



Laurea Triennale in informatica-Università di Salerno  
Corso di *Ingegneria del Software*- Prof. C.Gravino



# SDD System Design Document

UniPass

Versione	1.4
Data	23/01/2023
Destinatario	Prof. C. Gravino
Presentato da	Barretta Alessandro Di Lauro Antonio Malafronte Sabato Zampaglione Gerardo
Approvato da	



## Revision History

Data	Versione	Cambiamenti	Autori
21/11/2022	0.1	Prima stesura	[Gruppo]
22/11/2022	0.2	Aggiunta Design Goals	[Gruppo]
24/11/2022	0.3	Stesura capitolo 1	Antonio Di Lauro
26/11/2022	0.5	Aggiunta Decomposizione in sottosistemi	[Gruppo]
29/11/2022	0.5	Aggiunta Design Architetturale	[Gruppo]
1/12/2022	0.7	Stesura capitolo 2	Alessandro Barretta
2/12/2022	0.6	Modifica della Decomposizione in sottosistemi e del Design Architetturale	[Gruppo]
4/12/2022	0.7	Stesura capitolo 3	Gerardo Zampaglione
6/12/2022	0.8	Aggiunta Deployment Diagram	[Gruppo]
7/12/2022		Stesura capitolo 4 (Servizi dei sottosistemi)	Sabato Malafronte
8/12/2022	0.9	Aggiunta Matrice d'accesso	Alessandro Barretta, Antonio Di Lauro
9/12/2022	0.10	Definizione dei Trade-offs	Sabato Malafronte, Gerardo Zampaglione
11/12/2022	1.0	Revisione	[Gruppo]
16/12/2022	1.1	Revisione pre-consegna intermedia	[Gruppo]
03/01/2023	1.2	Aggiunta dei Design Goals dei nuovi RNF aggiunti nel RAD	[Gruppo]
04/01/2023	1.3	Aggiunta Class Diagram Ristrutturato	Gerardo Zampaglione
23/01/2023	1.4	Revisione	[Gruppo]



# Sommario

I.	<b>Revision History .....</b>	<b>2</b>
II.	<b>1. Introduzione .....</b>	<b>4</b>
	1.1 Obiettivi del sistema .....	4
	1.2 Design Goals & Trade-offs .....	4
	1.3 Definizioni, acronimi e abbreviazioni .....	6
	1.4 Riferimenti .....	6
	1.5 Panoramica .....	6
III.	<b>2. Architettura del Sistema corrente .....</b>	<b>7</b>
IV.	<b>3. Architettura del Sistema proposto .....</b>	<b>7</b>
	3.1 Panoramica .....	7
	3.2 Decomposizione in sottosistemi .....	8
	3.3 Mapping Hardware/Software .....	16
	3.3.1 Diagramma di deployment .....	16
	3.4 Gestione dati persistenti .....	16
	3.5 Controllo degli accessi e sicurezza .....	18
	3.6 Controllo flusso globale del sistema .....	18
	3.7 Condizione limite .....	19
	3.7.1 Avvio del Sistema .....	19
	3.7.2 Arresto del Sistema .....	19
V.	<b>4. Servizi dei sottosistemi .....</b>	<b>19</b>

# 1. Introduzione

## 1.1 Obiettivi del sistema

Il sistema che si vuole realizzare ha i seguenti obiettivi:

- Realizzare un applicativo destinato agli studenti dell'Università di Salerno per organizzare viaggi di ritorno a casa in auto
- Consentire un ritorno parziale delle spese agli studenti guidatori
- Prevedere funzionalità di creazione viaggi, ricerca viaggi, prenotazione viaggi e un meccanismo di valutazione degli utenti che hanno partecipato ai viaggi

## 1.2 Design Goals & Trade-offs

In questa sezione vengono descritti i Design Goals, ossia le caratteristiche di qualità principali che il Sistema dovrà rispettare. Ciascun design goal è descritto da:

- **Rank:** valore che specifica una priorità compresa tra 1 e 8 (1 massima e 8 minima).
- **ID:** identificatore univoco del Design Goal seguito dal suo nome.
- **Descrizione:** descrizione del Design Goal.
- **Categoria:** categoria di appartenenza del Design Goal.
- **RNF di origine:** Requisito non funzionale che lo ha generato.

Rank	ID	Descrizione	Categoria	RNF di origine
1	<b>DG_1</b> Disponibilità del sistema	UniPass deve essere disponibile 24h al giorno.	Performance	RNF_P1
2	<b>DG_2</b> Fallimento di sistema	Un crash del sistema non deve causare la perdita dei dati.	Dependability	RNF_A1
3	<b>DG_3</b> Multiplatforma	Il sistema sarà fruibile sia su PC sia su dispositivi mobili.	End User	RNF_S1
4	<b>DG_4</b> Lista viaggi	La lista dei viaggi disponibili deve essere mostrata al passeggero entro 20 secondi dalla ricerca.	Performance	RNF_P3



5	<b>DG_5</b> Notifica fallimento	Il sistema, in situazioni di fallimento, deve notificare l'utente tramite appositi messaggi.	Dependability	RNF_A2
6	<b>DG_6</b> Documentazione	Il sistema metterà a disposizione dei futuri sviluppatori una completa documentazione del processo di sviluppo del software, per eventuali modifiche e/o estensioni.	Maintenance	RNF_S2
7	<b>DG_7</b> Tempo di rilascio	Il sistema sarà rilasciato entro i tempi prestabiliti.	Cost	RNF_PA1
8	<b>DG_8</b> Interfaccia utente	L'applicativo sarà completamente navigabile attraverso link accessibili mediante l'interfaccia grafica.	End User	RNF_U1



#### Trade-Offs:

Trade-off	Descrizione
<b>Tempo di rilascio vs funzionalità</b>	Al fine di rispettare la scadenza concordata per il rilascio del Sistema, si preferirà escludere le funzionalità con bassa/media priorità nel caso in cui la loro implementazione implichi un ritardo nella consegna.
<b>Spazio vs velocità</b>	Verrà utilizzata più memoria per garantire un accesso diretto ai dati più utilizzati.

### 1.3 Definizioni, acronimi e abbreviazioni

- DB: Database.
- DBMS: Database Management System.
- GUI: Graphic User Interface.
- DG: Design Goal.
- RAD: Requirements Analysis Document
- Sottosistema: un sottoinsieme dei servizi del dominio applicativo, formato da servizi legati da una relazione funzionale.
- Dati persistenti: dati che vengono salvati permanentemente all'interno del DB.

### 1.4 Riferimenti

- Requisiti non funzionali: Sezione 3.3 del RAD
- Libro:
  - Object-Oriented Software Engineering (Using UML, Patterns, and Java) Third Edition
- Autori:
  - Bernd Bruegge & Allen H. Dutoit

### 1.5 Panoramica

Il documento è composto da tre parti:

#### 1. Introduzione:

Nella prima parte viene fornita una breve panoramica sugli obiettivi del Sistema, seguita dalla definizione dei Design Goals e dei Trade-offs in cui si descrivono le qualità che caratterizzano maggiormente il Sistema.



Sono infine definiti gli acronimi, le abbreviazioni e le definizioni utilizzate all'interno del Documento.

## 2. Architettura del Sistema corrente:

In questa sezione vengono analizzate le architetture di sistemi simili a quello da noi proposto, evidenziando e discutendo differenze e similitudini.

## 3. Architettura del Sistema proposto:

Nella terza parte viene inizialmente descritta sommariamente l'architettura del Sistema proposto, dettagliata nei punti successivi i quali contengono: Decomposizione in sottosistemi, Design Architetturale, Diagramma di Deployment.

## 2. Architettura del Sistema corrente

---

Attualmente non è presente alcun software di car pooling che operi nell'ambiente universitario, sono state quindi analizzate le architetture di sistemi simili.

Nello specifico, applicazioni come Uber, BlaBlaCar e Lyft, utilizzano sistemi di tracciamento della posizione per offrire funzionalità relative alla gestione dei percorsi. Questi Sistemi sono quindi dotati, oltre che di un Web Server, di un LIS (Location Information Server) utilizzato per l'ottenimento di informazioni sulla posizione dei vari utenti che fanno uso del Sistema.

Il nostro applicativo non fa uso di informazioni sulla posizione degli utenti, non ha quindi come ulteriore scopo quello di gestire i percorsi dei diversi viaggi organizzati. Ne consegue che gli unici elementi in comune con le suddette applicazioni sono la presenza di un Web Server (per la comunicazione e lo scambio di informazioni tra il Sistema e i suoi utenti) e di un Database per la memorizzazione dei Viaggi organizzati e altri dati persistenti.

## 3. Architettura del Sistema proposto

---

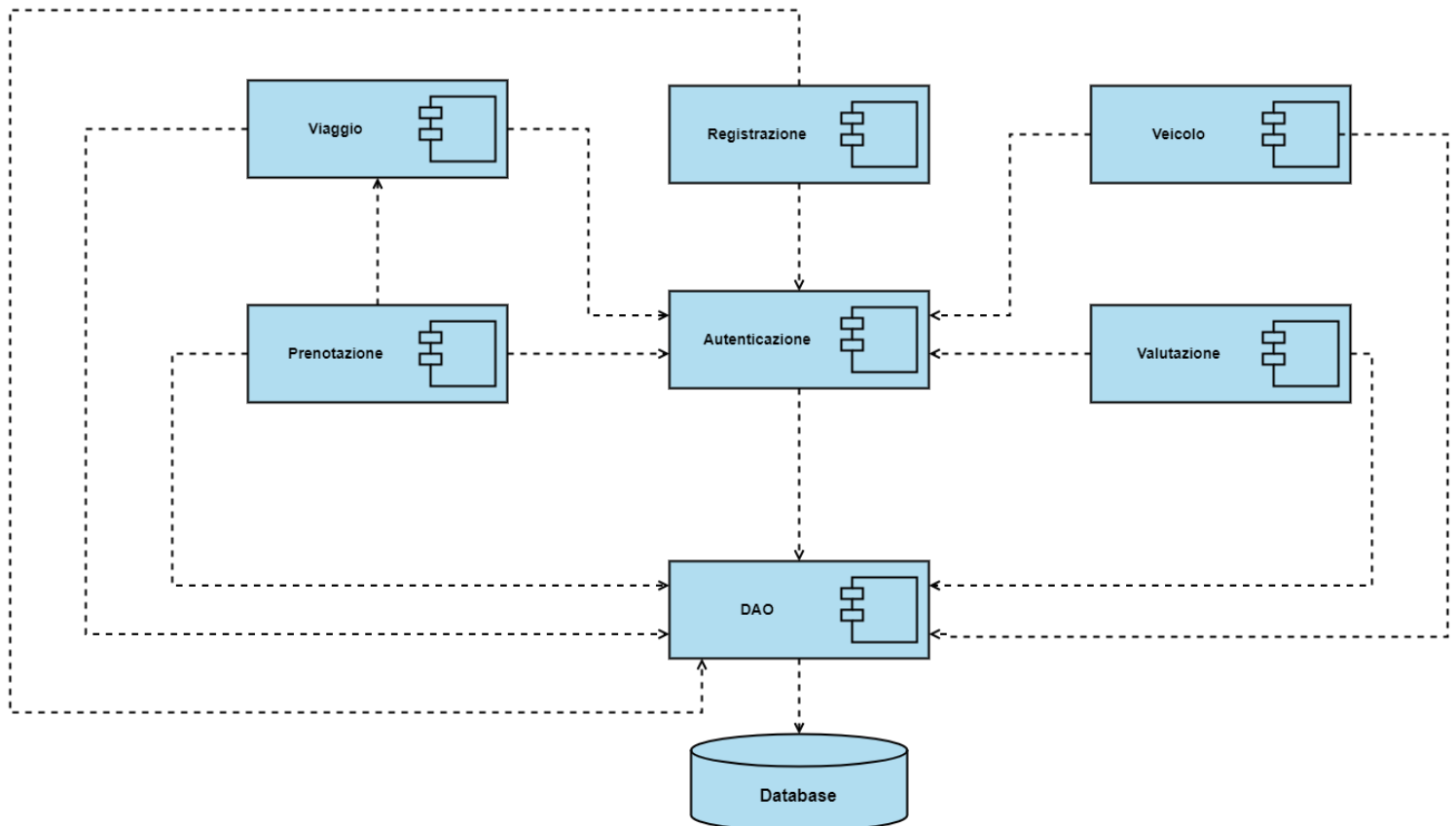
### 3.1 Panoramica

Il sistema proposto è basato sull'architettura Three-Tier.

Questo tipo di architettura risulta essere ottima per lo sviluppo di Web Application come il nostro sistema essendo presente una netta separazione tra logica di presentazione e logica di elaborazione.

Nello sviluppo del sistema verranno usati HTML e CSS per lo sviluppo della parte di front-end mentre per il back-end utilizzeremo Java SPRING.  
Per la gestione del Database verrà utilizzato MySQL.

### 3.2 Decomposizione in sottosistemi

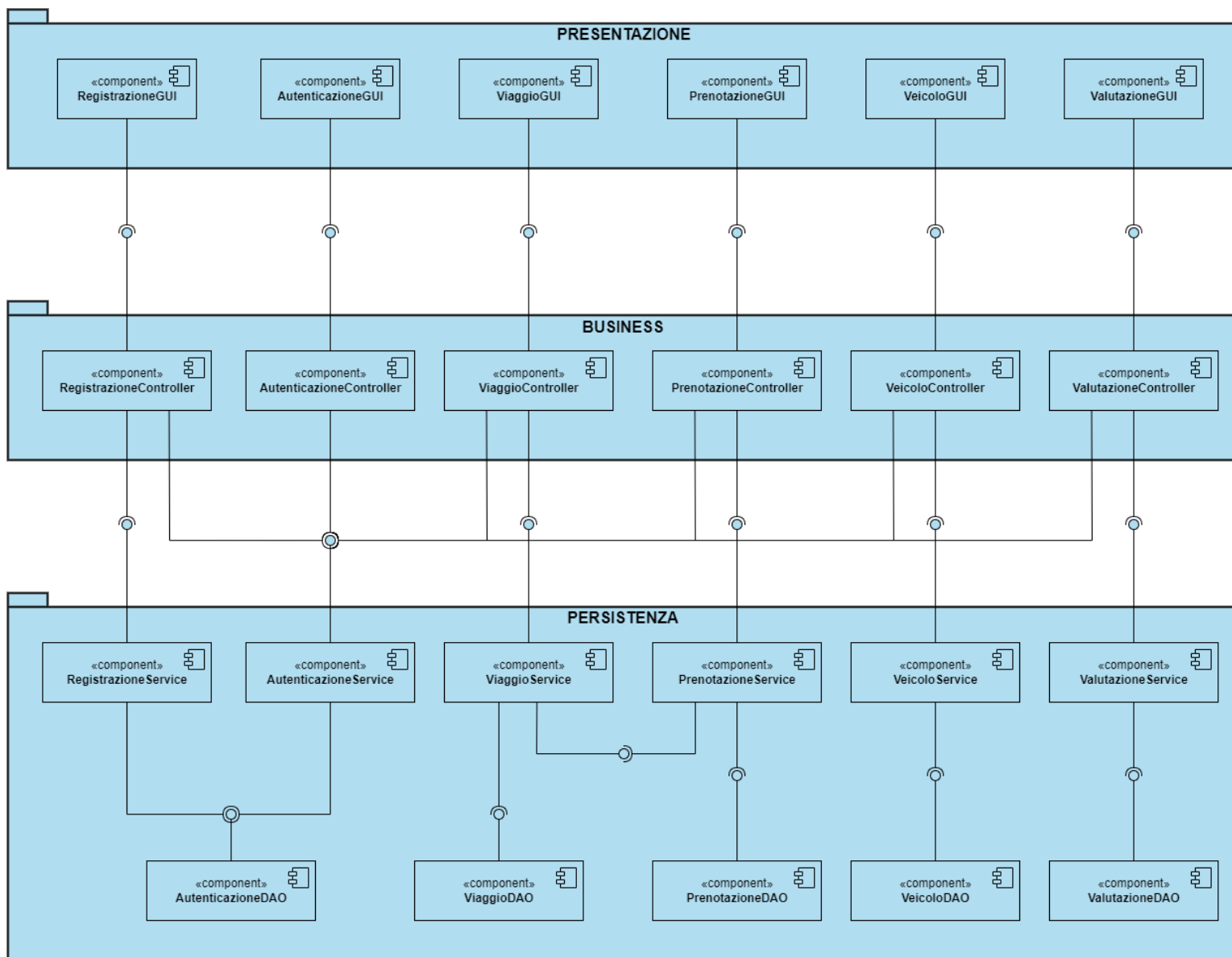


I sottosistemi individuati sono:

- **Registrazione:** Gestisce la registrazione dei vari tipi di utente: Passeggero e Guidatore.
- **Autenticazione:** Si occupa di gestire le funzionalità di Login, Logout, visualizzazione e modifica profilo
- **Viaggio:** Modella e raggruppa tutte le funzionalità relative alla gestione dei Viaggi.
- **Prenotazione:** Si occupa della gestione delle prenotazioni ai Viaggi.
- **Veicolo:** Gestisce l'aggiunta e la modifica dei veicoli associati ai Guidatori.
- **Valutazione:** Definisce e gestisce i servizi di valutazione degli utenti.
- **DAO:** Costituisce un'interfaccia tra i sottosistemi e il sottosistema Database.
- **Database:** Si occupa di gestire la persistenza dei dati.

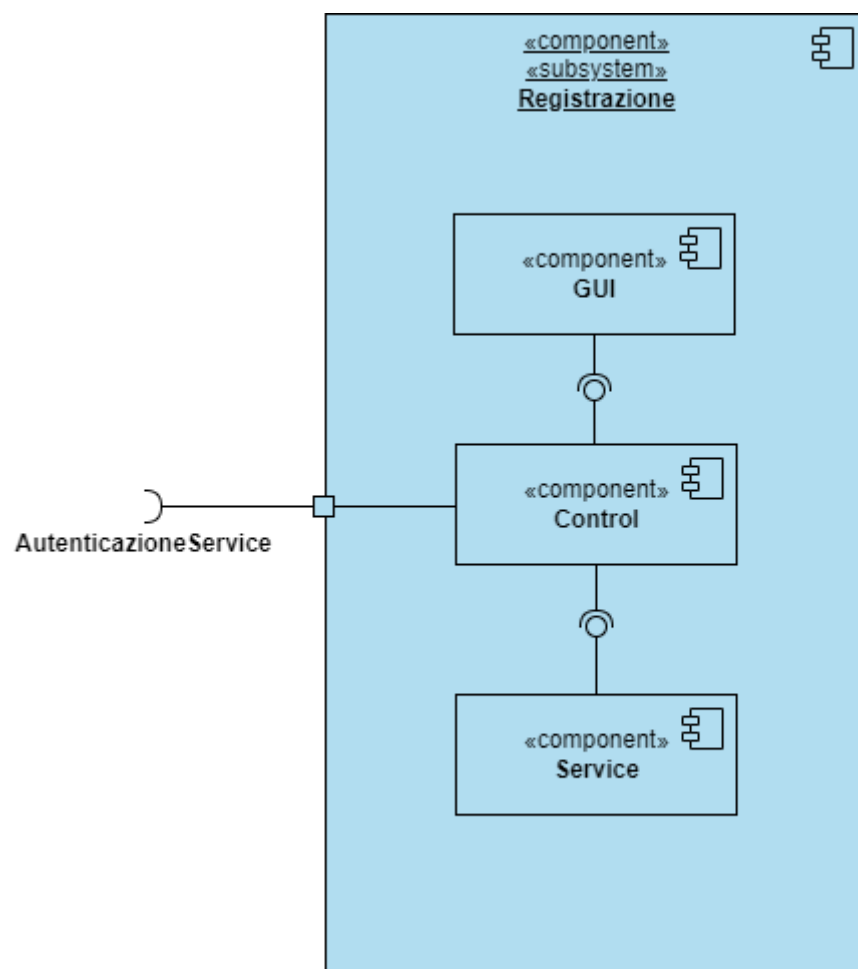


## Diagramma Architeturale

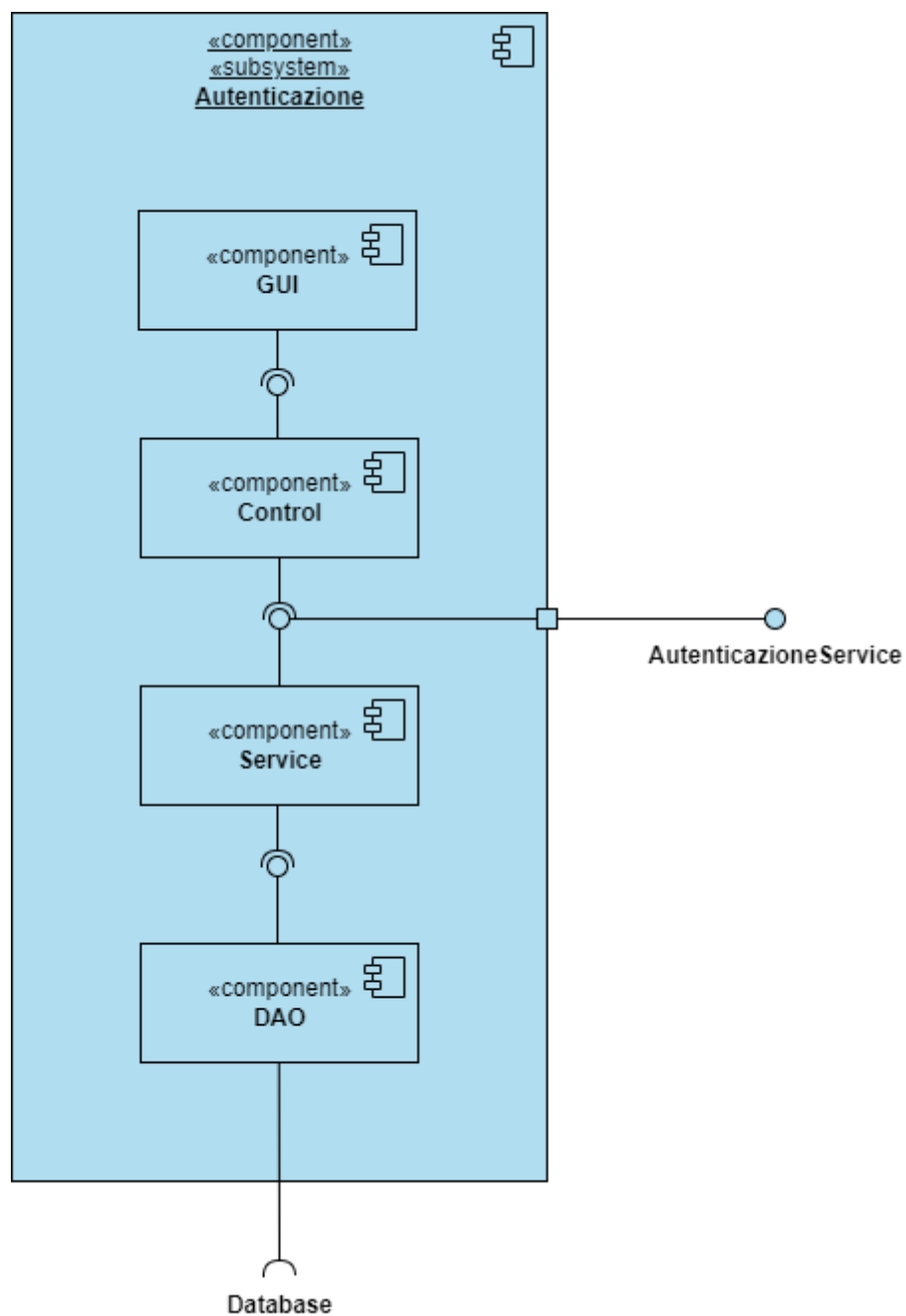


Di seguito una visione più dettagliata dei singoli sottosistemi, di cui si descrivono le componenti principali:

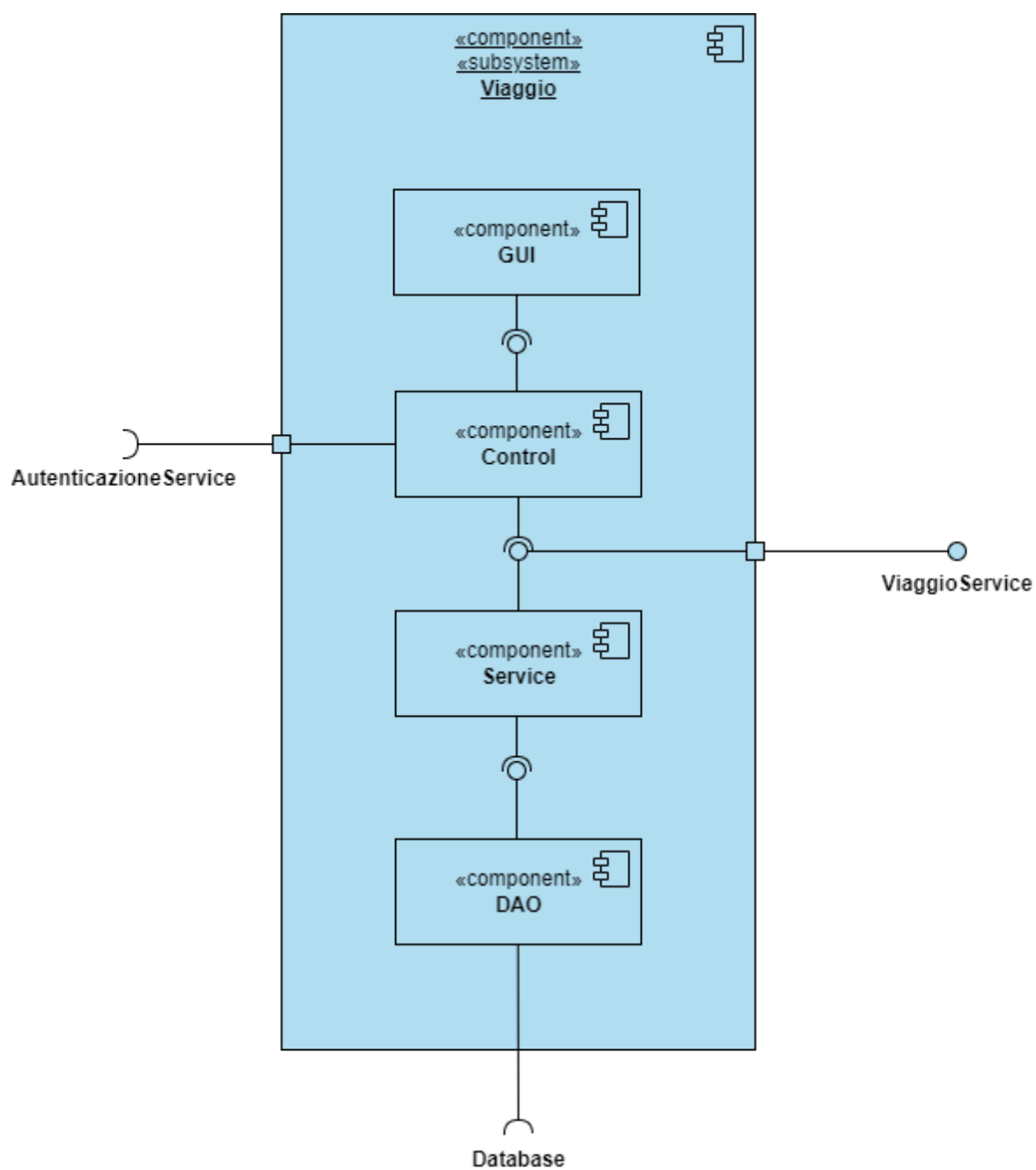
### Sottosistema Registrazione



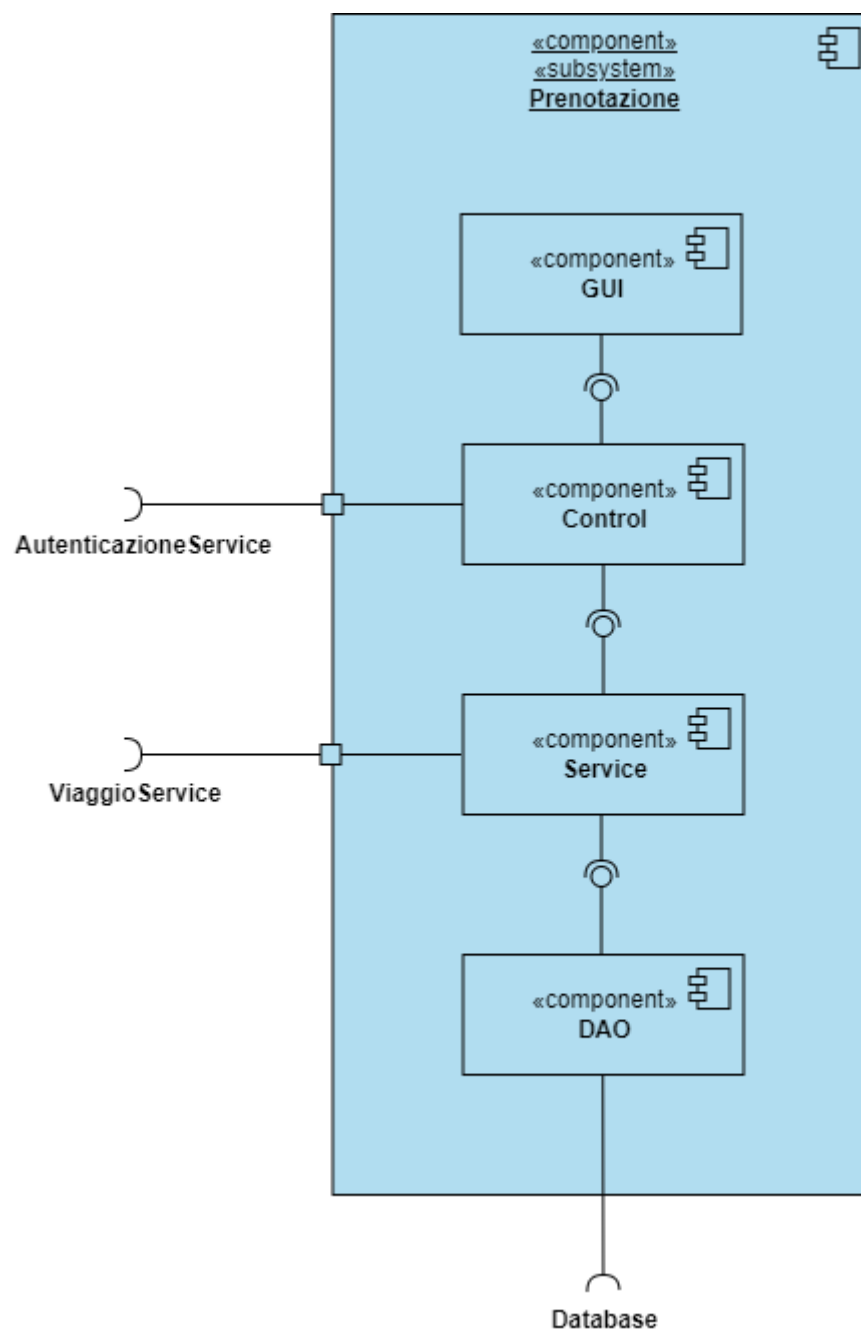
### Sottosistema Autenticazione



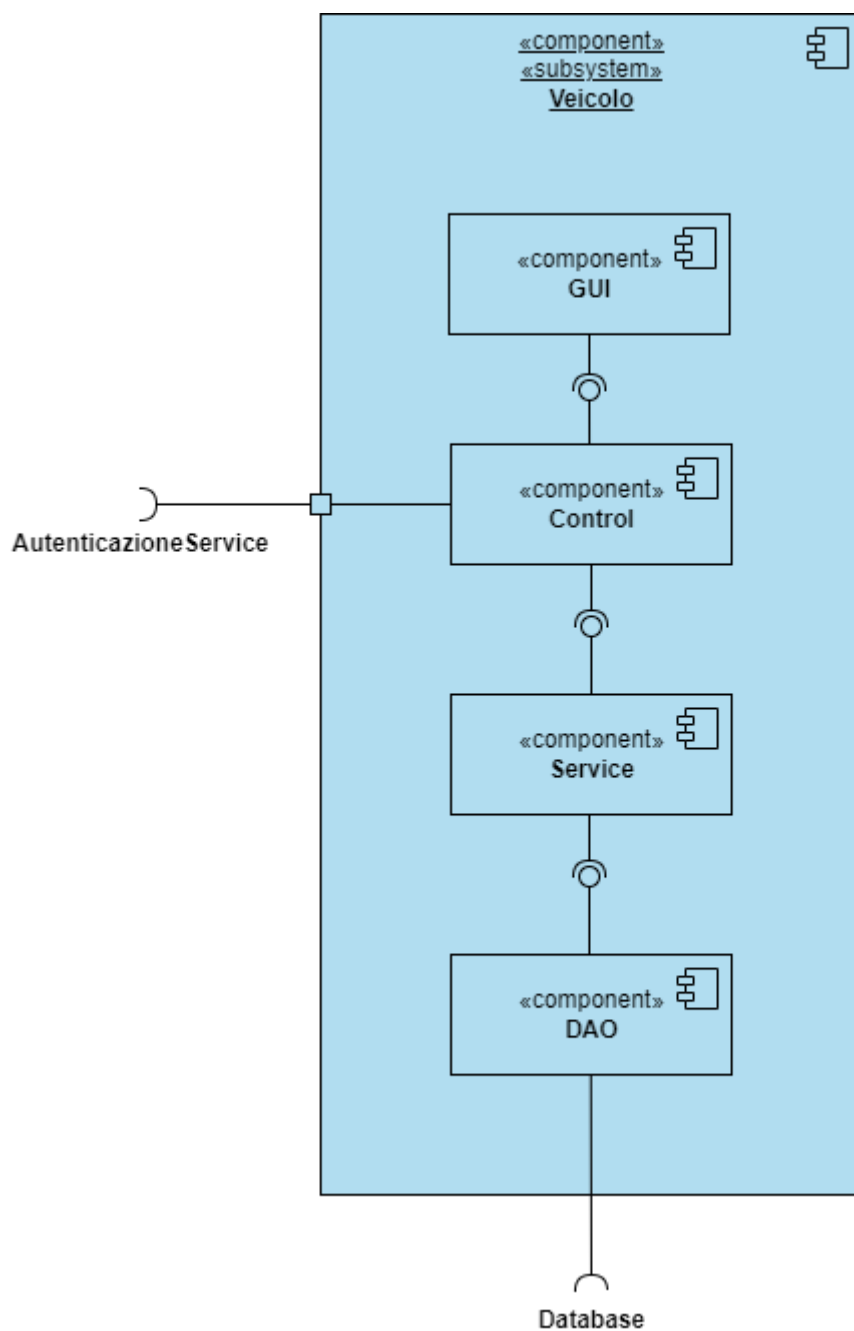
### Sottosistema Viaggio



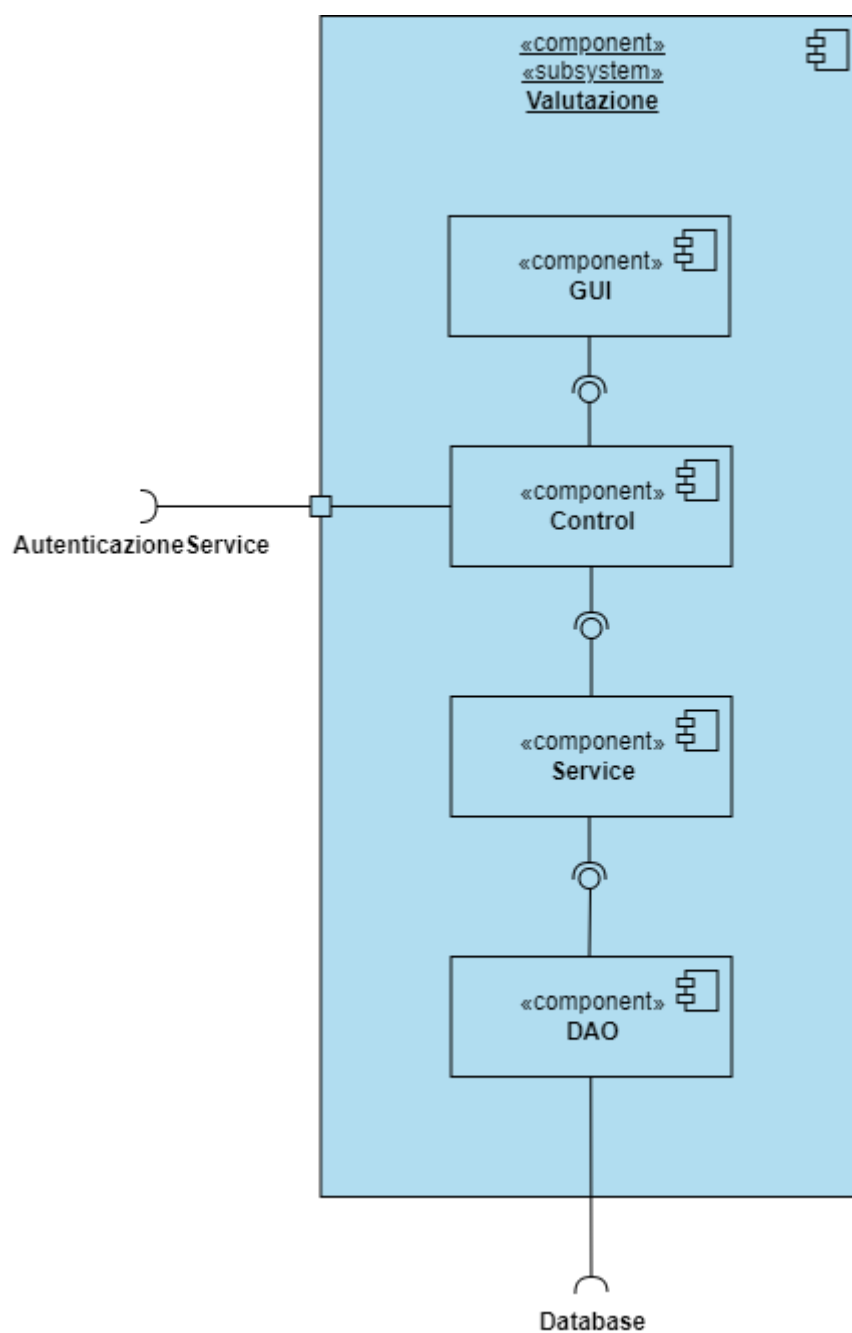
### Sottosistema Prenotazione



### Sottosistema Veicolo



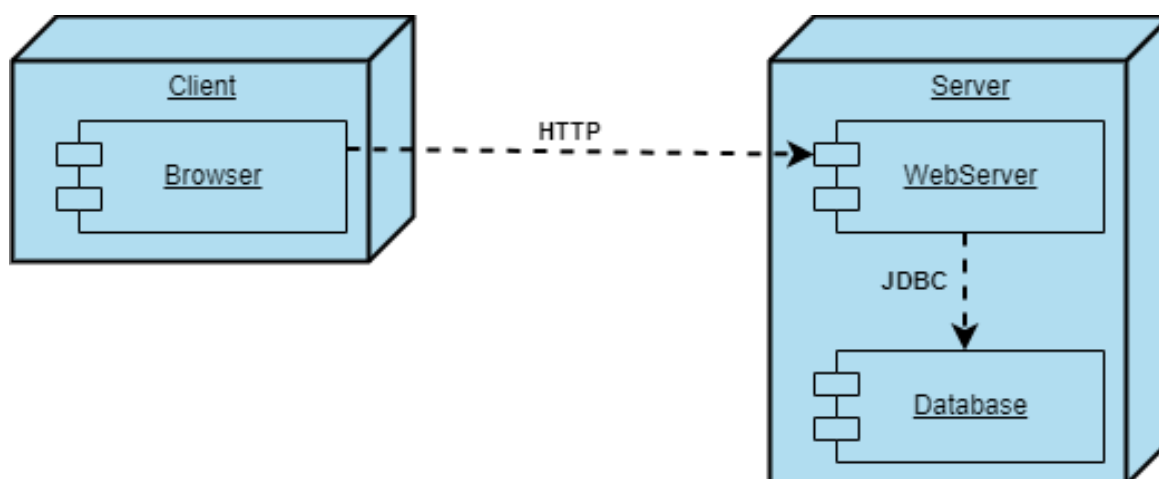
### Sottosistema Valutazione



### 3.3 Mapping Hardware/Software

Il Sistema da noi sviluppato è una Web Application, e in quanto tale la strategia di Mapping Hardware/Software a noi sembrata più conveniente è quella di prevedere un nodo Server su cui saranno “hostati” sia l'applicazione che il Database da essa utilizzato. Il Web Server sarà poi accessibile da tutti gli utenti dotati di una connessione Internet e di un qualsiasi dispositivo con un browser installato.

#### 3.3.1 Diagramma di deployment



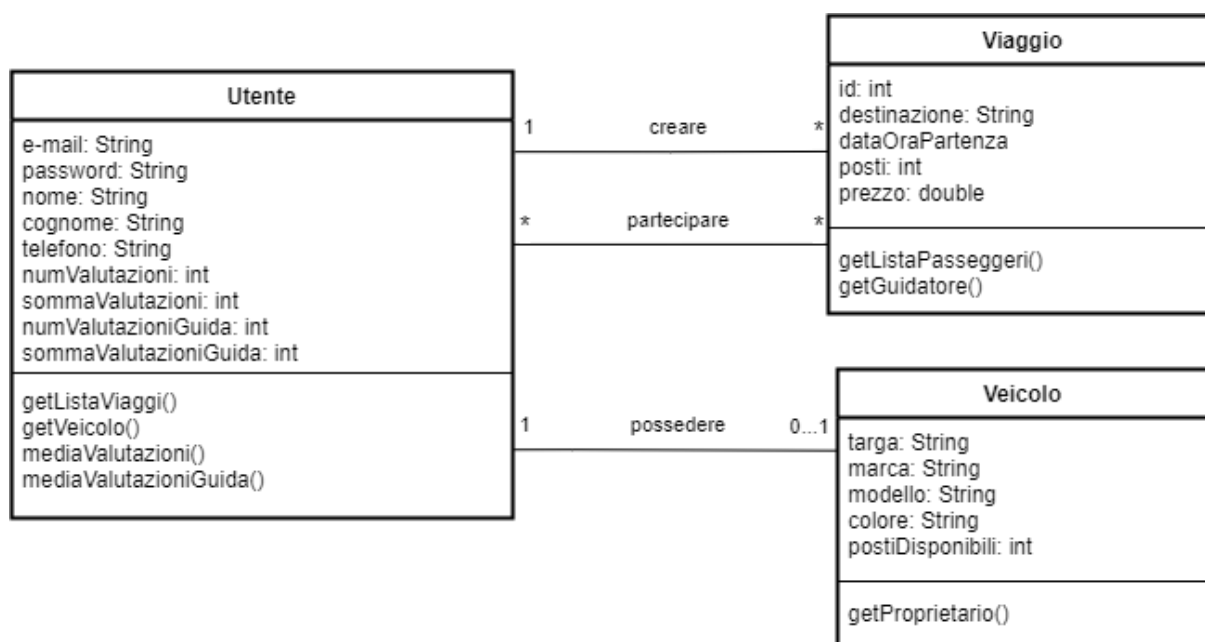
### 3.4 Gestione dati persistenti

Per la gestione dei dati persistenti si utilizzerà un database relazionale, poiché, tra i diversi tipi di database a disposizione, questo rappresenta un modo intuitivo e chiaro per la rappresentazione di dati persistenti in tabelle. Si utilizzerà inoltre un DBMS che ci garantirà sia di registrare, organizzare ed estrarre i dati contenuti nel DB che di usufruire di funzionalità quali integrità dei dati, gestione della concorrenza ed accesso affidabile ai dati stessi. Questo è importante poiché vari utenti potrebbero accedere contemporaneamente alle informazioni del DB.



## Class Diagram Ristrutturato

Tenendo conto dei Design Goals e dei Trade-off, il Class Diagram presente nel **Requirements Analysis Document** è stato ristrutturato. Nello specifico, è stata rimossa la specializzazione della classe “Utente” in “Passeggero” e “Guidatore”, riportando alla classe padre tutti i loro metodi e attributi.



### 3.5 Controllo degli accessi e sicurezza

Di seguito è riportata la matrice degli accessi, la quale indica i servizi accessibili per ciascun attore del Sistema:

ATTORI OGGETTI	UTENTE REGISTRATO	GUIDATORE	PASSEGGERO
REGISTRAZIONE	Registrazione		
AUTENTICAZIONE	Login Logout VisualizzaProfilo ModificaProfilo VisualizzaStoricoViaggi		
VIAGGIO		CreaViaggio CancellaViaggio EscludiPasseggero	CercaViaggi
PRENOTAZIONE		ChiudiPrenotazioni	PrenotazioneViaggio CancellaPrenotazione
VEICOLO		AggiungiVeicolo ModificaVeicolo RimuoviVeicolo	
VALUTAZIONE		ValutaPasseggero	ValutaGuidatore

### 3.6 Controllo flusso globale del sistema

La piattaforma UniPass è un'applicazione in cui si fa largo uso di interfacce grafiche. L'utente quindi, quando vuole utilizzare una funzionalità lo farà per mezzo della rispettiva Graphic User Interface. Ognuna di esse avrà il suo controllo dedicato, il quale eseguirà le operazioni necessarie per soddisfare al meglio la richiesta dell'utente. Data quindi la natura dell'applicazione è stato previsto un meccanismo di controllo del flusso di tipo event-driven, poiché in ogni situazione il Sistema risponderà ad eventi scaturiti da azioni dell'utente sulle varie GUI.



### 3.7 Condizione limite

#### 3.7.1 Avvio del Sistema

Lo Use Case relativo è stato definito nel punto 3.4.2 del RAD

#### 3.7.2 Arresto del Sistema

Lo Use Case relativo è stato definito nel punto 3.4.2 del RAD

## 4. Servizi dei sottosistemi

In questa sezione sono mostrati i servizi offerti da ognuno dei sottosistemi definiti in precedenza:

### Sottosistema Registrazione

Servizio	Descrizione	Interfaccia
<b>Registrazione</b>	Questa funzionalità permette di registrarsi alla piattaforma come utente.	RegistrazioneService

### Sottosistema Autenticazione

Servizio	Descrizione	Interfaccia
<b>Login</b>	Questa funzionalità permette ad un utente di accedere al sistema fornendo le proprie credenziali.	AutenticazioneService
<b>Logout</b>	Questa funzionalità permette ad un utente di disconnettersi dal Sistema.	AutenticazioneService
<b>Visualizza profilo</b>	Questa funzionalità permette ad un utente di visualizzare le sue informazioni personali.	AutenticazioneService
<b>Modifica profilo</b>	Questa funzionalità permette ad un utente di modificare le sue informazioni personali.	AutenticazioneService
<b>Visualizza storico</b>	Questa funzionalità permette ad un utente di visualizzare il proprio storico viaggi.	AutenticazioneService



#### Sottosistema Viaggio

Servizio	Descrizione	Interfaccia
<b>Crea viaggio</b>	Questa funzionalità permette ad un guidatore di creare un viaggio.	ViaggioService
<b>Elimina viaggio</b>	Questa funzionalità permette ad un guidatore di cancellare un viaggio.	ViaggioService
<b>Cerca viaggi</b>	Questa funzionalità permette ad un passeggero di ricercare dei viaggi mediante dei filtri di ricerca.	ViaggioService
<b>Escludi Passeggero</b>	Questa funzionalità permette ad un guidatore di escludere un passeggero dal proprio viaggio.	ViaggioService

#### Sottosistema Prenotazione

Servizio	Descrizione	Interfaccia
<b>Prenotazione viaggio</b>	Questa funzionalità permette ad un passeggero di prenotarsi ad un viaggio.	PrenotazioneService
<b>Cancella prenotazione</b>	Questa funzionalità permette ad un passeggero di cancellare la propria prenotazione ad un viaggio.	PrenotazioneService
<b>Chiudi prenotazioni</b>	Questa funzionalità permette ad un guidatore di chiudere le prenotazioni al proprio viaggio.	PrenotazioneService

#### Sottosistema Veicolo

Servizio	Descrizione	Interfaccia
<b>Aggiungi veicolo</b>	Questa funzionalità permette ad un guidatore di associare un veicolo al proprio profilo.	VeicoloService
<b>Modifica veicolo</b>	Questa funzionalità permette ad un guidatore di modificare un veicolo associato al proprio profilo.	VeicoloService
<b>Rimuovi veicolo</b>	Questa funzionalità permette ad un guidatore di rimuovere un veicolo associato al proprio profilo.	VeicoloService



#### Sottosistema Valutazione

Servizio	Descrizione	Interfaccia
<b>Valuta guidatore</b>	Questa funzionalità permette ad un passeggero di valutare il guidatore di un viaggio a cui ha partecipato.	ValutazioneService
<b>Valuta passeggero</b>	Questa funzionalità permette ad un guidatore di valutare un passeggero di un suo viaggio.	ValutazioneService