PROGETTAZIONE

CARLOAN

Versione 1.0

Data di rilascio:

INGEGNERIA DEL SOFTWARE A.A. 2014-2015

Realizzato da

Piemonte Christopher 587662 ITPS
chris.piemonte@hotmail.it

Saracino Paolo 592949 ITPS
paolo.saracino.ps@gmail.com

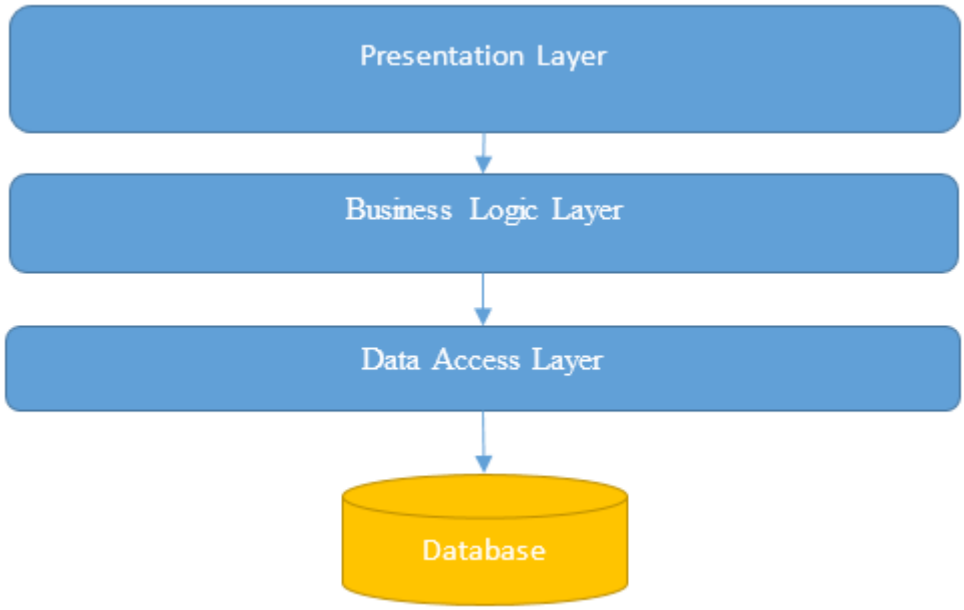
INDICE

1. ARCHITETTURA

1.1 Livelli Architetturali

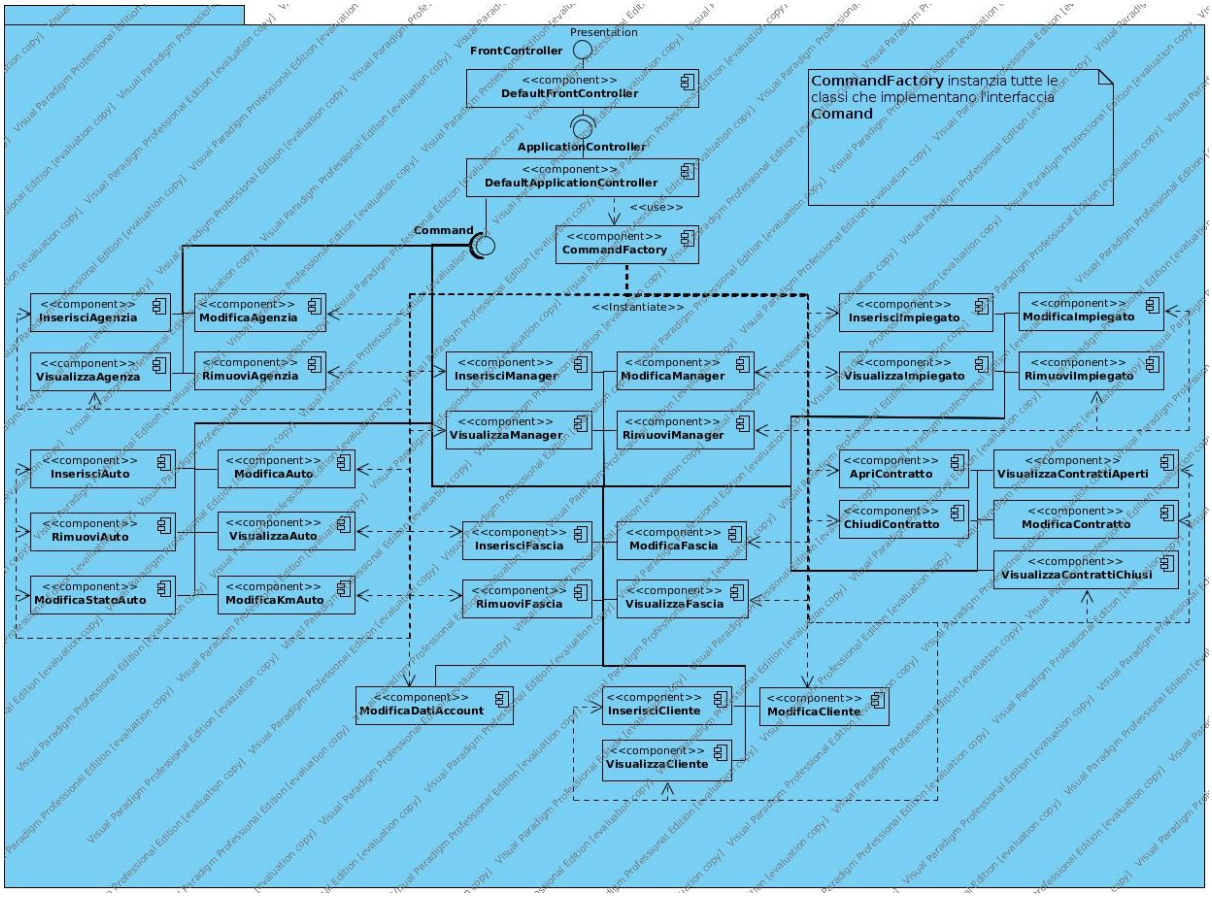
Il sistema CarLoan è stato sviluppando seguendo un architettura Client-Server che lavora in modalità Stand Alone, così come richiesto nei requisiti dell'ambiente di esecuzione.

Il sistema è descritto tramite un'architettura a tre livelli:

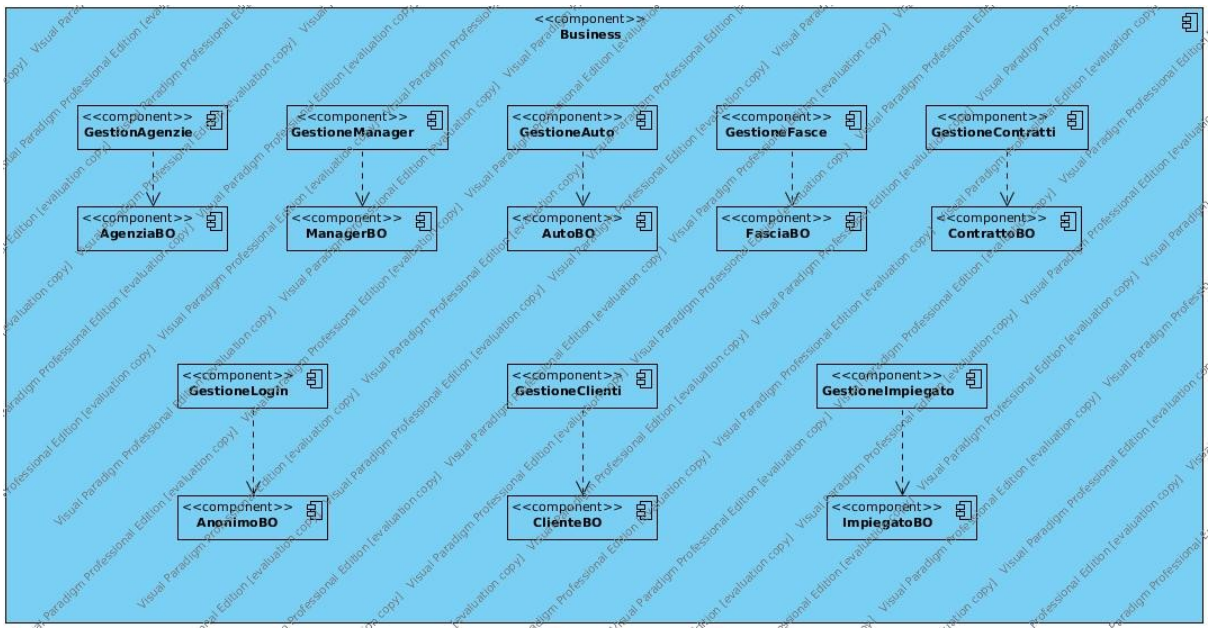


1.2 Diagramma delle Componenti

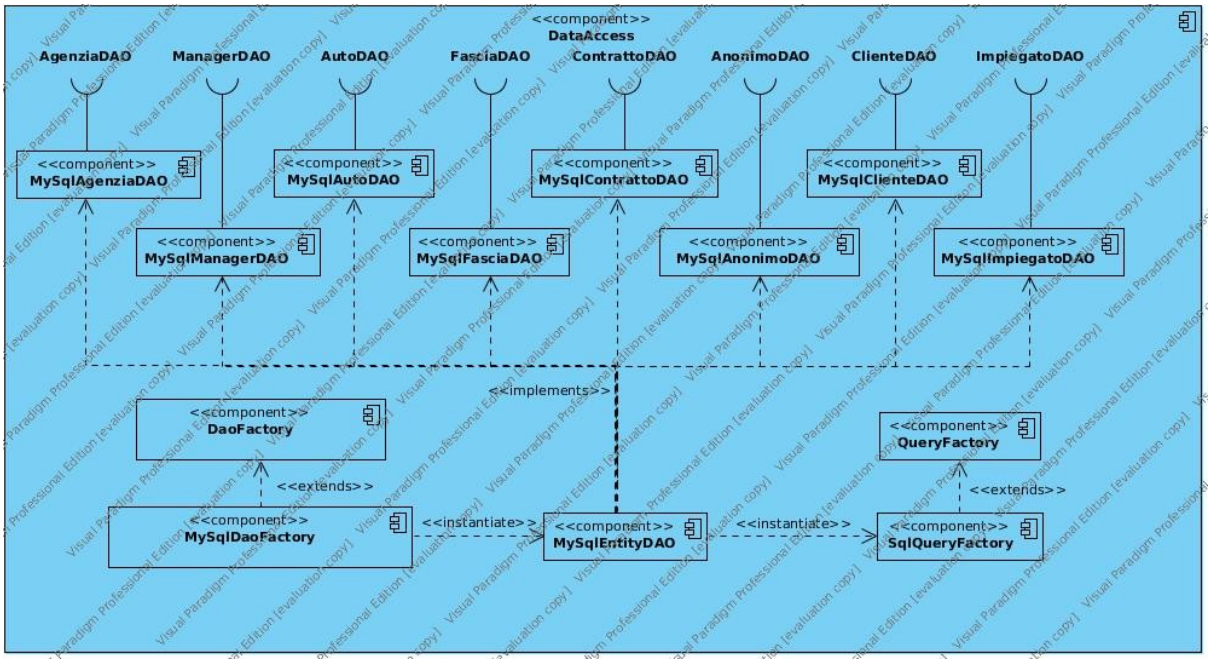
1.2.1 Presentation



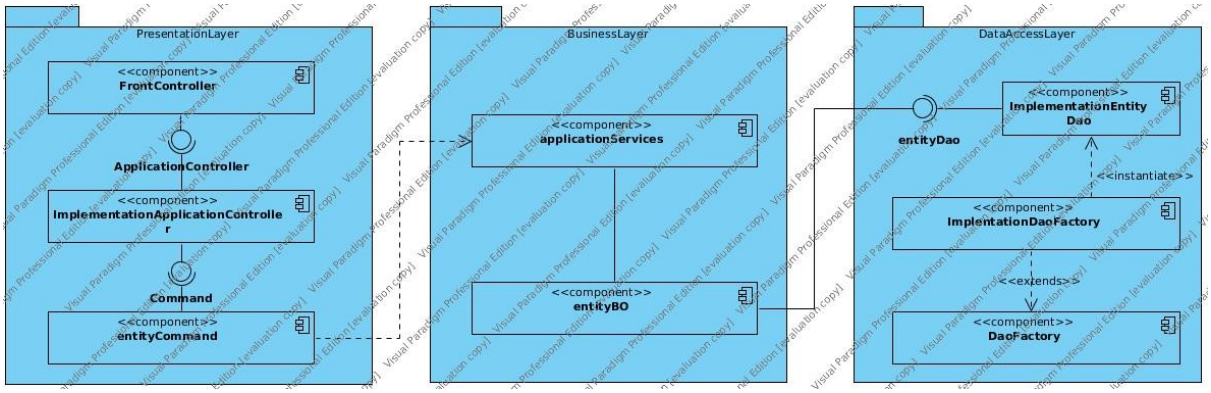
1.2.2 Business



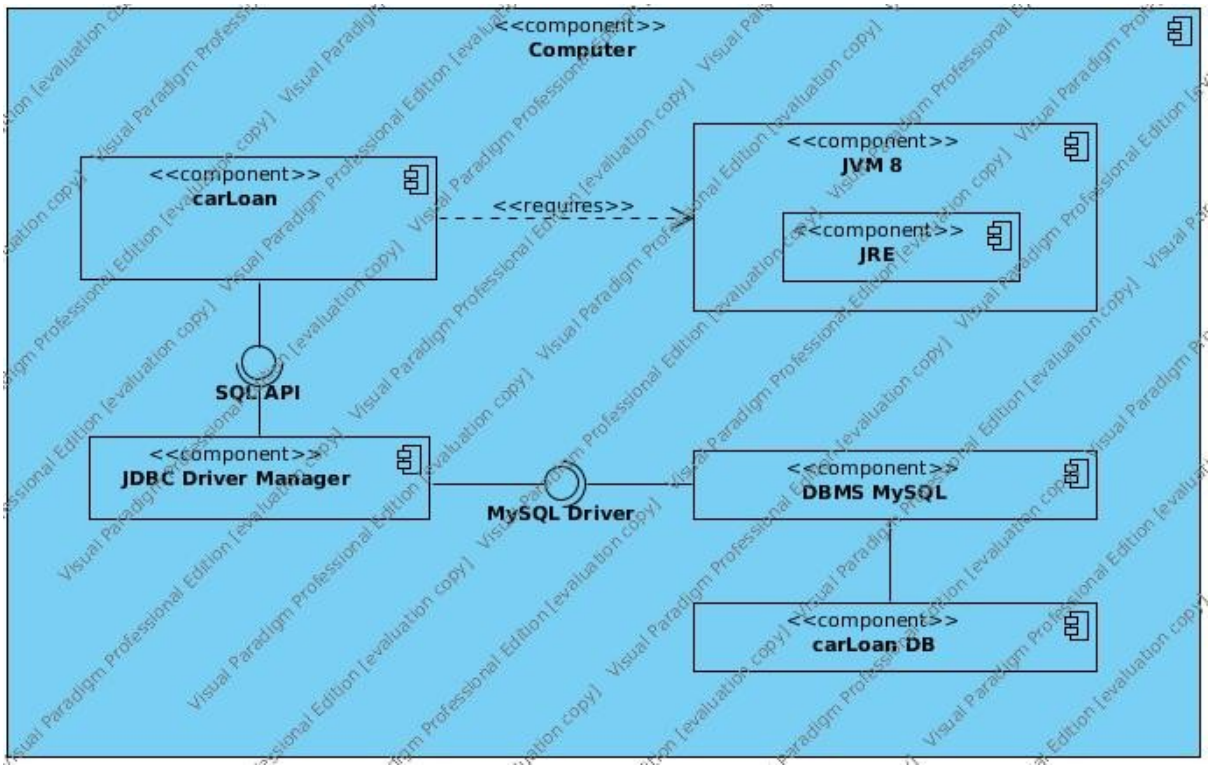
1.2.3 Integration



1.2.4 Comunicazione fra livelli



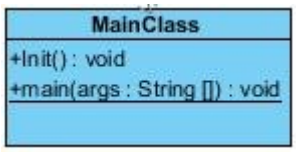
1.3 Diagramma di Configurazione



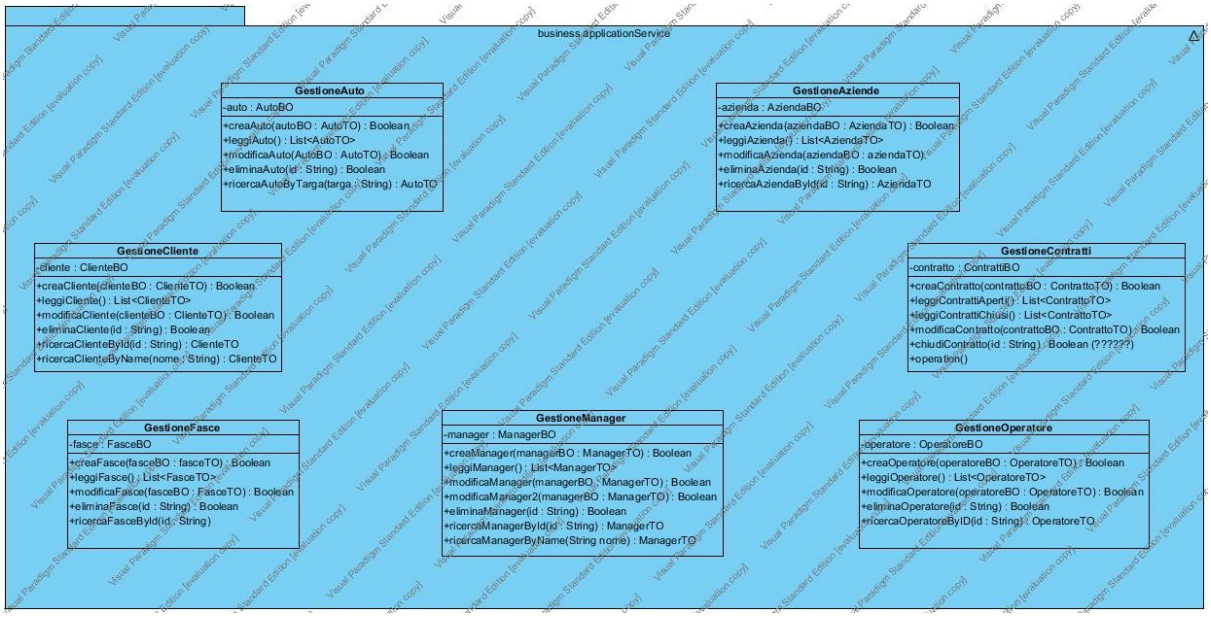
2. PROGETTO DI DETTAGLIO

2.1 Diagramma delle Classi

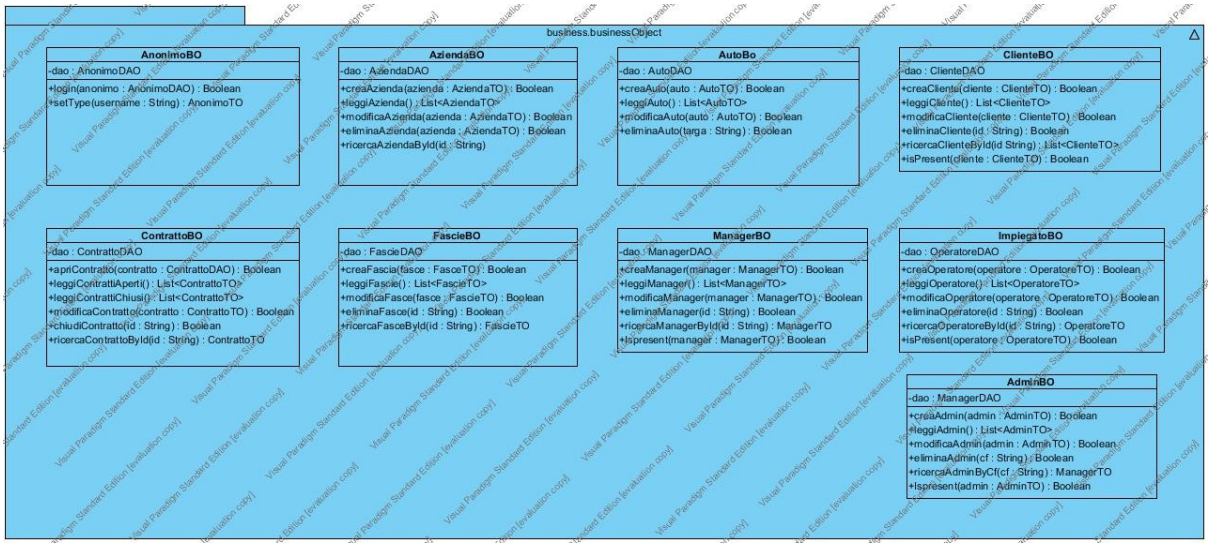
2.1.1 application



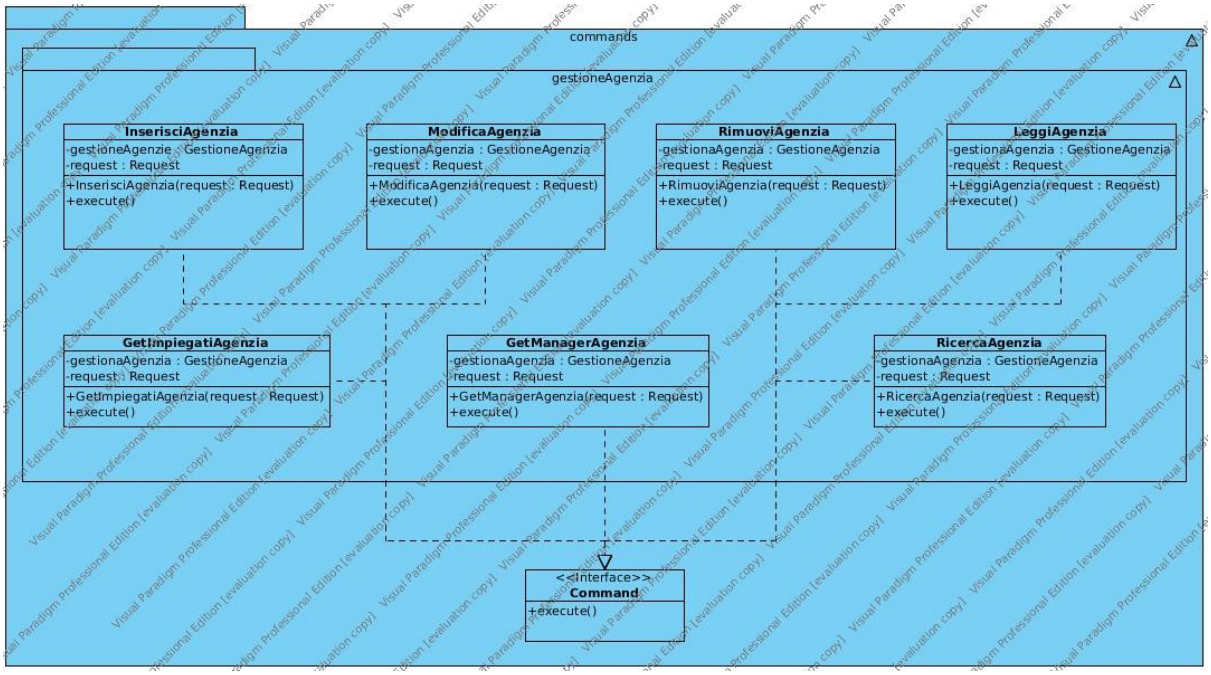
2.2.1 business.applicationServices



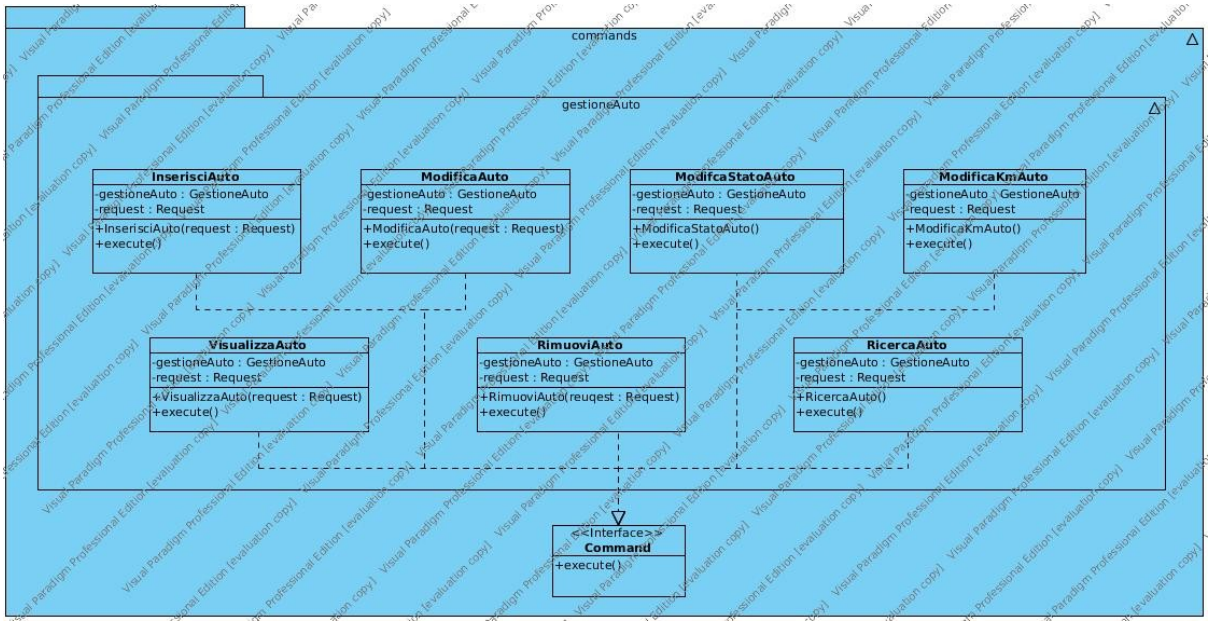
2.2.2 business.businessObjects



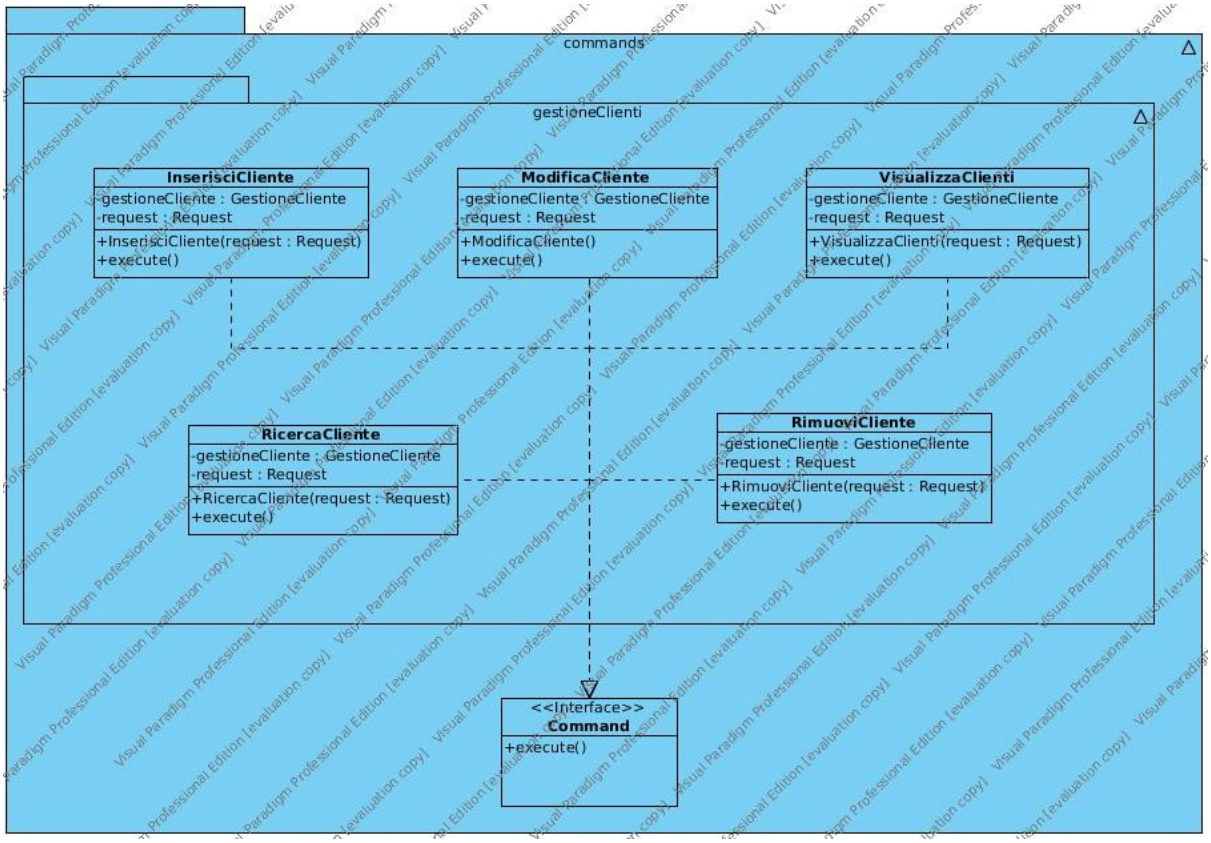
2.3.1 presentation.commands.gestioneAgenzie



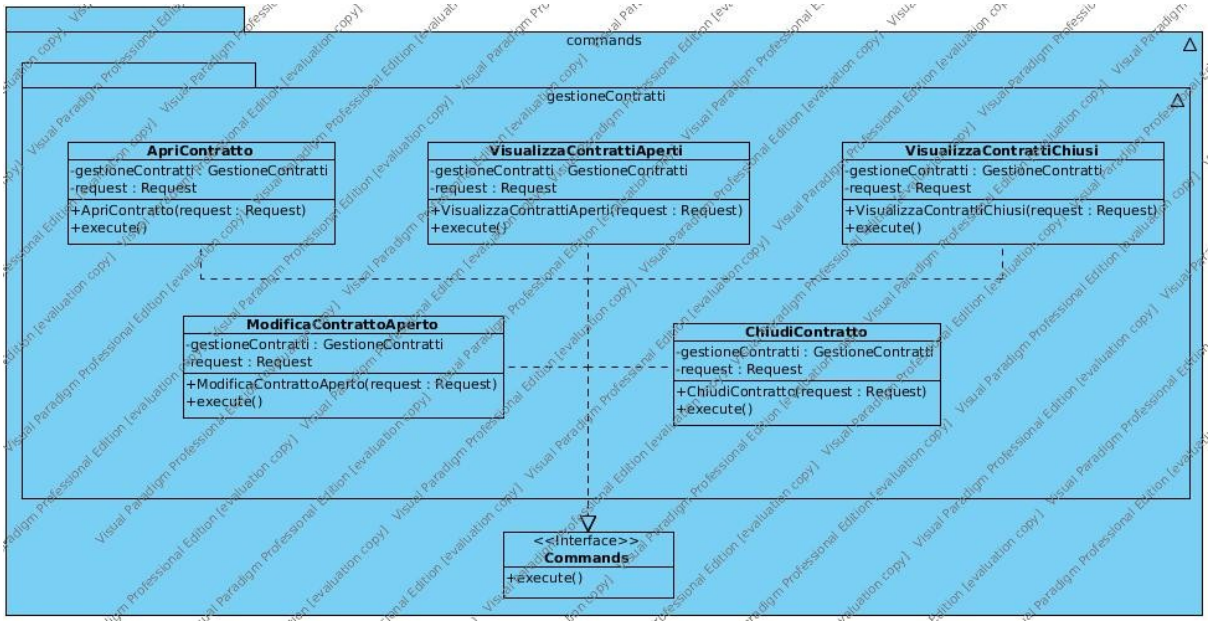
2.3.2 presentation.commands.gestioneAuto



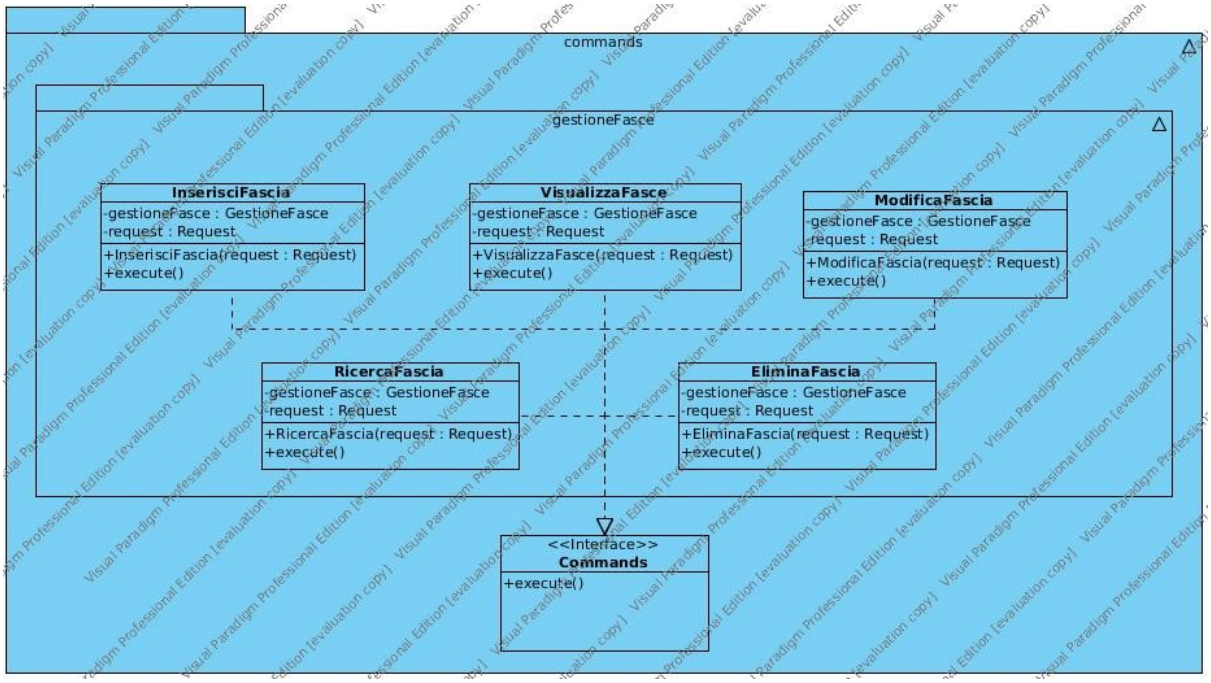
2.3.3 presentation.commands.gestioneClienti



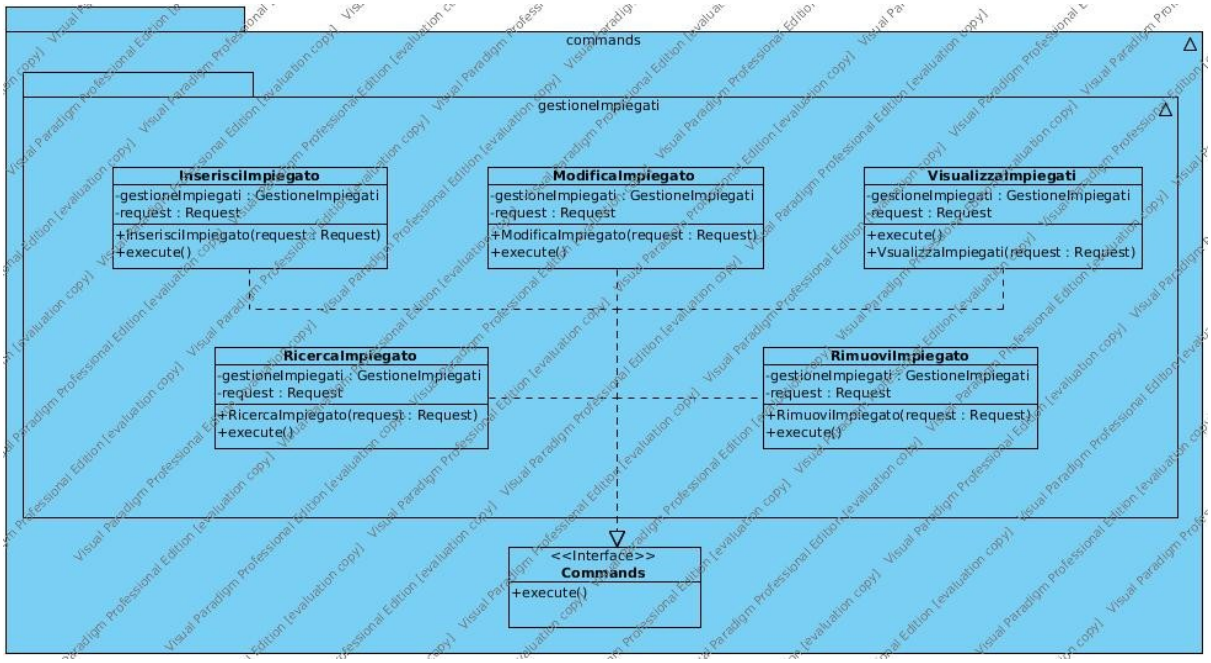
2.3.4 presentation.commands.gestioneContratti



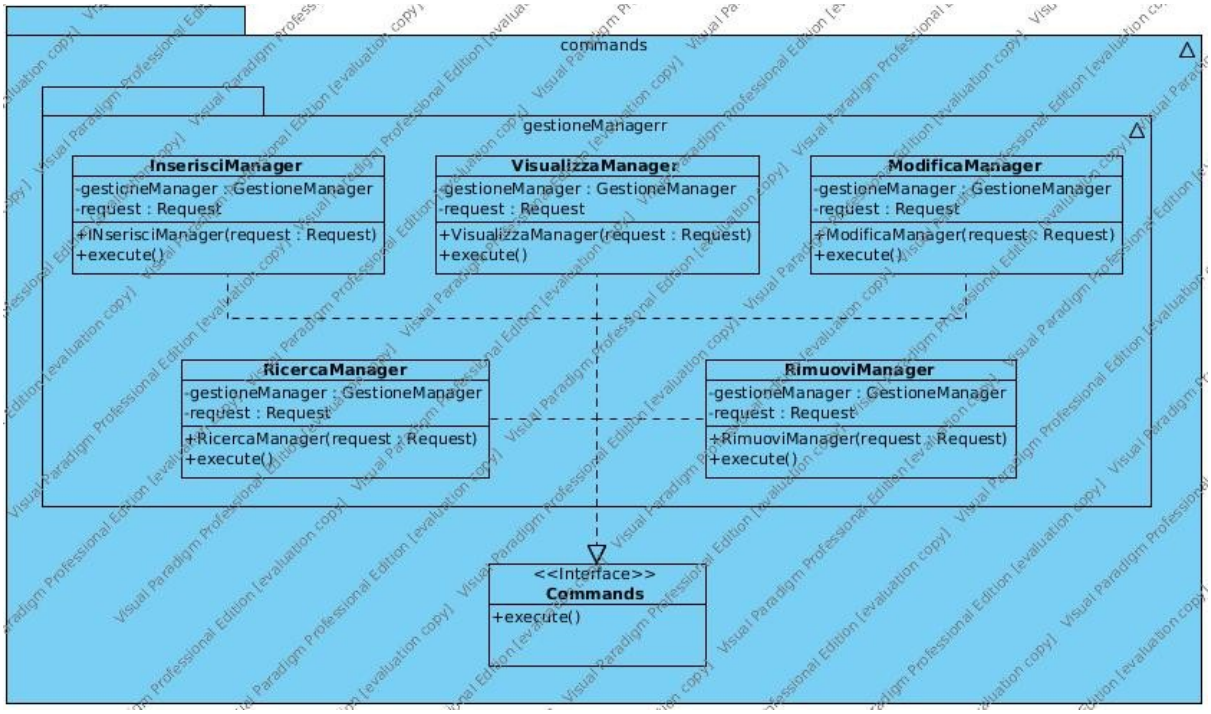
2.3.5 presentation.commands.gestioneFasce



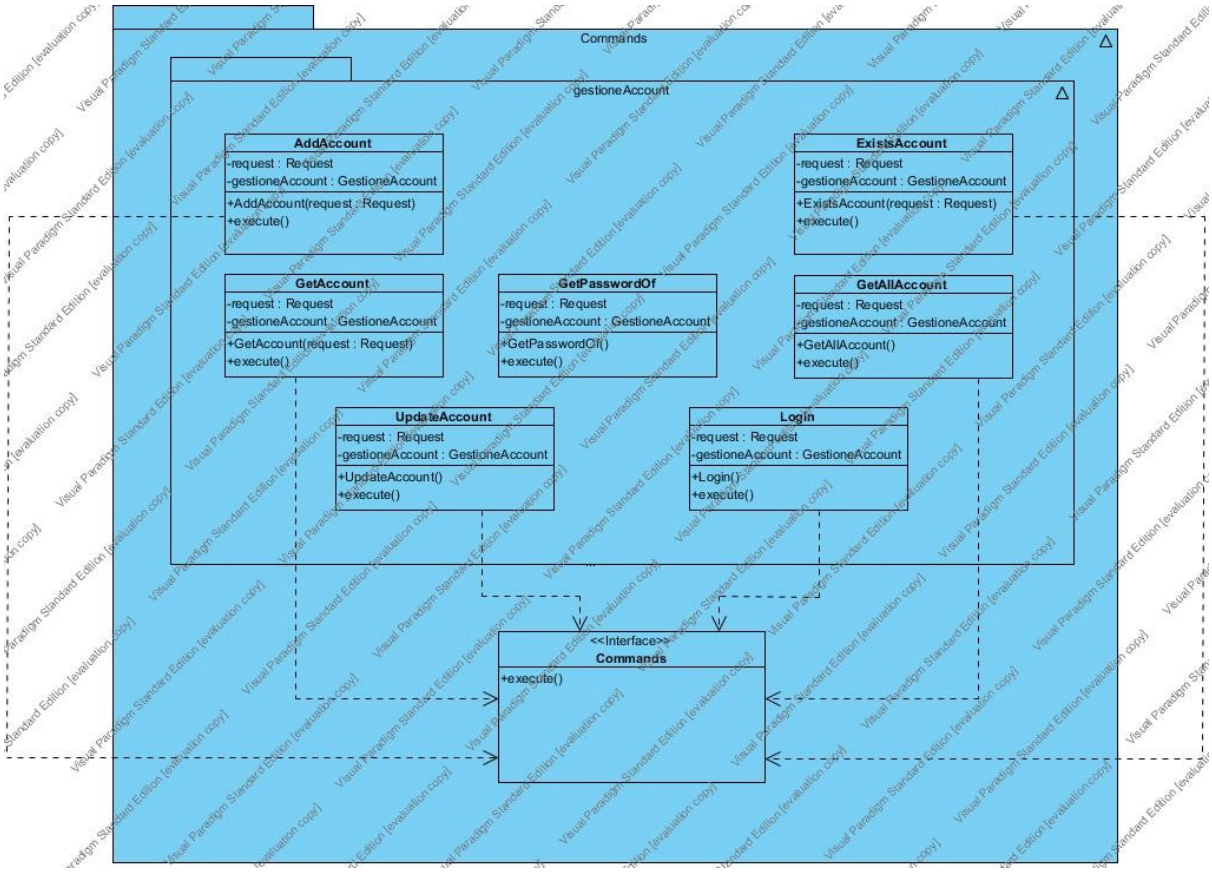
2.3.6 presentation.commands.gestioneImpiegati



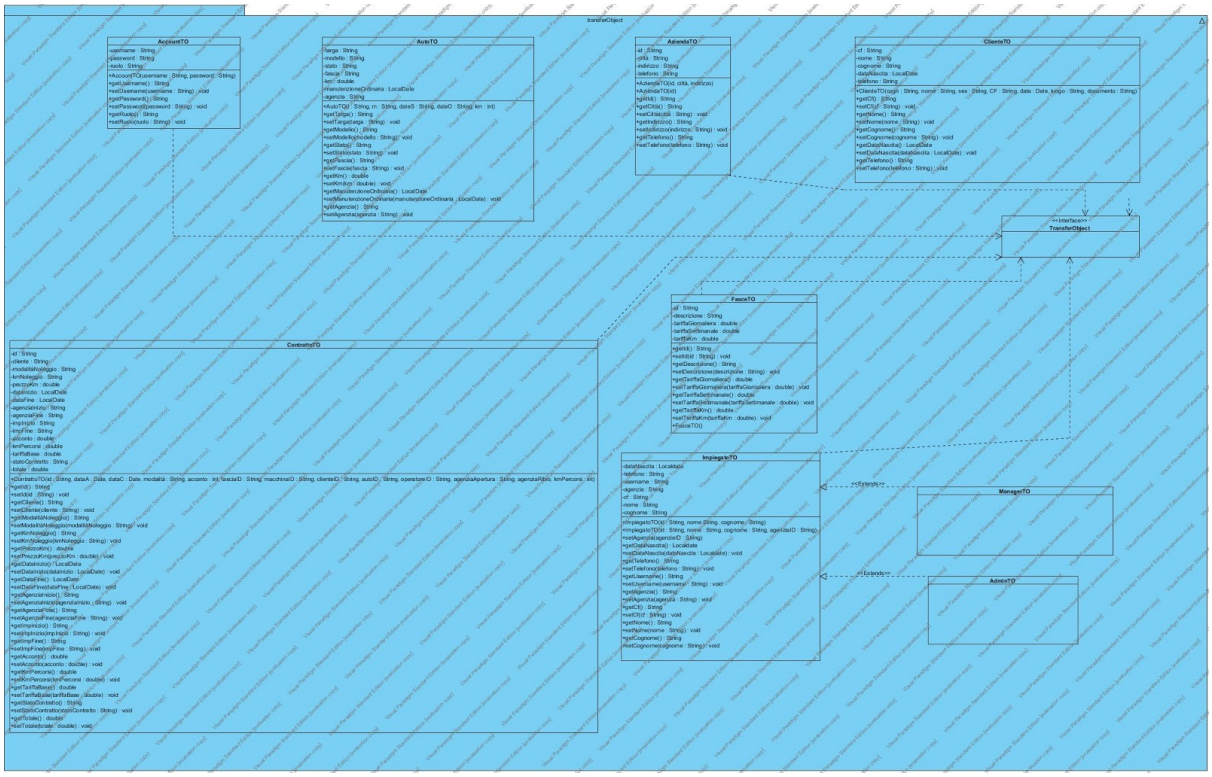
2.3.7 presentation.commands.gestioneManager



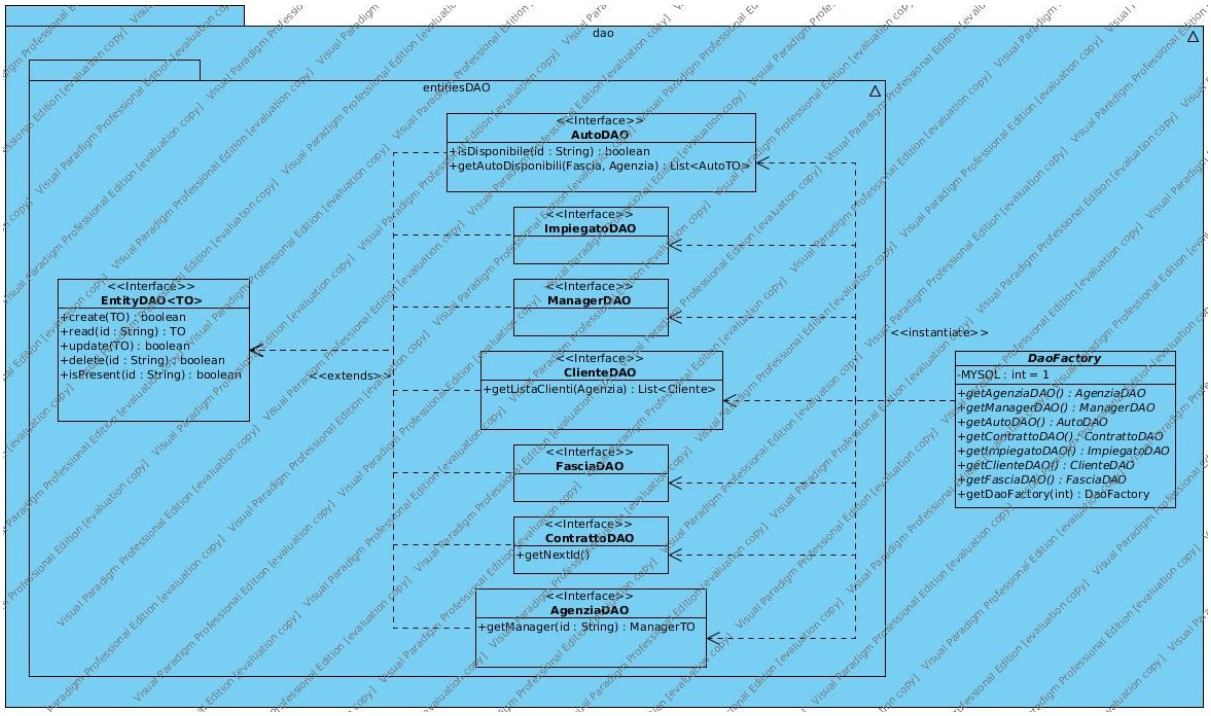
2.3.8 presentation.commans.gestioneAccount



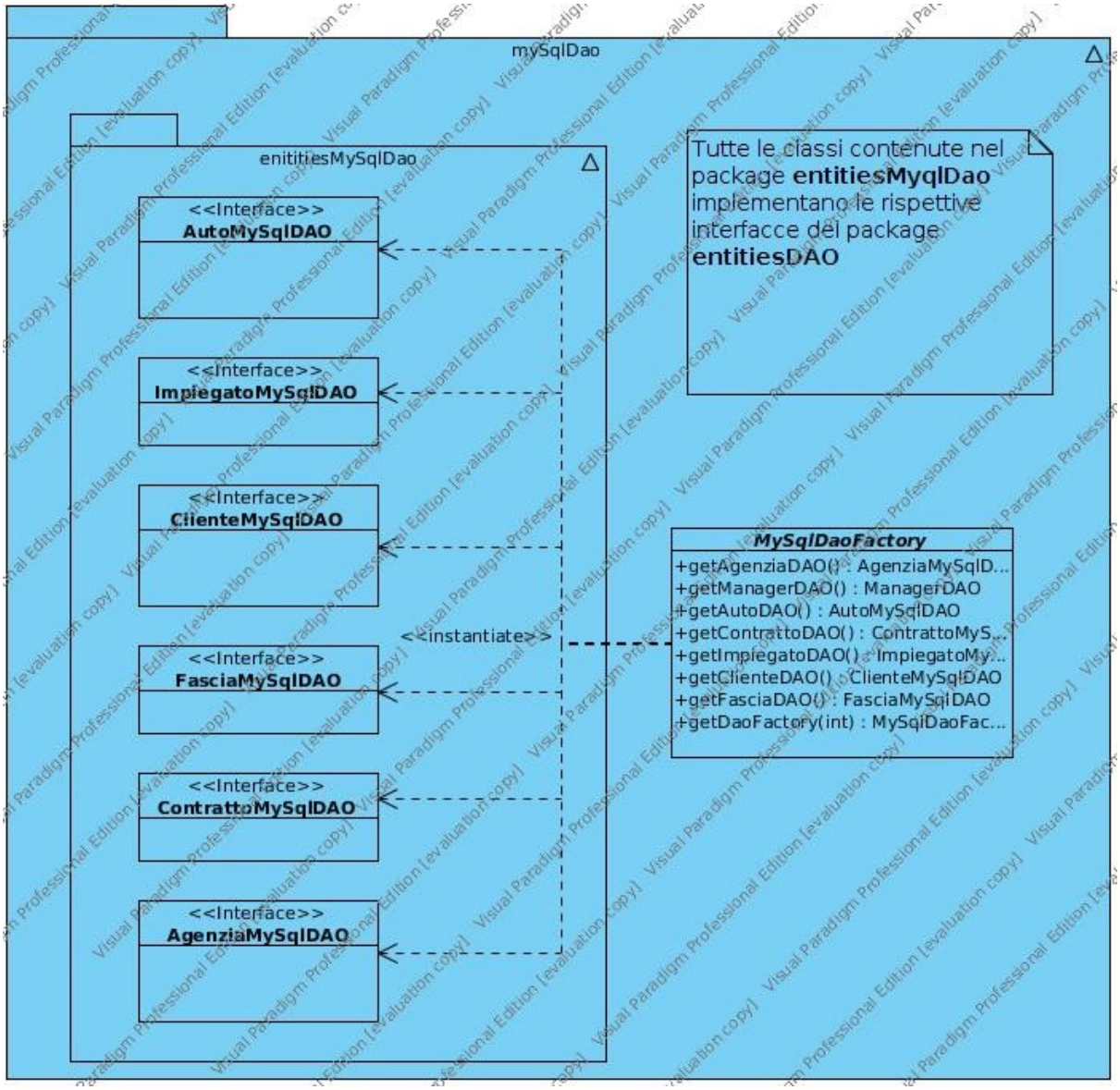
2.4 transferObjects



2.5.1 integration.dao



2.5.2 integration.mySqlDao

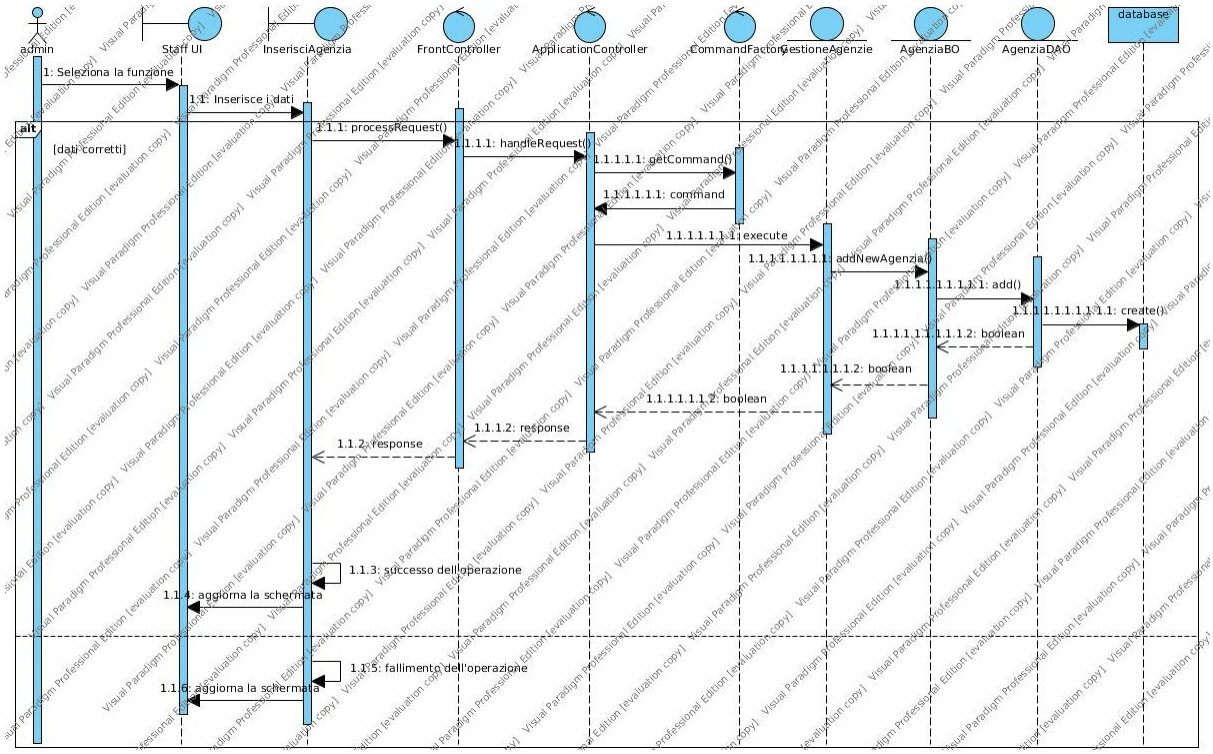


2.2 Specifiche delle Classi

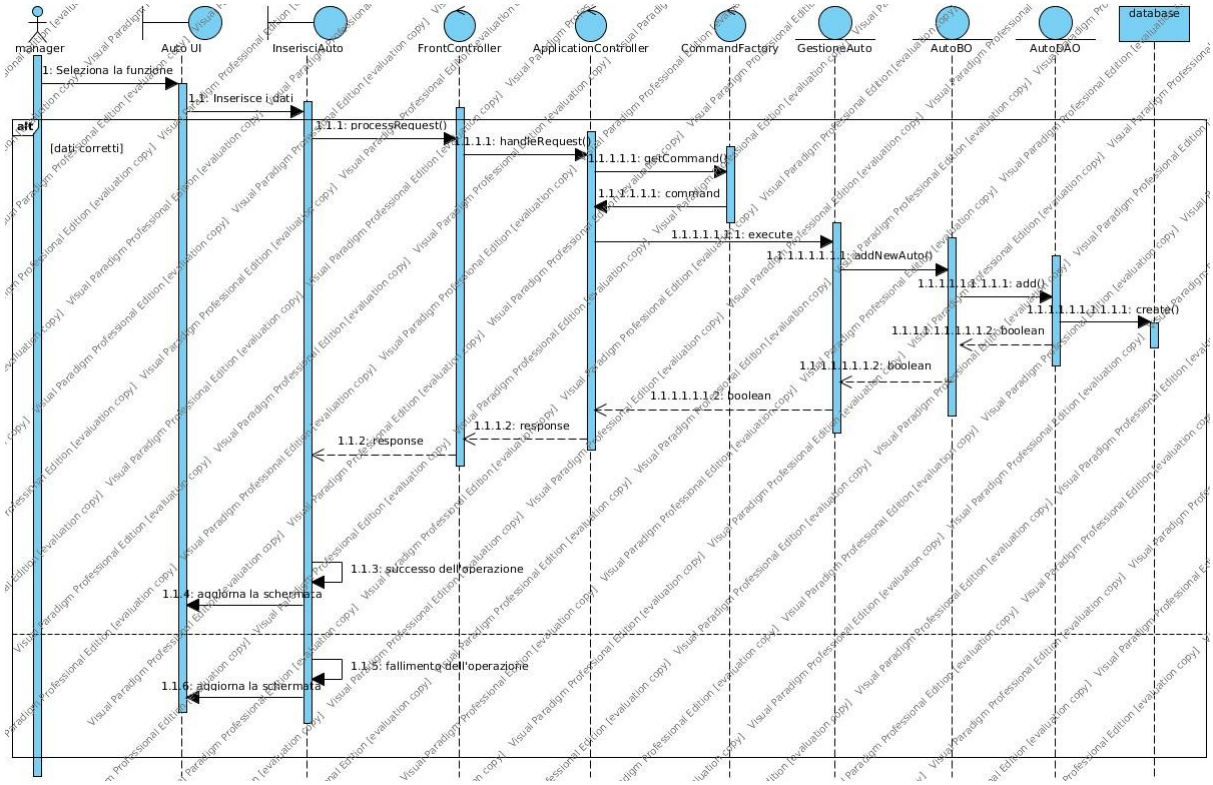
Vedere la cartella Progettazione/javadoc.

2.3 Diagrammi di Sequenza

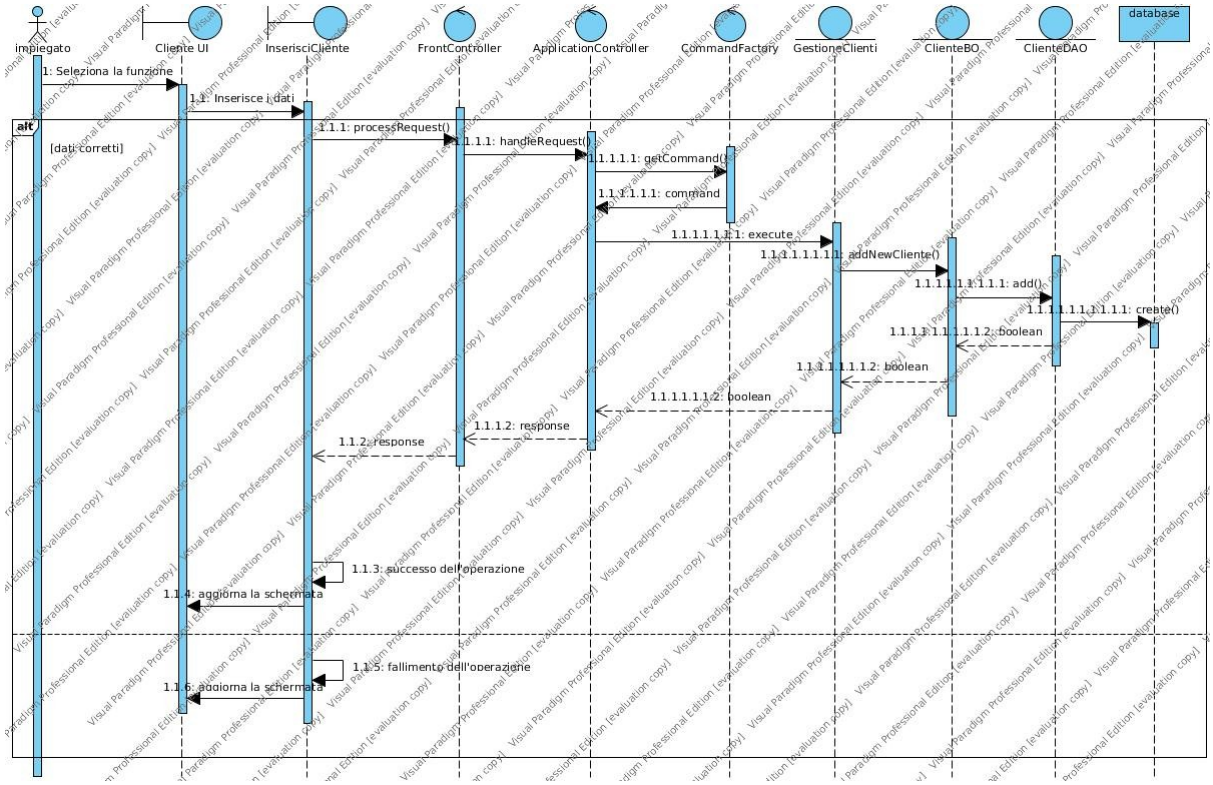
InserisciAgenzia



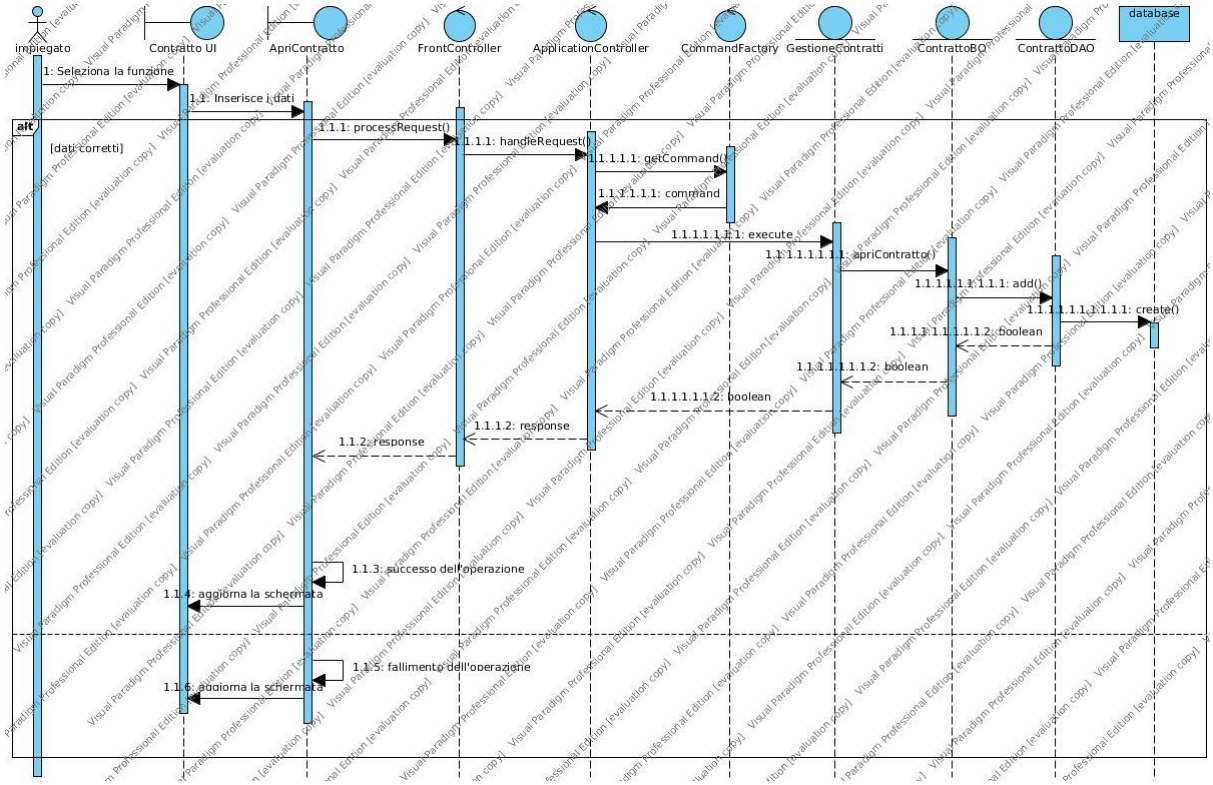
inserisciAuto



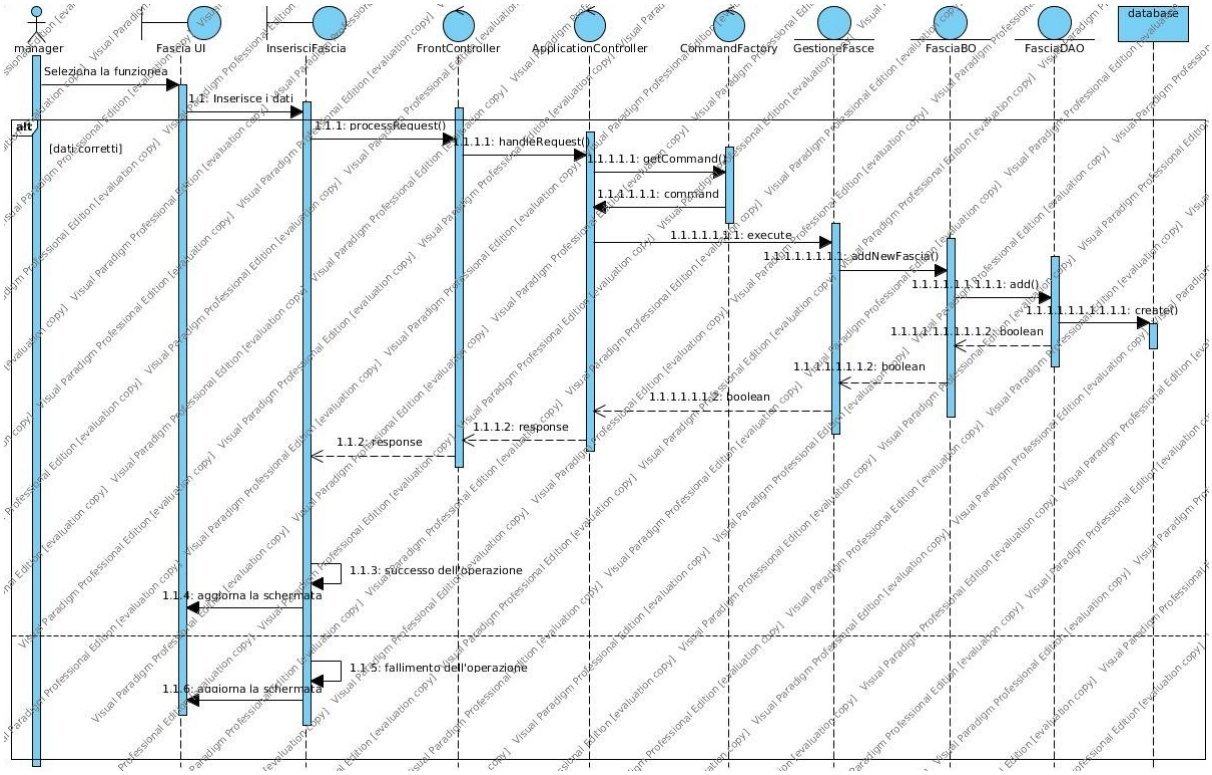
inserisciCliente



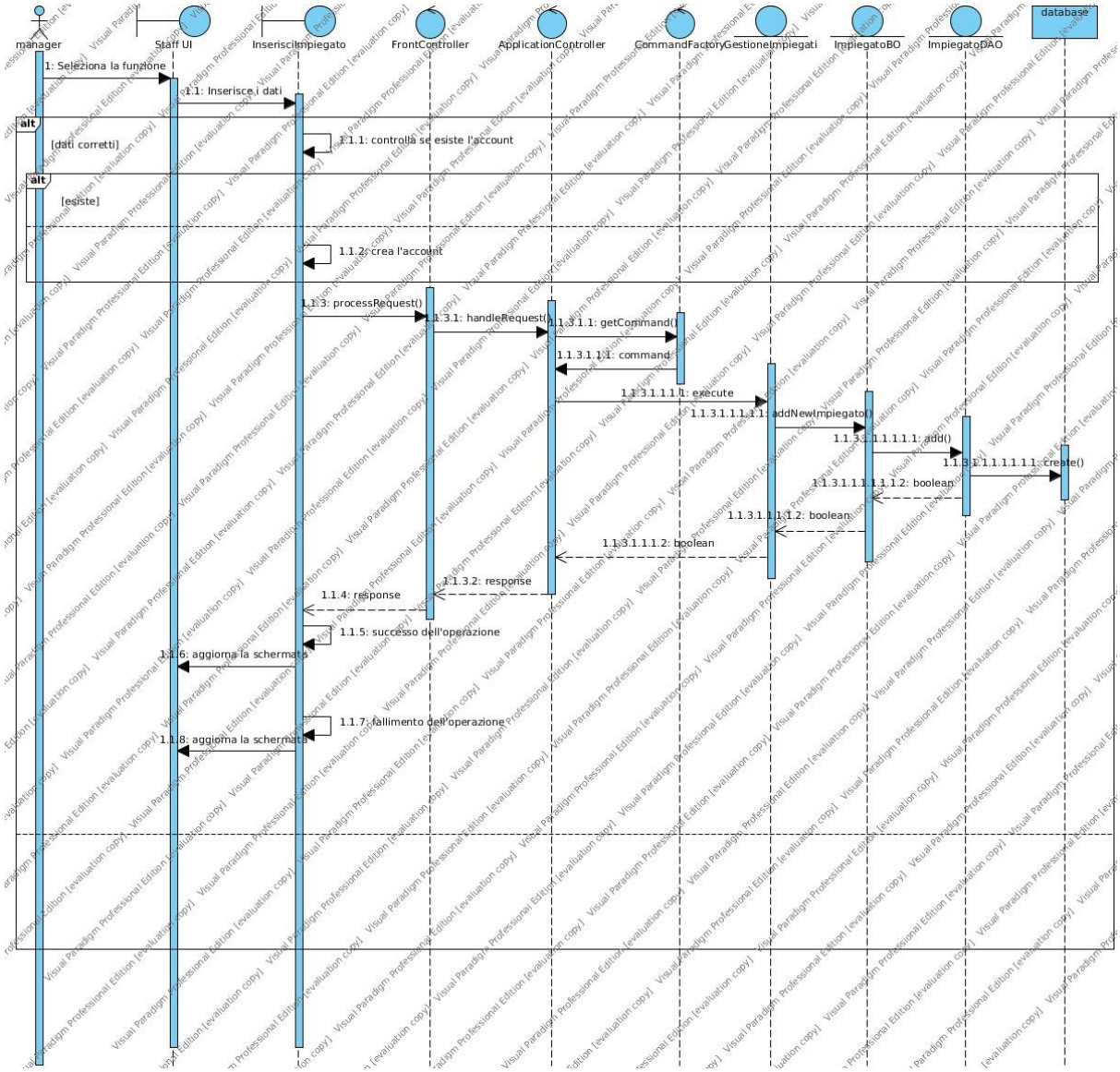
inseriscContratto



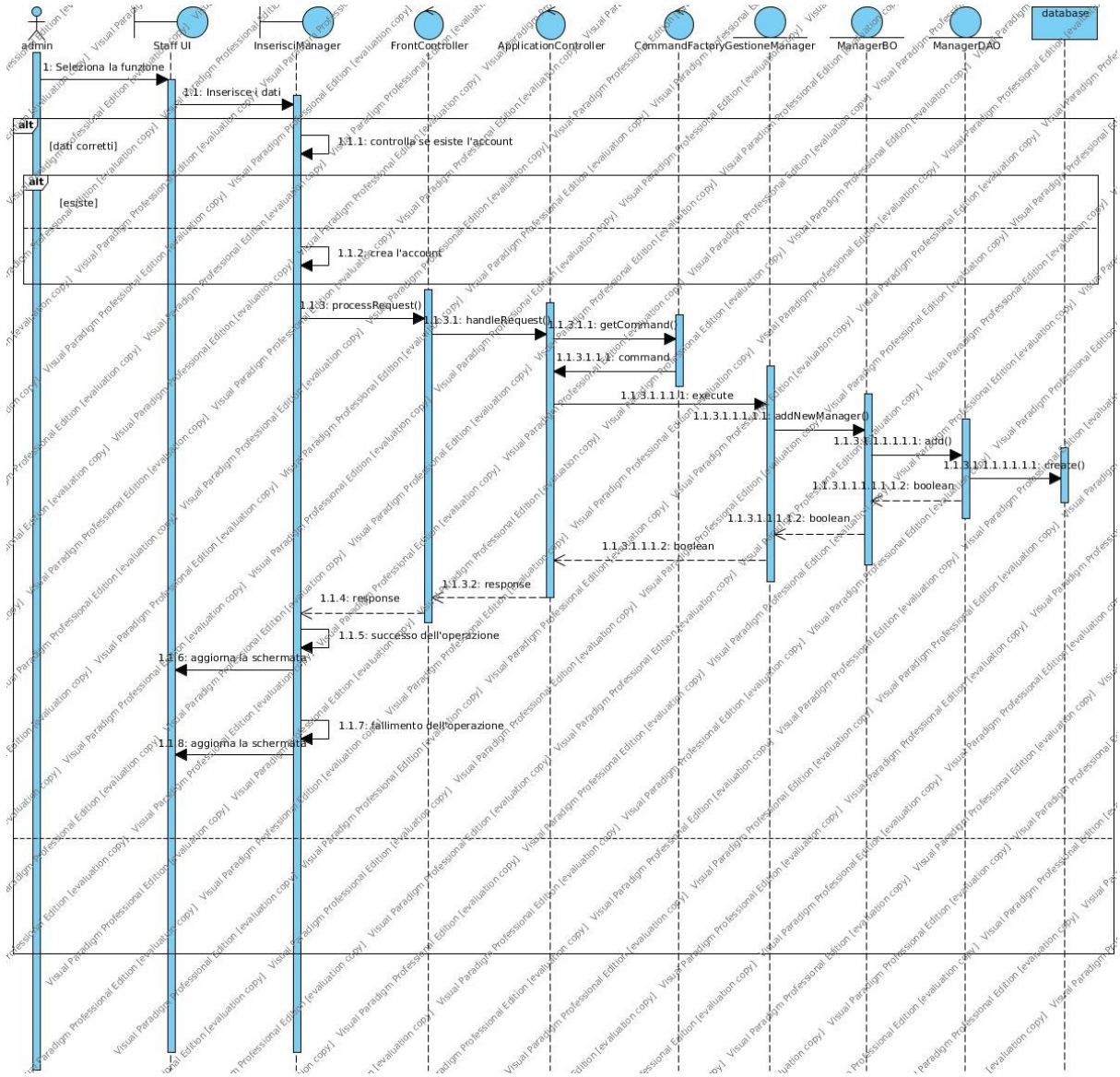
inserisciFascia



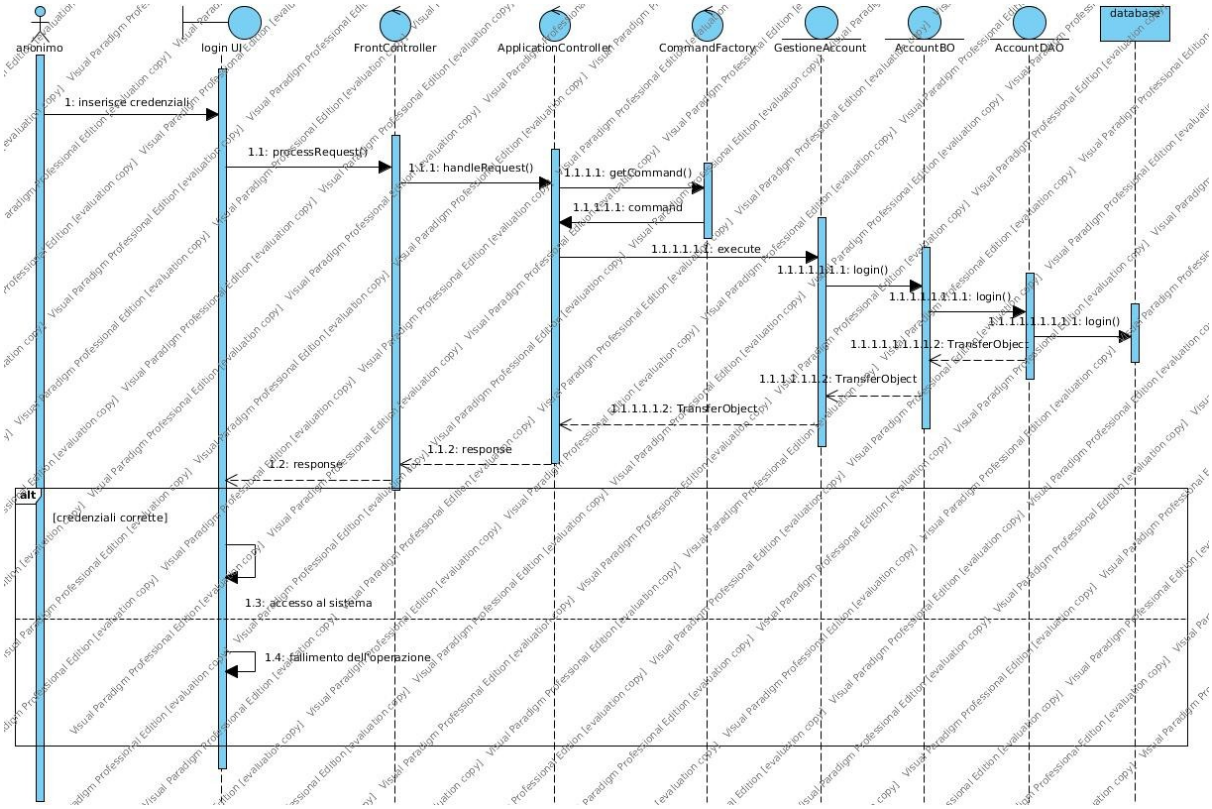
inserisciImpiegato



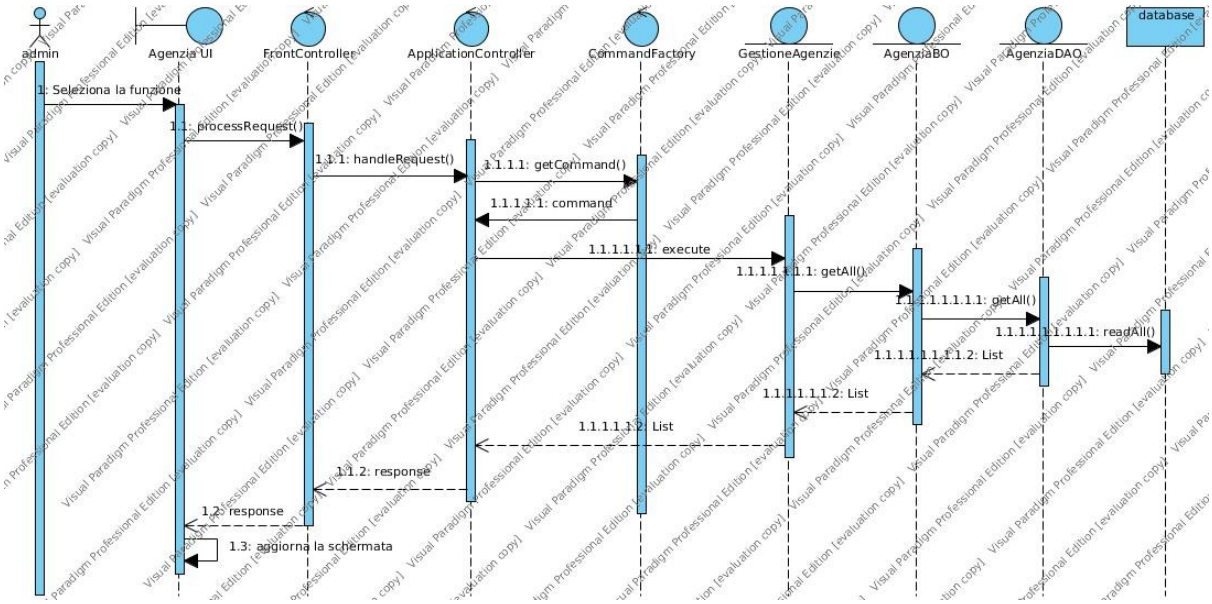
inserisciManager



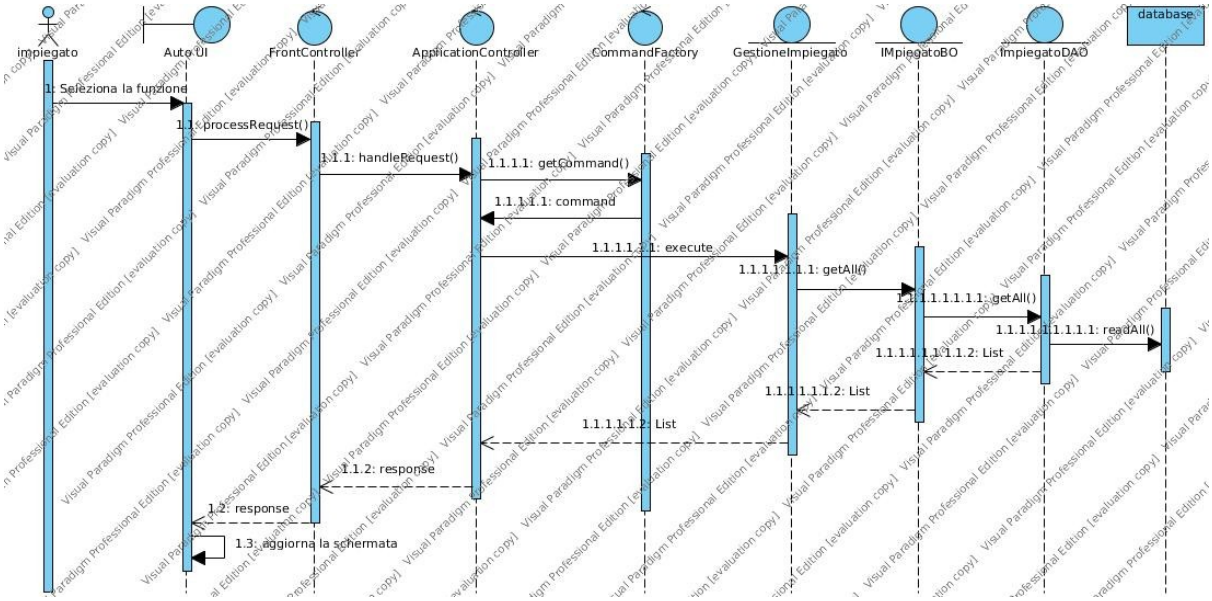
login



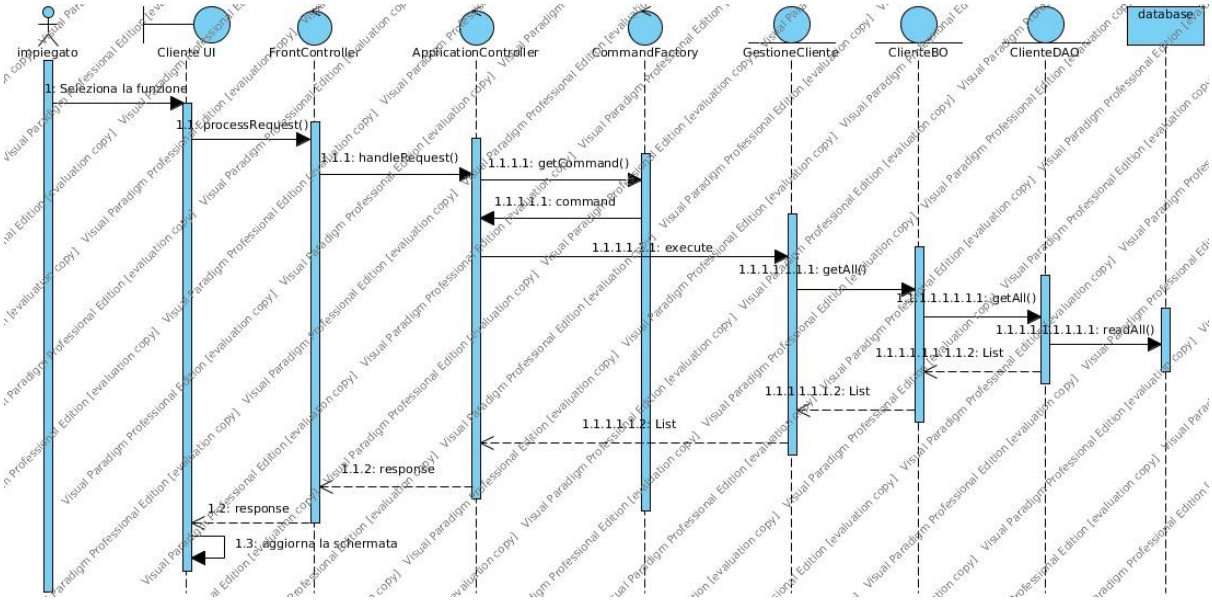
readAgenzia



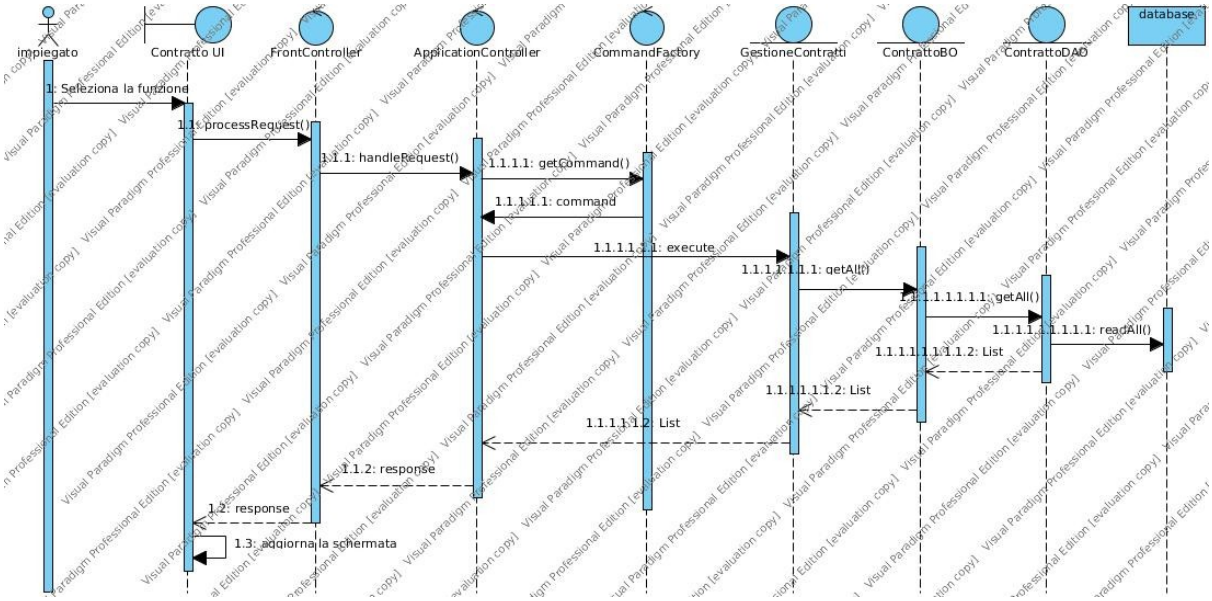
readAuto



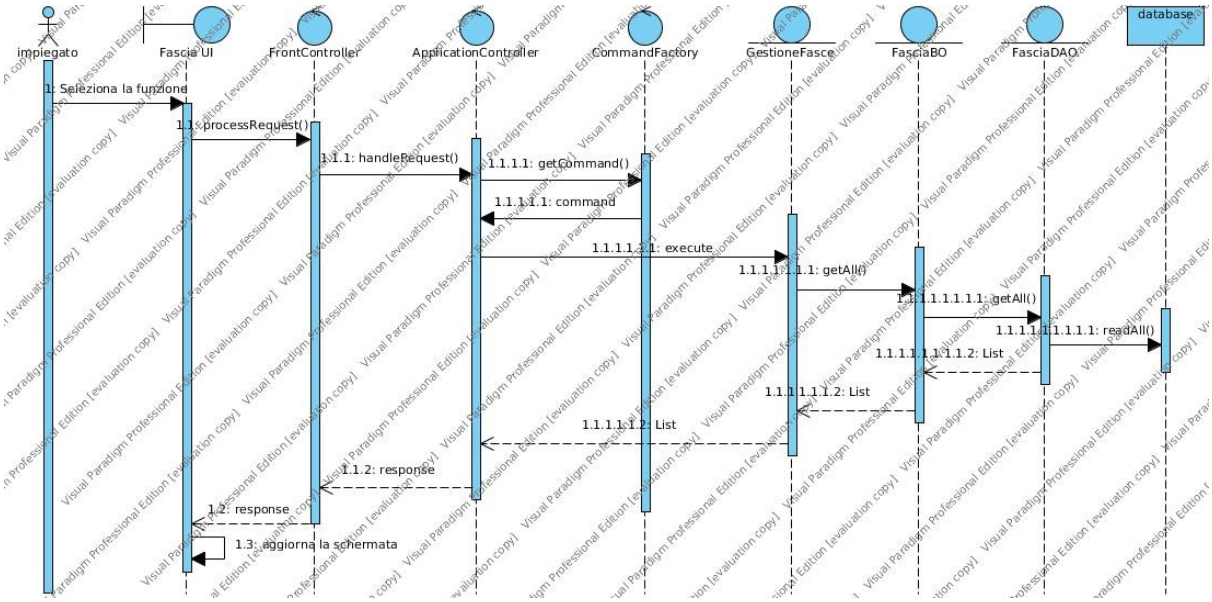
readCliente



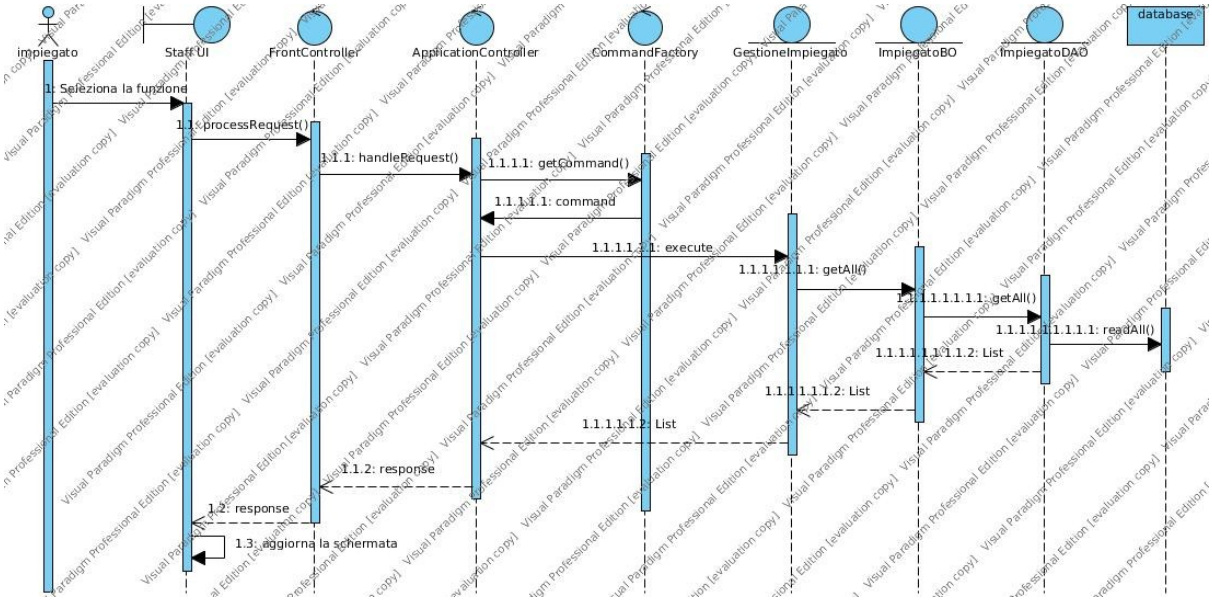
readContratto



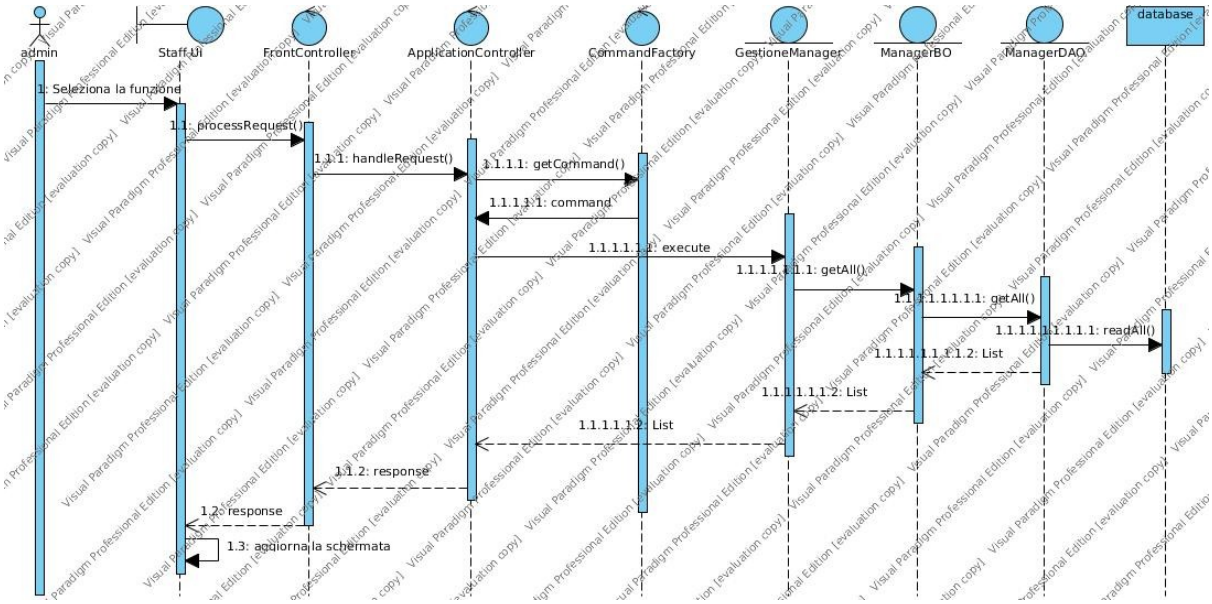
readFascia



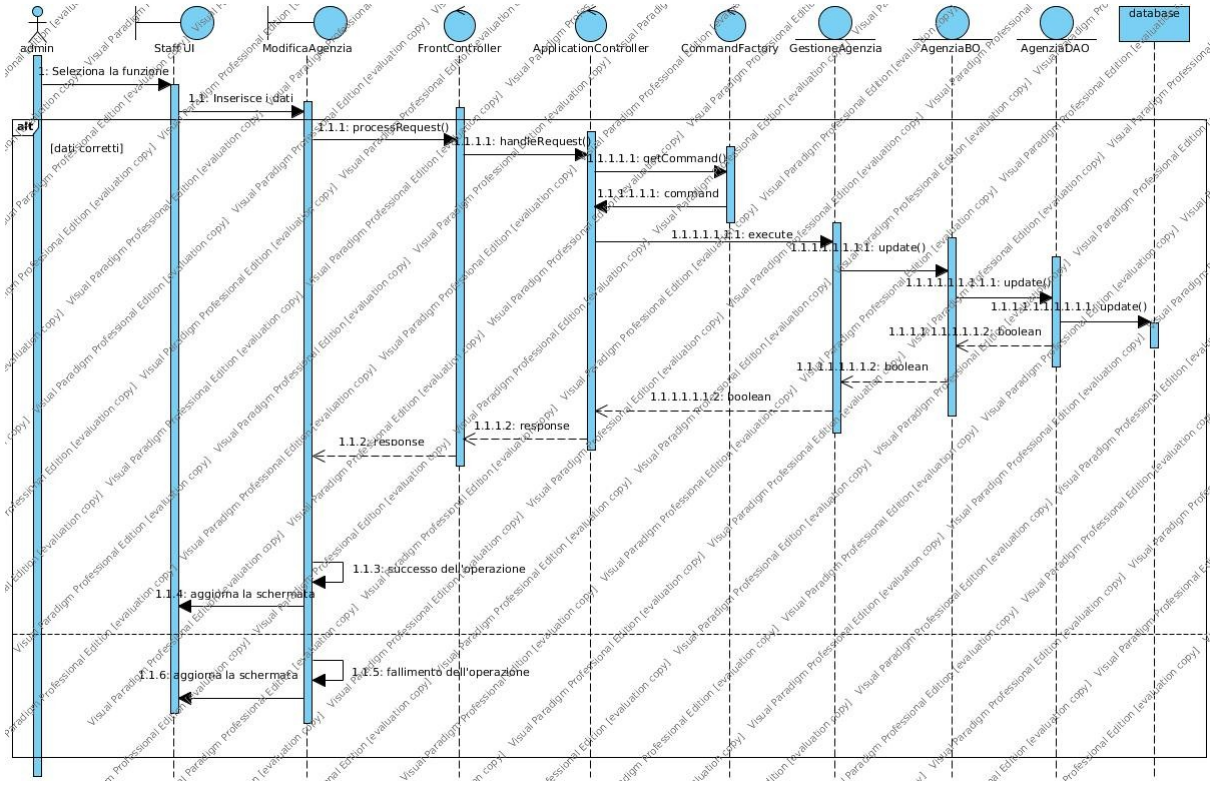
readImpiegato



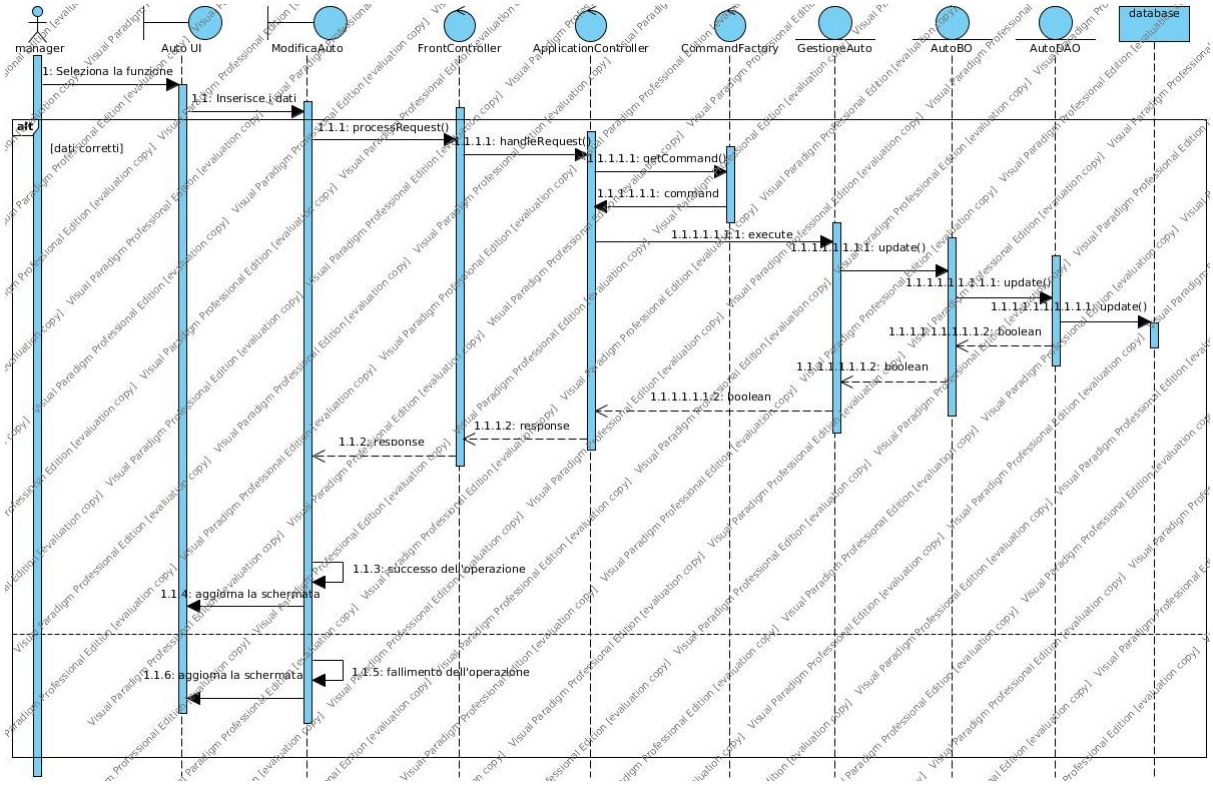
readManager



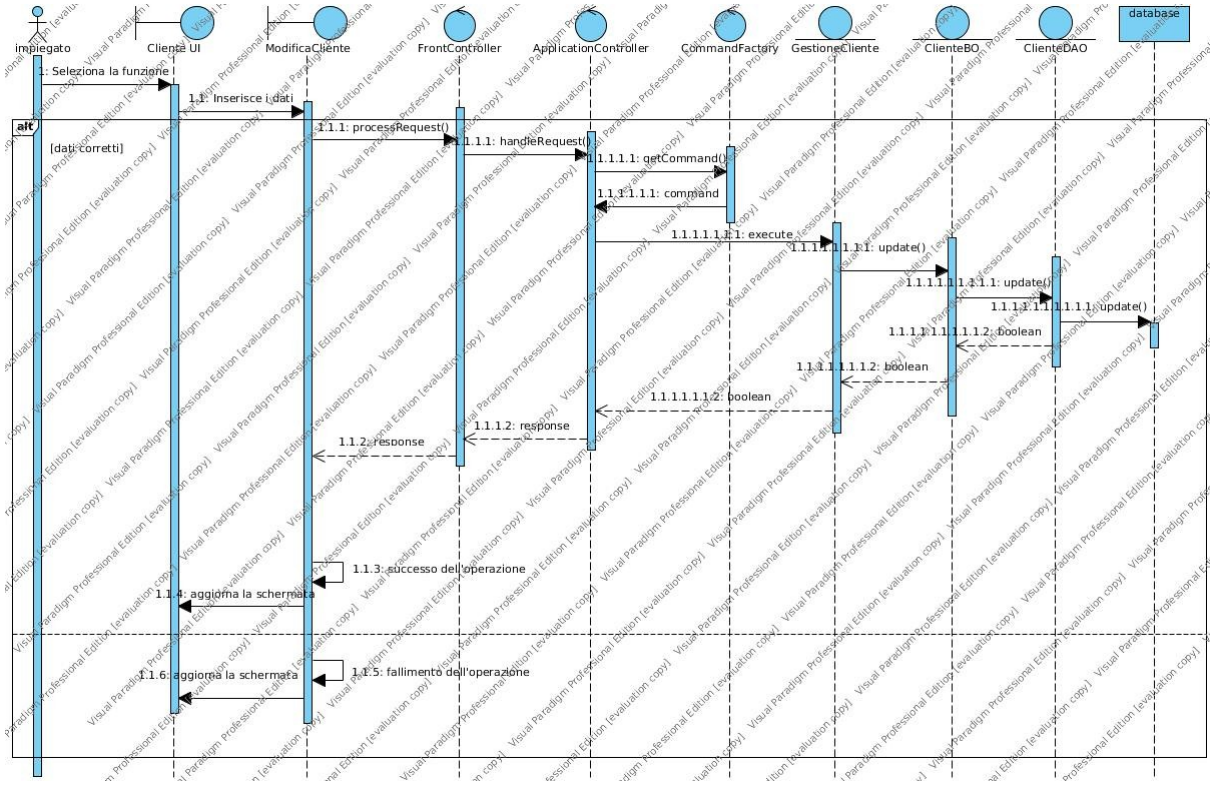
updateAgenzia



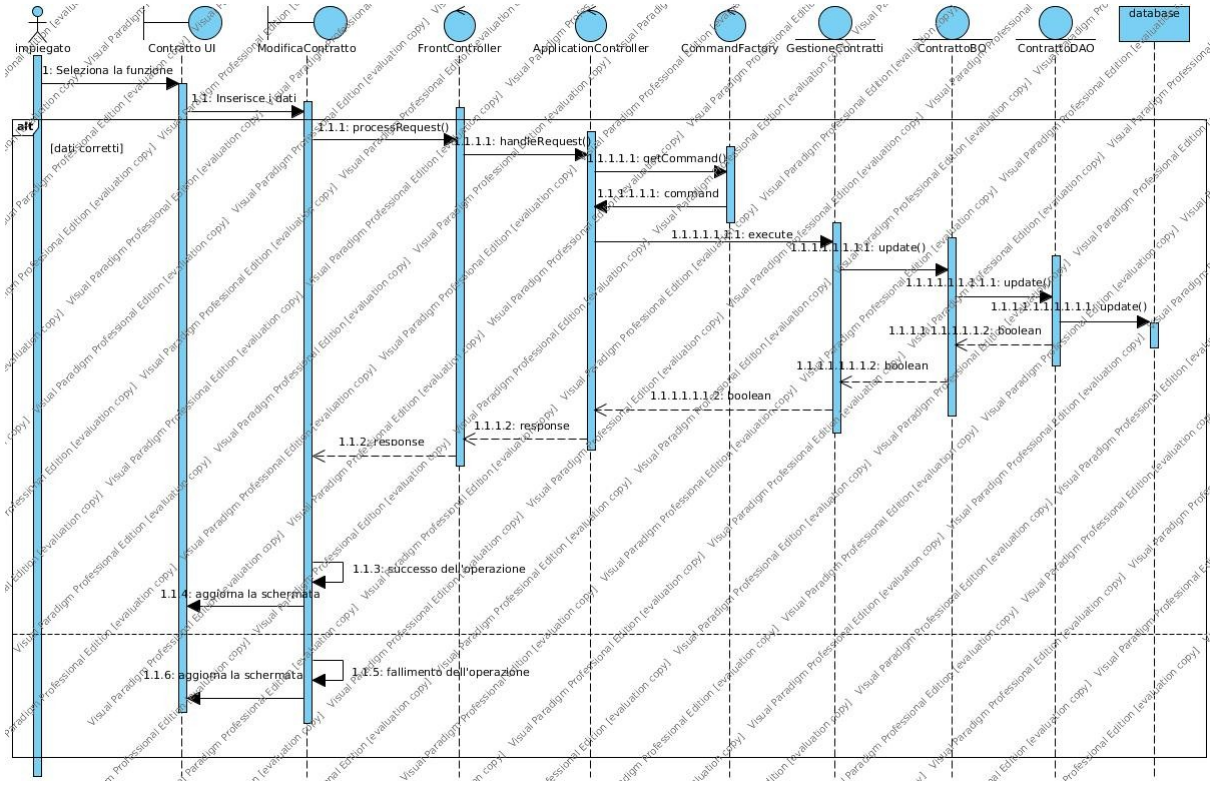
updateAuto



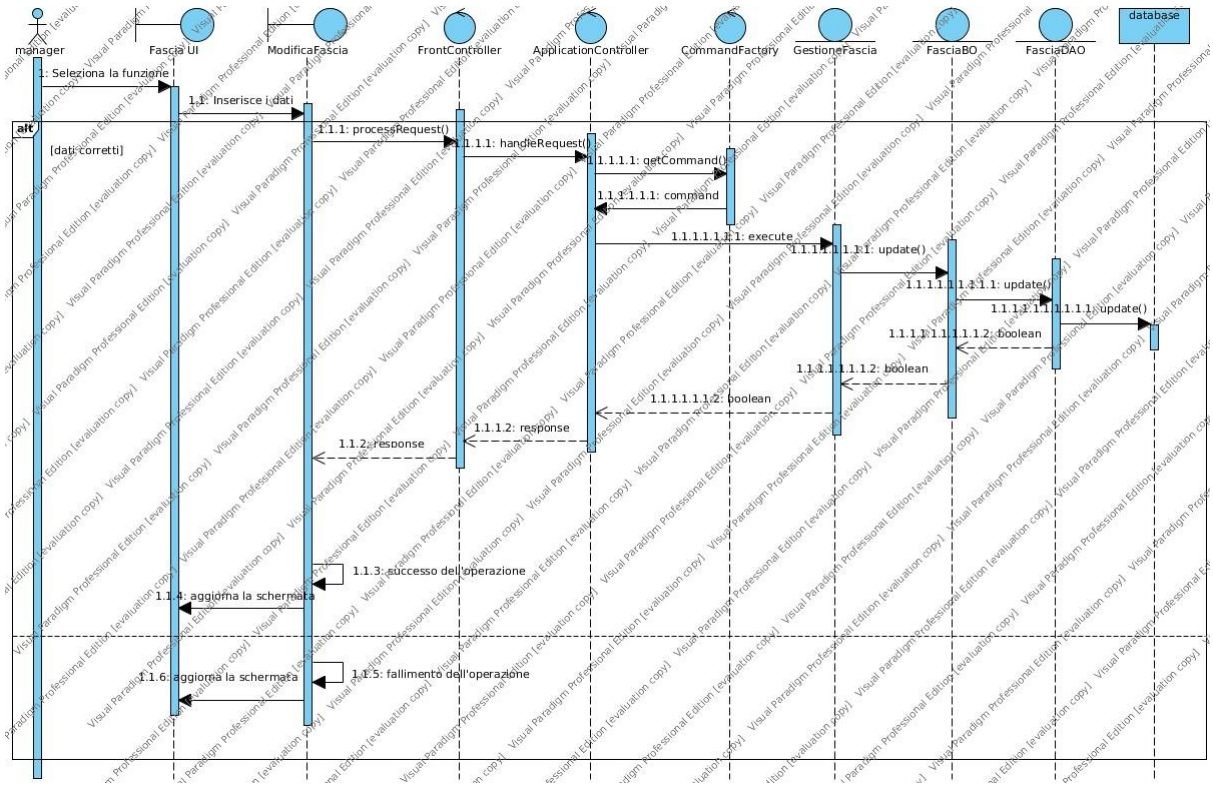
updateCliente



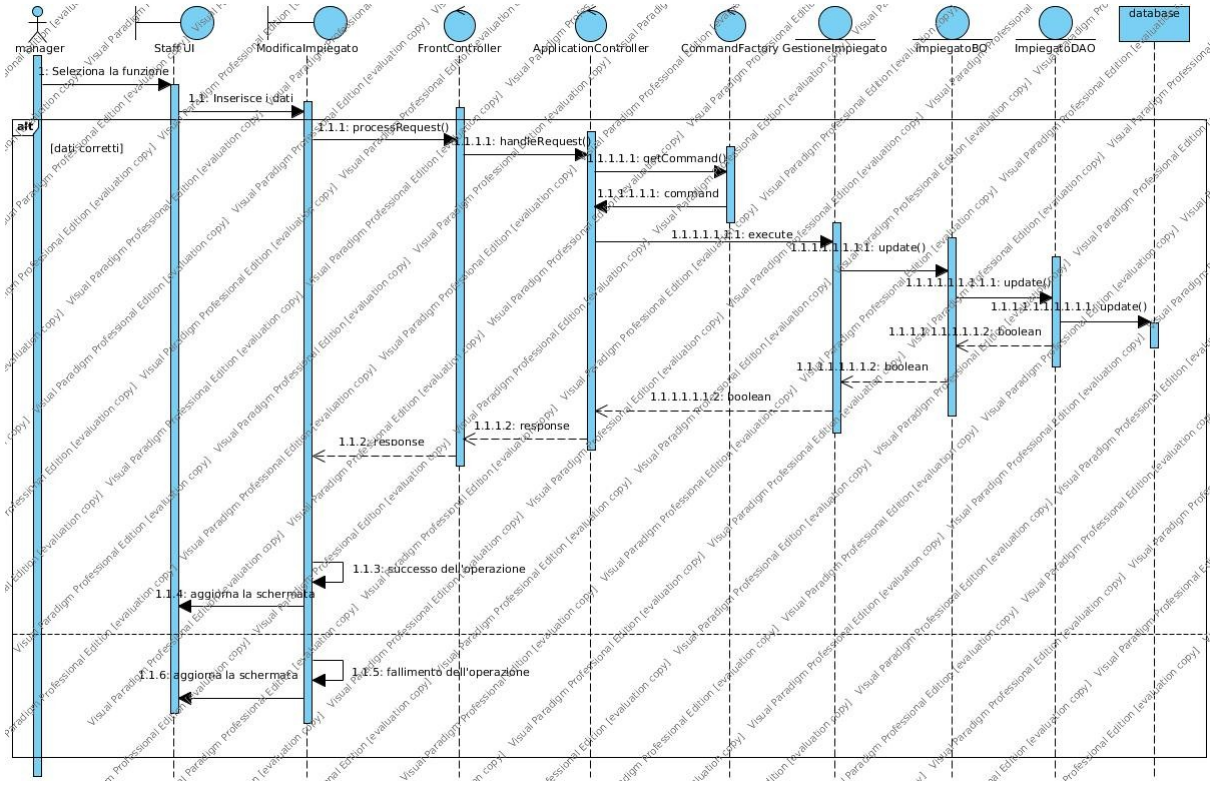
updateContratto



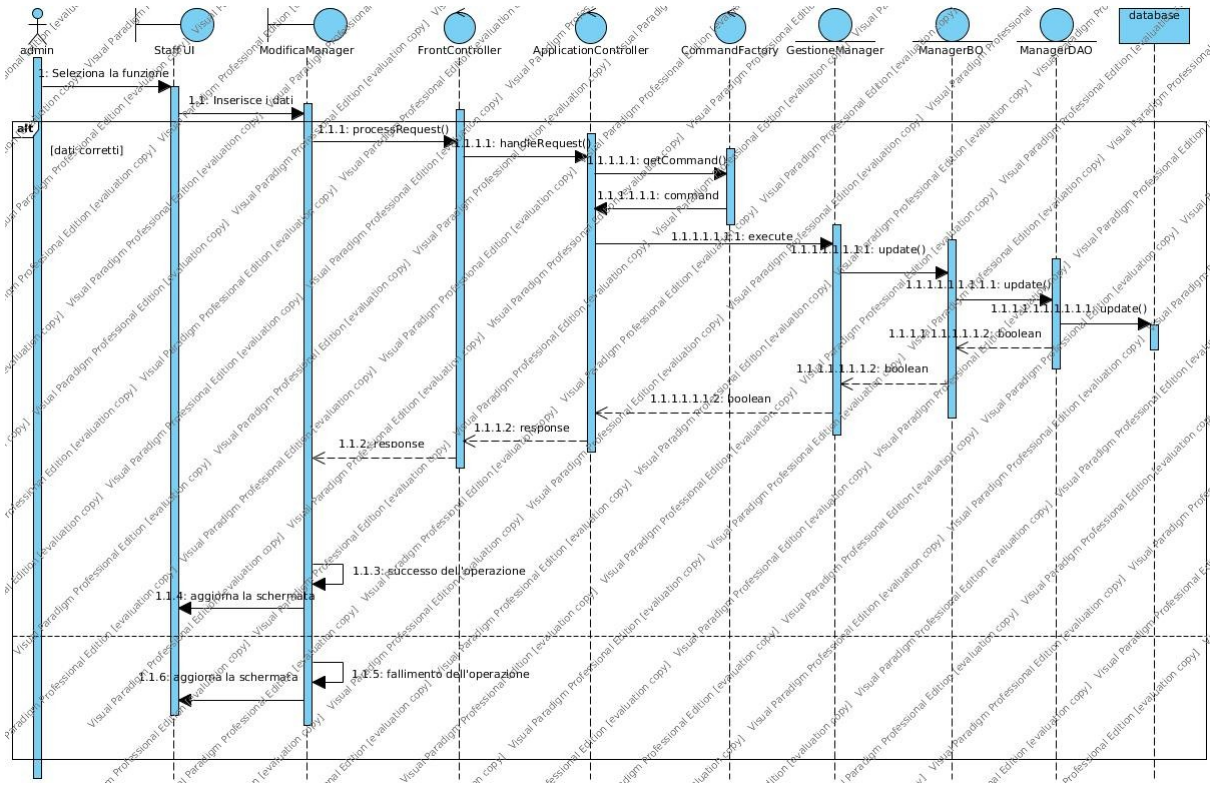
updateFascia



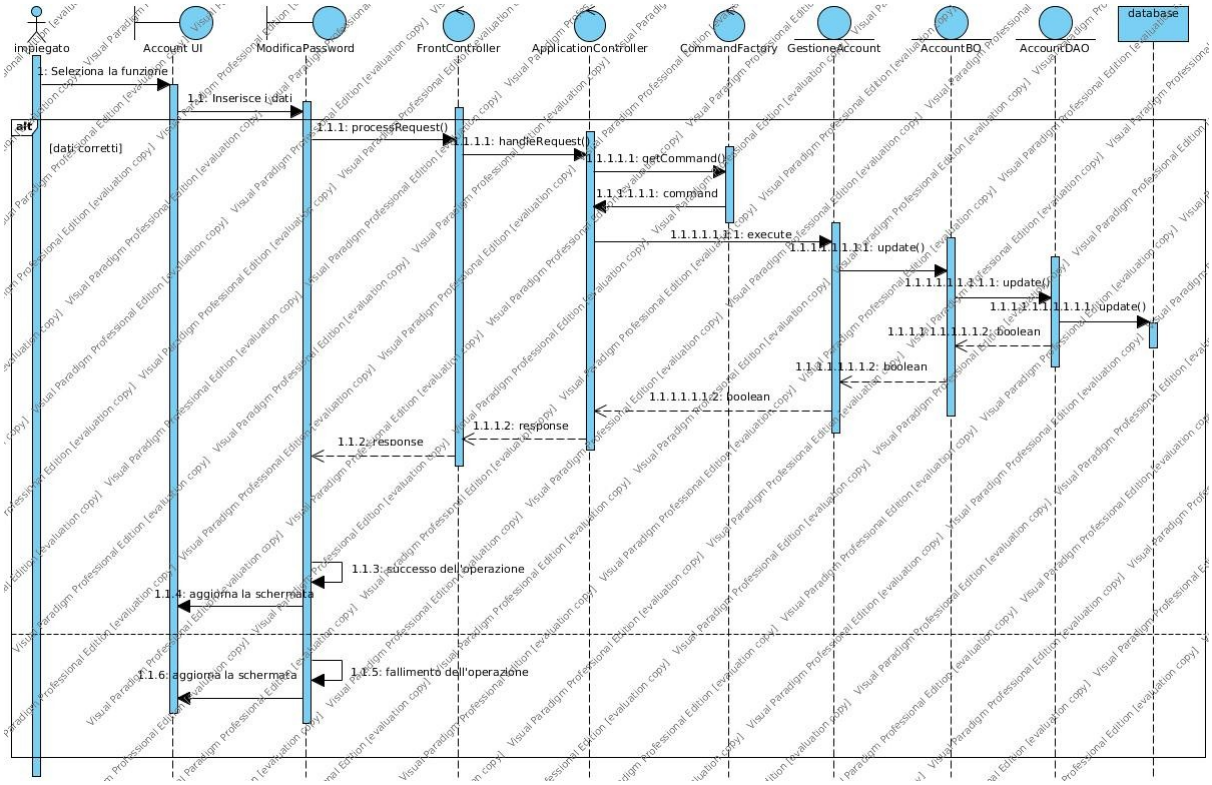
updateImpiegato



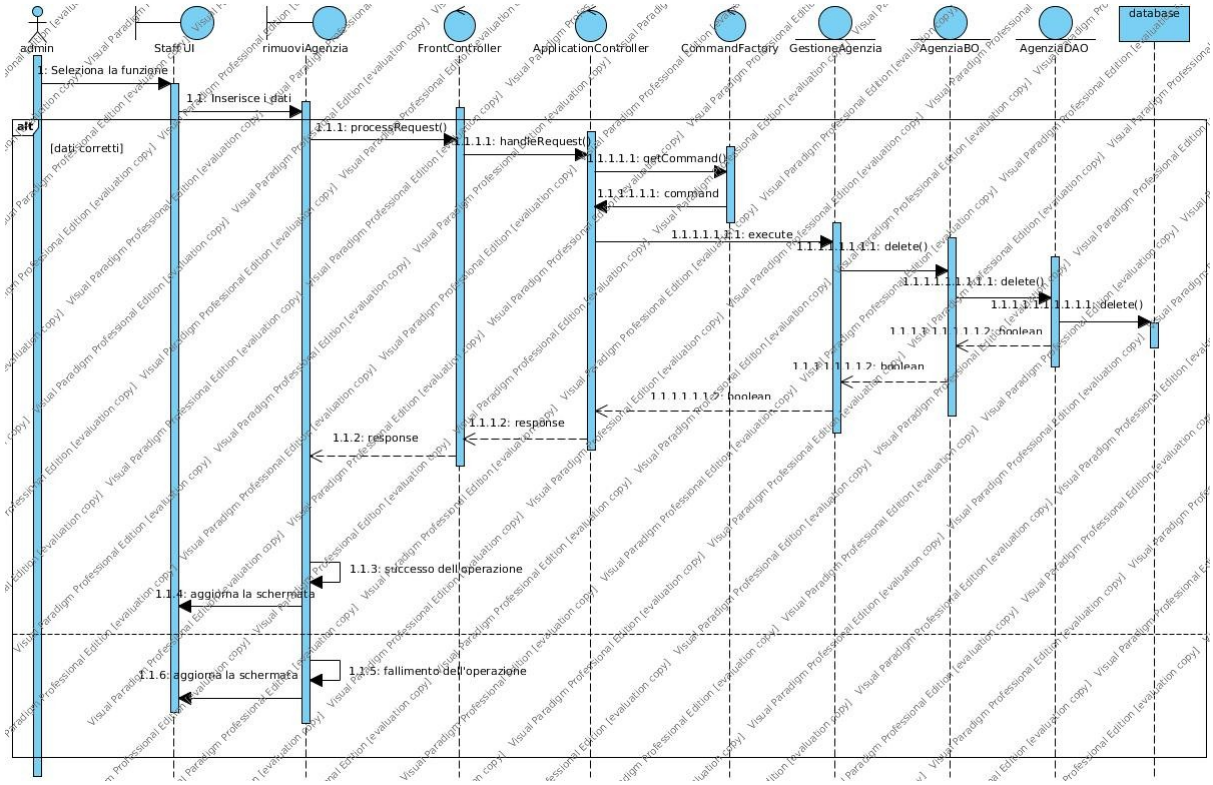
updateManager



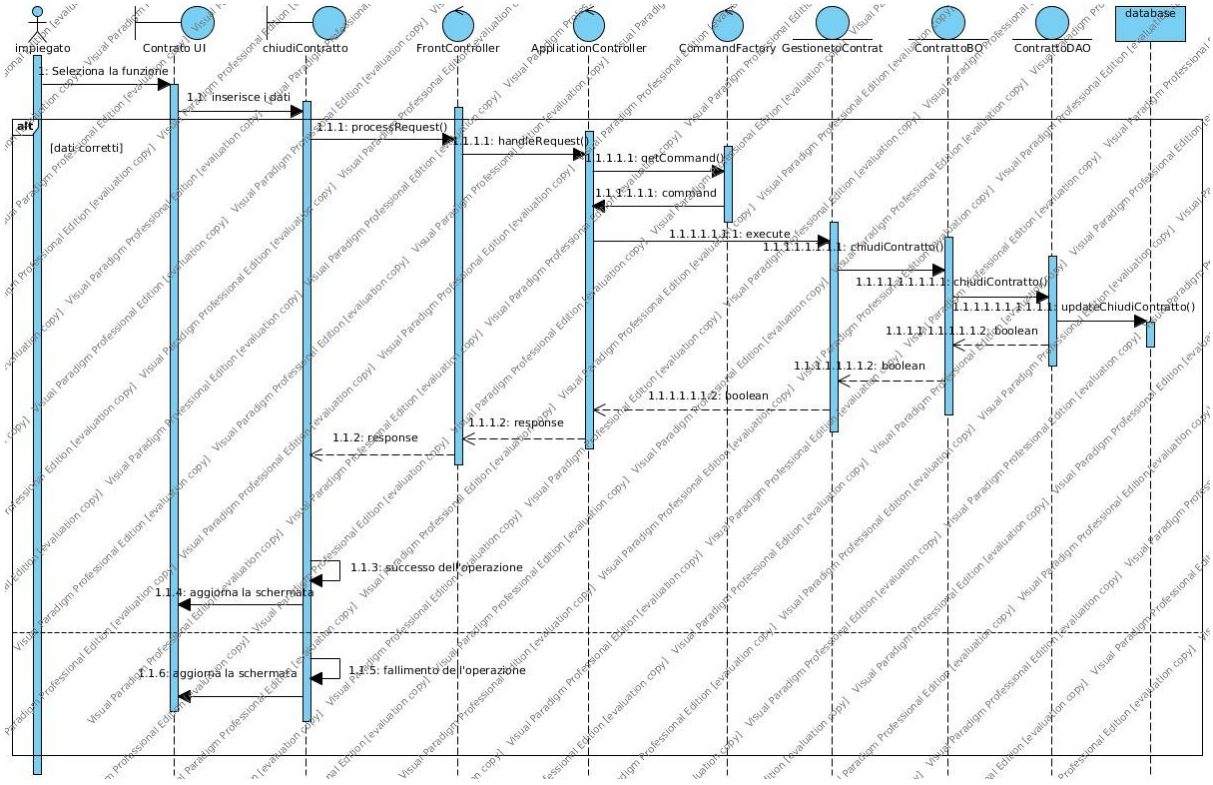
updatePassword



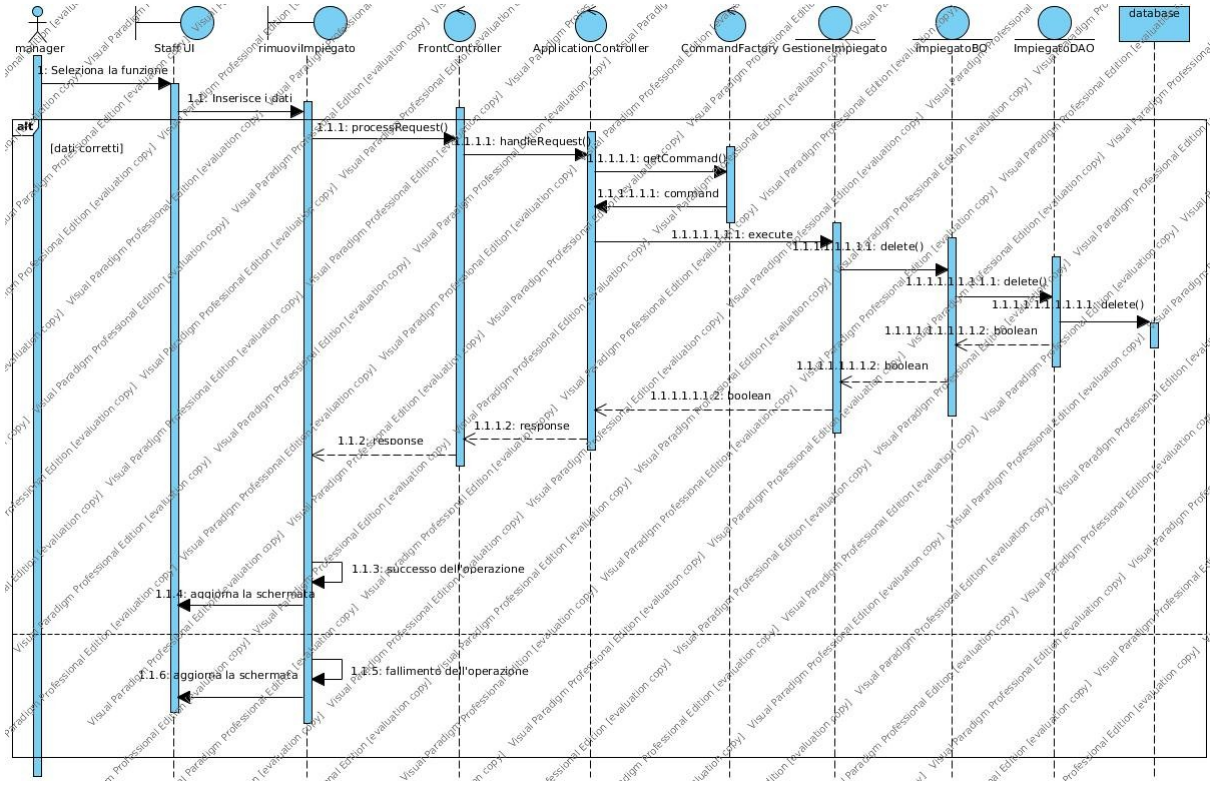
deleteAgenzia



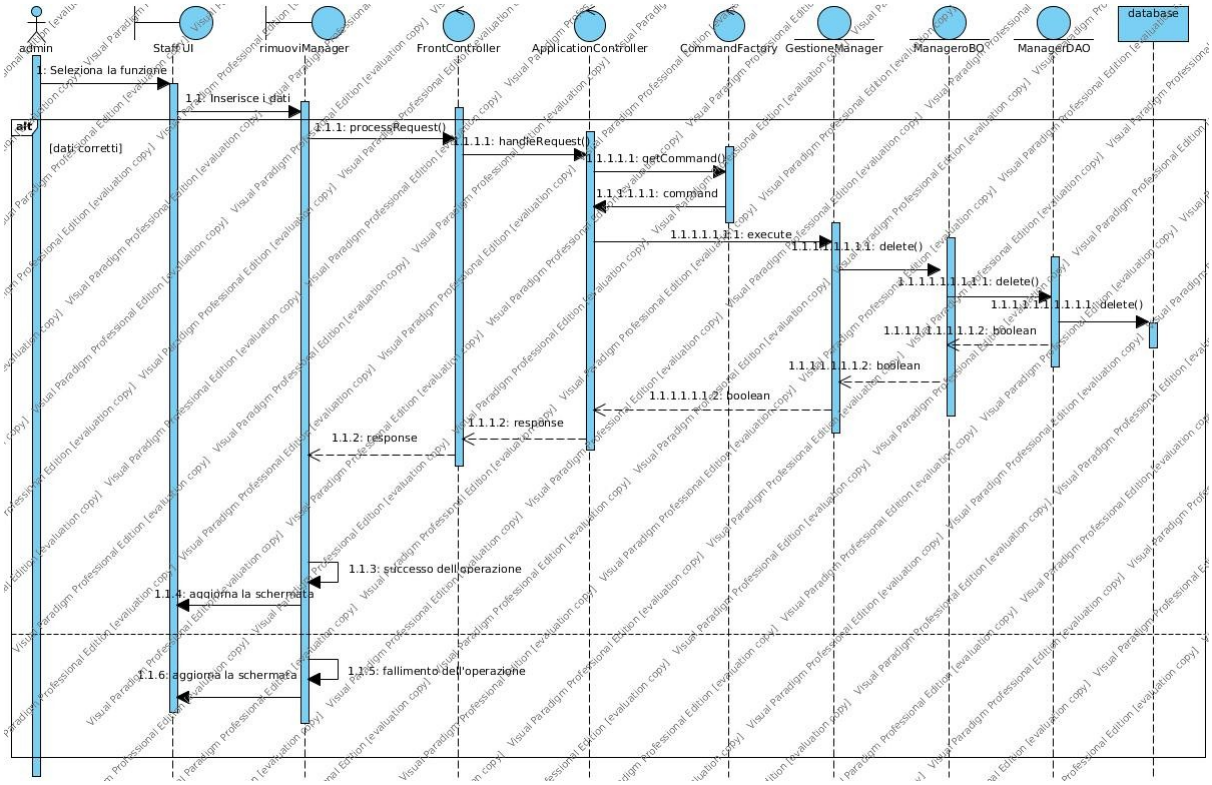
deleteContratto



deleteImpiegato



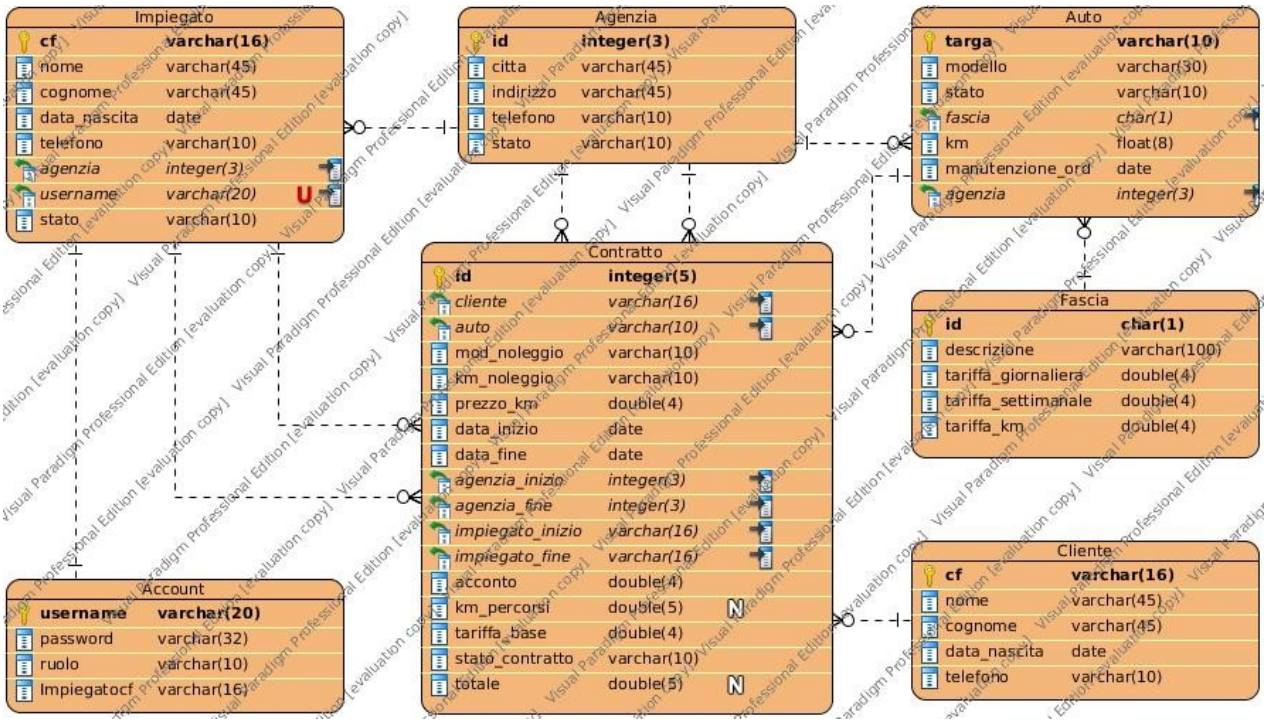
deleteManager



3. PROGETTO DEI DATI

3.1 Database

3.1.1 Modello del Database



3.1.2 Dettaglio dei Dati

Agenzia

Name	Type	Constraints	Nullable	Description
id	INT(3)	PK	NO	id agenzia
citta	VARCHAR(45)		NO	città agenzia
indirizzo	VARCHAR(45)		NO	indirizzo agenzia

telefono	VARCHAR(10)		NO	telefono agenzia
stato	ENUM(aperta, chiusa)		NO	stato agenzia

Account

Name	Type	Constraints	Nullable	Description
username	VARCHAR(20)	PK	NO	username di un impiegato
password	VARCHAR(32)		NO	password
ruolo	ENUM(admin, manager, impiegato)		NO	ruolo

Fascia

Name	Type	Constraints	Nullable	Description
id	CHARACTER(1)	PK	NO	id fascia
descrizione	VARCHAR(100)		NO	descrizione fascia
tariffa_giornaliera	DOUBLE(4,2)		NO	tariffa giornaliera
tariffa_settimanale	DOUBLE(4, 2)		NO	tariffa settimanale

tariffa_km	DOUBLE(4, 2)		NO	tariffa al chilometro

Cliente

Name	Type	Constraints	Nullable	Description
cf	VARCHAR(16)	PK	NO	codice fiscale
nome	VARCHAR(45)		NO	nome cliente
cognome	VARCHAR(45)		NO	cognome cliente
data_nascita	DATE		NO	data di nascita
telefono	VARCHAR(10)		NO	telefono

Auto

Name	Type	Constraints	Nullable	Description
targa	VARCHAR(10)	PK	NO	targa auto
modello	VARCHAR(30)		NO	modello auto
stato	ENUM(D, M, N)		NO	stato auto
fascia	CHARACTER	FK	NO	fascia auto

	(1)			
km	FLOAT(8, 2)		NO	chilometraggio auto
manutenzione_ord	DATE		NO	prossima data di manutenzione
agenzia	INT(3)	FK	NO	agenzia in cui si trova l'auto

Impiegato

Name	Type	Constraints	Nullable	Description
cf	VARCHAR(16)	PK	NO	codice fiscale
nome	VARCHAR(45)		NO	nome impiegato
cognome	VARCHAR(45)		NO	cognome impiegato
data_nascita	DATE		NO	data di nascita
telefono	VARCHAR(10)		NO	telefono
agenzia	INT(3)	FK	NO	agenzia di appartenenza
userna	VARCHAR(20)	FK	NO	username

me				
stato	ENUM(attivo, licenziato)		NO	stato

Contratto

Name	Type	Constraints	Nullable	Description
id	INT(5)	PK	NO	id contratto
cliente	VARCHAR(16)	FK	NO	cliente che ha noleggiato l'auto
auto	VARHCAR(10)	FK	NO	auto noleggiata
mod_noleggio	ENUM(giornaliero, settimanale)		NO	modalità di noleggio
km_noleggio	ENUM(limitato, illimitato)		NO	km di noleggio
prezzo_km	DOUBLE(5,2)		NO	prezzo al km
data_inizio	DATE		NO	inizio noleggio
data_fine	DATE		NO	data massima di rientro
agenzia_inizio	INT(3)	FK	NO	agenzia di noleggi

agenzia_fine	INT(3)	FK	NO	agenzia di rietro
impiegato_inizio	VARHCAR(16)	FK	NO	impiegato che apre il contratto
impiegato_fine	VARHCAR(16)	FK		impiegato che chiude il contratto
acconto	DOUBLE(5, 2)		NO	acconto versato
km_percorsi	DOUBLE(6, 2)			km percorsi alla chiusura
tariffa_base	DOUBLE(5,2)		NO	importo di partenza
stato_contratto	ENUM(aperto, chiuso)		NO	stato del contratto
totale	DOUBLE(5, 2)			prezzo totale

3.2 File System

3.2.1 Altri file

Le tipologie di file utilizzate sono:

- File .properties:
    - mySql.properties, contiene informazioni per la connessione al database

- File .fxml per la realizzazione delle interfacce con JavaFX
- File .json:
    - queries.json, contiene le query utilizzate
    - views.json, contiene i path delle view utilizzate
- File .sql:
    - db_schema.sql, contiene lo schema SQL del database
    - db:populate.sql, utilizzato per popolare il database

4. APPENDICE

4.1 Pattern utilizzati

Pattern Architetturali

Front Controller
Application Controller
View Dispatcher
Application Service
Business Object
Transfer Object
Data Access Object

Design Pattern

Command
Command Factory
Singleton
Factory