

# Object Design Document

*Team Plemento*

# 1. Introduzione

Lo scopo del documento è di illustrare gli oggetti del sistema nel contesto dell'architettura MVC.

Le componenti MVC sono rappresentati come oggetti BCE nel seguente modo:

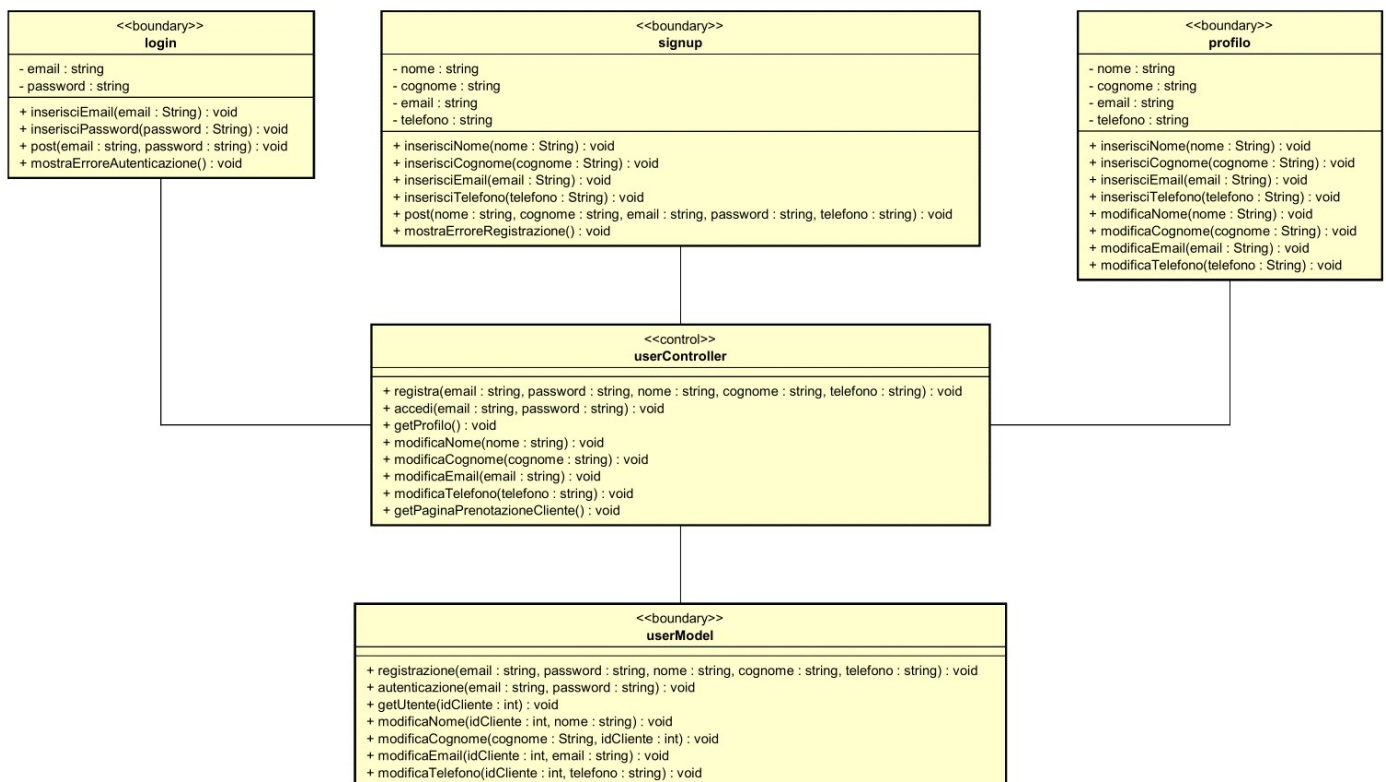
- View -> Boundary (interfaccia con l'utente)
- Controller -> Control
- Model -> Boundary (interfaccia con il DBMS)

L'architettura proposta è (EJS, Bootstrap, native Javascript) – (Node.js) – (MySQL)

## 2. Object Diagrams

Di seguito i diagrammi degli oggetti raggruppati per funzionalità.

### Gestione Account Cliente



- Registrazione Cliente (signup.ejs)
- Autenticazione Cliente (login.ejs)
- Visualizzazione e modifica dati (profilo.ejs)

# Prenotazione Cliente



- Effettuare prenotazione (map.ejs)
- Effettuare pagamento (effettuarePagamento.ejs)
- Visualizzazione prenotazioni (prenotazioniCliente.ejs)
- Invio motivazione e stima consegna del ritardo (notificaRitardo.ejs)

La pagina per effettuare la prenotazione (map.ejs) contiene l'algoritmo per il controllo della disponibilità dei veicoli. L'algoritmo assicura che nessun cliente occupi un veicolo necessario ad una prenotazione già effettuata, ma può occuparlo e consegnarlo prima dell'inizio della prenotazione in cui quel veicolo è richiesto. Questo garantisce utilizzo efficiente dei veicoli e una loro alta disponibilità.

L'algoritmo restituisce se è possibili prenotare, avendo come parametri:

- Tipo di veicolo
- Parcheggio di partenza
- Parcheggio di arrivo
- Data e ora di partenza
- Data e ora di arrivo

Di seguito lo pseudocodice dell'algoritmo:

N = veicoliDisponibiliOra //numero di veicoli nel parcheggio selezionato al momento della prenotazione

per ogni (prenotazione che comincia prima della dataora di partenza desiderata):

N = N - 1

per ogni (prenotazione che termina prima della dataora di partenza desiderata):

N = N + 1

// N diventa il numero di veicoli disponibili alla dataora di partenza desiderata

se N > 0

per ogni (prenotazione che comincia durante il periodo di noleggio desiderato)

N = N - 1 // veicolo che sarà utilizzato, quindi non prenotabile

// N diventa il numero di veicoli che è possibile prenotare

se N > 0

è possibile prenotare

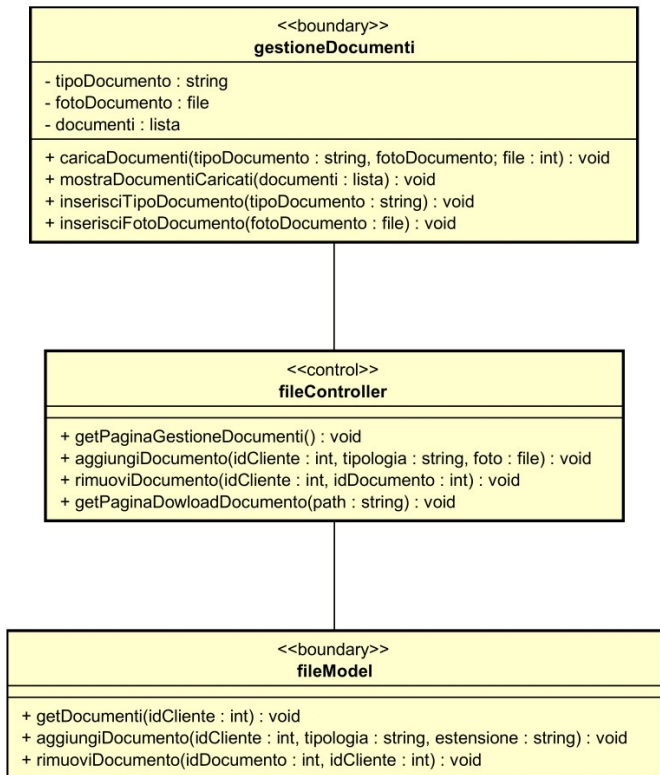
altrimenti

nessun veicolo disponibile

altrimenti

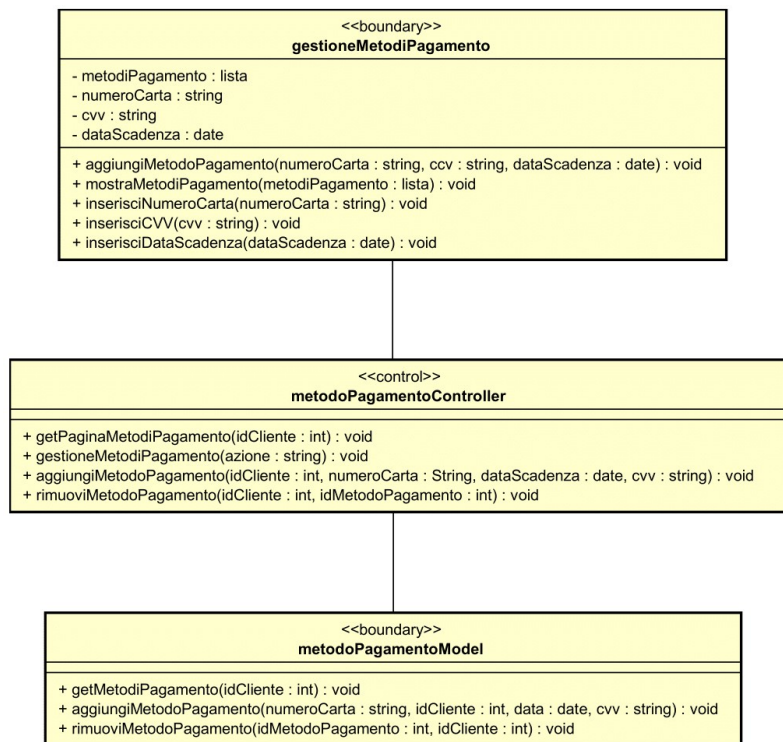
nessun veicolo disponibile

## Gestione Documenti



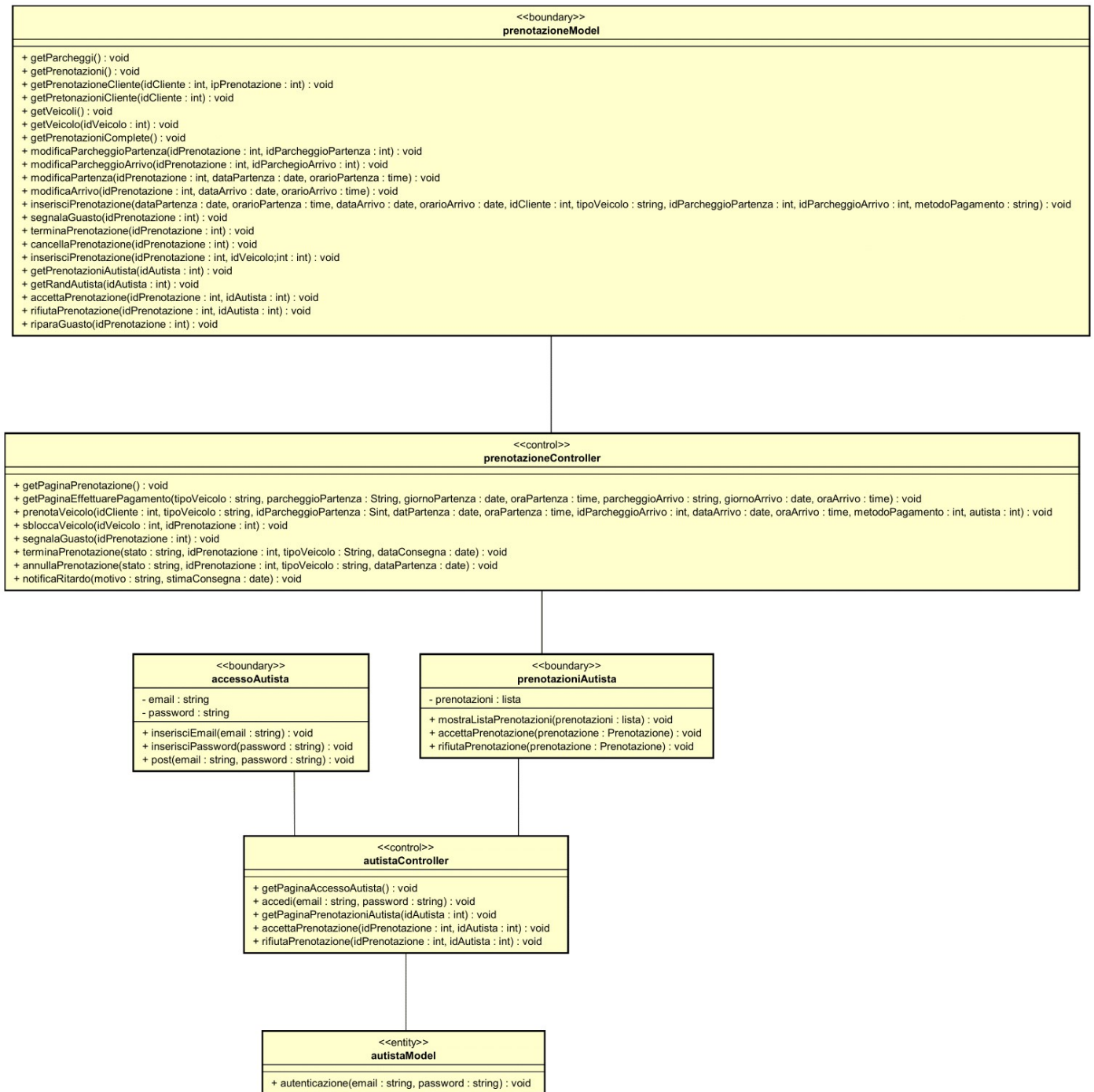
- Visualizzazione e inserimento documenti (gestioneDocumenti.ejs)

## Gestione Metodi di Pagamento



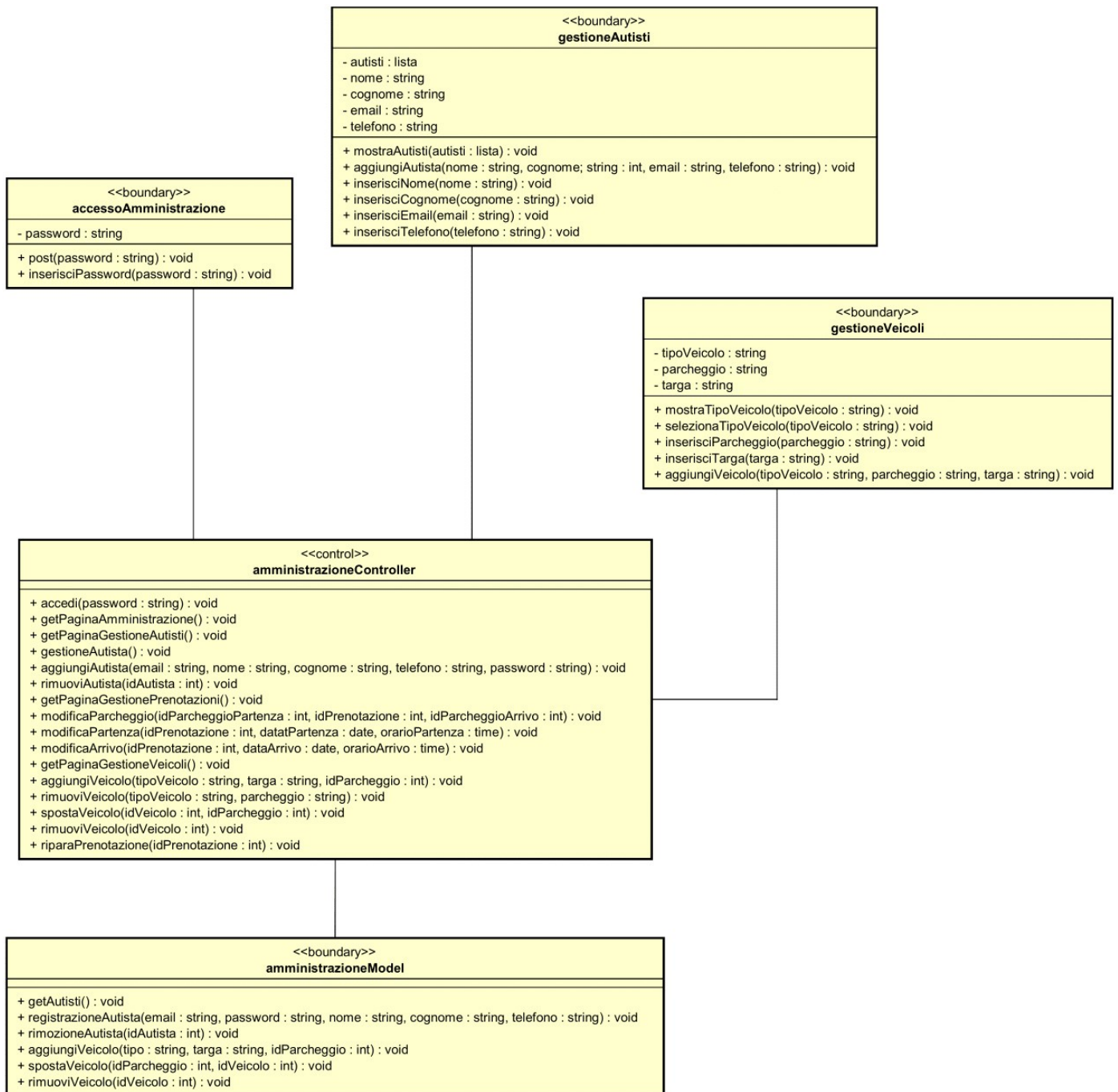
- Visualizzazione e inserimento metodi di pagamento (gestioneMetodiPagamento.ejs)

# Autista



- Autenticazione autista (accessoAutista.ejs)
- Visualizzazione e accettazione/rifiuto prenotazioni (prenotazioniAutista.ejs)

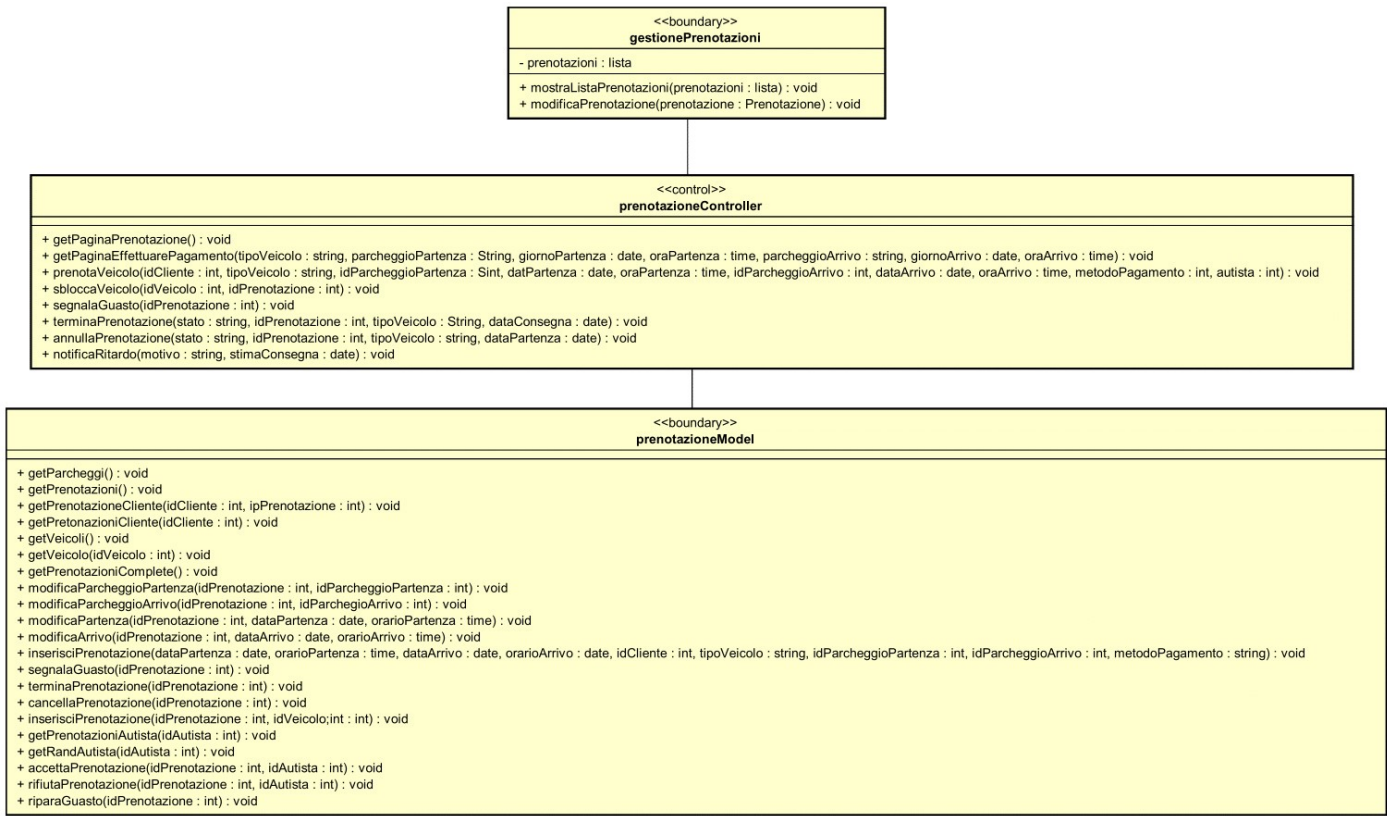
# Amministrazione



- Accesso amministratore (accessoAmministrazione.ejs)
- Visualizzazione e inserimento autisti (gestioneAutisti.ejs)
- Visualizzazione e inserimento veicoli (gestioneVeicoli.ejs)



# Gestione Prenotazioni amministratore



- Visualizzazione e modifica prenotazioni (gestionePrenotazioni.ejs)