



Sviluppo Software

Noleggio Auto

Enrico Gentile

Hamza En Nakhly

Giuseppe Alfieri



N46006046
N46005845
N46005956

INDICE

Sistema di noleggio online di autovetture

1. Specifiche informali.....	2
2. Analisi e specifica dei requisiti	3
2.1 Analisi nomi-verbi.....	3
2.2 Revisione dei requisiti	4
2.3 Glossario dei termini.....	5
2.4 Classificazione dei requisiti.....	5
2.4.1 Requisiti funzionali	5
2.4.2 Requisiti sui dati.....	6
2.4.3 Vincoli / Altri requisiti.....	6
2.5 Modellazione dei casi d'uso.....	7
2.6 Diagramma dei casi d'uso	9
2.7 Scenari.....	9
2.8 Diagramma delle classi.....	16
2.9 Diagrammi di sequenza.....	16
2.10 Verifica della completezza dei requisiti	19
3. Stima dei costi.....	20
4. Piano di test funzionale	23
5. Progettazione.....	37
5.1 Diagramma delle classi.....	37
5.2 Diagrammi di sequenza.....	37
6. Implementazione.....	39
7. Testing.....	41
7.1 Test strutturale.....	41
7.2 Test funzionale.....	44
8. Manutenzione	46

1. Specifiche informali

Sistema noleggio auto

Si vuole realizzare un sistema informatico per una società di noleggio auto. Sul sistema agiscono gli operatori della società di noleggio, che gestiscono il parco auto, e i guidatori, che possono verificare la disponibilità (e i prezzi) delle auto e procedere al noleggio.

Il sistema deve registrare per ogni veicolo: la targa, il prezzo fissato per giorno-noleggio, il numero di passeggeri per cui l'auto è omologata, l'alimentazione (benzina/gasolio) e la potenza del motore. La compagnia gestisce i segmenti di veicoli B, C e D, che corrispondono alle auto small, medium e large.

Gli operatori possono aggiungere o modificare i dati relativi ai veicoli della compagnia. Poiché il sistema deve registrare tutte le auto che sono state impiegate per il noleggio, non è possibile rimuovere le auto dal sistema ma i funzionari all'atto di modifica possono alterare lo stato dei veicoli tra in-servizio o dismesso.

Gli utenti possono ricercare sul sistema le auto in base all'intervallo temporale di noleggio (data ritiro e data consegna) e al segmento desiderato. Il sistema deve mostrare loro le auto per il segmento richiesto che sono disponibili nel periodo indicato: un'auto è effettivamente disponibile per il noleggio se risulta al momento della ricerca come in-servizio e se non esistono altre prenotazioni che la impegnano anche parzialmente con le date indicate per la prenotazione.

Un utente che non possiede le credenziali può registrarsi in ogni momento sulla piattaforma specificando: nome, cognome, data di nascita, email, anno di conseguimento, scadenza e numero della patente di guida.

Un utente registrato può interagire col sistema per noleggiare un'auto. Il noleggio consiste di una fase di ricerca di un'auto, seguita dalla specifica di accessori da includere alla prenotazione: gli accessori si distinguono in servizi assicurativi e optional auto. Un utente può scegliere al più un servizio assicurativo accessorio, perché tali servizi vanno in conflitto tra loro (ad esempio un servizio assicurativo prevede una riduzione della franchigia, un altro la elimina); d'altro canto, non esistono limiti di scelta per l'utente riguardo gli optional auto.

Tutti gli accessori hanno una descrizione testuale e un prezzo prefissato che contribuisce a determinare il costo totale della prenotazione. Il costo della prenotazione è dato dal prodotto del prezzo giorno-noleggio per i giorni di prenotazione, sommato con i prezzi di tutti i servizi accessori scelti. Infine, all'atto della prenotazione è possibile aggiungere gratuitamente un ulteriore utente tra quelli registrati nel sistema associandolo alla prenotazione come guidatore supplementare.

2. Analisi e specifica dei requisiti

2.1 Analisi nomi-verbi

Sistema noleggio auto

Si vuole realizzare un sistema informatico per una società di noleggio auto. Sul sistema agiscono gli operatori della società di noleggio, che gestiscono il parco auto, e i guidatori, che possono verificare la disponibilità (e i prezzi) delle auto e procedere al noleggio.

Il sistema deve registrare per ogni veicolo: la targa, il prezzo fissato per giorno-noleggio, il numero di passeggeri per cui l'auto è omologata, l'alimentazione (benzina/gasolio) e la potenza del motore. La compagnia gestisce i segmenti di veicoli B, C e D, che corrispondono alle auto small, medium e large.

Gli operatori possono aggiungere o modificare i dati relativi ai veicoli della compagnia. Poiché il sistema deve registrare tutte le auto che sono state impiegate per il noleggio, non è possibile rimuovere le auto dal sistema ma i funzionari all'atto di modifica possono alterare lo stato dei veicoli tra in-servizio o dismesso.

Gli utenti possono ricercare sul sistema le auto in base all'intervallo temporale di noleggio (data ritiro e data consegna) e al segmento desiderato. Il sistema deve mostrare loro le auto per il segmento richiesto che sono disponibili nel periodo indicato: un'auto è effettivamente disponibile per il noleggio se risulta al momento della ricerca come in-servizio e se non esistono altre prenotazioni che la impegnano anche parzialmente con le date indicate per la prenotazione.

Un utente che non possiede le credenziali può registrarsi in ogni momento sulla piattaforma specificando: nome, cognome, data di nascita, email, anno di conseguimento, scadenza e numero della patente di guida.

Un utente registrato può interagire col sistema per noleggiare un'auto. Il noleggio consiste di una fase di ricerca di un'auto, seguita dalla specifica di accessori da includere alla prenotazione: gli accessori si distinguono in servizi assicurativi e optional auto. Un utente può scegliere al più un servizio assicurativo accessorio, perché tali servizi vanno in conflitto tra loro (ad esempio un servizio assicurativo prevede una riduzione della franchigia, un altro la elimina); d'altro canto, non esistono limiti di scelta per l'utente riguardo gli optional auto.

Tutti gli accessori hanno una descrizione testuale e un prezzo prefissato che contribuisce a determinare il costo totale della prenotazione. Il costo della prenotazione è dato dal prodotto del prezzo giorno-noleggio per i giorni di prenotazione, sommato con i prezzi di tutti i servizi accessori scelti. Infine, all'atto della prenotazione è possibile aggiungere gratuitamente un ulteriore utente tra quelli registrati nel sistema associandolo alla prenotazione come guidatore supplementare.

LEGENDA:

Classe

Attributo

Funzionalità

Attore

Classe-Attore

2.2 Revisione dei requisiti

1. Il sistema deve verificare la disponibilità delle auto in fase di ricerca.
2. Il sistema deve consentire ad un utente registrato di noleggiare un'auto.
3. Per ogni auto si deve memorizzare: la targa, il prezzo fissato per giorno-noleggio, il numero di passeggeri per cui l'auto è omologata, l'alimentazione (benzina/gasolio) e la potenza del motore.
4. Il parco auto gestisce i segmenti di auto B, C, D (small, medium, large).
5. Il sistema deve consentire agli operatori di aggiungere i dati relativi alle auto della compagnia.
6. Il sistema deve consentire agli operatori di modificare i dati relativi alle auto della compagnia.
7. Non è possibile rimuovere le auto dal sistema.
8. Un operatore può alterare lo stato di un'auto da in-servizio a dismesso o viceversa.
9. Un'auto può essere in stato in-servizio o dismesso.
10. Il sistema deve consentire all'utente di ricercare le auto sul sistema.
11. La ricerca di un'auto sul sistema viene effettuata in base ad un intervallo temporale di noleggio (data ritiro e data consegna) e un segmento specificato.
12. Un'auto è disponibile per il noleggio se risulta al momento della ricerca come in-servizio e se non esistono altre prenotazioni che la impegnano anche parzialmente con le date indicate per la prenotazione
13. Il sistema deve consentire ad un utente senza credenziali di registrarsi.
14. Il sistema deve memorizzare per ogni utente registrato: nome, cognome, data di nascita, email, anno di conseguimento, scadenza e numero della patente di guida.
15. Il sistema deve consentire in fase di noleggio di specificare gli accessori.
16. Gli accessori si distinguono in servizi assicurativi e optional auto.
17. Non ci sono limiti sulla scelta degli optional auto.

18. Un utente può scegliere al più un servizio assicurativo accessorio.

19. Gli accessori hanno una descrizione testuale e un prezzo prefissato.

20. Il costo totale della prenotazione è calcolato come il prodotto del prezzo giornaliero di noleggio per il numero di giorni di prenotazione, sommato ai prezzi di tutti gli accessori scelti.

21. Il sistema deve permettere ad un utente, all'atto del noleggio, di specificare optionalmente un utente registrato come guidatore supplementare.

2.3 Glossario dei termini

Termine	Descrizione	Sinonimi
Auto	Elementi di cui è composto il parco d'auto	Veicolo
Utenti	Un generico utente che può visualizzare le informazioni del parco auto.	
Utente registrato	Un utente che ha effettuato la procedura di registrazione	Guidatori
Segmenti	Sistemi di classificazione delle automobili.	
Accessori	Servizi offerti dal sistema di noleggio auto che possono essere aggiunti all'interno di una prenotazione previo sovrapprezzo.	
Guidatore Supplementare	Guidatore del veicolo che può legalmente sostituire il titolare della prenotazione alla guida del veicolo.	

2.4 Classificazione dei requisiti

2.4.1 Requisiti funzionali

ID	Requisito	Origine (n. frase dei requisiti revisionati)
RF01	Il sistema deve verificare la disponibilità delle auto in fase di ricerca.	
RF02	Il sistema deve consentire ad un utente registrato di noleggiare un'auto.	
RF03	Il sistema deve consentire agli operatori di aggiungere i dati relativi ai veicoli della compagnia.	
RF04	Il sistema deve consentire agli operatori di modificare i dati relativi ai veicoli della compagnia.	

RF05	Il sistema deve consentire a un utente di ricercare un'auto sul sistema.	
RF06	Il sistema deve consentire a un utente senza credenziali di registrarsi.	
RF07	Il sistema deve consentire in fase noleggio di specificare gli accessori.	
RF08	Il sistema deve consentire in fase di noleggio di specificare optionalmente un utente registrato come guidatore supplementare.	
RF09	Il sistema deve consentire ad un utente di effettuare il login.	

2.4.2 Requisiti sui dati

ID	Requisito	Origine (n. frase dei requisiti revisionati)
RD01	Per ogni auto si deve memorizzare: la targa, il prezzo fissato per giorno-noleggio, il numero di passeggeri per cui l'auto è omologata, l'alimentazione (benzina/gasolio) e la potenza del motore.	
RD02	I veicoli sono gestiti in segmenti A, B, C (small, medium, large).	
RD03	Un'auto può essere in stato in-servizio o dismesso.	
RD04	Per ogni utente registrato si deve memorizzare: nome, cognome, data di nascita, email, anno di conseguimento, scadenza e numero della patente di guida.	
RD05	Gli accessori si distinguono in servizi assicurativi e optional auto.	
RD06	Gli accessori hanno una descrizione testuale e un prezzo prefissato.	
RD07	Il costo totale della prenotazione è calcolato come il prodotto del prezzo giornaliero di noleggio per il numero di giorni di prenotazione, sommato ai prezzi di tutti gli accessori scelti.	
RD08	La ricerca di un'auto sul sistema viene effettuata in base ad un intervallo temporale di noleggio (data ritiro e data consegna) e un segmento specificato.	
RD09	Un'auto è disponibile per il noleggio se risulta al momento della ricerca come in-servizio e se non esistono altre prenotazioni che la impegnano anche parzialmente con le date indicate per la prenotazione.	
RD10	Un utente può scegliere al più un servizio assicurativo accessorio.	
RD11	Non ci sono limiti sulla scelta degli optional auto.	

2.4.3 Vincoli/Altri requisiti

ID	Requisito	Origine (n. frase dei requisiti revisionati)
V01	Non è possibile rimuovere le auto dal sistema.	

2.5 Modellazione dei casi d'uso

Attori Primari:

- Utente
- Utente Registrato
- Operatore

Attori Secondari:

Casi d'uso:

- UC1: NoleggiaAuto
- UC2: Autenticazione
- UC3: Registrazione
- UC4: RicercaAuto
- UC5: ModificaDatiAuto
- UC6: AggiungiDatiAuto

Casi d'uso di inclusione:

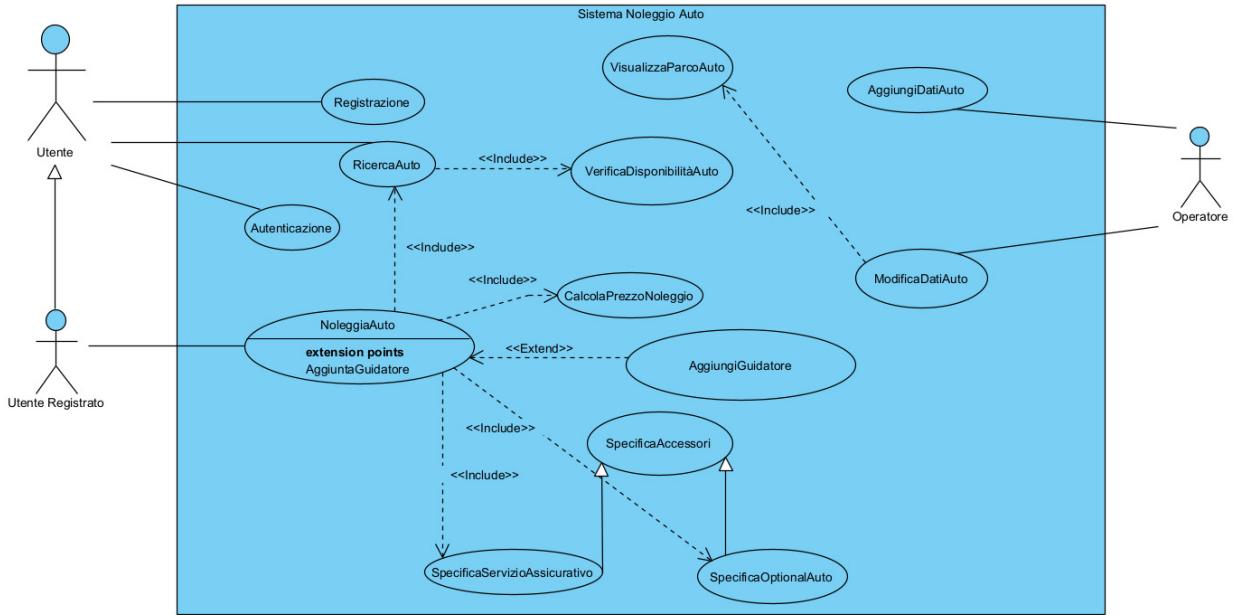
- UC4: RicercaAuto
- UC7: VisualizzaParcoAuto
- UC8: VerificaDisponibilitàAuto
- UC9: CalcolaPrezzoNoleggio
- UC10: SpecificaAccessori
- UC11: SpecificaServizioAssicurativo
- UC12: SpecificaOptionalAuto

Casi d'uso di estensione:

- UC13: AggiungiGuidatore

Caso d'uso	Attori Primari	Attori Secondari	Incl. / Ext.	Requisiti corrispondenti
UC1: NoleggiaAuto	Utente Registrato	-	Include: RicercaAuto CalcolaPrezzoNoleggio SpecificaAccessori	RF02
UC2: Autenticazione	Utente	-	-	RF09
UC3: Registrazione	Utente	-	-	RF06
UC4: RicercaAuto	Utente	-	Include: VerificaDisponibilitàAuto Incluso in: NoleggiaAuto	RF05
UC5: ModificaDatiAuto	Operatore	-	Include: VisualizzaParcoAuto	RF04
UC6: AggiungiDatiAuto	Operatore	-	-	RF03
UC7: VisualizzaParcoAuto	Operatore	-	Incluso in: ModificaDatiAuto	RF04
UC8: VerificaDisponibilitàAuto	Utente Registrato	-	Incluso in: RicercaAuto	RF01
UC9: CalcolaPrezzoNoleggio	-	-	Incluso in: NoleggiaAuto	RF02
UC10: SpecificaAccessori	Utente Registrato	-	Generalizzazione di: SpecificoServizioAssicurativo SpecificOptionalAuto Incluso in: NoleggiaAuto	RF07
UC11: SpecificoServizioAssicurativo	Utente Registrato	-	-	RF07
UC12: SpecificOptionalAuto	Utente Registrato	-	-	RF07
UC13: AggiungiGuidatore	Utente Registrato	-	Estensione di: NoleggiaAuto	RF08

2.6 Diagramma dei casi d'uso



2.7 Scenari

Di seguito troviamo l'analisi degli scenari di tutti i casi d'uso visti in precedenza:

<i>Caso d'uso:</i>	<i>Caso d'uso di inclusione: CalcolaPrezzoNoleggio</i>
<i>Attore primario</i>	Utente Registrato
<i>Attore secondario</i>	-
<i>Descrizione</i>	Il sistema calcola il prezzo totale del noleggio auto
<i>Pre-condizioni</i>	-
<i>Sequenza di eventi principale</i>	<ol style="list-style-type: none"> Il sistema calcola il prezzo del noleggio considerando il prezzo del servizio assicurativo specificato Il sistema calcola il prezzo del noleggio considerando il prezzo degli optional auto specificati Il sistema calcola il prezzo del noleggio considerando l'intervallo temporale di noleggio e il prezzo giorno-noleggio dell'auto Il sistema mostra all'utente il prezzo complessivo al termine della procedura
<i>Post-condizioni</i>	-
<i>Casi d'uso correlati</i>	NoleggiaAuto
<i>Sequenza di eventi alternativi</i>	-

<i>Caso d'uso:</i>	<i>NoleggiaAuto</i>
<i>Attore primario</i>	Utente Registrato
<i>Attore secondario</i>	-
<i>Descrizione</i>	Un utente registrato può noleggiare un'auto del segmento scelto tra quelle visualizzate che sono disponibili nell'intervallo temporale scelto
<i>Pre-Condizioni</i>	-
<i>Sequenza di eventi Principale</i>	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando l'utente registrato richiede di fare il noleggio 2. include(RicercaAuto) 3. L'utente registrato sceglie l'auto desiderata tra quelle disponibili. 4. include(SpecificaServizioAssicurativo) 5. include(SpecificaOptionalAuto) 6. include(CalcolaPrezzoNoleggio) 7. Il sistema richiede all'utente registrato se vuole aggiungere un guidatore supplementare alla prenotazione. Punto di estensione : aggiungiGuidatore 8. Il sistema mostra all'utente registrato il costo totale del noleggio. 9. L'utente registrato conferma il pagamento 10. Il sistema mostra un messaggio di conferma del noleggio
<i>Post-Condizioni</i>	Il sistema rende l'auto selezionata non disponibile nell'intervallo di tempo scelto dall'utente registrato.
<i>Casi d'uso correlati</i>	nessuno
<i>Sequenza di eventi alternativi</i>	-
<i>Caso d'uso:</i>	<i>Caso d'uso di inclusione: RicercaAuto</i>
<i>Attore primario</i>	Utente
<i>Attore secondario</i>	-
<i>Descrizione</i>	Il sistema ricerca le auto disponibili per il noleggio
<i>Pre-condizioni</i>	-
<i>Sequenza di eventi Principale</i>	<ol style="list-style-type: none"> 5. L'utente specifica l'intervallo temporale di noleggio e il segmento 6. Per ogni auto presente nel parco auto il sistema <ol style="list-style-type: none"> 6.1 include(VerificaDisponibilitàAuto) 7. Il sistema mostra una lista delle auto disponibili all'utente 8. Il sistema mostra un messaggio di ERRORE se non ci sono auto disponibili per il noleggio
<i>Post-condizioni</i>	-
<i>Casi d'uso correlati</i>	<i>NoleggiaAuto</i>

Sequenza di eventi alternativi

-

Caso d'uso:

Caso d'uso di inclusione: VerificaDisponibilitàAuto

<i>Attore primario</i>	Utente
<i>Attore secondario</i>	-
<i>Descrizione</i>	Il sistema verifica se un'auto è disponibile per il noleggio
<i>Pre-condizioni</i>	-
<i>Sequenza di eventi Principale</i>	<ol style="list-style-type: none"> 1. Se il sistema verifica che l'auto è in servizio e che non ci sono altri noleggi che la impiegano nell'intervallo di tempo specificato <ol style="list-style-type: none"> 1.1 Il sistema segnala che l'auto è disponibile 2. Altrimenti 3. Il sistema segnala che l'auto non è disponibile
<i>Post-condizioni</i>	-
<i>Casi d'uso correlati</i>	<i>RicercaAuto</i>

Sequenza di eventi alternativi

Caso d'uso di inclusione: SpecificaServizioAssicurativo

<i>Attore primario</i>	Utente Registrato
<i>Attore secondario</i>	-
<i>Descrizione</i>	L'utente in fase di noleggio può specificare un servizio assicurativo
<i>Pre-condizioni</i>	-
<i>Sequenza di eventi Principale</i>	<ol style="list-style-type: none"> 1. Il sistema mostra una vista dei servizi assicurativi. 2. L'utente specifica un servizio assicurativo. 3. Il sistema invia un messaggio di conferma all'utente al termine della specifica
<i>Post-condizioni</i>	-
<i>Casi d'uso correlati</i>	<i>NoleggiaAuto</i>

Sequenza di eventi alternativi

<i>Caso d'uso:</i>	<i>Caso d'uso di inclusione: SpecificaOptionalAuto</i>
<i>Attore primario</i>	Utente Registrato
<i>Attore secondario</i>	-
<i>Descrizione</i>	L'utente in fase di noleggio può specificare degli optional auto
<i>Pre-condizioni</i>	L'utente ha iniziato un noleggio.
<i>Sequenza di eventi</i>	<ol style="list-style-type: none"> 1. Il sistema mostra una vista degli optional auto 2. L'utente specifica uno o più optional auto 3. Il sistema invia un messaggio di conferma all'utente al termine della specifica
<i>Principale</i>	
<i>Post-condizioni</i>	-
<i>Casi d'uso correlati</i>	<i>NoleggiaAuto</i>
<i>Sequenza di eventi alternativi</i>	-

<i>Caso d'uso:</i>	<i>Caso d'uso di estensione: AggiungiGuidatoreSupplementare</i>
<i>Attore primario</i>	Utente Registrato
<i>Attore secondario</i>	-
<i>Descrizione</i>	L'utente in fase di noleggio può specificare un utente registrato come guidatore supplementare
<i>Pre-condizioni</i>	Durante la fase di noleggio, l'utente sceglie di inserire un guidatore supplementare.
<i>Sequenza di eventi</i>	<ol style="list-style-type: none"> 1. L'utente specifica l'email del guidatore supplementare 2. Il sistema verifica se il guidatore supplementare è registrato nel sistema 3. Se il sistema verifica che il guidatore supplementare è un utente registrato <ol style="list-style-type: none"> 3.1 Il sistema invia un messaggio di conferma all'utente 4. Altrimenti 5. Il sistema invia un messaggio di ERRORE all'utente
<i>Principale</i>	
<i>Post-condizioni</i>	-Il sistema associa il guidatore supplementare al noleggio effettuato dal sistema
<i>Casi d'uso correlati</i>	<i>NoleggiaAuto</i>
<i>Sequenza di eventi alternativi</i>	-

*Caso d'uso:**Autenticazione*

<i>Attore primario</i>	Utente
<i>Attore secondario</i>	-
<i>Descrizione</i>	Un utente già registrato può accedere al sistema in modo da usufruire delle funzionalità che il sistema mette a disposizione
<i>Pre-condizioni</i>	
<i>Sequenza di eventi Principale</i>	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando l'utente richiede al sistema di autenticarsi <ol style="list-style-type: none"> 1.1. L'utente inserisce l'email. 2. Se l'utente che vuole autenticarsi non è registrato <ol style="list-style-type: none"> 2.1. Il sistema invia all'utente un messaggio di ERRORE. 3. Altrimenti <ol style="list-style-type: none"> 3.1. Il sistema invia all'utente un messaggio di conferma della procedura.
<i>Post-condizioni</i>	L'utente accede alla piattaforma come utente registrato.
<i>Casi d'uso correlati</i>	nessuno
<i>Sequenza di eventi alternativi</i>	-

*Caso d'uso:**Registrazione*

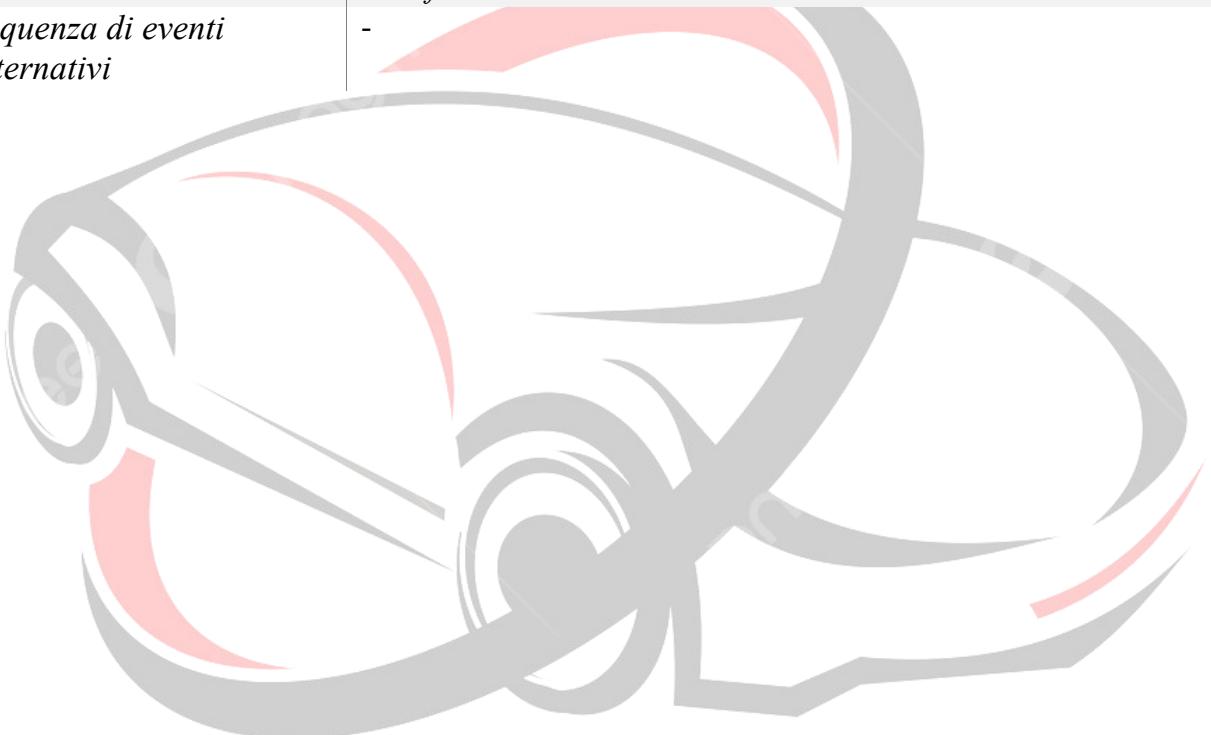
<i>Attore primario</i>	Utente
<i>Attore secondario</i>	-
<i>Descrizione</i>	Un utente senza credenziali può registrarsi sulla piattaforma con i propri dati
<i>Pre-condizioni</i>	-
<i>Sequenza di eventi Principale</i>	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando l'utente richiede al sistema di registrarsi <ol style="list-style-type: none"> 1.1. L'utente inserisce nome, cognome, data di nascita, email, anno di conseguimento, scadenza e numero della patente di guida 2. Se esiste un utente registrato con la stessa email <ol style="list-style-type: none"> 2.1. Il sistema invia all'utente un messaggio di ERRORE 3. Altrimenti <ol style="list-style-type: none"> 3.1. Il sistema invia all'utente un messaggio di conferma della procedura
<i>Post-condizioni</i>	Il sistema memorizza un nuovo utente registrato.
<i>Casi d'uso correlati</i>	nessuno
<i>Sequenza di eventi alternativi</i>	-

<i>Caso d'uso:</i>	<i>AggiungiDatiAuto</i>
<i>Attore primario</i>	Operatore
<i>Attore secondario</i>	-
<i>Descrizione</i>	Un operatore vuole aggiungere i dati di una nuova auto nel sistema
<i>Pre-condizioni</i>	-
<i>Sequenza di eventi</i>	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando l'operatore richiede di aggiungere i dati di un'auto <ol style="list-style-type: none"> 1.1 L'operatore aggiunge la targa, il prezzo fissato per giorno-noleggio, il numero di passeggeri per cui l'auto è omologata, l'alimentazione (benzina/gasolio) la potenza del motore e il segmento a cui l'auto appartiene 2. Se il sistema verifica che i dati inseriti siano invalidi <ol style="list-style-type: none"> 2.1 Il sistema mostra un messaggio di ERRORE all'operatore. 3. Altrimenti <ol style="list-style-type: none"> 3.1 Il sistema invia un messaggio di conferma all'operatore dell'avvenuta aggiunta
<i>Post-condizioni</i>	Il sistema memorizza una nuova auto
<i>Casi d'uso correlati</i>	nessuno
<i>Sequenza di eventi alternativi</i>	-
<i>Caso d'uso:</i>	<i>ModificaDatiAuto</i>
<i>Attore primario</i>	Operatore
<i>Attore secondario</i>	-
<i>Descrizione</i>	Un operatore vuole modificare dati di un'auto presente nel sistema
<i>Pre-condizioni</i>	-
<i>Sequenza di eventi</i>	<ol style="list-style-type: none"> 1. Il caso d'uso inizia quando l'operatore richiede di modificare i dati di un'auto presente nel sistema 2. Include(VisualizzaParcoAuto) 3. L'operatore specifica l'auto e i relativi dati da modificare. 4. Il sistema ricerca l'auto richiesta per la modifica. 5. Se il sistema trova l'auto richiesta per la modifica. <ol style="list-style-type: none"> a. Il sistema invia un messaggio di ERRORE all'operatore se i dati sono invalidi. 6. Altrimenti <ol style="list-style-type: none"> a. Il sistema invia un messaggio di ERRORE all'operatore se l'auto non è stata trovata. 7. Al termine dell'operazione di modifica il sistema invia un messaggio di conferma all'operatore
<i>Post-condizioni</i>	Il sistema memorizza le modifiche apportate dall'operatore
<i>Casi d'uso correlati</i>	nessuno

Sequenza di eventi alternativi

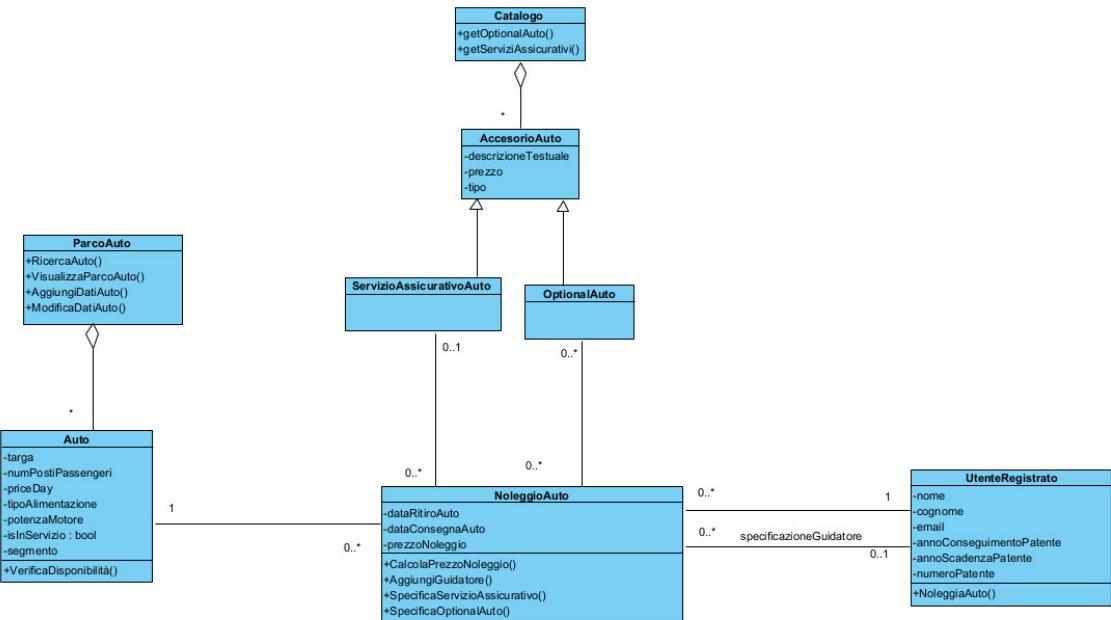
-

<i>Caso d'uso:</i>	<i>Caso d'uso di inclusione: VisualizzaParcoAuto</i>
<i>Attore primario</i>	Operatore
<i>Attore secondario</i>	-
<i>Descrizione</i>	Un operatore può visualizzare il parco auto
<i>Pre-condizioni</i>	-
<i>Sequenza di eventi Principale</i>	<ol style="list-style-type: none"> 1. Il sistema accede al parco auto 2. Il sistema mostra all'operatore i dati delle varie auto del parco auto
<i>Post-condizioni</i>	
<i>Casi d'uso correlati</i>	<i>ModificaDatiAuto</i>
<i>Sequenza di eventi alternativi</i>	-



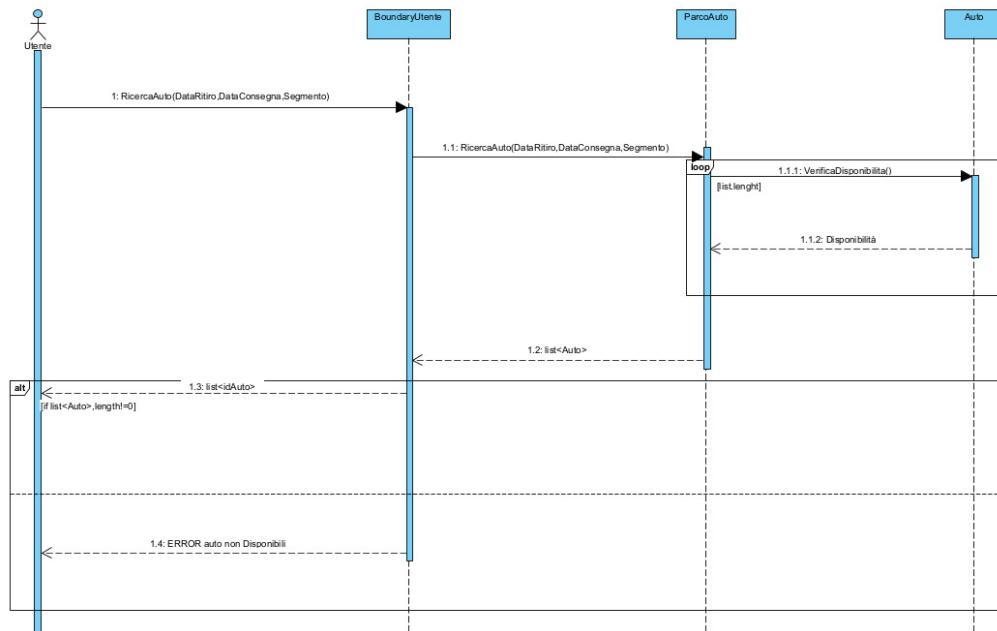
2.8 Diagramma delle classi

Diagramma delle classi di analisi.

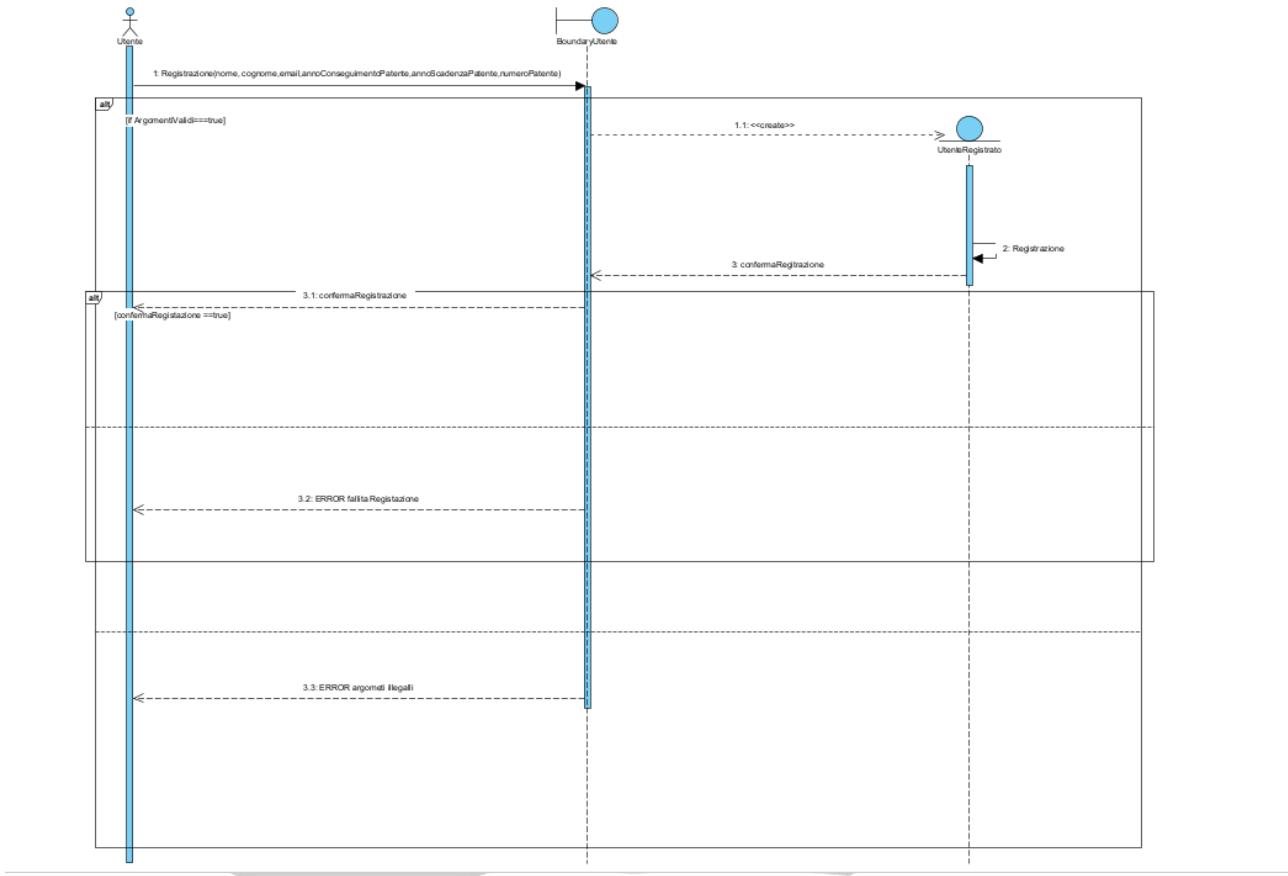


2.9 Diagrammi di sequenza

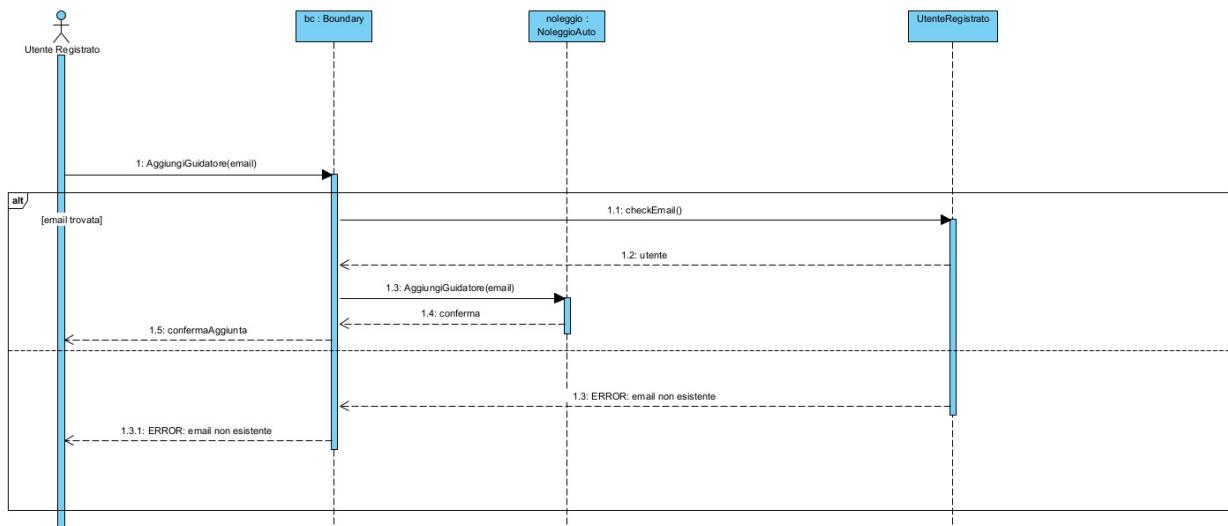
- Diagramma di analisi per il caso d'uso "RicercaAuto"



- Diagramma di sequenza di analisi per il caso d'uso “Registrazione”



- Diagramma di sequenza di analisi di “AggiungiGuidatore”



- Diagramma di sequenza di analisi di “NoleggioAuto”

Nota : Data la grandezza del diagramma non è stato riportato...consultare il file vpp in cui si trova la realizzazione

2.10 Verifica della completezza dei requisiti

Legenda: UCD = Use Case Diagram, CD = Class Diagram, SD = Sequence Diagram

- **RF₀₁** è modellato nell'UCD con l'attore "UtenteRegistrato" e con il caso d'uso UC8
- **RF₀₂** è modellato nell'UCD con l'attore "UtenteRegistrato" e con i casi d'uso UC1,UC9
- **RF₀₃** è modellato nell'UCD con l'attore "Operatore" e con il caso d'uso UC7
- **RF₀₄** è modellato nell'UCD con l'attore "Operatore" e con il caso d'uso UC6
- **RF₀₅** è modellato nell'UCD con l'attore "Utente" e con il caso d'uso UC4
- **RF₀₆** è modellato nell'UCD con l'attore "Utente" e con il caso d'uso UC3
- **RF₀₇** è modellato nell'UCD con l'attore "UtenteRegistrato" e con i casi d'uso UC10,UC11,UC12
- **RF₀₈** è modellato nell'UCD con l'attore "UtenteRegistrato" con il caso d'uso UC13
- **RF₀₉** è modellato nell'UCD con l'attore "Utente" con il caso d'uso UC2
- **RD₀₁** è modellato nel CD con gli attributi della classe "Auto"
- **RD₀₂** è modellato nel CD con l'attributo "segmento" nella classe "Auto"
- **RD₀₃** è modellato nel CD con l'attributo "inServizio" della classe "Auto"
- **RD₀₄** è modellato nel CD con gli attributi della classe "UtenteRegistrato"
- **RD₀₅** è modellato nel CD con la classe "AccessorioAuto" e la relazione di generalizzazione -specializzazione con le classi "ServizioAssicurativo" e "OptionalAuto"
- **RD₀₆** sono modellati nel CD con gli attributi della classe "Accessoriato"
- **RD₀₇** è modellato nel SD mediante la funzione "calcolaPrezzoNoleggio"
- **RD₀₈** è modellato nel SD con i parametri della funzione "ricercaAuto"
- **RD₀₉** è modellato nel SD mediante la funzione "VerificaDisponibilità"
- **RD₁₀** è modellato nel CD mediante una associazione con cardinalità [0..1,0..*] tra la classe "ServizioAssicurativo" e la classe "NoleggioAuto"
- **RD₁₁** è modellato nel CD mediante una associazione con cardinalità [0..*,0..*] tra la classe "OptionalAuto" e la classe "NoleggioAuto"

3. Stima dei costi

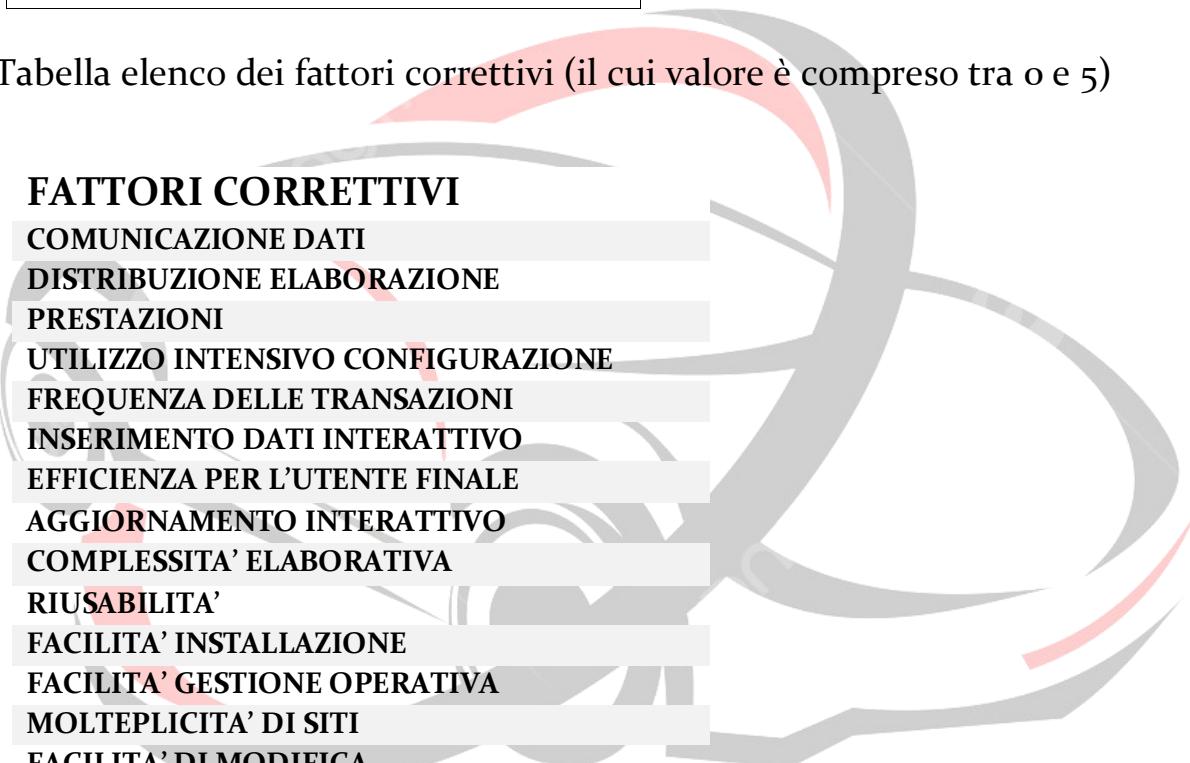
Riportare la stima dei costi secondo il metodo dei Punti Funzione.

- Tabella di riferimento per le complessità di dati e transazioni

	SEMPLICE	MEDIO	COMPLESSO
NILF	7	10	15
NEIF	5	7	10
NEI	3	4	6
NEO	4	5	7
NEQ	4	4	6

- Tabella elenco dei fattori correttivi (il cui valore è compreso tra 0 e 5)

FATTORI CORRETTIVI

- 
- COMUNICAZIONE DATI**
DISTRIBUZIONE ELABORAZIONE
PRESTAZIONI
UTILIZZO INTENSIVO CONFIGURAZIONE
FREQUENZA DELLE TRANSAZIONI
INSERIMENTO DATI INTERATTIVO
EFFICIENZA PER L'UTENTE FINALE
AGGIORNAMENTO INTERATTIVO
COMPLESSITA' ELABORATIVA
RIUSABILITA'
FACILITA' INSTALLAZIONE
FACILITA' GESTIONE OPERATIVA
MOLTEPLICITA' DI SITI
FACILITA' DI MODIFICA

NOLEGGIO AUTO

	VALORE	SEMPLICE	MEDIO	COMPLESSO	TOT
NILF	1		10		10
NEIF	0				
NEI	7	3			21
NEO	2		5		10
NEQ	1	4			4

NILF: Le auto vengono create dal sistema, le identifichiamo come ILF. [1 medio]

UFP= 45

LLOC/FP = 2.385

NEI: Data ritiro, Data consegna, segmento, email guidatore supplementare, auto ,optional auto, servizio assicurativo [7 semplici]

NEO: Visualizzazione delle auto disponibili, ottenuta mediante elaborazione, calcolo del prezzo del noleggio ottenuto tramite elaborazione [2 medio]

NEQ: Visualizzazione degli accessori, ottenuto tramite semplice richiesta [1 semplice]

FATTORI CORRETTIVI

COMUNICAZIONE DATI	3
DISTRIBUZIONE ELABORAZIONE	3
PRESTAZIONI	2
UTILIZZO INTENSIVO CONFIGURAZIONE	3
FREQUENZA DELLE TRANSAZIONI	2
INSERIMENTO DATI INTERATTIVO	3
EFFICIENZA PER L'UTENTE FINALE	4
AGGIORNAMENTO INTERATTIVO	4
COMPLESSITÀ ELABORATIVA	3
RIUSABILITÀ	1
FACILITÀ INSTALLAZIONE	2
FACILITÀ GESTIONE OPERATIVA	2
MOLTECIPLITÀ DI SITI	0
FACILITÀ DI MODIFICA	2
	34

FP= 44.55

JAVA = 2.361

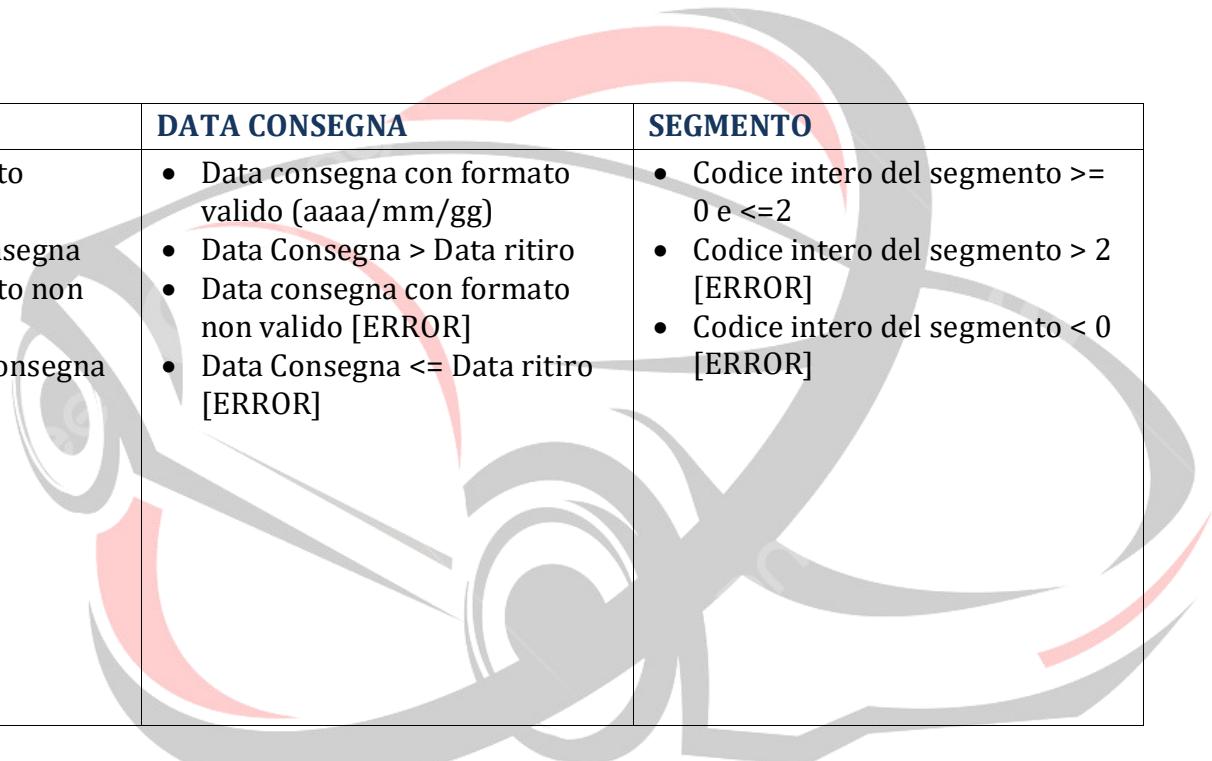
- LLOC/FP con Java è 53
- $FP=UFP*(0.65+0.01*\sum_{i=1}^{14} F_i)$

Per la scelta dei fattori correttivi si sono tenute in considerazione le seguenti caratteristiche:

1. **COMUNICAZIONE DATI:** Media. È necessario che il sistema permetta l'aggiunta di un nuovo noleggio alla lista di quelli effettuati.
2. **DISTRIBUZIONE ELABORAZIONE:** Media. Il sistema distribuisce l'elaborazione su diversi componenti per velocizzare l'elaborazione.
3. **PRESTAZIONI:** Non troppo basso. Il sistema deve garantire una risposta all'utente in tempi ragionevoli ma non necessariamente troppo brevi.
4. **UTILIZZO INTENSIVO CONFIGURAZIONE:** Medio. Per poter consentire all'utente di noleggiare un'auto e di arricchire tale noleggio con accessori vari è richiesto un certo grado di configurazione
5. **FREQUENZA DELLE TRANSAZIONI:** Non troppo bassa. L'aggiunta di un nuovo noleggio non è fatta frequentemente.
6. **INSERIMENTO DATI INTERATTIVO:** Medio. L'utente inserisce in maniera regolare i diversi dati in diverse iterazioni con il sistema durante lo svolgimento della funzionalità.
7. **EFFICIENZA PER L'UTENTE FINALE:** Alta. Il sistema deve essere progettato per semplificare al massimo e assistere le operazioni del cliente.
8. **AGGIORNAMENTO INTERATTIVO:** Alto. Il sistema deve aggiornare le informazioni visualizzate dall'utente in tempo reale per potergli fornire sempre dati aggiornati.
9. **COMPLESSITA' ELABORATIVA:** Media. L'aggiunta di un nuovo noleggio alla lista richiede l'utilizzo di diverse iterazioni con l'utente e l'utilizzo di diverse tipologie di dati
10. **RIUSABILITA':** Bassa. La funzionalità del sistema è strutturata per il noleggio di un'auto da un parco auto con l'aggiunta di vari accessori, risultando essere molto specifica per una data funzionalità.
11. **FACILITA' INSTALLAZIONE:** Non troppo bassa. L'installazione della funzionalità non troppe modifiche nel sistema esistente.
12. **FACILITA' GESTIONE OPERATIVA:** Non troppo alta. L'operazione maneggia contemporaneamente diversi dati eterogenei.
13. **MOLTEPLICITA' DI SITI:** Nulla. La funzionalità è sempre centralizzata nel sistema.
14. **FACILITA' DI MODIFICA:** Non troppo bassa. Non è facile modificare la funzionalità in modo da renderla adattabile ad altre circostanze

4. Piano di test funzionale

4.1 PIANO DI TEST UTILIZZANDO IL METODO DEL *CATEGORY-PARTITION TESTING* PER LA FUNZIONALITÀ “RicercaAuto”.



DATA RITIRO	DATA CONSEGNA	SEGMENTO
<ul style="list-style-type: none"> • Data ritiro con formato valido(aaaa/mm/gg) • Data ritiro < Data consegna • Data ritiro con formato non valido [ERROR] • Data Ritiro >= Data consegna [ERROR] 	<ul style="list-style-type: none"> • Data consegna con formato valido (aaaa/mm/gg) • Data Consegnna > Data ritiro • Data consegna con formato non valido [ERROR] • Data Consegnna <= Data ritiro [ERROR] 	<ul style="list-style-type: none"> • Codice intero del segmento >= 0 e <=2 • Codice intero del segmento > 2 [ERROR] • Codice intero del segmento < 0 [ERROR]

Il numero di test da effettuarsi senza particolari vincoli è: $4 * 4 * 3 = 48$

Introduciamo i vincoli [ERROR].

Il numero di test da eseguire per testare singolarmente i vincoli è 6(2 per Data Ritiro, 2 per Data consegna, 2 per Segmento)

Il numero di test risultante è: $(1*1*1) + 6 = 7$

TEST SUITE

Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Tutti input validi	Data ritiro valida Data consegna valida Segmento valido		{ Data ritiro: "2023-07-13". Data consegna: "2023-07-15". Segmento: 1. }	Il sistema mostra le auto disponibilità nell'intervallo di tempo specificato e di quel segmento	
2	Data ritiro formato non valida	Data ritiro con formato non valido [ERROR] Data consegna Segmento		{ Data ritiro: "2023/luglio/13". Data consegna: "2023-07-15". Segmento: 1. }	Data ritiro non corretta!	
3	Data consegna formato non valido	Data ritiro Data consegna con formato non valido [ERROR] Segmento		{ Data ritiro: "2023-07-13". Data consegna: "2023/luglio/15". Segmento: 1. }	Data consegna non corretta!	

4	Data consegna < data ritiro	Data ritiro \geq Data consegna [ERROR] Data consegna \leq Data ritiro [ERROR] Segmento		<pre>{ Data ritiro: "2023-07-13". Data consegna: "2023-07-11". Segmento: 1. }</pre>	Data consegna deve essere successiva a quella di ritiro!	
5	Data ritiro > data consegna	Data ritiro \geq Data consegna [ERROR] Data consegna \leq Data ritiro [ERROR] Segmento		<pre>{ Data ritiro: "2023-08-13". Data consegna: "2023-07-13". Segmento: 1. }</pre>	Data ritiro non deve essere successiva alla data consegna!	
6	Codice intero del segmento > 2	Data ritiro Data consegna Codice intero del segmento > 2 [ERROR].		<pre>{ Data ritiro: "2023-07-13". Data consegna: "2023-07-15". Segmento: 4. }</pre>	Segmento non valido!	
7	Codice intero del segmento < 0	Data ritiro Data consegna Codice intero del segmento < 0 [ERROR].		<pre>{ Data ritiro: "2023-07-13". Data consegna: "2023-07-15". Segmento: -1. }</pre>	Segmento non valido!	

4.2 PIANO DI TEST UTILIZZANDO IL METODO DEL **CATEGORY-PARTITION TESTING** PER LA FUNZIONALITÀ “AggiungiDatiAuto”

TARGA	NUMERO POSTI PASSEGGERI	PREZZO PER GIORNO	TIPO ALIMENTAZIONE	POTENZA MOTORE	SEGMENTO
<ul style="list-style-type: none"> Stringa di caratteri di lunghezza ≤ 10 Stringa di caratteri di lunghezza > 10 [ERROR] 	<ul style="list-style-type: none"> Numero di posti passeggeri > 0 Numero di posti passeggeri ≤ 0 [ERROR] 	<ul style="list-style-type: none"> Prezzo Auto > 0.0 Prezzo Auto ≤ 0.0 [ERROR] 	<ul style="list-style-type: none"> Tipo di alimentazione = “benzina” Tipo di alimentazione = “gasolio” Tipo di alimentazione diverso da “benzina” e “gasolio” [ERROR] 	<ul style="list-style-type: none"> Potenza Motore > 0 Potenza Motore ≤ 0 [ERROR] 	<ul style="list-style-type: none"> Codice intero del segmento ≥ 0 e ≤ 2 Codice intero del segmento > 2 [ERROR] Codice intero del segmento < 0 [ERROR]

Il numero di test da effettuarsi senza particolari vincoli è $2 \cdot 2 \cdot 2 \cdot 3 \cdot 2 \cdot 3 = 144$

Introduciamo i vincoli [ERROR].

Il numero di test da eseguire per testare singolarmente i vincoli è 7(1 per Targa, 1 per Numero posti passeggeri, 1 per Prezzo per giorno , 1 per Tipo alimentazione , 1 per Potenza motore, 2 per segmento)

Il numero di test risultante è: $(1*1*1*1*1*1) + 7 = 8$

TEST SUITE

Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Tutti input validi	Targa Valida Numero posti passeggeri valido Prezzo per giorno valido Tipo alimentazione valido Potenza motore valido Segmento valido		{ Targa : "GA 129KM" Numero posti passeggeri: 4 Prezzo per giorno : 30.0 Tipo alimentazione : "benzina" Potenza motore : 100 Segmento: 1. }	Aggiunta confermata	Viene aggiunta una nuova auto nel sistema

2	Targa non valida	Targa come stringa di caratteri > 10 [ERROR] Numero posti passeggeri valido Prezzo per giorno valido Tipo alimentazione valido Potenza motore valido Segmento valido		{ Targa : "AA 000KKKMMMM" Numero posti passeggeri: 4 Prezzo per giorno : 30.0 Tipo alimentazione : "benzina" Potenza motore : 75 Segmento: 1. }	Targa non valida!	
3	Numero posti passeggeri < 0	Targa valida Numero posti passeggeri <=0 [ERROR] Prezzo per giorno valido Tipo alimentazione valido Potenza motore valido Segmento valido		{ Targa : "AA 000AA" Numero posti passeggeri: -1 Prezzo per giorno : 34.99 Tipo alimentazione : "gasolio" Potenza motore : 69 Segmento: 0. }	Numero posti passeggeri non valido	
4	Prezzo per giorno = 0.0	Targa valida Numero posti passeggeri valido Prezzo per giorno <=0 [ERROR]		{ Targa : "EF 123AA" Numero posti passeggeri: 3 Prezzo per giorno : 0 Tipo alimentazione : "benzina" }	Prezzo non valido!	

		Tipo alimentazione valido Potenza motore valido Segmento valido		Potenza motore : 135 Segmento: 2 }		
5	Tipo alimentazione differente da "gasolio" e "benzina"	Targa valida Numero posti passeggeri valido Prezzo per giorno valido Tipo alimentazione diverso da "benzina" e "gasolio" [ERROR] Potenza motore valido Segmento valido		{ Targa : "AB 200KA" Numero posti passeggeri: 4 Prezzo per giorno : 30.0 Tipo alimentazione : "elettrico" Potenza motore : 100 Segmento: 1. }	Tipo di alimentazione invalido!	
6	Potenza motore < 0 (in cavalli)	Targa valida Numero posti passeggeri valido Prezzo per giorno valido Tipo alimentazione valido Potenza motore <=0 [ERROR] Segmento valido		{ Targa : "GC 150KM" Numero posti passeggeri: 4 Prezzo per giorno : 30.0 Tipo alimentazione : "benzina" Potenza motore : -100 Segmento: 1. }	Potenza motore invalida!	

7	Codice intero del segmento maggiore di 2	Targa valida Numero posti passeggeri valido Prezzo per giorno valido Tipo alimentazione valido Potenza motore valida Codice intero del segmento > 2 [ERROR]		{ Targa : "ER 764XX" Numero posti passeggeri: 3 Prezzo per giorno : 56.99 Tipo alimentazione : "gasolio" Potenza motore : 120 Segmento: 6 }	Segmento invalido!	
8	Codice intero del segmento minore di 0	Targa valida Numero posti passeggeri valido Prezzo per giorno valido Tipo alimentazione valido Potenza motore valida Codice intero del segmento <0 [ERROR]		{ Targa : "ER 764XX" Numero posti passeggeri: 3 Prezzo per giorno : 56.99 Tipo alimentazione : "gasolio" Potenza motore : 120 Segmento: -9 }	Segmento invalido!	

4.3 PIANO DI TEST UTILIZZANDO IL METODO DEL CATEGORY-PARTITION TESTING PER LA FUNZIONALITÀ “Registrazione”

NOME	COGNOME	DATA DI NASCITA	EMAIL	ANNO DI CONSEGUIMENTO	ANNO DI SCADENZA	NUMERO DELLA PATENTE
<ul style="list-style-type: none"> • Stringa di caratteri > 1 • Stringa di caratteri <= 1 [ERROR] 	<ul style="list-style-type: none"> • Stringa di caratteri > 1 • Stringa di caratteri <= 1 [ERROR] 	<ul style="list-style-type: none"> • Data di nascita con formato valido(aaaa/mm/gg) • Data di nascita con formato non valido [ERROR] 	<ul style="list-style-type: none"> • Email con il carattere @ • Email senza il carattere @ [ERROR] • Email già presente nel database [ERROR] 	<ul style="list-style-type: none"> • Anno di conseguimento > 1900 • Anno di conseguimento >= Anno di conseguimento + 10 • Anno di conseguimento < 1900 [ERROR] • Anno di conseguimento < Anno di conseguimento +10 [ERROR] • Anno di conseguimento > Anno di scadenza -10 [ERROR] 	<ul style="list-style-type: none"> • Anno di scadenza > 1900 • Anno di scadenza >= Anno di conseguimento + 10 • Anno di scadenza < 1900 [ERROR] • Anno di scadenza < Anno di conseguimento +10 [ERROR] 	<ul style="list-style-type: none"> • Stringa alfanumerica di caratteri = 10 • Stringa alfanumerica di caratteri diversa da 10 [ERROR]

Il numero di test da effettuarsi senza particolari vincoli è $2*2*2*3*4*4*3=1152$
 Introduciamo i vincoli [ERROR].

Il numero di test da eseguire per testare singolarmente i vincoli è 10 (1 per nome, 1 per cognome, 1 per data di nascita, 2 per email, 2 per anno di conseguimento, 2 per anno di scadenza, 1 per numero della patente)

Il numero di test risultante è: $(1*1*1*1*1*1*1) + 1 = 11$

TEST SUITE

Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese
1	Tutti input validi	Nome valido Cognome valido Data di nascita valida Email valida Anno di conseguimento valido Anno di scadenza valida Numero di patente valido	L'email inserita non è già presente nel sistema	{ Nome: Mario Cognome: Rossi Data di nascita: 1985/01/01 Email: MarioRossi85@gmail.com Anno di conseguimento: 2004 Anno di scadenza: 2024 Numero di patente: CTR67542UT }	Il Sistema registra l'utente come nuovo utente registrato del sistema.	L'utente registrato viene salvato nel sistema
2	Nome non valido	Nome stringa di caratteri <= 1 [ERROR] Cognome Data di nascita Email Anno di conseguimento Anno di scadenza Numero di patente		{ Nome: M Cognome: Rossi Data di nascita: 1985/01/01 Email: MarioRossi85@gmail.com Anno di conseguimento: 2004 Anno di scadenza: 2024 Numero di patente: CTR67542UT }	Nome inserito non valido!	

3	Cognome non valido	<p>Nome Cognome stringa di caratteri <= 1 [ERROR] Data di nascita Email Anno di conseguimento Anno di scadenza Numero di patente</p>		{ Nome: Mario Cognome: R Data di nascita: 1985/01/01 Email: MarioRossi85@gmail.com Anno di conseguimento: 2004 Anno di scadenza: 2024 Numero di patente: CTR67542UT }	Cognome inserito non valido!	
4	Data di nascita con formato non valido	<p>Nome Cognome Data di nascita con formato non valido [ERROR] Email Anno di conseguimento Anno di scadenza Numero di patente</p>		{ Nome: Mario Cognome: Rossi Data di nascita: 1985-genn-primo Email: MarioRossi85@gmail.com Anno di conseguimento: 2004 Anno di scadenza: 2024 Numero di patente: CTR67542UT }	La data inserita non è nel formato valido!	
5	Email senza il carattere @	<p>Nome Cognome Data di nascita Email senza il carattere @ [ERROR] Anno di conseguimento Anno di scadenza</p>		{ Nome: Mario Cognome: Rossi Data di nascita: 1985-01-01 Email: MarioRossi85gmail.com Anno di conseguimento: 2004 Anno di scadenza: 2024 Numero di patente: CTR67542UT }	Email non valida!	

		Numero di patente				
6	Email già presente nel database	Nome Cognome Data di nascita Email già presente nel database [ERROR] Anno di conseguimento Anno di scadenza Numero di patente	L'email inserita è già presente nel sistema	{ Nome: Mario Cognome: Rossi Data di nascita: 1985-01-01 Email: MarioRossi85@gmail.com Anno di conseguimento: 2004 Anno di scadenza: 2024 Numero di patente: CTR67542UT }	C'è già una registrazione effettuata con quella email!	
7	Anno di conseguimento < 1900	Nome Cognome Data di nascita Email Anno di conseguimento < 1900 [ERROR] Anno di scadenza Numero di patente		{ Nome: Mario Cognome: Rossi Data di nascita: 1985/01/01 Email: MarioRossi85@gmail.com Anno di conseguimento: 1899 Anno di scadenza: 2024 Numero di patente: CTR67542UT }	Anno di conseguimento della patente non valido!	
8	Anno di conseguimento > Anno di scadenza -10	Nome Cognome Data di nascita Email Anno di conseguimento > Anno di scadenza - 10 [ERROR] Anno di scadenza		{ Nome: Mario Cognome: Rossi Data di nascita: 1985/01/01 Email: MarioRossi85@gmail.com Anno di conseguimento: 2004 Anno di scadenza: 2008 Numero di patente: CTR67542UT }	Anno di conseguimento e anno di scadenza non validi!	

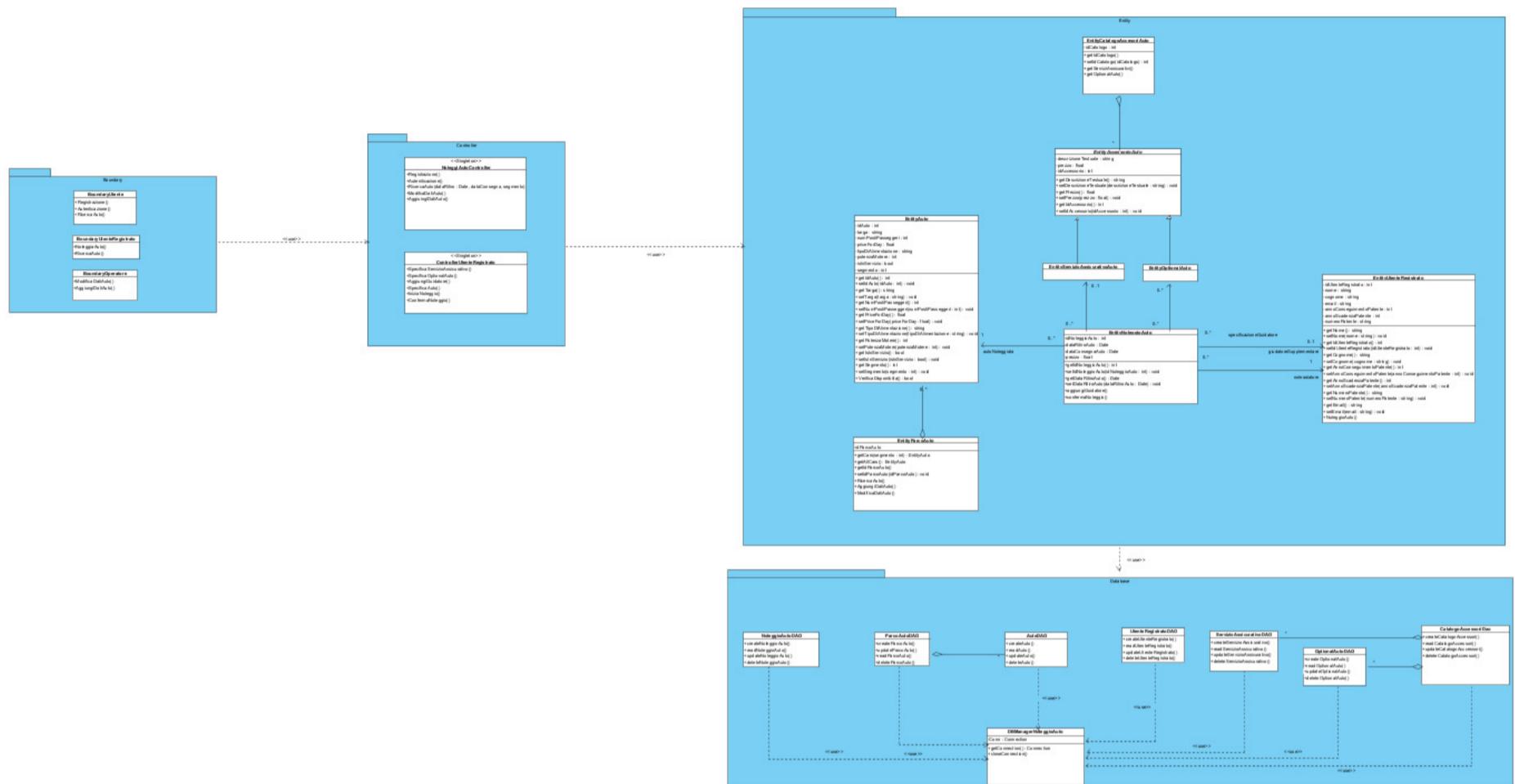
		Numero di patente				
9	Anno di scadenza < 1900	Nome Cognome Data di nascita Email Anno di conseguimento Anno di scadenza < 1900 [ERROR] Numero di patente		{ Nome: Mario Cognome: Rossi Data di nascita: 1985/01/01 Email: MarioRossi85@gmail.com Anno di conseguimento: 2004 Anno di scadenza: 1899 Numero di patente: CTR67542UT }	Anno di scadenza della patente non valido!	
10	Anno di scadenza < Anno di conseguimento + 10	Nome Cognome Data di nascita Email Anno di conseguimento Anno di scadenza < Anno di conseguimento + 10 [ERROR] Numero di patente		{ Nome: Mario Cognome: Rossi Data di nascita: 1985/01/01 Email: MarioRossi85@gmail.com Anno di conseguimento: 2004 Anno di scadenza: 2010 Numero di patente: CTR67542UT }	Anno di conseguimento della patente non valido!	
11	Numero di patente è una stringa alfanumerica di caratteri diversa da 10	Nome Cognome Data di nascita Email Anno di conseguimento Anno di scadenza		{ Nome: Mario Cognome: Rossi Data di nascita: 1985/01/01 Email: MarioRossi85@gmail.com Anno di conseguimento: 2004 Anno di scadenza: 2024 Numero di patente: CT81 }	Numero di patente non valido!	

		Stringa alfanumerica di caratteri diversa da 10 [ERROR]		}		
--	--	--	--	---	--	--



5. Progettazione

5.1 Diagramma delle classi



5.2 Diagrammi di sequenza

Diagramma di progetto del caso d'uso “RicercaAuto”

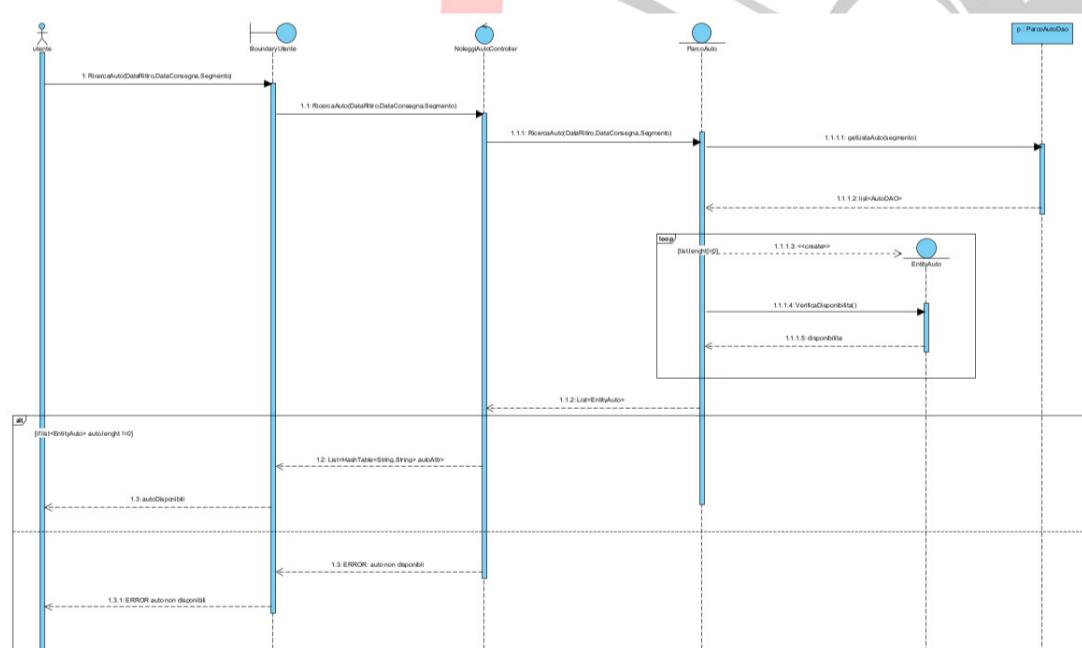


Diagramma di progetto del caso d'uso “AggiungiGuidatore”

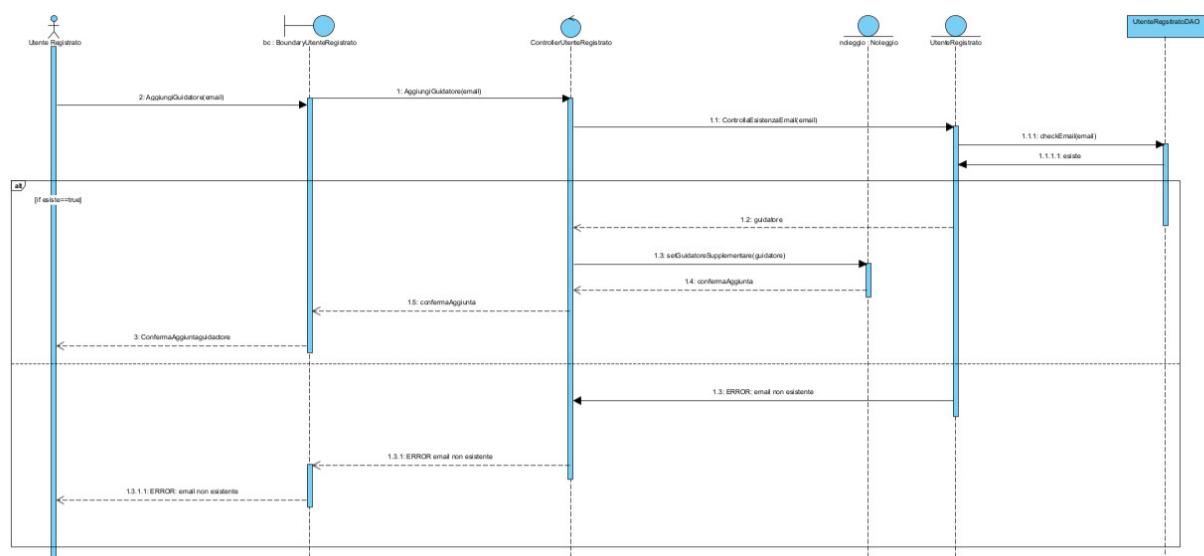


Diagramma di progetto del caso d'uso “NoleggioAuto”

Nota: Data la grandezza del diagramma non è stato riportato...consultare il file vpp in cui si trova la realizzazione



6. Implementazione

1. Package

- boundary, control, entity , exceptions modellati con il Pattern BCED

2. Classi

2.1. Classi del package **boundary**

- MainWindow è la finestra principale della GUI del sistema software
- BoundaryOperatore è la finestra con cui un operatore puo interagire con il sistema per operazioni di inserimento e modifica di dati di auto
- BoundaryUtente è la finestra con cui un utente puo interagire con il sistema per operazioni di Registrazione , Autenticazione e RicercaAuto
- BoundaryUtenteRegistrato è la finestra con cui un utente Registrato puo interagire con il sistema per operazioni di Ricerca Auto Noleggia Auto

2.2 Classi del package **control**

- NoleggiAutoController offre operazioni senza privilegi di controllo
- ControllerUtenteRegistrato offre operazioni con i privilegi di un utente registrato di controllo

2.3 Classi del package **Entity**

- EntityAuto è responsabile della gestione e conservazione dei dati dell' entità modellata Auto
- EntityAccessorioAuto è la classe astratta per la gestione dei dati dell' entità modellata AccessorioAuto
- EntityServizioAssicurativi è la classe figlia di EntityAccessorioAuto per la gestione dei dati di della entità modellata ServizioAssicurativo
- EntityOptionalAuto è la classe figlia di EntityAccessorioAuto per la gestione dei dati di della entità modellata OptionalAuto
- EntityUtenteRegistrato è la classe responsabile della gestione e conservazione dei dati della entità modellata UtenteRegistrato
- EntityCatalogo è la classe responsabile della gestione di EntityAccessorioAuto che offre metodi per la accesso di essi
- EntityParco è la classe responsabile della gestione di EntityAuto che offre metodi per l' accesso e conservazione di essi

2.4 Classi del package **database**

- AutoDAO consente di accedere alla tabella Auto e compiere le operazioni di create, read e altro.
- UtenteRegistratoDAO consente di accedere alla tabella UtentiRegistrati e compiere le operazioni di create, read di un record e altro
- ParcoAutoDAO consente di prelevare dal DB tutti i record della tabella Auto
- CatalogoDAO consente di prelevare dal DB tutti i record delle tabelle ServiziAssicurativi e OptionalAuto
- ServizioAssicurativoDAO consente di accedere alla tabella ServiziAssicurativi e compiere le operazioni di create, read di un record e altro.
- OptionalAutoDAO consente di accedere alla tabella OptionalAuto e compiere le operazioni di create, read di un record e altro.
- NoleggioAutoDAO consente di accedere alla tabella NoleggiAuto ,compiere le operazioni di create,read ,memorizzare le associazioni tra i noleggi e gli optional auto accedendo alla tabella associativa OptionalAuto_has_NoleggiAuto e altro
- DbConnectionManager consente di aprire e chiudere la connessione con il DB

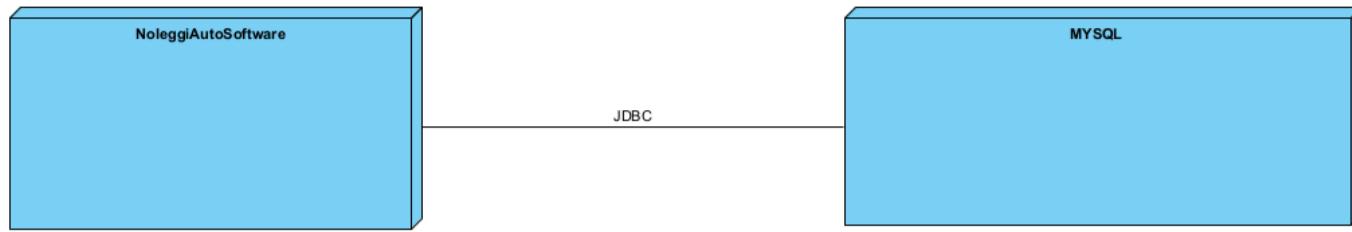
3. Eccezioni

- EmailUtenteNonValida
- FormatoDataNonValido
- NumeroPassegeriNonValido
- NumeroPatenteNonValido
- PotenzaMotoreNonValida
- PrezzoNonValido
- SegmentoNonValido
- TargaInvalida
- TipoAlimentazioneNonValido

4. Artefatti necessari per il programma

- 4.1 hamcaster-core-1.3.jar
- 4.2 junit-4.13.2.jar per il testing con il FrameWork Junit
- 4.3 mysql-connector-java-8.0.29.jar per il collegamento con il DBMS utilizzato MYSQL

5. Deployment Diagram



6. Confronto con la stima dei costi

LOC=2302

LLOC/FP=51.75

Il numero di linee di codice è 2302 mentre il numero di linee di codice stimato è 2361 da cui si conclude che la stima dei costi è stata adeguata



7. Testing

7.1 Test strutturale

Di seguito è riportato il testing white box su due metodi.

Control Flow Graph

- CFG del metodo “RicercaAuto”

```
public static ArrayList<Hashtable<String, String>> RicercaAuto(String dataRitiro , String dataConsegna , int segmento ) throws Exception {
    if(segmento < 0 || segmento>2)
        throw new Exception("Segmento non valido");
    if(!isValidDate(dataConsegna)|| !isValidDate(dataRitiro)) {
        throw new Exception("Formato data non valida yyyy-mm-dd");
    }

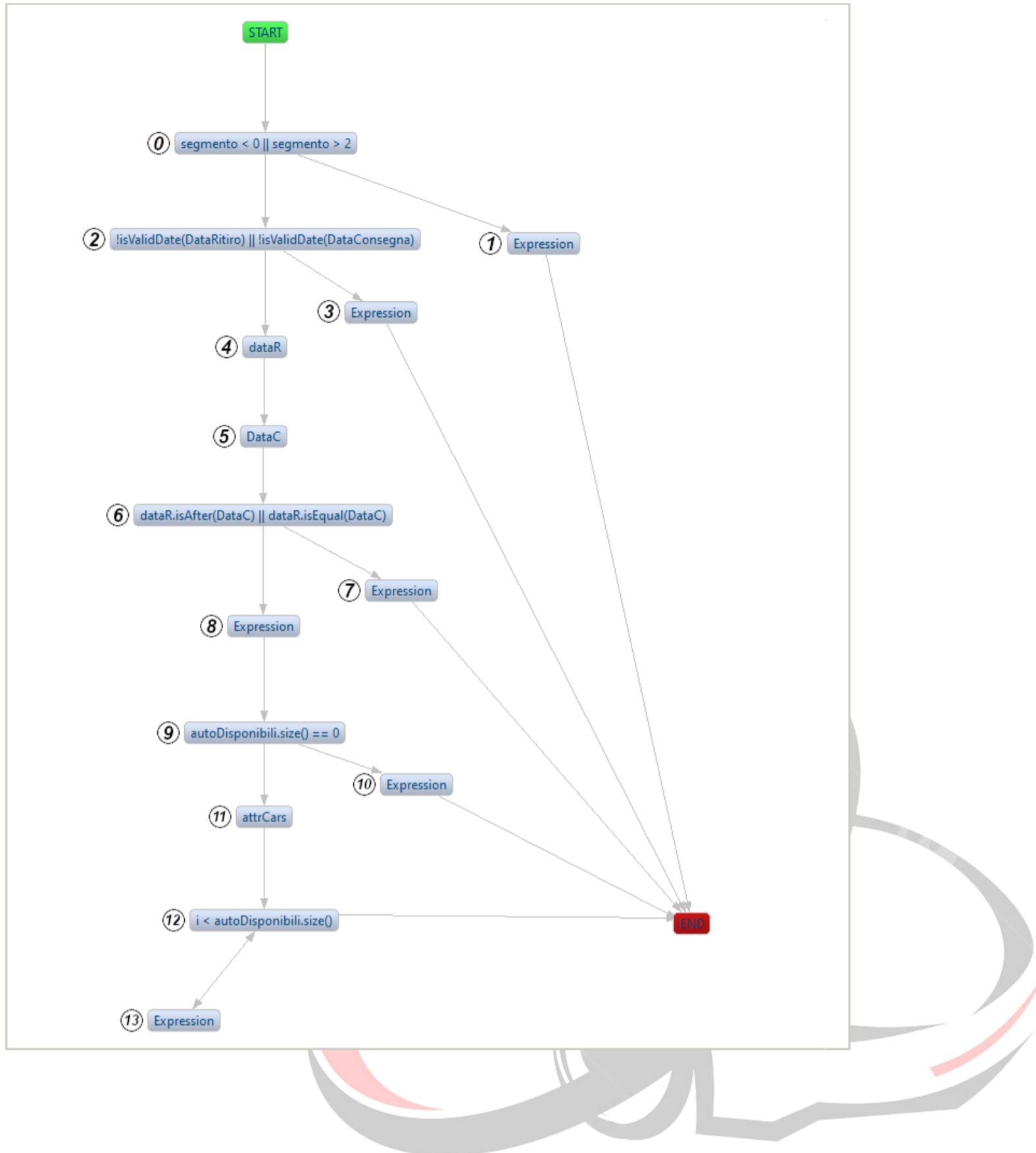
    LocalDate dataR = parsingDate(dataRitiro);
    LocalDate dataC= parsingDate(dataConsegna);

    if(dataR.isAfter(dataC)|| dataR isEqual(dataC))
        throw new Exception("Intervallo non valido");

    ArrayList<EntityAuto> autoDisponibili= EntityParcoAuto.RicercaAuto(dataR, dataC, segmento);

    if(autoDisponibili.size()==0)
        throw new Exception("Auto Non disponibili");
    ArrayList<Hashtable<String, String>> attrCars= new ArrayList<>();
    for (EntityAuto entityAuto : autoDisponibili) {
        attrCars.add(entityAuto.getAttribute());
    }
    return attrCars ;
}
```





NUMERO CICLOMATICO:

numero di regioni chiuse del grafo = 6
 numero di nodi predicati (0,2,6,9,12) +1 = 6
 $\# \text{ archi} - \# \text{ nodi} + 2 = (20 - 16) + 2 = 6$

CAMMINI:

- 1) I-0-1-F
- 2) I-0-2-3-F
- 3) I-0-2-4-5-6-7-F
- 4) I-0-2-4-5-6-8-9-10-F
- 5) I-0-2-4-5-6-8-9-11-12-F
- 6) I-0-2-4-5-6-8-9-11-12-13-12-F

- CFG del metodo "VerificaDisponibilitaAuto"

```

public boolean VerificaDisponibilita(LocalDate dataRitiro , LocalDate dataConsegna) {

    if(isInServizio==false)
        return false;

    ArrayList<EntityNoleggioAuto> noleggiAuto = caricaNoleggiAuto(new AutoDAO(idAuto));

    for (EntityNoleggioAuto noleggio : noleggiAuto) {

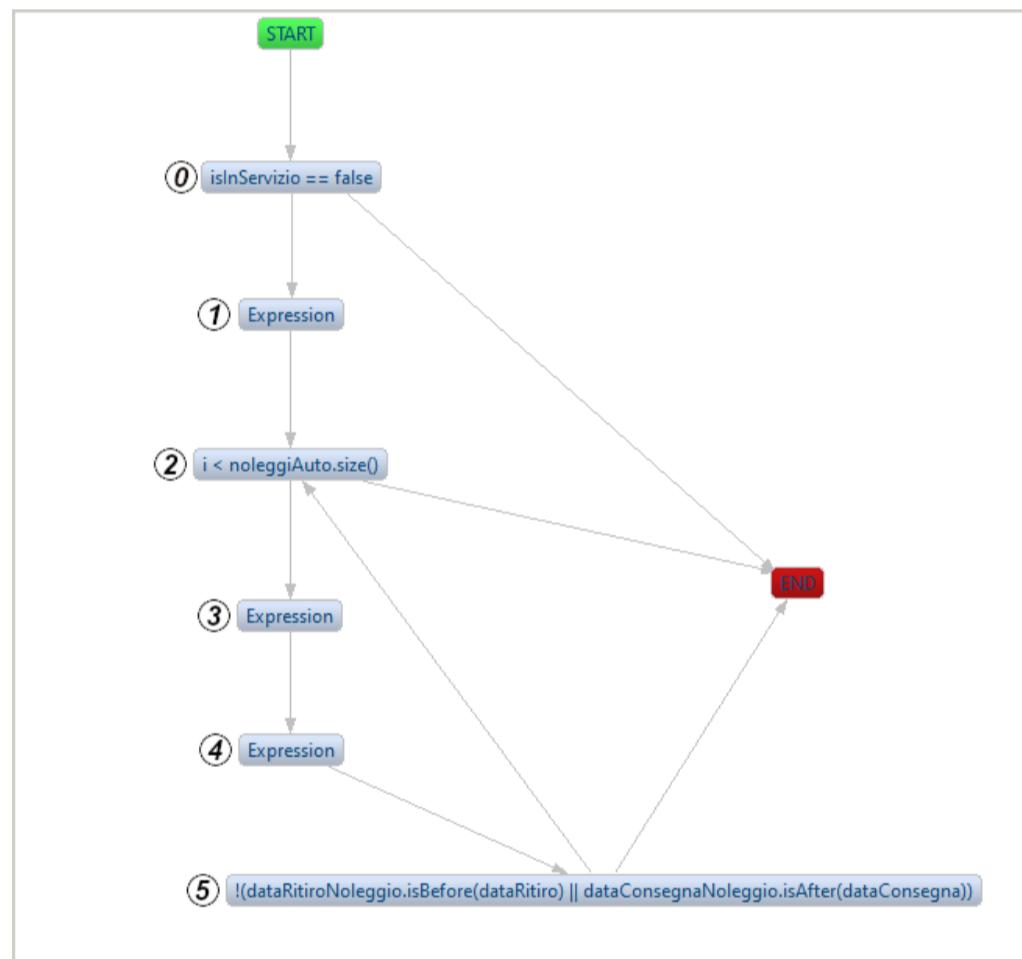
        LocalDate dataRitiroNoleggio = noleggio.getDataRitiroAuto();
        LocalDate dataConsegnaNoleggio = noleggio.getDataConsengnaAuto();

        if(!( dataConsegnaNoleggio.isBefore(dataRitiro) || dataRitiroNoleggio.isAfter(dataConsegna) )) {

            return false;
        }
    }

    return true ;
}

```



NUMERO CICLOMATICO:

numero di regioni chiuse del grafo = 4
 numero di nodi predicati (0, 2, 5) +1 = 4
 $\# \text{ archi} - \# \text{ nodi} + 2 = (10 - 8) + 2 = 4$

CAMMINI:

- 1) I-0-F
- 2) I-0-1-2-F
- 3) I-0-1-2-3-4-5-F
- 4) I-0-1-2-3-4-5-2-F

7.2 Test funzionale

Qui sotto sono descritti i risultati dell'esecuzione dei test funzionali precedentemente pianificati per la funzionalità "Ricerca Auto", adoperando lo schema di tabella seguente.

Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese	Output Ottenuuti	Post-condizioni Ottenuute	Esito (FAIL, PASS)
1	Tutti input validi	Targa Valida Numero posti passeggeri valido Prezzo per giorno valido Tipo alimentazione valido Potenza motore valido Segmento valido		Targa = "GA 129KM" Numero posti passeggeri: 4 Prezzo per giorno= 30.0 Tipo alimentazione="benzina" Potenza motore =100 Segmento= 1.	Aggiunta confermata	Viene aggiunta una nuova auto nel sistema	GOOD INSERT	Viene aggiunta una nuova auto nel sistema	PASS
2	Targa non valida	Targa come stringa di caratteri > 10 Numero posti passeggeri valido Prezzo per giorno valido Tipo alimentazione valido Potenza motore valido Segmento valido		Targa = "AA 000KKKKKK" Numero posti passeggeri = 4 Prezzo per giorno= 30.0 Tipo alimentazione= "benzina" Potenza motore= 75 Segmento: 1.	Targa non valida!		[Eccezione] Targa AA 000KKKKKK non valida		PASS
3	Numero posti passeggeri < 0	Targa valida Numero posti passeggeri <=0 Prezzo per giorno valido Tipo alimentazione valido Potenza motore valido Segmento valido		Targa = "AA 000AA" Numero posti passeggeri= -1 Prezzo per giorno= 34.99 Tipo alimentazione= "gasolio" Potenza motore= 69 Segmento= 0.	Numero posti passeggeri non valido		[Eccezione] Numero di passeggeri non valido. Inserire una quantità maggiore di 0.		PASS
4	Prezzo per giorno = 0.0	Targa valida Numero posti passeggeri valido Prezzo per giorno <=0 Tipo alimentazione valido Potenza motore valido Segmento valido		Targa= "EF 123AA" Numero posti passeggeri= 3 Prezzo per giorno= 0 Tipo alimentazione= "benzina" Potenza motore= 135 Segmento= 2	Prezzo non valido!		[Eccezione] Prezzo non valido. Inserire una quantità maggiore di 0.		PASS
5	Tipo alimentazione differente da "gasolio" e "benzina"	Targa valida Numero posti passeggeri valido Prezzo per giorno valido		Targa= "AB 200KA" Numero posti passeggeri: 4 Prezzo per giorno= 30.0 Tipo alimentazione= "elettrico" Potenza motore= 100	Tipo di alimentazione invalido!		[Eccezione] Tipo di alimentazione non valido. Inserire : (benzina/gasolio)		PASS

		Tipo alimentazione diverso da "benzina" e "gasolio" Potenza motore valido Segmento valido		Segmento= 1.					
6	Potenza motore < 0 (in cavalli)	Targa valida Numero posti passeggeri valido Prezzo per giorno valido Tipo alimentazione valido Potenza motore <=0 Segmento valido		Targa= "GC 150KM" Numero posti passeggeri: 4 Prezzo per giorno= 30.0 Tipo alimentazione= "benzina" Potenza motore= -100 Segmento: 1.	Potenza motore invalida!		[Eccezione] Potenza non valida. Inserire una quantità maggiore di 0.		PASS
7	Codice intero del segmento maggiore di 2	Targa valida Numero posti passeggeri valido Prezzo per giorno valido Tipo alimentazione valido Potenza motore valida Codice intero del segmento > 2		Targa= "ER 764XX" Numero posti passeggeri= 3 Prezzo per giorno= 56.99 Tipo alimentazione= "gasolio" Potenza motore= 120 Segmento= 6	Segmento invalido!		[Eccezione] Codice segmento non valido: usare 0-small, 1-medium e 2-large		PASS
8	Codice intero del segmento minore di 0	Targa valida Numero posti passeggeri valido Prezzo per giorno valido Tipo alimentazione valido Potenza motore valida Codice intero del segmento <0		{ Targa= "ER 764XX" Numero posti passeggeri: 3 Prezzo per giorno= 56.99 Tipo alimentazione= "gasolio" Potenza motore= 120 Segmento= -9	Segmento invalido!		[Eccezione] Codice segmento non valido: usare 0-small, 1-medium e 2-large		PASS

8. Manutenzione

1. Il Cliente richiede di fornire al sistema la possibilità, agli utenti registrati, di visualizzare il proprio storico prenotazioni.
 - 1.1. In questo caso, la funzionalità offrirà al all'utente registrato la possibilità di visualizzare le prenotazioni effettuate dall'utente.
2. Il software, su richiesta del cliente, può essere esteso per gestire diversi parchi auto discriminati in base alla regione geografica.
 - 2.1. In questo caso, il sistema introdurrà più parchi auto.
 - 2.2. In questo caso, se richiesto dal cliente, il sistema potrà gestire diversi cataloghi di accessori per i diversi parchi auto.
3. Per migliorare la qualità del sistema software, si possono introdurre maggiori controlli sui dati inseriti da applicazioni o utenti che interagiscono col sistema.
 - 3.1. Per esempio, il sistema può controllare i dati relativi al documento di patente inserito dall'utente, accertandosi che un utente che inserisce un documento di patente sia legalmente possessore di quel documento
4. Il sistema, nel tempo, dovrà aggiornare la propria GUI, per essere più attrattivo nell'utilizzo.

9. Appendice

- Modello ER usato

