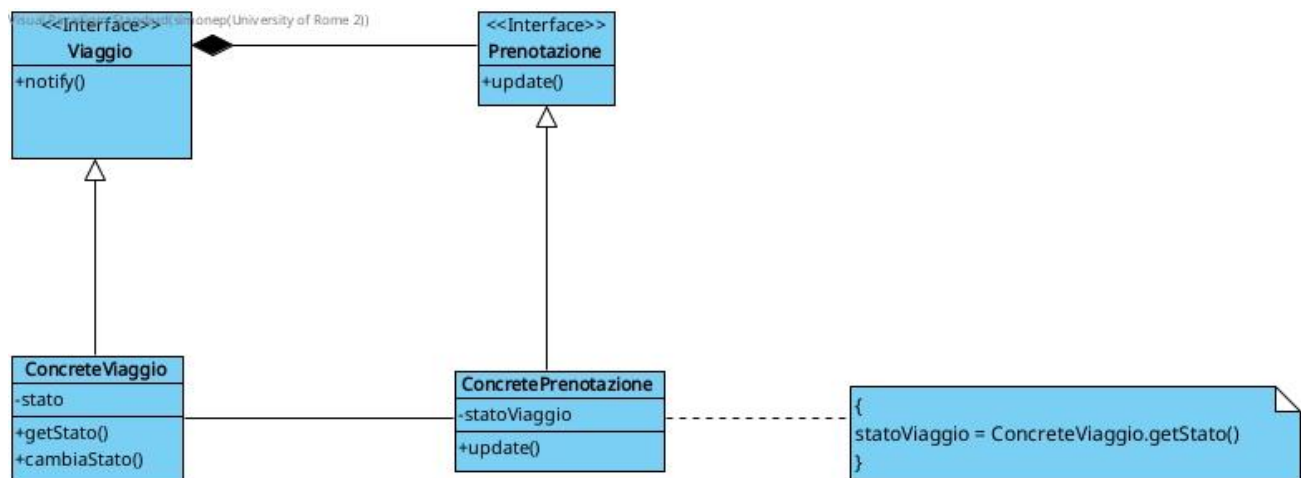
Eclipse EE (Enterprise Edition): Eclipse è un ambiente di sviluppo integrato multi-linguaggio e multiplatforma. È utilizzabile per la produzione di software di vario genere, fornendo un IDE completo per Java e plug-in che supportano XML, JavaScript, PHP, e altri linguaggi. Supporta l'uso di JUnit, un framework per il testing delle unità che utilizza un archivio .jar contenente collezioni di classi, offrendo vantaggi come risparmio di tempo e riduzione dell'errore umano durante l'esecuzione dei test. Eclipse è dotato di numerosi strumenti, tra cui quelli per il refactoring rapido ed efficiente del codice. L'uso della versione Enterprise Edition è motivato dalla possibilità di estendere la classe HttpServlet per implementare la parte di Application Server, fornendo servizi in risposta alle richieste del client.

Android Studio: Android Studio è un ambiente di sviluppo integrato per la piattaforma Android, basato su IntelliJ IDEA di JetBrains. Supporta linguaggi come Java, Kotlin e C++. È utilizzato per sviluppare la parte client delle applicazioni Android, sfruttando l'ART (Android Runtime) al posto della JVM (Java Virtual Machine), che offre prestazioni superiori. Il debugging è migliorato grazie alla diagnostica immediata dell'intero progetto e alla visualizzazione delle attività in una sezione chiamata logcat. Android Studio include un SDK (Software Development Kit) che fornisce vari strumenti per lo sviluppo di applicazioni Java, come emulatori per la simulazione e il testing, e piattaforme differenziate per livello di API, permettendo l'implementazione di funzionalità specifiche

MySQL: MySQL è un sistema di gestione di database relazionali (RDBMS) open source utilizzato per archiviare e gestire grandi quantità di dati in modo strutturato. È noto per la sua velocità, affidabilità e facilità d'uso. MySQL utilizza il linguaggio SQL (Structured Query Language) per l'interazione con il database, supportando operazioni come l'inserimento, l'aggiornamento, la cancellazione e il recupero dei dati. È ampiamente utilizzato in applicazioni web grazie alla sua capacità di gestire grandi volumi di traffico e alla compatibilità con vari linguaggi di programmazione come PHP, Java, e Python. MySQL offre funzionalità avanzate come la replica dei dati, il clustering per la scalabilità orizzontale, e strumenti per il backup e il ripristino dei dati. Inoltre, fornisce un sistema di permessi e accesso basato su utenti, che garantisce la sicurezza dei dati. Le sue prestazioni possono essere ottimizzate mediante l'uso di indici, tabelle partizionate e configurazioni di caching. MySQL è disponibile in diverse edizioni, tra cui la Community Edition, completamente gratuita, e le edizioni commerciali con supporto tecnico aggiuntivo e funzionalità avanzate.

DESIGN PATTERNS

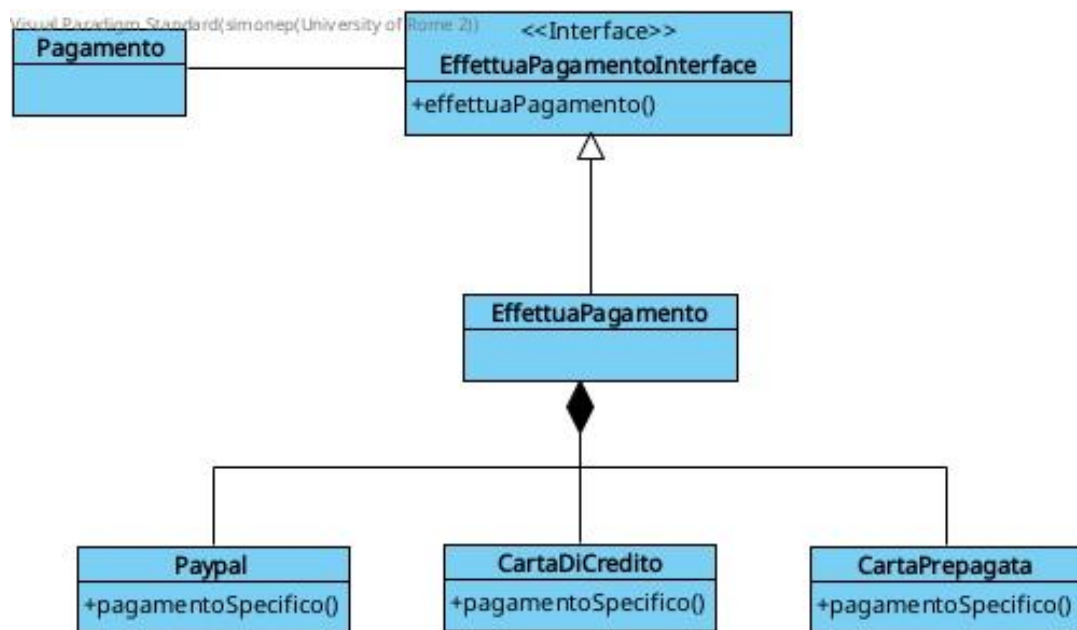
Observer pattern



Nome e classificazione	Observer - è un pattern comportamentale basato su oggetti
Scenario	Utente guidatore crea o cancella viaggi, ogni utente può creare istanze di prenotazione riferite a viaggi precedentemente creati e con stato attivo.
Motivazione di utilizzo	<p>Per notificare alle istanze di prenotazione se l'istanza di viaggio a cui si riferiscono è attiva o meno facciamo utilizzo del design pattern observer.</p> <p>Utilizziamo questo strumento per gestire la relazione di dipendenza uno-a-molti tra viaggio e prenotazioni. In altre parole, la variazione dello stato di una istanza di viaggio deve essere "osservata" dalle istanze di prenotazioni da cui è riferita, in modo che possano "aggiornarsi" automaticamente. La conoscenza dello stato di una istanza di viaggio consente un aggiornamento immediato e automatizzato delle istanze di prenotazione che si riferiscono a tale viaggio, in modo da adattarsi dinamicamente a qualsiasi cambiamento.</p> <p>Nel nostro caso, la modifica dello stato del viaggio influenza direttamente le prenotazioni associate a quel viaggio: se un viaggio è attivo possono essere create o cancellate istanze di prenotazione che si riferiscono a tale viaggio, se il viaggio non è attivo non saranno consentite eventuali richieste di cancellazione da parte dei prenotati in quanto fuori dal tempo limite o accettate nuove prenotazioni.</p>
Partecipanti	Viaggio, ConcreteViaggio, Prenotazione, ConcretePrenotazione

Conseguenze	<p>Sotto il profilo comportamentale, il pattern consente l'aggiornamento e la notifica di eventuali cambiamenti avvenuti nel soggetto. Nel nostro caso, ad esempio, quando lo stato di un viaggio viene modificato, viene inviata una notifica di tipo broadcast agli osservatori interessati (prenotazioni). Così, quando lo stato passa da attivo a non attivo, questa azione viene comunicata alle prenotazioni associate, che ad esempio perdono la possibilità di essere cancellate.</p>
--------------------	---

Adapter:



Nome e classificazione	Adapter – strutturale basato su oggetti
Scenario	<p>Possibile utilizzo del design pattern adapter per gestire il pagamento specificando meglio come funziona questa operazione di pagamento rispetto a quanto fatto nel class diagram precedente (dove avviene implicitamente nel metodo prenotaViaggio). In questo nuovo scenario la funzione di pagamento deve supportare diversi metodi di pagamento come PayPal, carta di credito e carta prepagata. Il design pattern Adapter viene utilizzato per unificare l'interfaccia di pagamento, in modo che il sistema possa gestire i pagamenti senza conoscere le specifiche di ciascun metodo.</p>
Motivazione di utilizzo	Utilizziamo il design pattern Adapter per gestire l'operazione di pagamento all'interno

	<p>dell'applicazione. Questo approccio consente di integrare diversi metodi di pagamento senza dover implementare logiche specifiche per ciascun metodo di pagamento nel sistema principale. L'adattatore si occupa di adattare ciascun metodo di pagamento all'interfaccia comune, semplificando così l'integrazione e la gestione dei pagamenti.</p>
Partecipanti	<p>Pagamento, EffettuaPagamentoInterface, EffettuaPagamento, Paypal, CartaDiCredito, CartaPrepagata</p>
Conseguenze	<p>Decoupling: Il pattern consente di separare l'implementazione specifica dei vari metodi di pagamento dal resto del sistema, garantendo un basso accoppiamento tra le componenti.</p> <p>Riutilizzo del codice: Gli adattatori consentono di riutilizzare le classi esistenti per i diversi metodi di pagamento, evitando la duplicazione del codice e promuovendo la modularità e la manutenibilità del sistema.</p> <p>Estensibilità: Aggiungere nuovi metodi di pagamento diventa più semplice, poiché è sufficiente creare un nuovo adattatore che implementi l'interfaccia comune per il pagamento.</p> <p>Facilità di test: Grazie all'interfaccia comune per i pagamenti, è più agevole scrivere test automatizzati per verificare il corretto funzionamento del sistema di pagamento.</p> <p>Consistenza: Tutti i metodi di pagamento vengono gestiti in modo uniforme all'interno dell'applicazione, garantendo una user experience coerente per gli utenti.</p>