# Progetto di Progettazione e algoritmi



Maffi Andrea, Sassi Annamaria, Testa Lorenzo

Anno accademico 2021-2022

Sommario

[Progetto di Progettazione e algoritmi 1](#_Toc111964088)

[Concessionario intelligente 1](#_Toc111964089)

[Iterazione 0 2](#_Toc111964090)

[Analisi dei requisiti 2](#_Toc111964091)

[Configurazione iniziale architettura 2](#_Toc111964092)

[Modellazione casi d’uso 2](#_Toc111964093)

[Tool chain 5](#_Toc111964094)

[Iterazione 1 6](#_Toc111964095)

[Design dell’architettura software 11](#_Toc111964096)

[Pseudocodice e analisi complessità tempo 14](#_Toc111964097)

[Analisi statica 14](#_Toc111964098)

[Analisi dinamica 15](#_Toc111964099)

[Iterazione intermedia 16](#_Toc111964100)

[Analisi requisiti e casi d’uso 16](#_Toc111964101)

[Design dell’architettura software 16](#_Toc111964102)

[Analisi statica 16](#_Toc111964103)

[Analisi dinamica 16](#_Toc111964104)

[Iterazione finale 17](#_Toc111964105)

[Analisi requisiti e casi d’uso 17](#_Toc111964106)

[Design dell’architettura software 17](#_Toc111964107)

[Analisi statica 17](#_Toc111964108)

[Analisi dinamica 17](#_Toc111964109)

[Guida all’installazione dell’applicazione 18](#_Toc111964110)

[Uso dell’applicazione 19](#_Toc111964111)

# Concessionario intelligente

In una concessionaria con diverse filiali sparse sul territorio, le auto usate possono essere vendute nella filiale in cui si trovano o in un un’altra. Le filiali hanno in comune un database contenente tutte le macchine usate presenti.

Una volta venduta una macchina, l’incaricato post-vendita determina tutti gli interventi che devono essere eseguiti e solo in seguito a ciò viene assegnato lo slot di lavorazione tramite l’algoritmo. Questi interventi possono essere effettuati nella filiale in cui si trova l’auto oppure in una qualsiasi altra filiale a seconda del metodo più efficiente (tenendo anche conto del tempo di trasporto tra le filiali).

Gli interventi possono essere modificati (può aumentare o diminuire il carico), cancellati oppure aggiunti; in tutti e tre i casi viene aggiornato lo slot di lavorazione.

Il venditore può anche inserire nel database nuove macchine in vendita.

Il capo officina visualizza tutti gli interventi che ha in carico la filiale e quindi può assegnare i singoli interventi ai meccanici.

I meccanici possono visualizzare le lavorazioni che gli sono state assegnate e una volta terminate devono segnalarlo tramite l’apposita interfaccia.

Il capo filiale può controllare il database contenente le macchine vendute (e relativi interventi) e quelle in vendita.

Nella filiale centrale è presente il capo di tutte le filiali e può controllare le macchine in transito tra le filiali, quelle in lavorazione, quelle in attesa di essere trasportate.

Per poter accedere all’applicazione web del sistema bisogna accedere con le proprie credenziali.

L’amministratore di rete può inserire nuovi utenti, eliminare account o modificare i dati di un utente.

Nella filiale centrale è contenuto il database centrale e l’algoritmo.

L’algoritmo prende in considerazione la capacità di lavoro di ogni officina e la lista degli interventi delle singole auto.

La filiale centrale deve poter vedere tutte le giacenze e, sulla base del numero delle auto in officina e dei tempi necessari al trasporto delle auto, decidere come smistare il carico di auto in attesa.

La filiale centrale conosce tutti i tempi di trasporto tra le filiali.

Il capo officina cambia lo stato delle auto.

## Iterazione 0

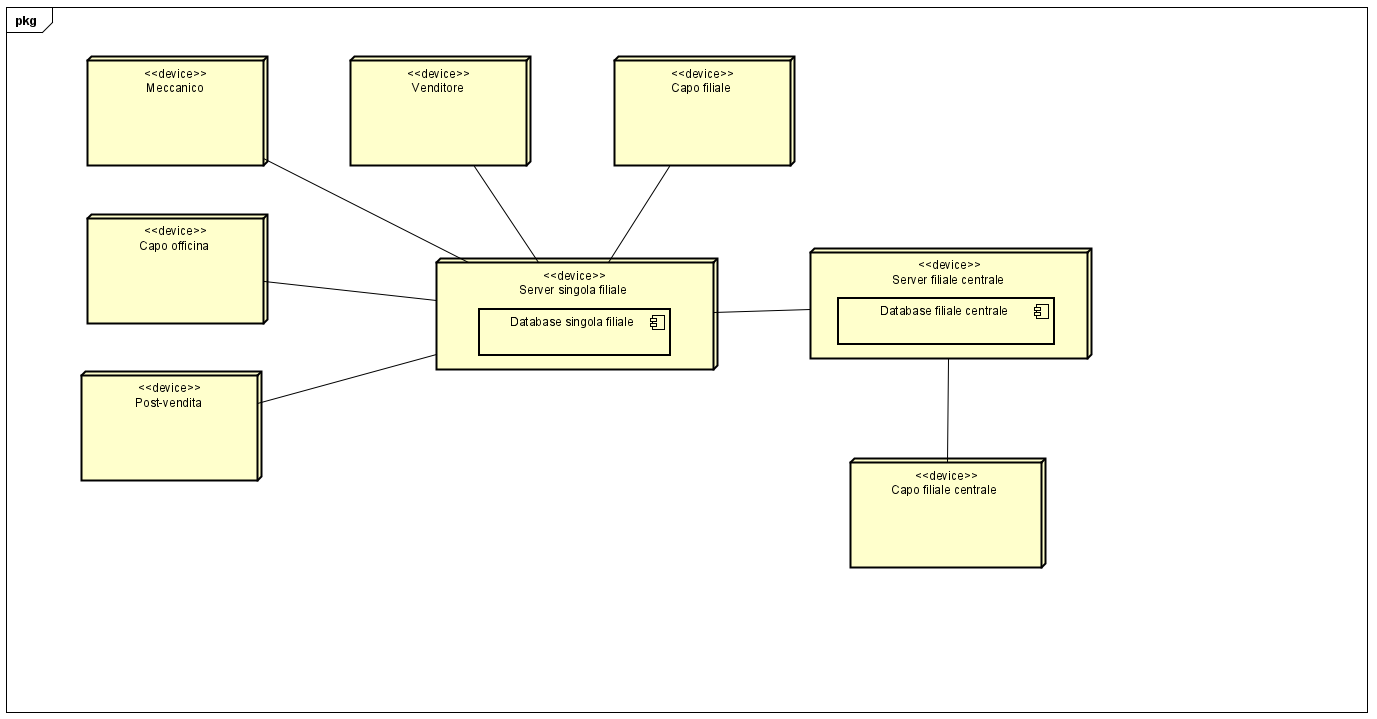
### Analisi dei requisiti

Il sistema è composto da: una serie di interfacce con cui le persone possono interagire, le varie componenti sottostanti che permettono ciò e l’algoritmo. Grazie a quest’ultimo è possibile determinare la schedulazione delle macchine vendute in modo che siano aggiustate nel più breve tempo possibile.

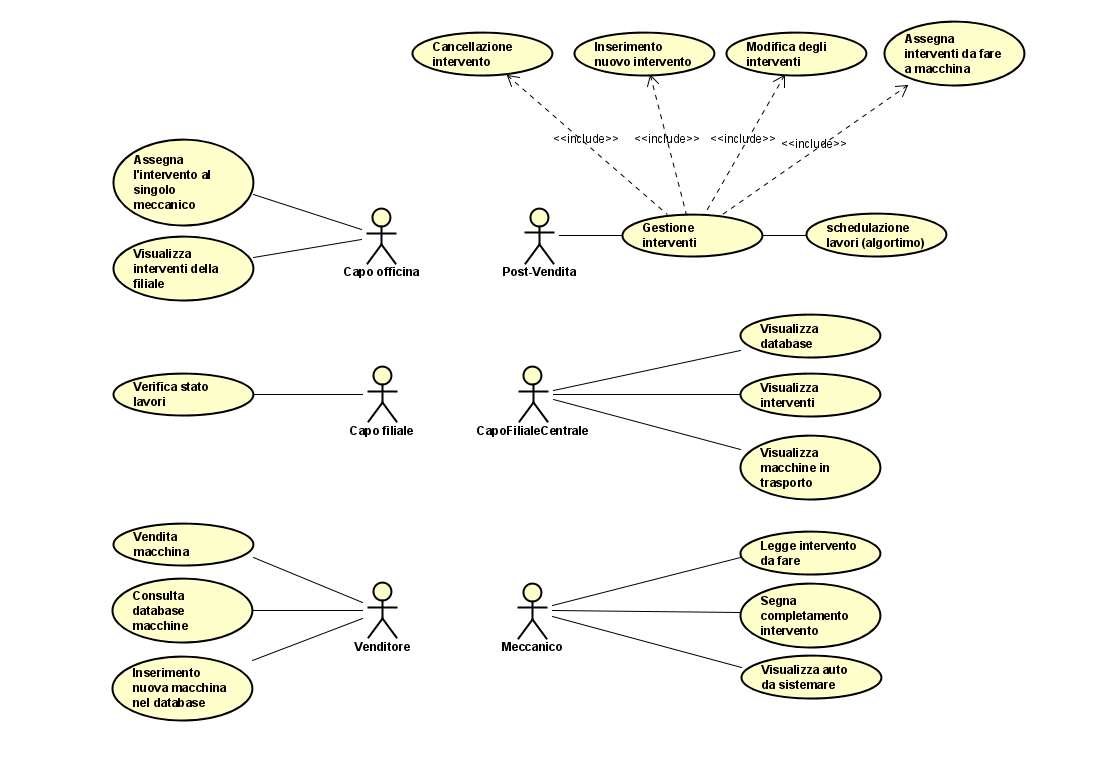
### Configurazione iniziale architettura

Abbiamo pensato a un’architettura nella quale ogni sottosistema rappresenta un tipo di utente e queste componenti interagiscono con il database della singola filiale.

Dato che ci sono più filiali, abbiamo individuato una filiale centrale nella quale c’è il server centrale e il capo filiale centrale che può controllare le altre filiali.



### Modellazione casi d’uso



|  |
| --- |
| Nome: PV |
| Attori: post-vendita |
| Eventi:   1. Gestione degli interventi (eliminazione, modificazione, aggiunta) |

|  |
| --- |
| Nome: V |
| Attori: Venditore |
| Eventi:   1. Visualizzare database contenente le macchine in vendita 2. Vendere una macchina 3. Inserisce nuova macchina nel database |

|  |
| --- |
| Nome: CO |
| Attori: Capo officina |
| Eventi:   1. Visualizza gli interventi programmati per quella filiale 2. Assegna gli interventi ai meccanici 3. Cambia stato delle auto (in coda, lavorazione, trasporto alla filiale di lavorazione e trasporto alla filiale di consegna) |
| Nome: CF |
| Attori: Capo filiale |
| Eventi:   1. Visualizza gli interventi programmati 2. Visualizza le macchine vendute 3. Visualizza le macchine in vendita |

|  |
| --- |
| Nome: M |
| Attori: Meccanico |
| Eventi:   1. Visualizza le macchine da sistemare (relative al suo ponte) 2. Visualizza gli interventi da effettuare 3. Segnalazione fine intervento |

|  |
| --- |
| Nome: CFC |
| Attori: Capo filiale centrale |
| Eventi:   1. Visualizza le macchine in transito 2. Visualizza le macchine vendute 3. Visualizza le macchine in vendita |

|  |
| --- |
| Nome: AR |
| Attori: Amministratore di rete |
| Eventi:   1. Inserisce nuovi utenti 2. Elimina utenti 3. Modifica i dati degli utenti |

### Tool chain

|  |  |
| --- | --- |
| Attività | Tool |
| Modellazione UML | Astah UML |
| Linguaggio di programmazione | Java |
| Framework backend | Spring Boot |
| DBMS |  |
| IDE | Eclipse |
| Analisi statica | STAN4J |
| Analisi dinamica | JUnit 4 |
| Versioning | Git & GitHub |

## Iterazione 1

In questa iterazione abbiamo sviluppato in modo più dettagliato i diagrammi e abbiamo iniziato a strutturare in eclipse il workspace.

### Casi d’uso

|  |
| --- |
| Nome: PV |
| Attori: post-vendita |
| Precondizioni: login |
| Eventi:   1. Gestione degli interventi    1. Cancellazione intervento    2. Inserimento nuovo intervento    3. Modifica degli interventi    4. Assegnare intervento a una macchina |
| Postcondizioni: logout |

|  |
| --- |
| Nome: V |
| Attori: Venditore |
| Precondizioni: login |
| Eventi:   1. Visualizzare database contenente le macchine in vendita 2. Vendere una macchina 3. Inserisce nuova macchina nel database |
| Postcondizioni: logout |

|  |
| --- |
| Nome: CO |
| Attori: Capo officina |
| Precondizioni: login |
| Eventi:   1. Visualizza gli interventi programmati per quella filiale 2. Assegna gli interventi ai meccanici 3. Cambia stato delle auto (in coda, lavorazione, trasporto alla filiale di lavorazione e trasporto alla filiale di consegna) |
| Postcondizioni: logout |

|  |
| --- |
| Nome: CF |
| Attori: Capo filiale |
| Precondizioni: login |
| Eventi:   1. Visualizza gli interventi programmati 2. Visualizza le macchine vendute 3. Visualizza le macchine in vendita |
| Postcondizioni: logout |

|  |
| --- |
| Nome: M |
| Attori: Meccanico |
| Precondizioni: login |
| Eventi:   1. Visualizza le macchine da sistemare (relative al suo ponte) 2. Visualizza gli interventi da effettuare 3. Segnalazione fine intervento |
| Postcondizioni: logout |

|  |
| --- |
| Nome: CFC |
| Attori: Capo filiale centrale |
| Precondizioni: login |
| Eventi:   1. Visualizza le macchine in transito 2. Visualizza le macchine vendute 3. Visualizza le macchine in vendita |
| Postcondizioni: logout |

|  |
| --- |
| Nome: AR |
| Attori: Amministratore di rete |
| Precondizioni: login |
| Eventi:   1. Inserisce nuovi utenti 2. Elimina utenti 3. Modifica i dati degli utenti |
| Postcondizioni: logout |

Raggruppamenti casi d’uso:

Si può osservare che i casi d’uso si dividono in quattro macroaree: interventi, vendita automobile, account manager, amministrazione, quindi possiamo suddividerli come segue.

|  |
| --- |
| Interventi:   1. Gestione interventi    1. Assegnamento interventi alla singola macchina    2. Modifica di un intervento    3. Inserimento di un intervento    4. Cancellazione di un intervento 2. Determinazione slot intervento 3. Visualizzazione degli interventi programmati in una determinata filiale 4. Assegnazione interventi ai meccanici 5. Segnalazione del completamento dell’intervento |
| Vendita automobile:   1. Visualizzare database contenente le macchine in vendita 2. Vendere una macchina 3. Inserimento nuova macchina in vendita nel database |
| Account Manager:   1. Login 2. Logout 3. Registrazione nuovo utente 4. Eliminazione account 5. Modifica dati utente |
| Amministrazione (capo filiale centrale):   1. Visualizzazione macchine vendute 2. Visualizzazione macchine in vendita 3. Visualizzazione interventi 4. Visualizzazione macchine in trasporto |

Nel seguito verrà descritto nel dettaglio ogni caso d’uso implementato. I codici utilizzati per identificarli sono composti dalla sigla dell’utente che può eseguirli e viene aggiunto un numero che segue la numerazione usata in precedenza. Ad esempio: inserimento nuovo intervento avrà codice PV1a che sta a significare che l’attore che può eseguire queste azioni è PV (post vendita), il numero 1 indica che è nel primo blocco di azioni e invece l’ultima lettera rappresenta la sotto categoria del blocco.

|  |  |
| --- | --- |
| ID: | PV1a |
| Nome: | Inserimento nuovo intervento |
| Attore/i: | Post vendita |
| Triggered by: |  |
| Precondizione: | Login |
| Sequenza di azioni: |  |
| Postcondizione: | Logout |
| Eccezione/i: |  |

|  |  |
| --- | --- |
| ID: | V1 |
| Nome: | Visualizzare database contenente le macchine in vendita |
| Attore/i: | Venditore |
| Triggered by: |  |
| Precondizione: | Login |
| Sequenza di azioni: |  |
| Postcondizione: | Logout |
| Eccezione/i: |  |
| ID: | V2 |
| Nome: | Vendere una macchina |
| Attore/i: | Venditore |
| Triggered by: |  |
| Precondizione: | Login |
| Sequenza di azioni: |  |
| Postcondizione: | Logout |
| Eccezione/i: |  |

|  |  |
| --- | --- |
| ID: | V3 |
| Nome: | Inserisce nuova macchina nel database |
| Attore/i: | Venditore |
| Triggered by: |  |
| Precondizione: | Login |
| Sequenza di azioni: |  |
| Postcondizione: | Logout |
| Eccezione/i: |  |

|  |  |
| --- | --- |
| ID: | CO1 |
| Nome: | Visualizza gli interventi programmati per quella filiale |
| Attore/i: | Capo officina |
| Triggered by: |  |
| Precondizione: | Login |
| Sequenza di azioni: |  |
| Postcondizione: | Logout |
| Eccezione/i: |  |

|  |  |
| --- | --- |
| ID: | CO2 |
| Nome: | Cambia stato delle auto |
| Attore/i: | Capo officina |
| Triggered by: |  |
| Precondizione: | Login |
| Sequenza di azioni: |  |
| Postcondizione: | Logout |
| Eccezione/i: |  |

|  |  |
| --- | --- |
| ID: | M1 |
| Nome: | Visualizza le macchine da sistemare (relative al suo ponte) |
| Attore/i: | Meccanico |
| Triggered by: |  |
| Precondizione: | Login |
| Sequenza di azioni: |  |
| Postcondizione: | Logout |
| Eccezione/i: |  |

|  |  |
| --- | --- |
| ID: | M2 |
| Nome: | Visualizza gli interventi da effettuare |
| Attore/i: | Meccanico |
| Triggered by: |  |
| Precondizione: | Login |
| Sequenza di azioni: |  |
| Postcondizione: | Logout |
| Eccezione/i: |  |

|  |  |
| --- | --- |
| ID: | M3 |
| Nome: | Segnalazione fine intervento |
| Attore/i: | Meccanico |
| Triggered by: |  |
| Precondizione: | Login |
| Sequenza di azioni: |  |
| Postcondizione: | Logout |
| Eccezione/i: |  |

### Design dell’architettura software

Diagramma UML di package:

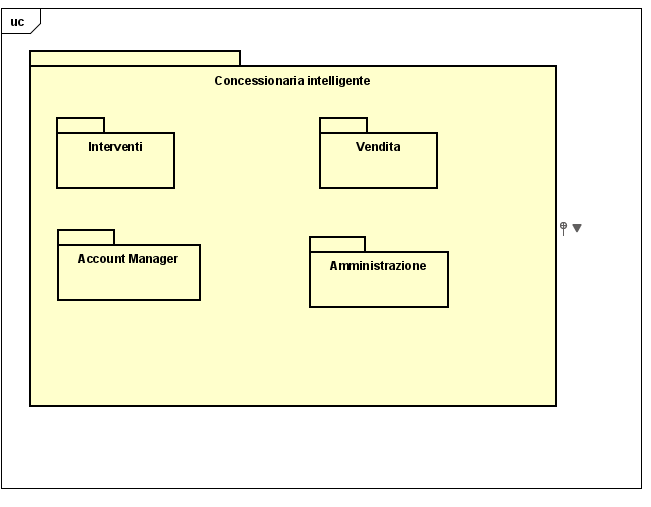


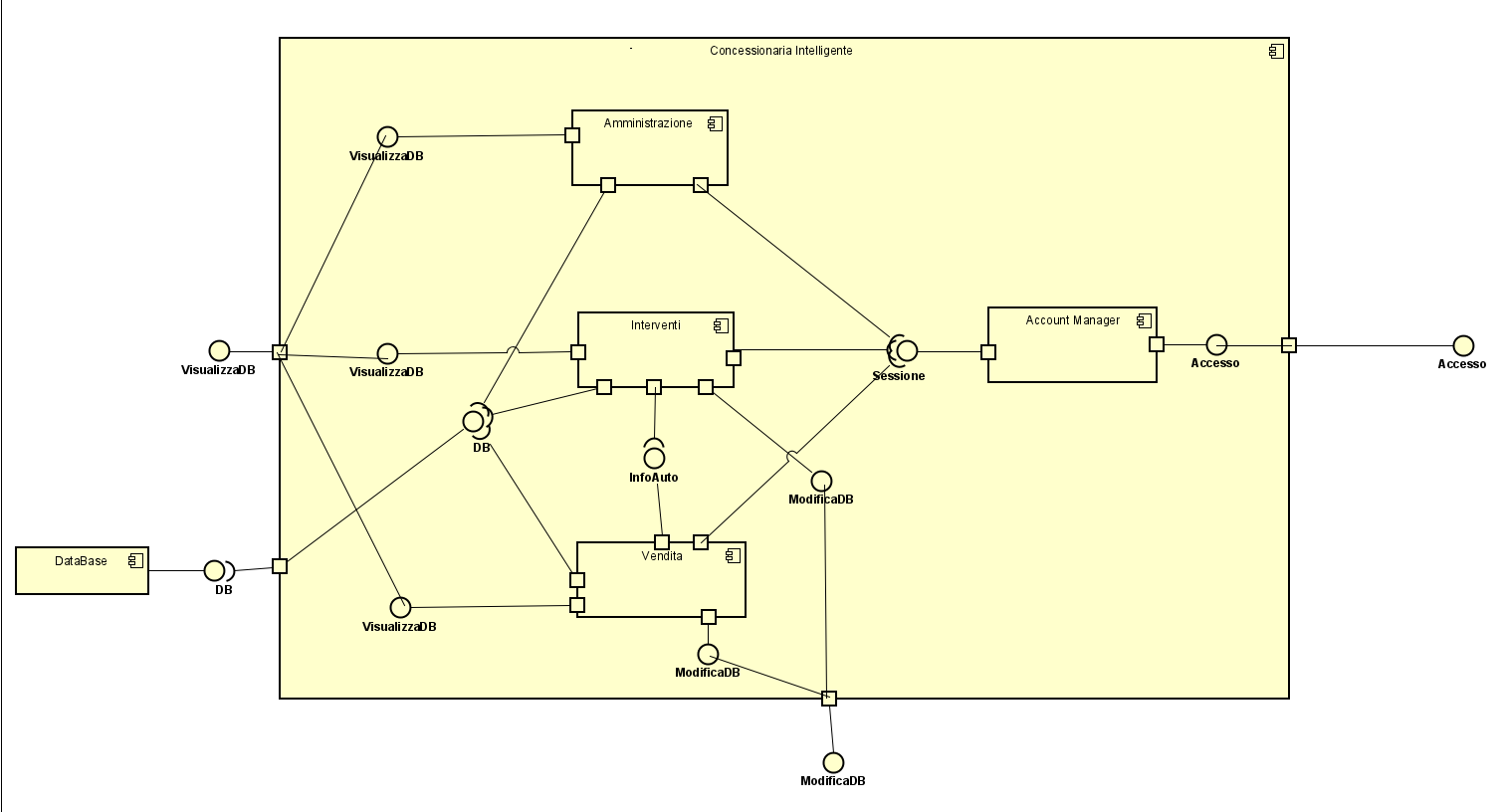
Diagramma UML delle componenti:

Nel primo diagramma si sono considerati i sottosistemi nel loro insieme, mentre nel secondo si sono rappresentate anche le componenti dei sottosistemi in modo da mostrare nel loro insieme tutte le componenti. In ogni caso le interfacce con cui comunicano le diverse parti non cambiano, perché varia solo il grado di dettaglio dei diagrammi.

Le interfacce sono: visualizza DB, modifica DB, DB, accesso. Quasi tutti riguardano la connessione con il database in modo da poter visualizzarlo e modificarlo. L’interfaccia accesso serve per poter effettuare il login, anche se non è implementata con la richiesta delle credenziali.

Come già osservato, i casi d’uso sono raggruppati in quattro macroaree, per questo motivo abbiamo sviluppato quattro package indipendenti tra di loro e con ruoli diversi, in modo che siano coesi al loro interno, ma poco accoppiati tra di loro.

Al loro interno i servizi seguono il pattern esagonale che è suddiviso in controller, service ed entity. Il controller è all’esterno e serve per poter comunicare con l’esterno e nasconde al suo interno le altre parti da cui è composto il pattern. I servizi servono per far comunicare le entità con i controller.



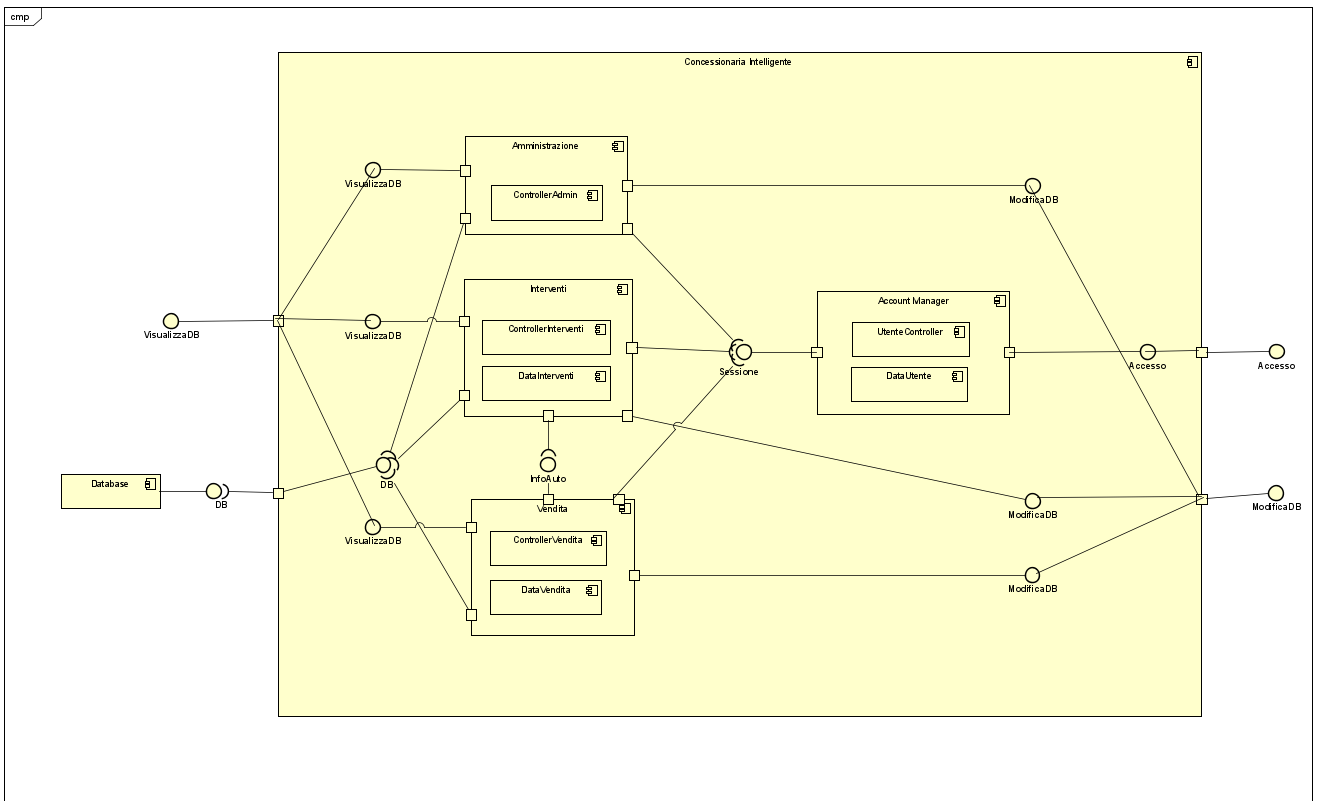


Diagramma UML delle classi per definire data type classes:

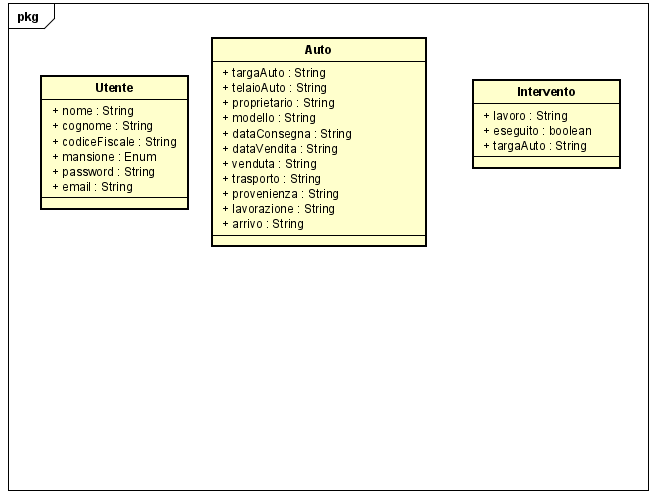
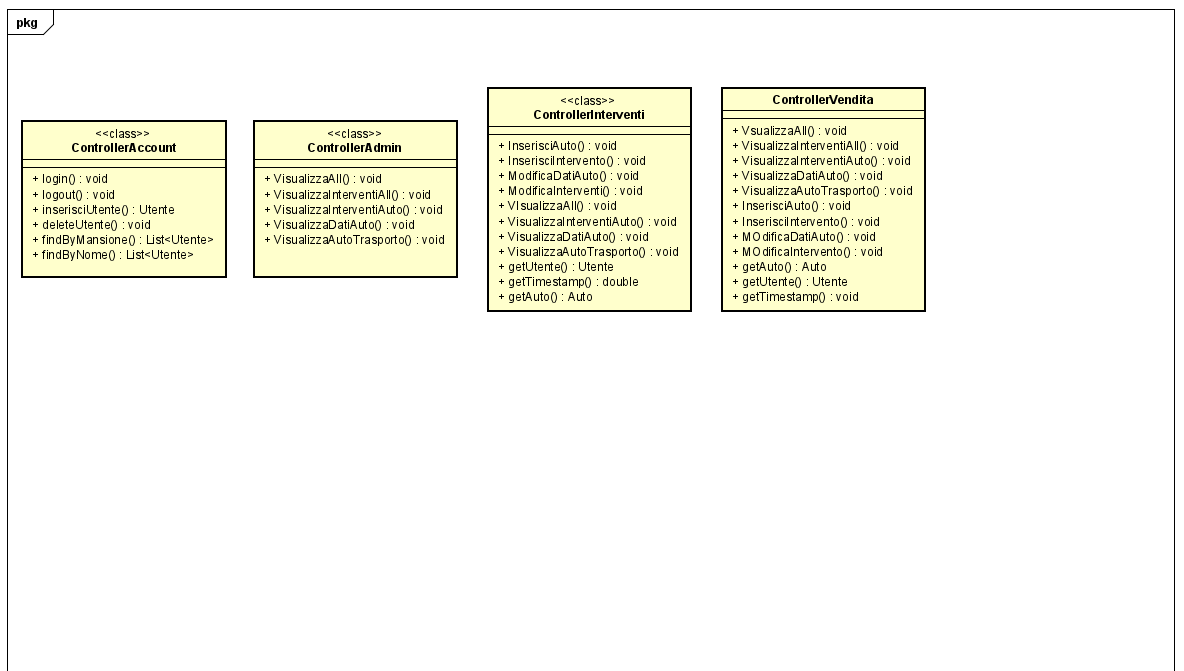


Diagramma UML delle classi per definire le interfacce:



### Pseudocodice e analisi complessità tempo

### Analisi statica

### Analisi dinamica

## Iterazione intermedia

### Analisi requisiti e casi d’uso

### Design dell’architettura software

### Analisi statica

### Analisi dinamica

## Iterazione finale

### Analisi requisiti e casi d’uso

### Design dell’architettura software

### Analisi statica

### Analisi dinamica

# Guida all’installazione dell’applicazione

# Uso dell’applicazione