

Corso di laurea in Informatica

## **Ingegneria del software**

### **Elaborato 2**

Applicazione per la gestione delle informazioni socio-sanitarie di  
monitoraggio per la prevenzione di contagi e pandemie

Brunello Simone: VR437366

Nicoletti Daniele: VR429618

Nicolis Nicholas: VR429666

Anno Accademico: 2019 - 2020



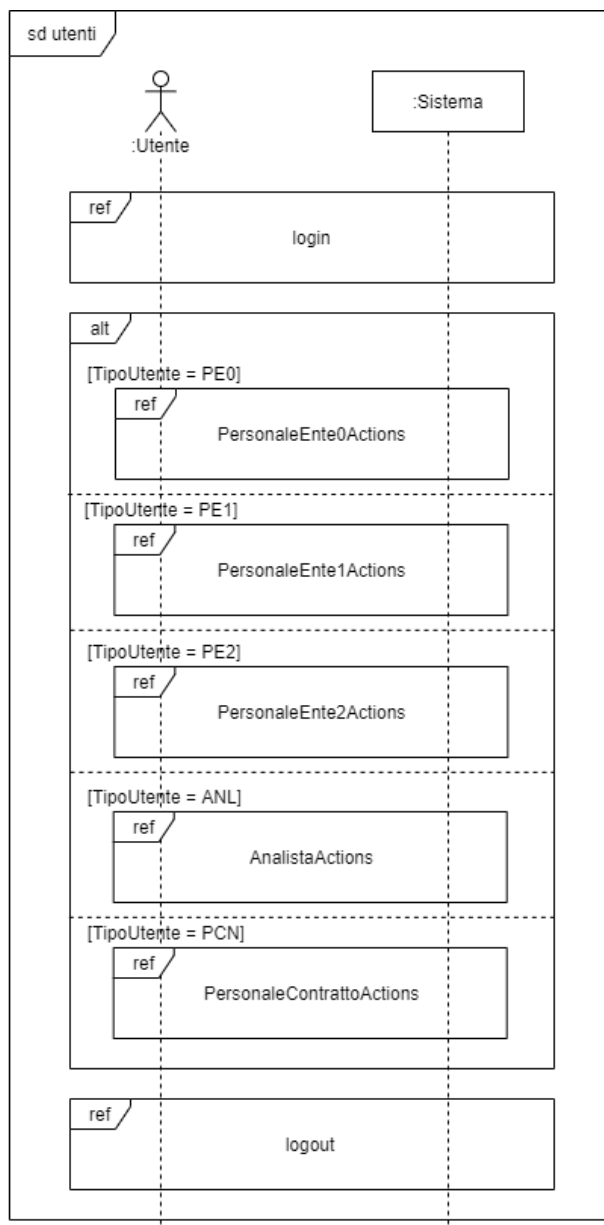
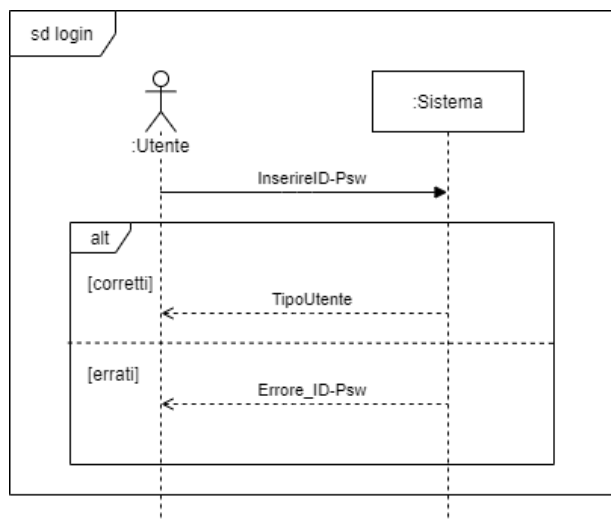
# INDICE

<b>1 Requisiti ed interazioni utente-sistema .....</b>	<b>4</b>
1.1 Note generali .....	4
 <b>2 Casi d'uso .....</b>	 <b>5</b>
2.1 Casi d'uso relativi al personale a contratto .....	6
2.2 Casi d'uso relativi al personale dell'ente .....	6
2.2.1 Casi d'uso relativi al personale dell'ente 0 .....	7
2.2.2 Casi d'uso relativi al personale dell'ente 1 .....	8
2.2.3 Casi d'uso relativi al personale dell'ente 2 .....	9
2.3 Casi d'uso relativi all'analista .....	10
 <b>3 Diagrammi di attività .....</b>	 <b>11</b>
3.1 Login .....	11
3.2 Personale a contratto .....	11
3.3.1 Personale dell'ente di tipo PE0 .....	12
3.3.2 Personale dell'ente di tipo PE1 .....	12
3.3.3 Personale dell'ente di tipo PE2 .....	13
3.4 Analista .....	13
 <b>4 Diagrammi delle classi .....</b>	 <b>14</b>
4.1 Controllori .....	15
4.2 Suddivisione geografica .....	16
4.3 Utenti .....	17
4.4 Analista .....	18
 <b>5 Diagrammi di sequenza del software .....</b>	 <b>19</b>
 <b>6 Sviluppo .....</b>	 <b>20</b>
 <b>7 Pattern .....</b>	 <b>21</b>
 <b>8 Attività di test e verifica .....</b>	 <b>22</b>

# 1 Requisiti ed interazioni utente-sistema

## 1.1 Note generali

Il sistema supporta l'utilizzo da parte del personale di prevenzione e monitoraggio epidemiologico del territorio nazionale. Ad ogni utente è riservato un profilo dedicato all'interno del sistema. Le macrocategorie di utenti sono "personale a contratto" PCN, "analista" ANL, "personale dell'ente 0" PEO, "personale dell'ente 1" PE1, "personale dell'ente 2" PE2. Previa autenticazione, con credenziali fornite dagli amministratori del sistema, ogni utente potrà accedere alla sua area riservata, ove sono presenti funzionalità esclusive relative al proprio impiego.



## 2 Casi d'uso

Dalla specifica abbiamo individuato i casi d'uso, ovvero le modalità con cui un attore può interagire con il sistema. Abbiamo deciso di rappresentare un caso d'uso per ogni attore appartenente al sistema.

### 2.1 Caso d'uso relativo al personale a contratto

**Attori:** Personale a contratto.

**Scopo e descrizione sintetica:** Il sistema permette al personale a contratto di inserire i dati epidemiologici relativi ai comuni di competenza. Inoltre, una volta a settimana, dovrà aggiornare il numero di persone in terapia intensiva ed in cura presso il medico di base. Queste saranno classificate sulla base della patologia riscontrata.

**Pre-condizione:** Il personale a contratto deve aver fatto log-in nel sistema.

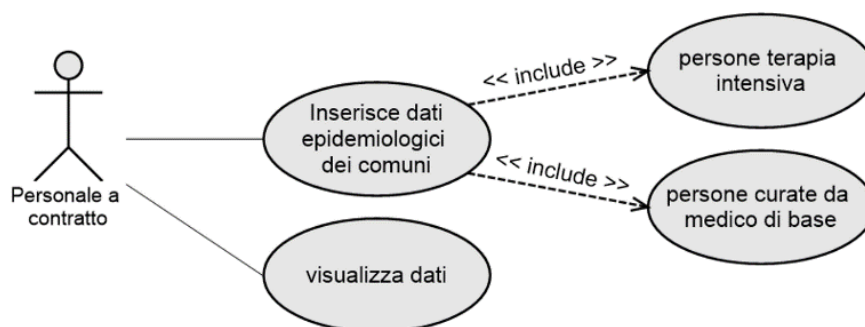
**Sequenza eventi:**

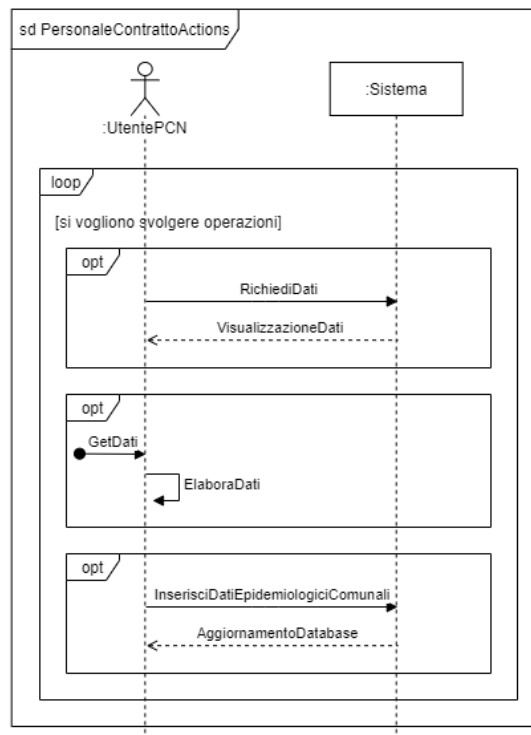
- I. Il personale a contratto sceglie un'operazione da svolgere.
- II. Uno tra i seguenti casi d'uso viene utilizzato:
  - Visualizzare i dati epidemiologici complessivi.
  - Inserire i dati epidemiologici del comune di appartenenza.

**Sequenza alternativa eventi:**

- Inserimento scorretto dei dati:
  - Se vengono inseriti dati errati o non coerenti il sistema visualizzerà un messaggio d'errore ed inviterà a ricompilare tali campi.
- Errore di visualizzazione:
  - Se non vengono trovati dati o vi è un errore nella visualizzazione, allora viene generato un messaggio d'errore.

**Post-condizione:** Se un nuovo dato viene inserito correttamente entra a far parte del database e successivamente potrà essere visualizzato dal resto del personale. Se il personale a contratto riceve tutte le elaborazioni disponibili, il sistema non mostrerà nessuna nuova elaborazione da inserire.





## 2.2 Caso d'uso relativo al personale dell'ente

**Attori:** Personale dell'ente.

**Scopo e descrizione sintetica:** Il sistema permette al personale dell'ente di visualizzare i dati contenuti nel sistema ed a seconda del proprio ruolo specifico, di svolgere ulteriori operazioni.

**Pre-condizione:** Il personale dell'ente deve aver fatto log-in nel sistema.

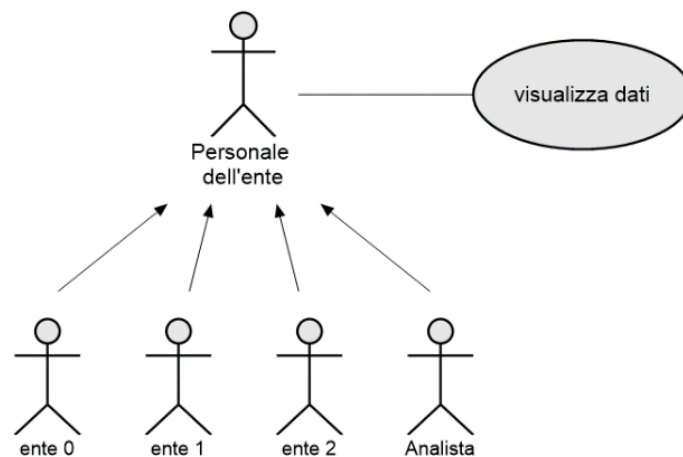
**Sequenza eventi:**

- I. Il personale dell'ente visualizza i dati

**Sequenza alternativa eventi:**

- Errore di visualizzazione:
  - Se non vengono trovati dati o vi è un errore nella visualizzazione viene generato un messaggio d'errore.

**Post condizione:** Se le operazioni di inserimento andranno a buon fine vi sarà una corretta visualizzazione dei dati.



## 2.2.1 Caso d'uso relativo all'ente 0

**Attori:** Personale dell'ente con abilitazione di tipo PEO.

**Scopo e descrizione sintetica:** Il sistema permette al personale dell'ente PEO di inserire e modificare i dati relativi alle regioni ed ai comuni.

**Pre-condizione:** Il personale dell'ente deve aver fatto log-in nel sistema.

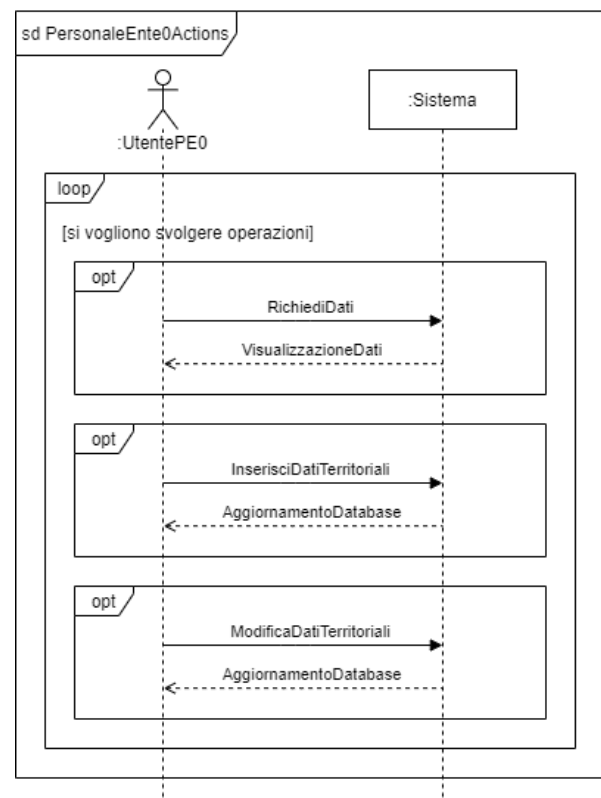
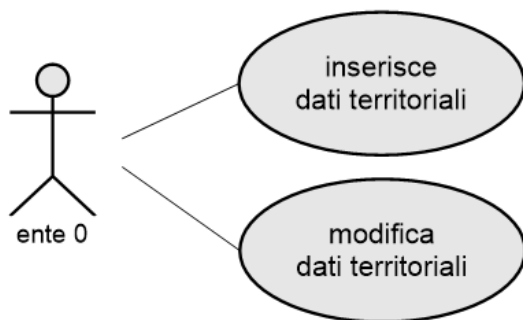
**Sequenza eventi:**

- I. Il personale dell'ente sceglie che operazione effettuare.
- II. Uno dei seguenti casi d'uso viene utilizzato:
  - Inserire dati territoriali.
  - Modificare dati territoriali.
  - Visualizzare dati territoriali.

**Sequenza alternativa eventi:**

- Inserimento errato dei dati:
  - Se vengono inseriti dati errati o non coerenti il sistema visualizzerà un messaggio di errore ed inviterà a ricompilare tali campi.
- Modifica incoerente dei dati:
  - Se vengono modificati dati in maniera errata o non coerente il sistema visualizzerà un messaggio di errore ed inviterà a effettuare nuovamente la modifica.

**Post-condizione:** Se il dato geografico viene inserito o modificato correttamente entrerà a far parte del database.



## 2.2.2 Caso d'uso relativi all'ente 1

**Attori:** Personale dell'ente con abilitazione di tipo PE1.

**Scopo e descrizione sintetica:** Il sistema permette al personale dell'ente PE1 di modificare i responsabili comunali.

**Pre-condizione:** Il personale dell'ente deve aver fatto log-in nel sistema.

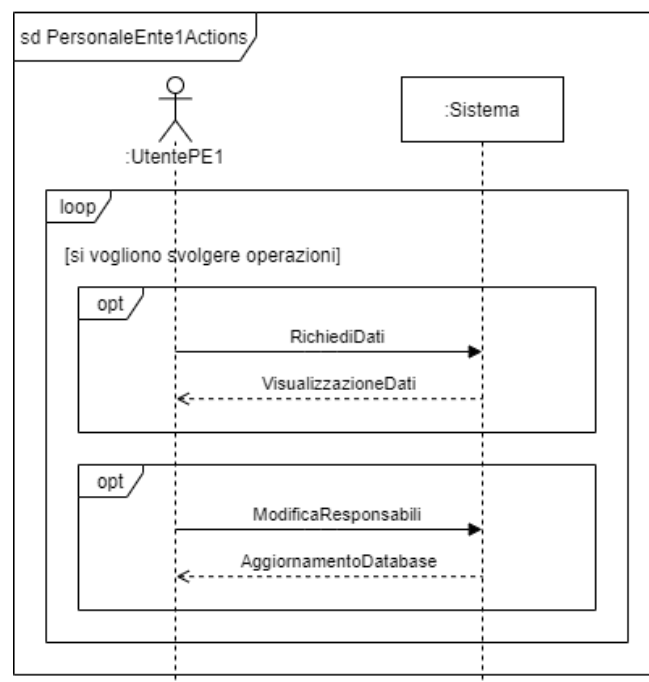
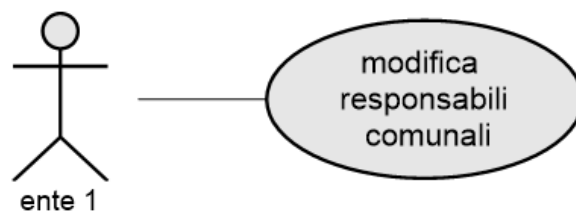
**Sequenza eventi:**

- I. Il personale dell'ente sceglie che operazione effettuare.
- II. Uno dei seguenti casi d'uso viene utilizzato:
  - Modificare i responsabili comunali.
  - Visualizzare i responsabili comunali.

**Sequenza alternativa eventi:**

- Modifica incoerente dei responsabili comunali:
  - Se i comuni assegnati vengono modificati in maniera errata il sistema visualizzerà un messaggio di errore ed inviterà a effettuare nuovamente la modifica.

**Post-condizione:** Se il comune di competenza del personale a contratto viene modificato correttamente questa informazione verrà inserita nel database.





### 2.2.3 Caso d'uso relativi all'ente 2

**Attori:** Personale dell'ente con abilitazione di tipo PE2.

**Scopo e descrizione sintetica:** Il sistema permette al personale dell'ente PE2 di inserire e modificare i dati epidemiologici provinciali.

**Pre-condizione:** Il personale dell'ente deve aver fatto log-in nel sistema.

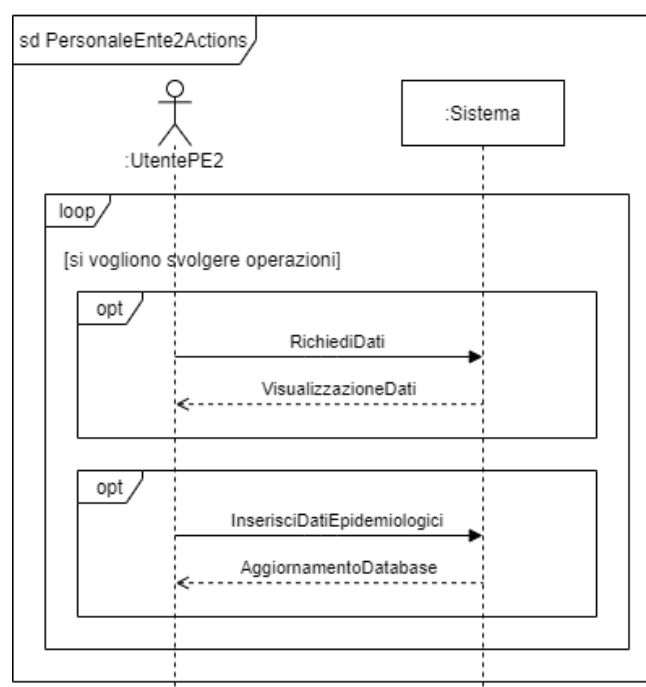
**Sequenza eventi:**

- III. Il personale dell'ente sceglie che operazione effettuare.
- IV. Uno dei seguenti casi d'uso viene utilizzato:
  - Inserire dati epidemiologici.
  - Modificare dati epidemiologici.
  - Visualizzare dati epidemiologici.

**Sequenza alternativa eventi:**

- Inserimento errato dei dati:
  - Se vengono inseriti dati errati o non coerenti il sistema visualizzerà un messaggio di errore ed inviterà a ricompilare tali campi.
- Modifica incoerente dei dati:
  - Se vengono modificati dati in maniera errata o non coerente il sistema visualizzerà un messaggio di errore ed inviterà a effettuare nuovamente la modifica.

**Post-condizione:** Se il dato viene inserito o modificato correttamente entrerà a far parte del database.



## 2.3 Caso d'uso relativo all'analista

**Attori:** Analista.

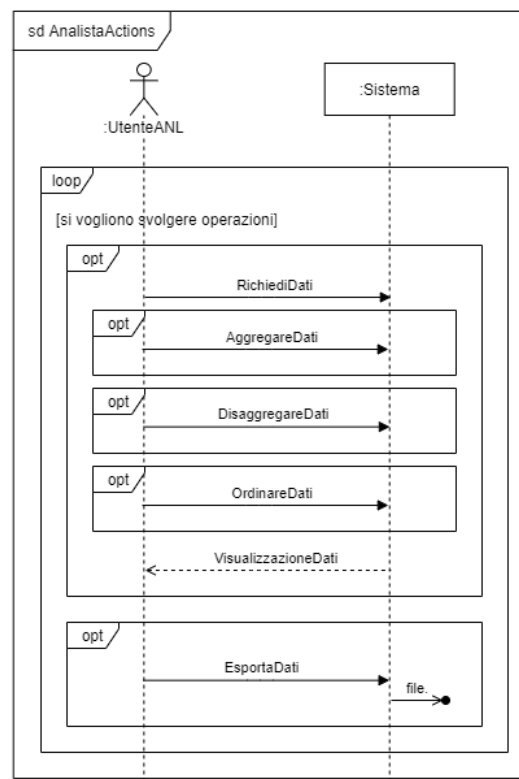
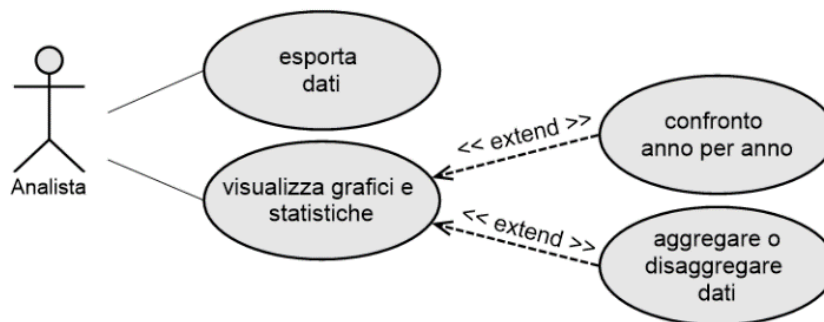
**Scopo e descrizione sintetica:** Il sistema permette agli analisti di visualizzare ed esportare i dati presenti nel database. Possono inoltre decidere che rappresentazione grafica visualizzare e come aggregare i dati tra loro, operando anche confronti su base temporale.

**Pre-condizione:** L'analista deve aver fatto log-in nel sistema.

**Sequenza eventi:**

- I. L'analista sceglie che operazione effettuare.
- II. Uno dei seguenti casi d'uso viene utilizzato:
  - Visualizzare statistiche o grafici secondo vari criteri.
  - Esportare i dati.

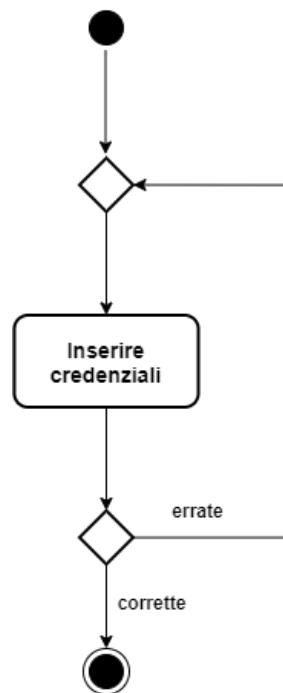
**Post-condizione:** Se la visualizzazione ed eventuale esportazione avviene in modo corretto, si avrà visualizzato a schermo i dati richiesti nella forma voluta, e/o si avrà un file di esportazione che conterrà i dati voluti.



### 3 Diagrammi di attività

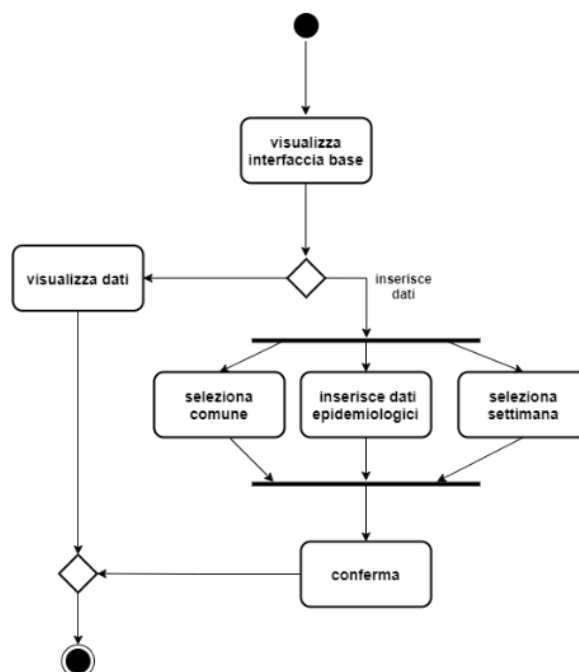
#### 3.1 Login

Diagramma di attività per il login di un utente.



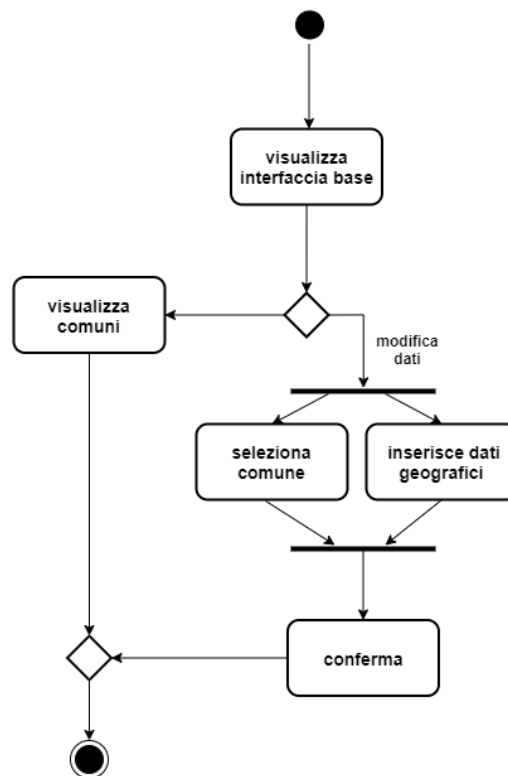
#### 3.2 Personale a contratto

Diagramma di attività che descrive le funzioni svolte dal personale a contratto PCN.



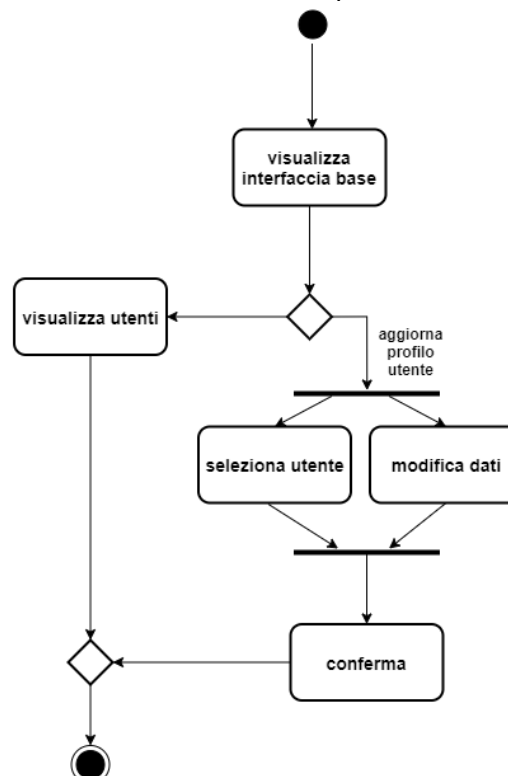
### 3.3.1 Personale dell'ente di tipo PE0

Diagramma di attività che descrive le funzioni svolte dal personale dell'ente di tipo PE0.



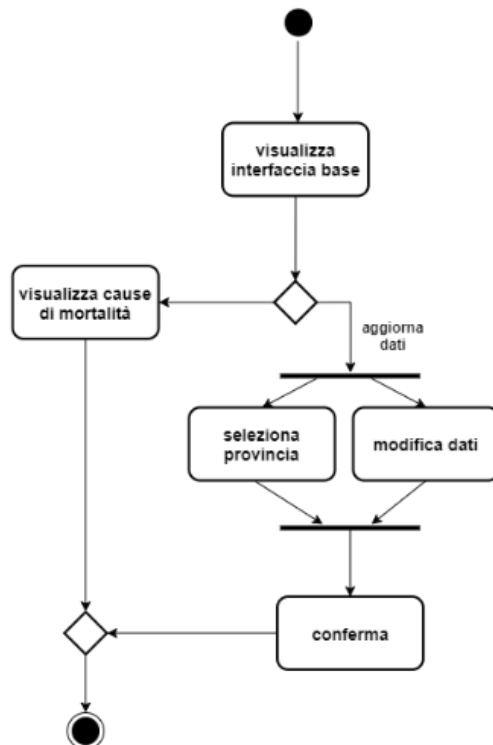
### 3.3.2 Personale dell'ente di tipo PE1

Diagramma di attività che descrive le funzioni svolte dal personale dell'ente di tipo PE1.



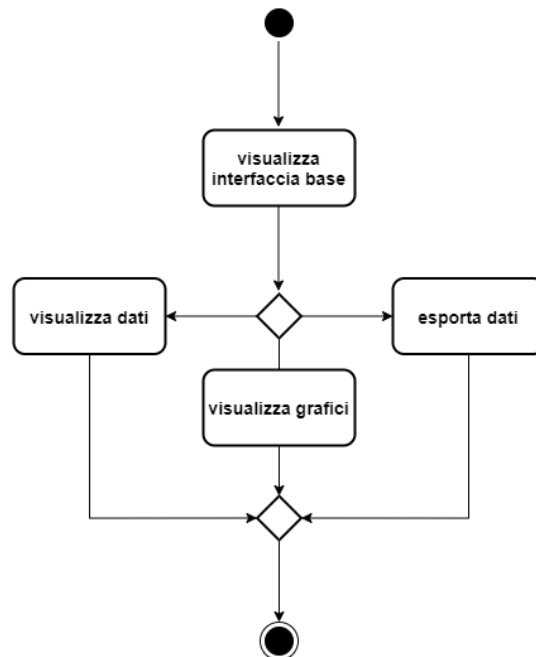
### 3.3.3 Personale dell'ente di tipo PE2

Diagramma di attività che descrive le funzioni svolte dal personale dell'ente di tipo PE2.



### 3.4 Analista

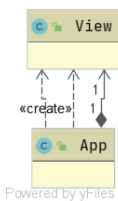
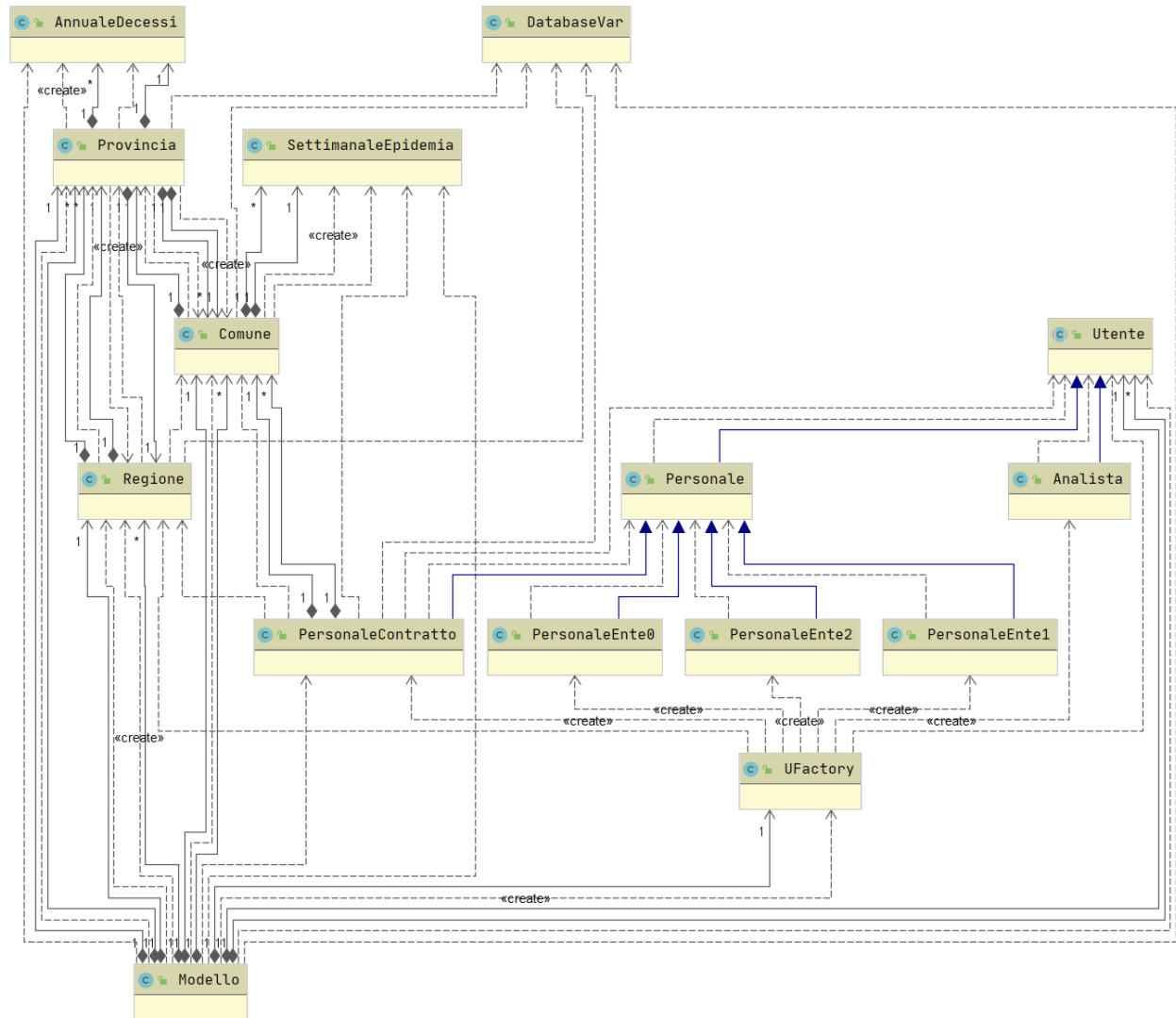
Diagramma di attività che descrive le funzioni svolte dall'analista ANL.



## 4 Diagrammi delle classi

Mostriamo ora il diagramma delle classi del sistema nella sua interezza. Successivamente andremo a visionare i gruppi di maggior interesse.

### Diagramma delle classi generale



Powered by yFiles

## 4.1 Controllori

Le seguenti classi sono addette alla gestione, modifica e visualizzazione dei dati presenti nel database. Ad ognuna di esse è legata un'interfaccia utente. I vari eventi generati dall'interazione utente-interfaccia vengono ricevuti ed elaborati da queste classi.

## 4.2 Suddivisione geografica

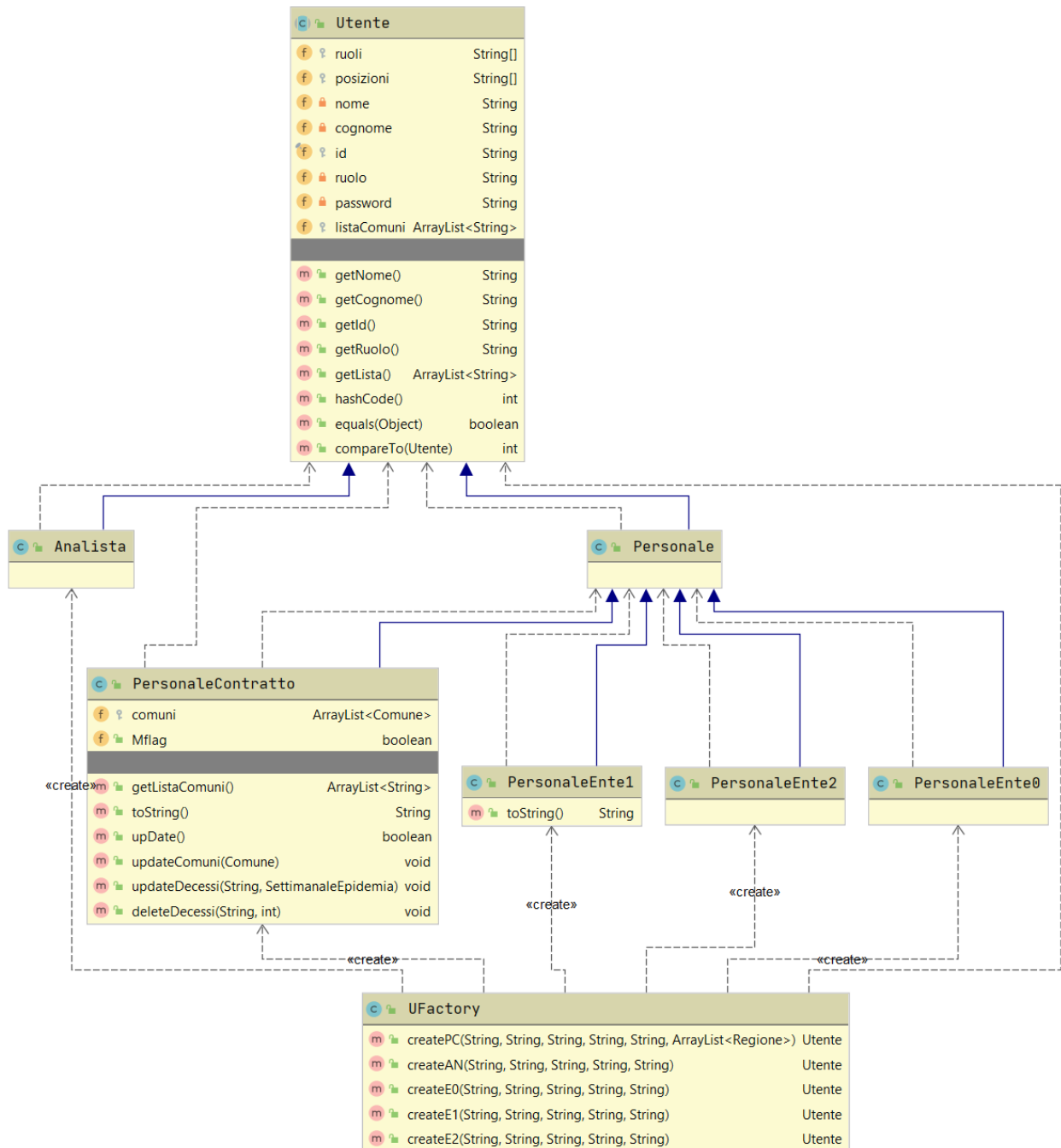
Questo insieme di classi permette l'archiviazione e manipolazione dei dati presenti all'interno del database. Le relazioni tra di esse costituiscono la gerarchia territoriale legata ai dati socio-sanitari.





### 4.3 Utenti

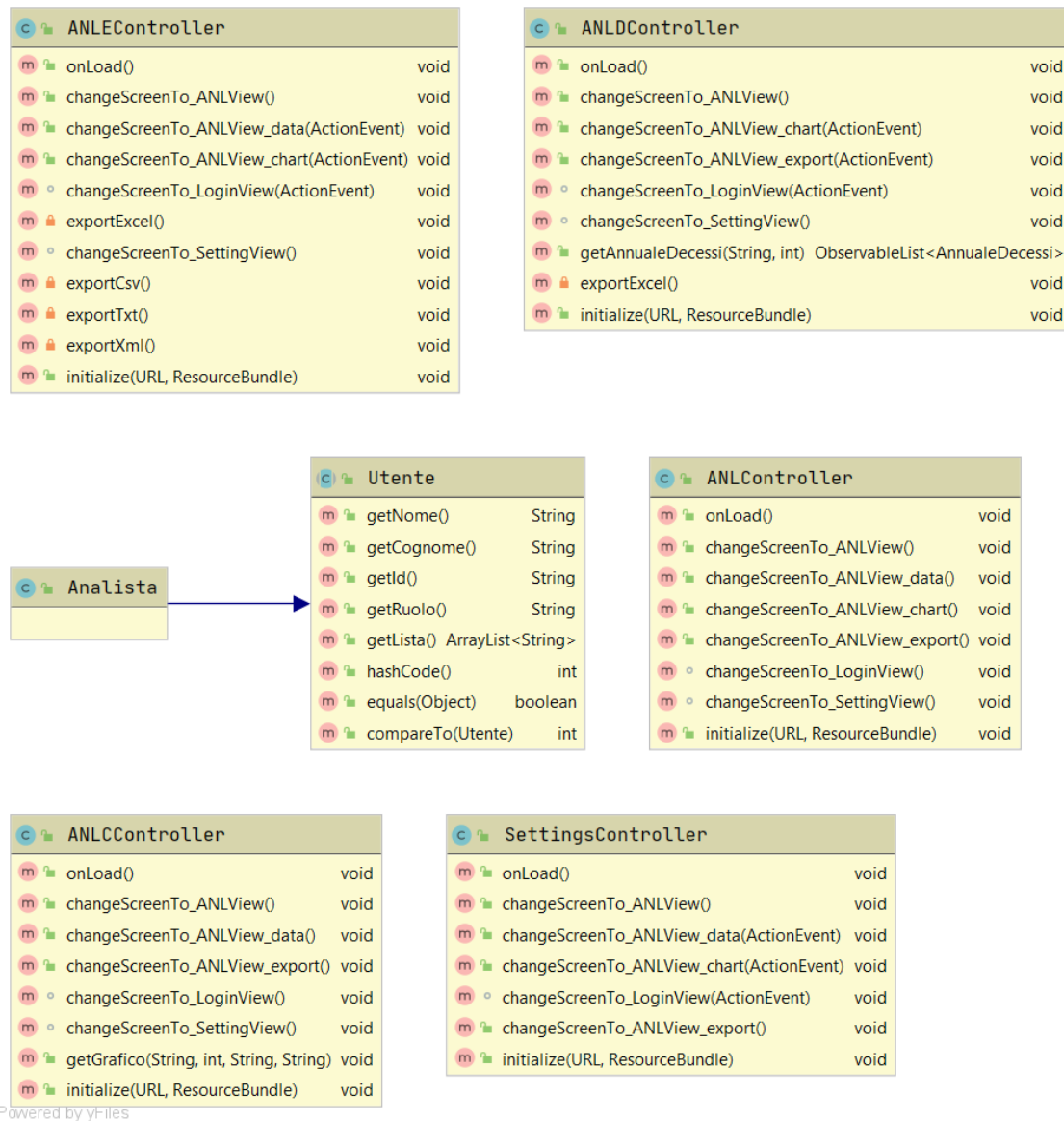
Queste classi gestiscono i dati relativi al personale. A partire dalla creazione come utente generico fino alla specializzazione del singolo ruolo, con relativo assegnamento delle varie autorizzazioni per operare all'interno del sistema informativo.



Powered by yFiles

## 4.4 Analista

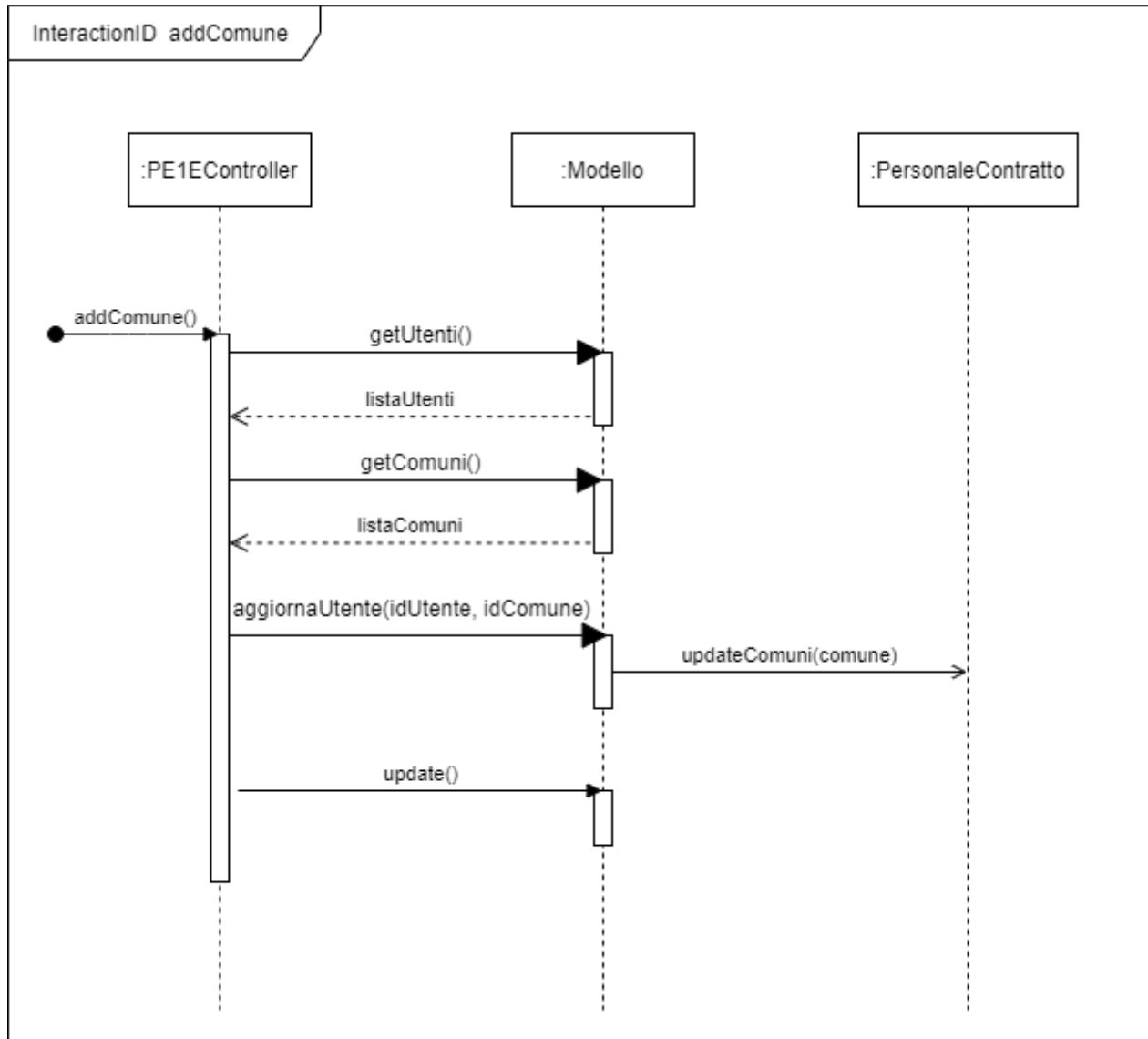
Un caso puntuale di utente è l'analista. Di seguito vengono riportate le classi che lo costituiscono e che gestiscono le sue interazioni con il sistema e conseguentemente con l'interfaccia grafica. I controllori dell'analista non sono direttamente collegati ad esso perché il modello svolge un ruolo di intermediazione.



## 5 Diagrammi di sequenza del software

Segue un diagramma di sequenza che mostra le dinamiche di alcune interazioni tra classi particolarmente interessanti. La seguente rappresentazione è in realtà un fax simile di molteplici casi all'interno del sistema che divergono solo per alcuni dettagli.

### *Modifica di un dato da parte del personale a contratto*



## 6 Sviluppo

Inizialmente abbiamo effettuato un'attenta analisi delle specifiche fornite per evidenziare le principali componenti da implementare. Abbiamo quindi suddiviso il progetto in micro porzioni ordinandole per priorità. In questo momento abbiamo anche cercato di prevedere le possibili complicanze di ogni segmento per agevolare il successivo lavoro di implementazione. Da qui in avanti lo sviluppo è stato sostanzialmente di tipo Agile ed incrementale. Abbiamo quindi deciso di cominciare con la realizzazione delle varie interfacce grafiche per chiarire quali sarebbero state poi le funzioni da implementare per garantire tutte le funzionalità richieste. A questo fine abbiamo utilizzato le librerie JavaFX e SceneBuilder per velocizzare il processo di design e scrittura del codice fxml. Ad ogni interfaccia fxml è stato collegato uno specifico controller per facilitare le comunicazioni interne al software. Il passo successivo è stato simulare un database attraverso l'uso di fogli di calcolo excel, gestiti attraverso la libreria apache-io. Abbiamo quindi scritto il file modello, e altre svariate classi java, che permette la comunicazione tra le interfacce fxml e i fogli excel.

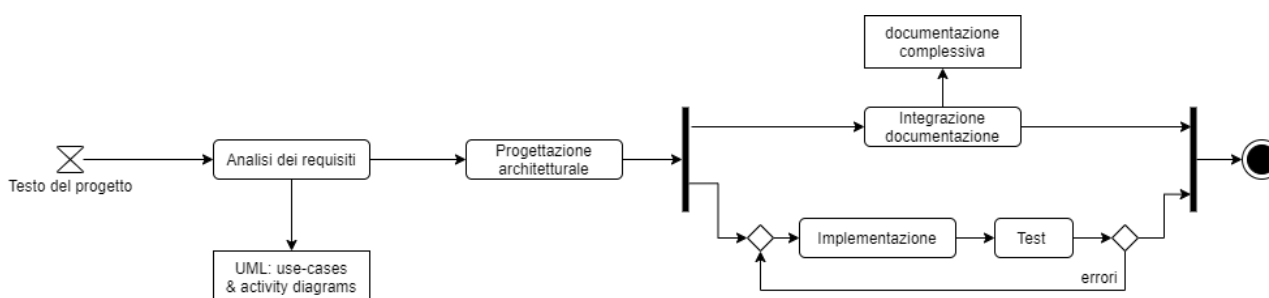
Per riuscire a lavorare tutti ad una versione aggiornata del progetto abbiamo utilizzato un sistema di versioning chiamato Space, dagli stessi sviluppatori di IntelliJ che è stato l'IDE sul quale abbiamo lavorato. Ogni push è stato preceduto da un'attenta revisione sulle modifiche apportate e da alcune simulazioni sul corretto funzionamento del segmento implementato.

La documentazione relativa alle classi interne è stata abbozzata nel momento precedente alla scrittura delle tali, per poi essere raffinata una volta concluso e riorganizzato il codice.

### Database

Per la gestione dei dati abbiamo deciso di creare quattro file excel:

- zone.xlsx : comprende 3 sheets contenenti le varie suddivisioni territoriali con i relativi identificatori univoci.
- utenti.xlsx : contiene i dati personali di ogni utente, quali credenziali ed incarichi/permessi. Per ogni personale a contratto è presente la lista dei suoi comuni di competenza.
- datiProvincia.xlsx : contiene i dati relativi ai decessi annuali suddivisi per provincia. Ogni provincia ha un suo sheet dedicato.
- datiComune.xlsx : contiene i dati epidemiologici settimanali suddivisi per comune. Ogni comune ha un suo sheet dedicato.



## 7 Pattern

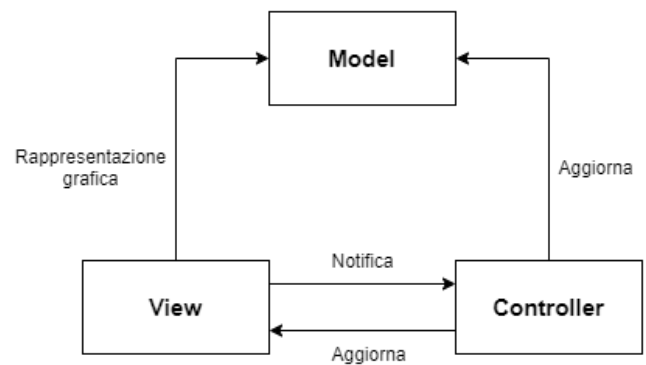
### MVC

Il sistema è stato realizzato seguendo lo schema architetturale del Model View Controller.

**Model:** è rappresentato dalla classe Modello.java, che permette la memorizzazione dell'informazione e la comunicazione tra tutte le componenti interne al sistema.

**View:** è composto da tutte le interfacce fxm, le quali attraverso generazione di eventi comunicano con i rispettivi controllori.

**Controller:** composto dall'insieme dei controllori, i quali recepiscono gli eventi da parte della View e conseguentemente aggiornano la stessa e il Model.



### SINGLETON

Il modello, che gestisce la comunicazione con tutti i fogli di calcolo, è stato realizzato seguendo il pattern singleton. Questo ci permette di avere, in qualsiasi punto del codice, il riferimento allo stesso oggetto modello. Essendo il modello l'unico oggetto in grado di accedere direttamente al database, riusciamo sempre a tenere quest'ultimo aggiornato e sincronizzato con le interazioni dei vari utenti.

### FACTORY

Per facilitare la creazione degli utenti abbiamo deciso di utilizzare questo pattern, che ci permette di avere maggiore chiarezza e diminuire ridondanza superflua. La creazione di un oggetto utente viene delegata all'oggetto UFactory il quale in base al tipo di utente va a chiamare il costruttore adeguato.

## **8 Attività di test e verifica**

Abbiamo testato l'applicazione durante tutto il suo sviluppo con semplici test mirati sulle nuove componenti appena realizzate. Inoltre, avendo effettuato la scrittura del codice sempre in parallelo ad altri membri del gruppo, c'è stato un controllo diretto anche durante ogni momento di scrittura. In una seconda fase l'abbiamo proposto anche a terze parti (sia nostri colleghi che familiari), raccogliendo le loro impressioni.

Non abbiamo utilizzato test JUnit.