



UNIVERSITÀ
degli STUDI
di CATANIA

Ingegneria del Software
Corso di laurea Magistrale in Ingegneria Informatica

AutoParking

Iterazione 2

Aurora Tallarita- Giulio Samperi

Sommario

| | |
|--|----------|
| Analisi e aggiornamenti | 3 |
| Diagramma delle classi | 4 |
| Diagramma di sequenza del sistema | 5 |
| Contratti delle operazioni | 5 |
| Diagrammi di sequenza | 6 |

Analisi e aggiornamenti

Nel corso di questa iterazione viene ripreso in esame il caso d'uso UC1, modificandone la struttura delle classi per ottenere una migliore gestione e divisione dei metodi, si è deciso di creare una classe Menù che si occupa dei metodi per la gestione dei vari casi del menù testuale, inoltre, si sono individuati e raggruppati metodi separati per promuovere la modularità e la leggibilità del codice.

Verrà dettagliato il caso d'uso UC2 relativo al ritiro di un veicolo

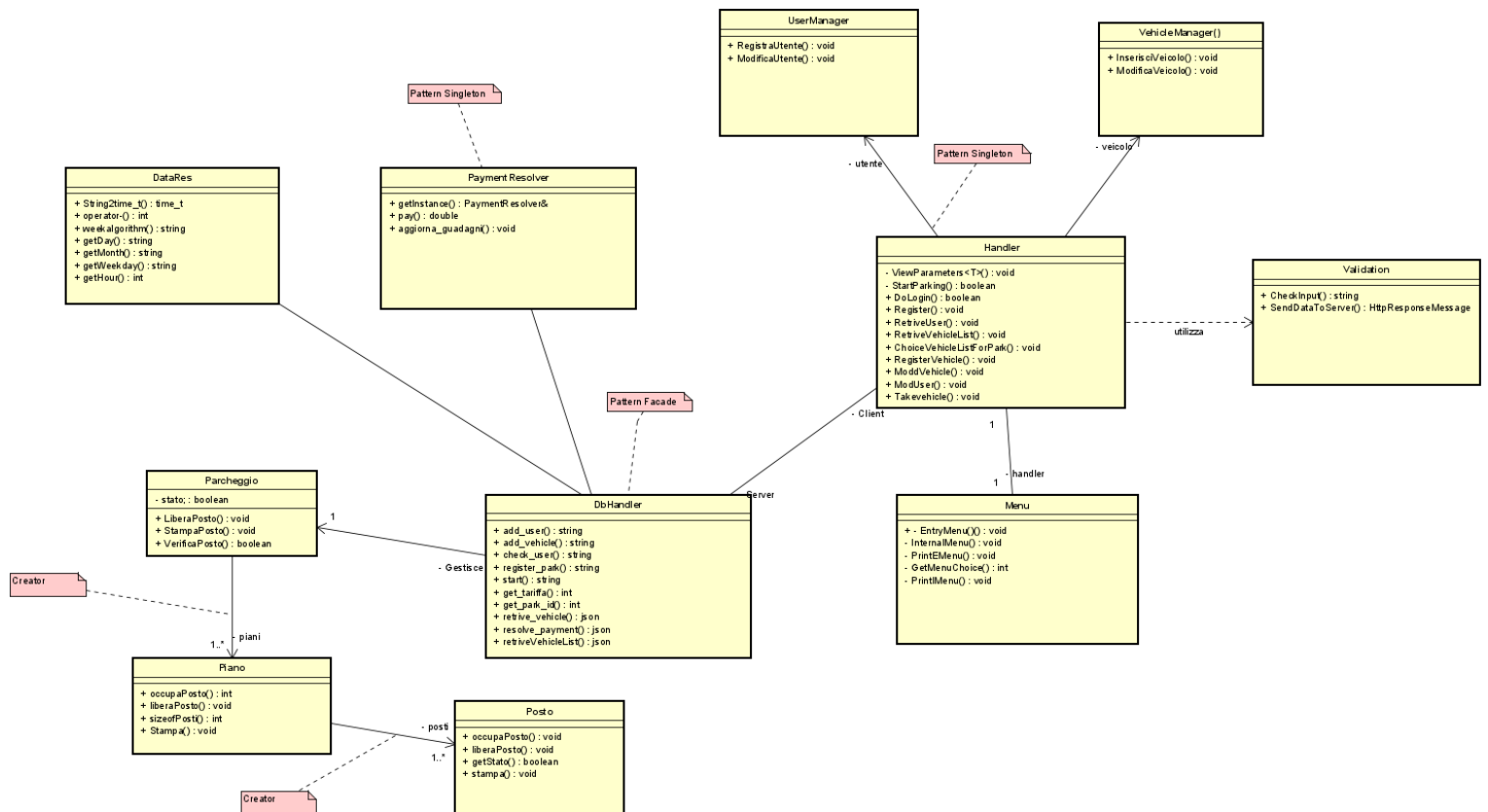
Riportiamo di seguito il caso d'uso in forma dettagliata:

UC2 Ritiro Veicolo:

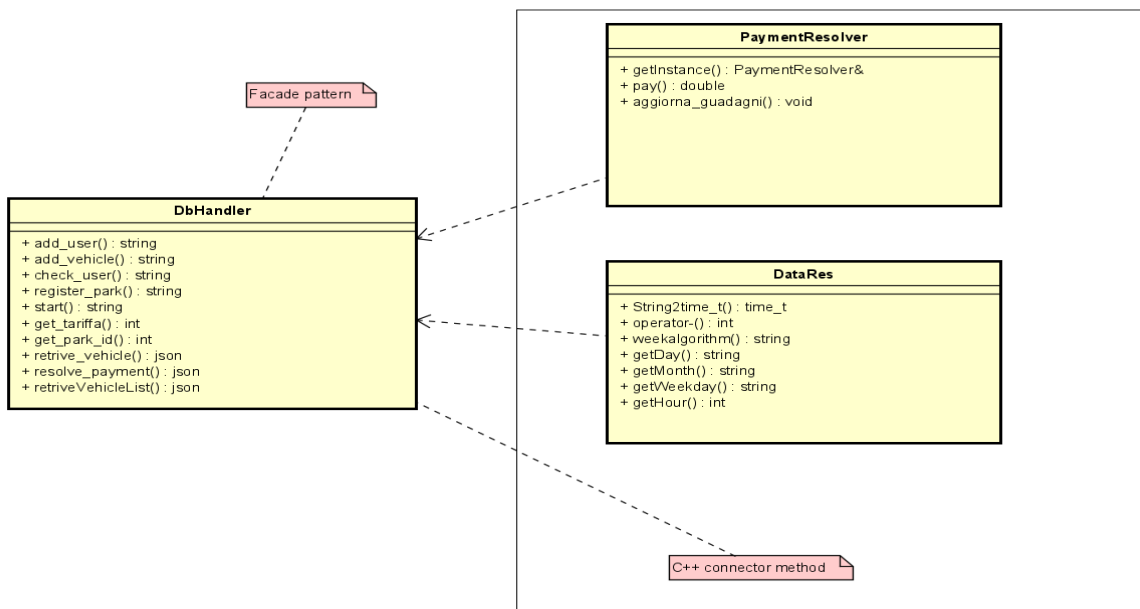
| | |
|--|--|
| Nome del caso d'uso | UC2 Ritiro veicolo |
| Portata | Applicazione AutoParking |
| Livello | Obbiettivo utente |
| Attore primario | Utente |
| Parti interessate e interessi | L'utente vuole richiedere il ritiro del veicolo in modo corretto e veloce. Il sistema vuole aggiornare le informazioni relative all'utente ed al suo veicolo |
| Pre-condizioni | L'utente deve essere già registrato ed aver effettuato un parcheggio. |
| Garanzia di successo | L'utente ritira il veicolo e paga, si aggiorna lo stato del posto |
| Scenario principale di successo | 1. L'utente seleziona "ritiro auto" dal terminale. 2. Il terminale richiede l'inserimento del codice univoco del ticket del parcheggio. 3. L'utente inserisce il codice univoco del ticket. 4. Il terminale restituisce il prezzo del ritiro calcolato tramite il tempo di inizio e di fine parcheggio e di una tariffa specifica. 5. L'utente paga la tariffa. 6. Il sistema accetta il pagamento e recupera il veicolo dell'utente. |
| Scenari alternativi | 4A. Il sistema non riconosce il codice e stampa un errore su terminale (ritorno al passo 2). |
| Frequenza di ripetizioni | Legata all'affluenza degli utenti |

Diagramma delle classi

Di seguito il diagramma delle classi di dominio ricavato dall'integrazione di nuove classi con quelle già esistenti.



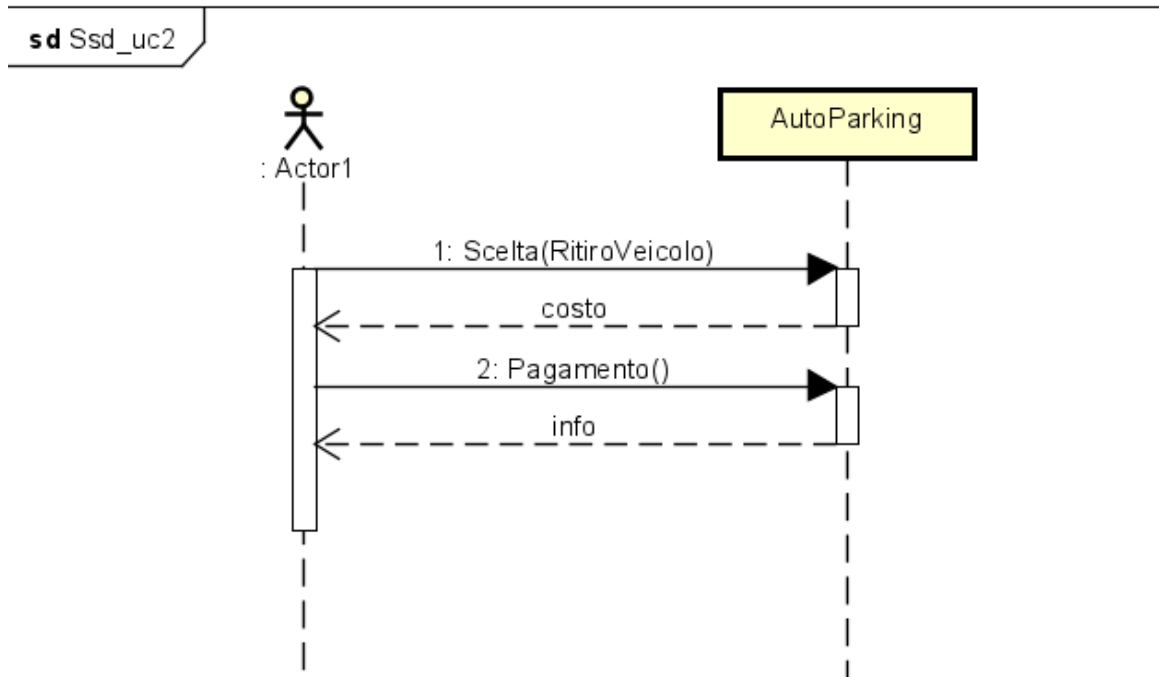
Si è deciso di utilizzare il pattern Singleton ai fini di limitare la creazione ad una sola istanza della classe `PaymentResolver`, per aggiungere integrità al calcolo dei profitti, e su `Handler` per aggiungere sicurezza alle richieste del server. È stato deciso di utilizzare il pattern `façade` nella classe `DbHandler`, semplifica l'interazione con il sistema di gestione del database sottostante. Incapsula l'accesso al database, la gestione delle connessioni, l'esecuzione di query e la manipolazione dei risultati. Espone metodi come `start`, `add_user`, `check_user`, `add_vehicle`, ecc funge da intermediario tra il sottosistema



del database e gli altri componenti del sistema, tra cui PaymentResolver usato per calcolare l'importo dovuto e DataRes per gestire la data del parcheggio e del ritiro.

Diagramma di sequenza del sistema

Indichiamo, tramite un diagramma di sequenza, il corso degli eventi, puntualizzando le interazioni tra il sistema e l'utente.



Contratti delle operazioni

Sono di seguito riportati i contratti delle operazioni dell'UC2

Contratto CO1: Scelta("Registrazione")

Operazione: Takevehicle()

Riferimenti: UC2 Ritiro Veiclo.

Precondizioni: L'utente deve essere già registrato ed aver effettuato un parcheggio.

Post-Condizioni: L'utente ritira il veicolo e paga

Contratto CO2: Pagamento

Operazione: Pay(object requestData)

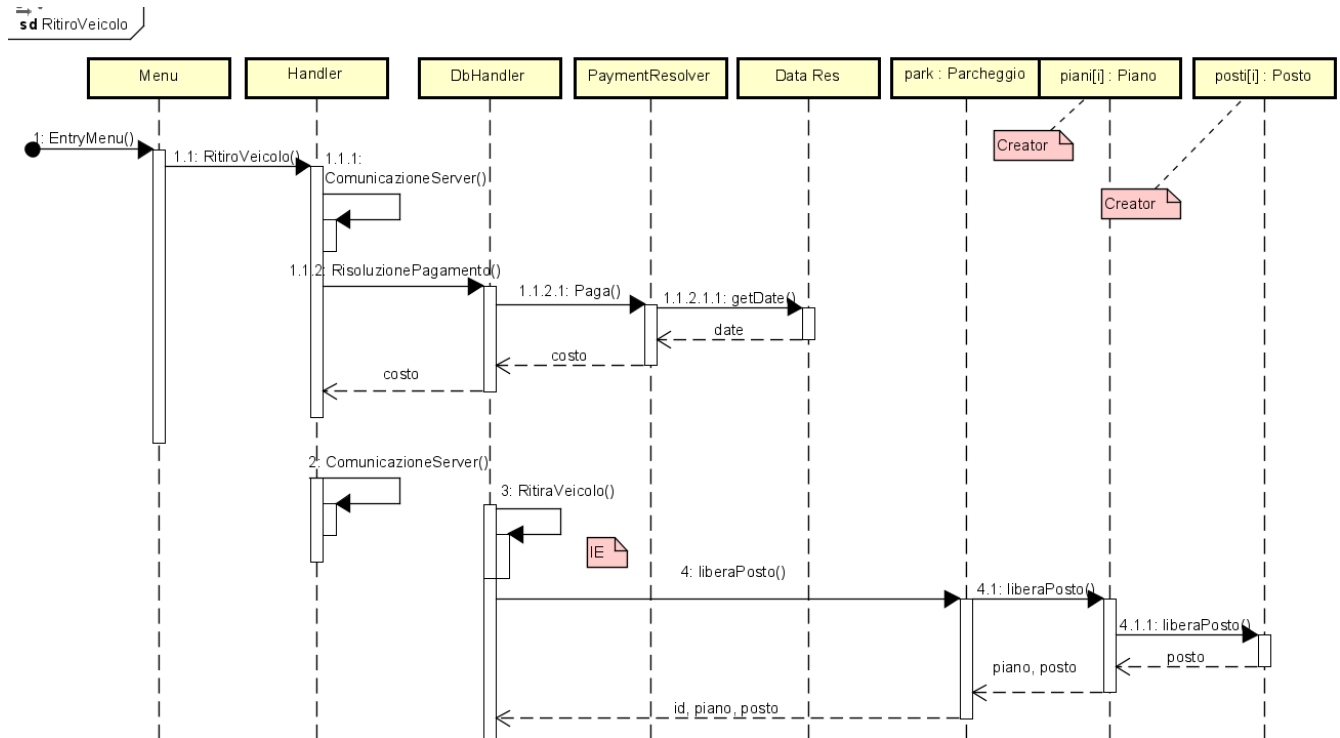
Riferimenti: UC2 Ritiro Veiclo.

Precondizioni: Il sistema accetta il pagamento e recupera il veicolo dell'utente.

Post-Condizioni: L'utente ritira il veicolo e paga, si aggiornalo stato del posto

Diagrammi di sequenza

UC2:



Di seguito i diagrammi aggiornati dell'uc1

