```
mirror object to mirror
mirror_object
peration == "MIRROR_X":
eirror_mod.use_x = True
mirror_mod.use_y = False
__mod.use_z = False
 _operation == "MIRROR_Y"
lrror_mod.use_x = False
 lrror_mod.use_y = True
 lrror_mod.use_z = False
 operation == "MIRROR_Z";
  rror_mod.use_x = False
  rror_mod.use_y = False
  rror_mod.use_z = True
 melection at the end -add
   ob.select= 1
   er ob.select=1
   ntext.scene.objects.action
  "Selected" + str(modifier
   irror ob.select = 0
  bpy.context.selected_obj
   lata.objects[one.name].sel
  int("please select exactle
  OPERATOR CLASSES ----
    vpes.Operator):
    X mirror to the selected
   ject.mirror_mirror_x"
  ext.active_object is not
```

# Progetto Architettura del software

Federico Pisapia 816481 Luca Amorosia 816636





L'obiettivo di questo progetto è realizzare un sistema di gestione operativa per un azienda che si occupa di progettare e installare centraline di distribuzione dell'energia elettrica.

Queste centraline sono dotate di sensori per la misurazione istantanea della potenza erogata

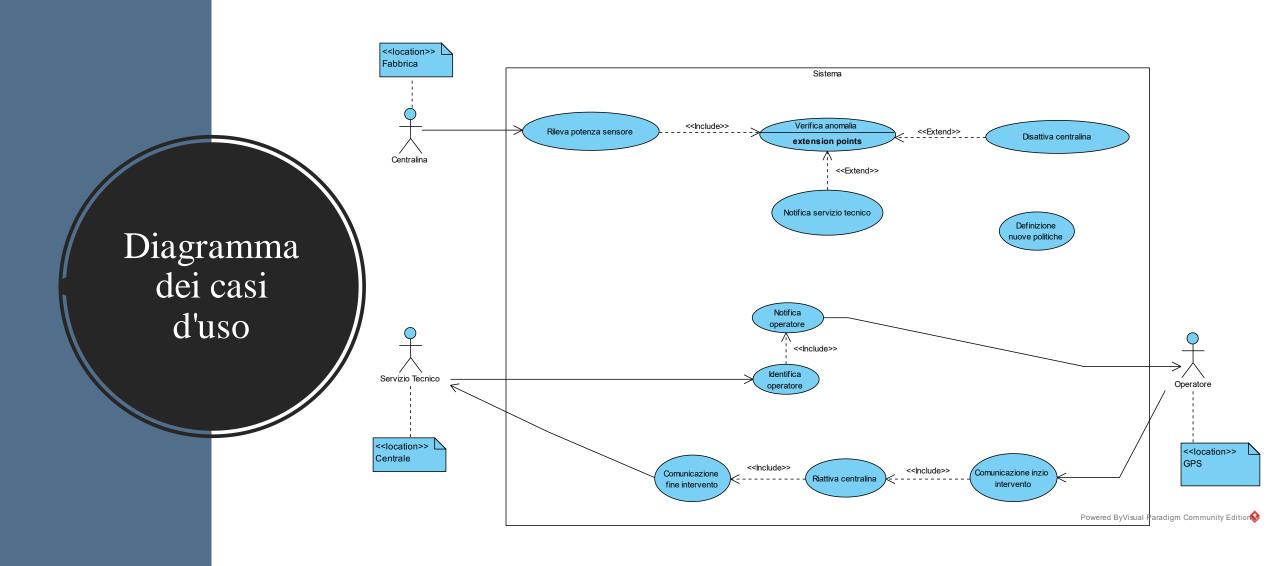
Le supposizioni iniziali sono state:

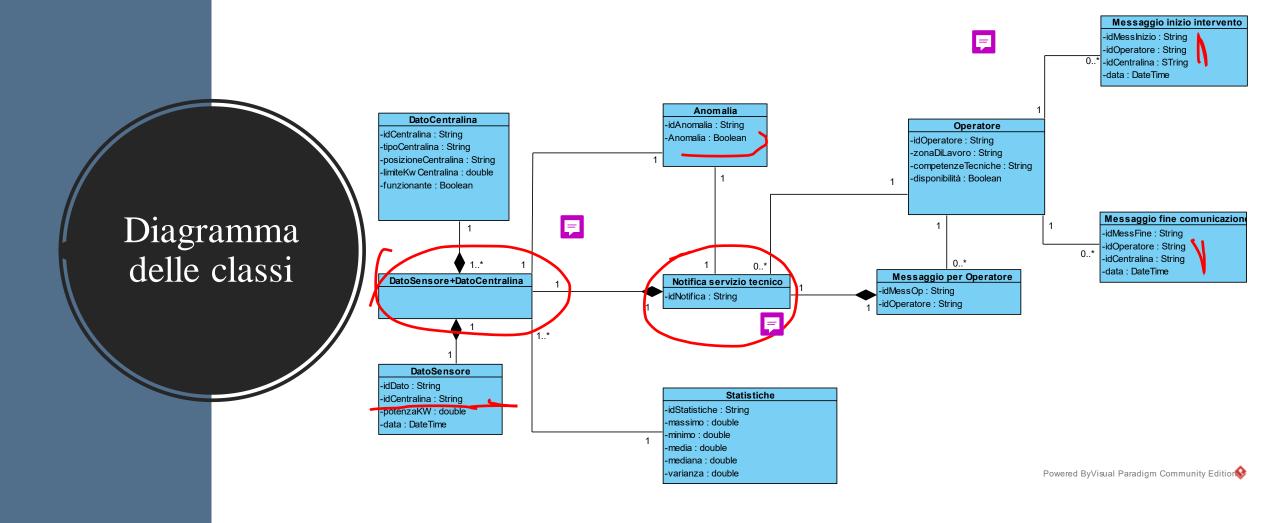
100 centraline

12 operatori

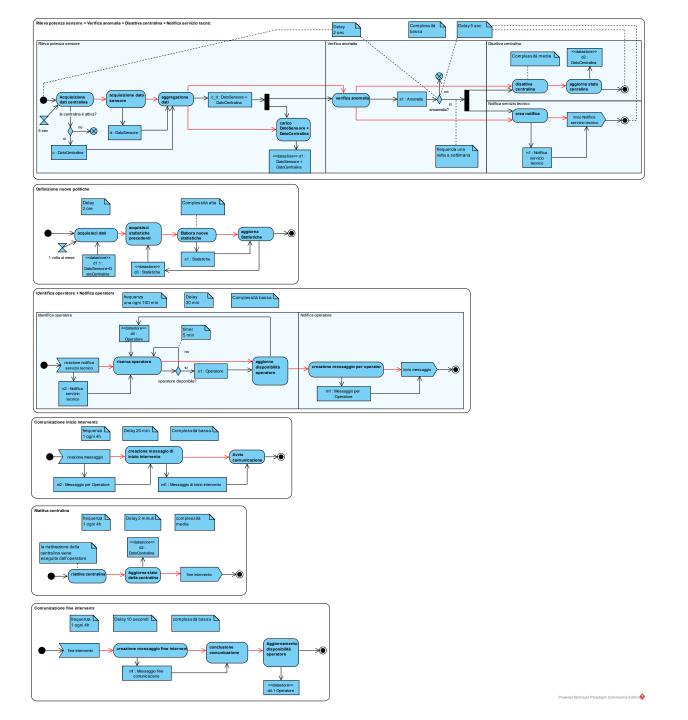
Ogni centralina ha in media un guasto a settimana

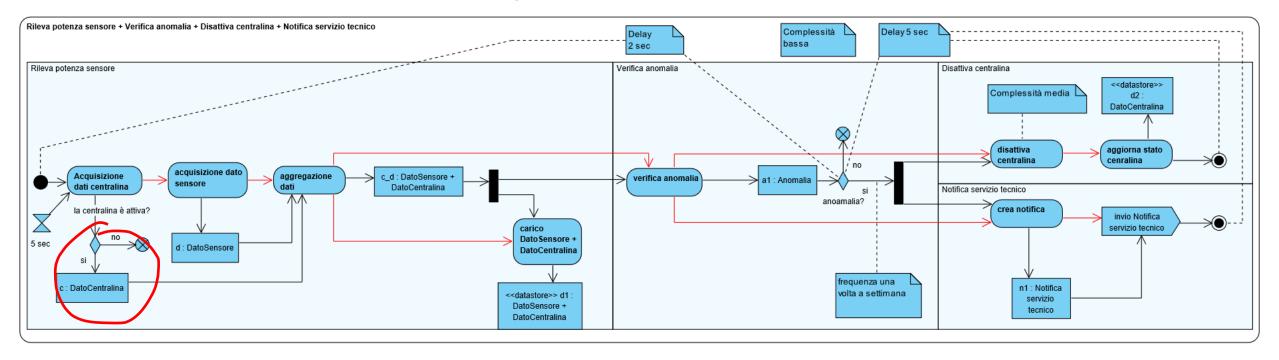
Queste supposizioni sono servite a definire le tempistiche del sistema.





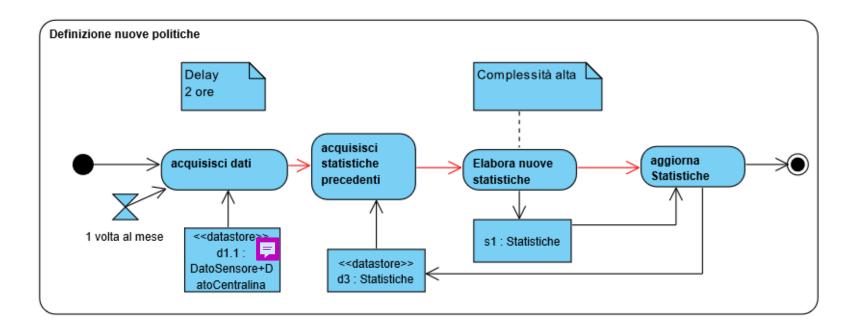






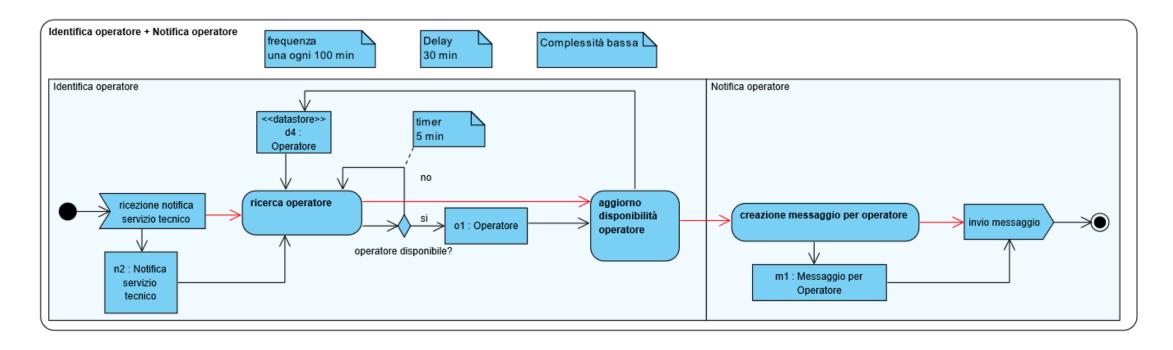
Questa activity rappresenta il caso d'uso *Rileva potenza sensore* che include il caso d'uso *Verifica anomalia* il quale è esteso dai casi d'uso *Disattiva centralina* e *Notifica servizio tecnico*. L'attività si occupa di:

- acquisire i dati della centralina:
- Se la centralina è spenta terminare l'esecuzione
- Se invece la centralina è attiva acquisire il valore del sensore
  - aggregare questi due dati e salvare il dato risultante nel datastore
  - controllare se è il dato è anomalo:
    - In caso positivo disattivare la centralina e mandare una notifica al servizio tecnico
    - In caso negativo terminare l'esecuzione



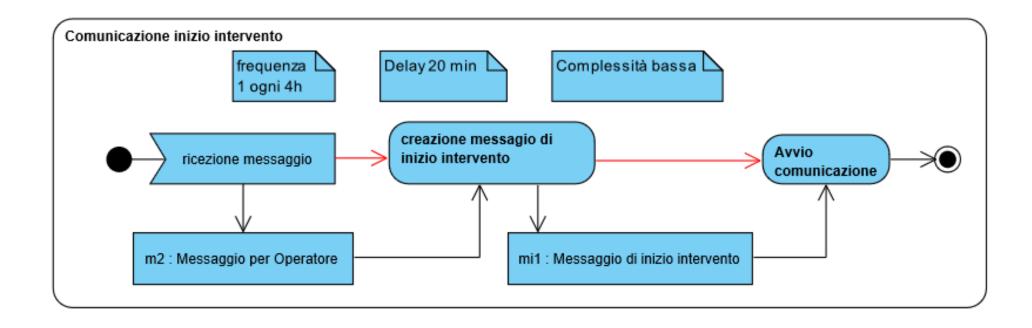
L'activity rappresenta il caso d'uso Definizione nuove politiche e si occupa di

- acquisire una volta al mese i dati aggregati della centralina e del sensore dal datastore
- acquisire le precedenti statistiche dal datastore
- elaborare delle nuove statistiche e aggiornare il datastore

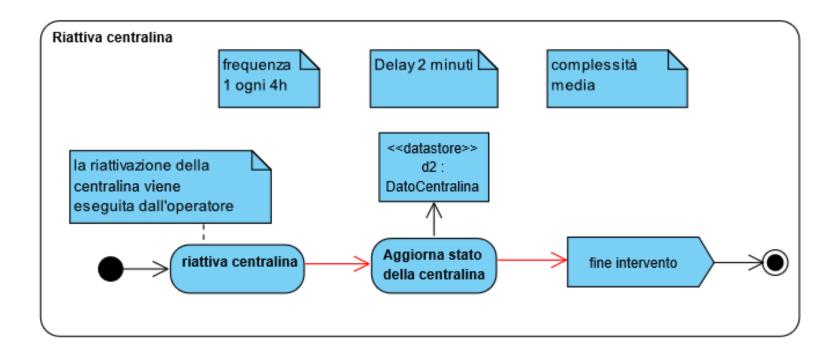


L'activity rappresenta il caso d'uso *Identifica operatore* che include il caso d'uso *Notifica operatore*. Si occupa di:

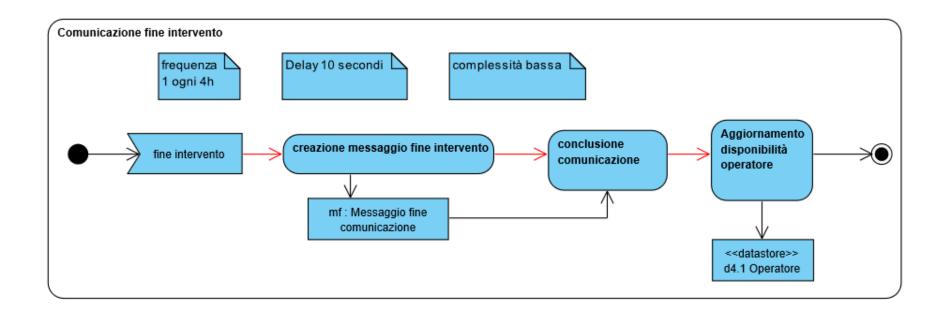
- ricevuta la notifica sull'anomalia, ricercare l'operatore adatto nel datastore
  - se l'operatore è disponibile, modificare la sua disponibilità, creare il messaggio per l'operatore e inviarlo
  - se invece l'operatore non è disponibile aspettare 5 min prima di rieseguire la ricerca dell'operatore



L'activity rappresenta il caso d'uso *Comunicazione inizio intervento* e si occupa di, ricevuto il messaggio dal servizio tecnico, creare il messaggio di inizio intervento e inviarlo al servizio tecnico per comunicare di aver iniziato l'intervento

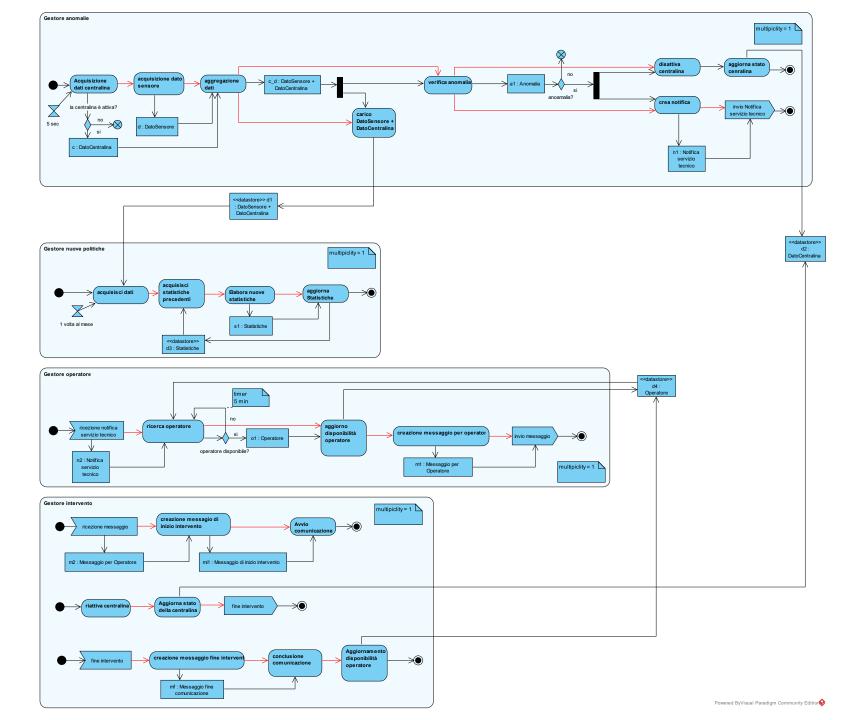


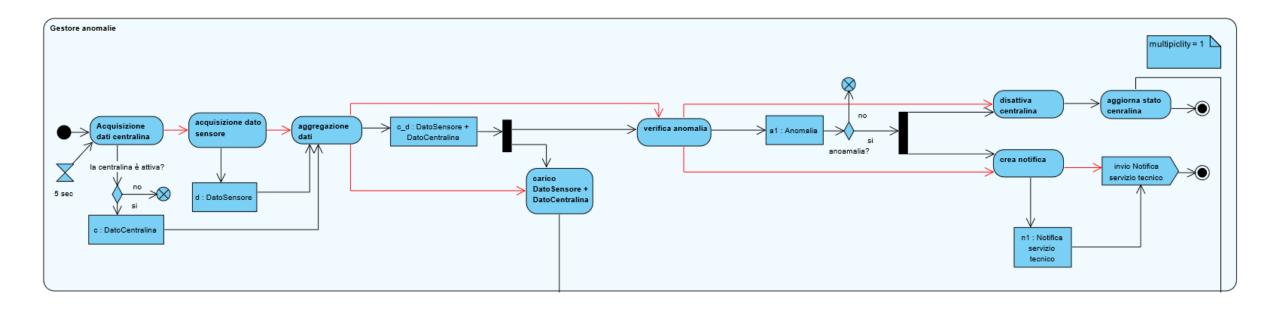
L'activity rappresenta il caso d'uso *Riattiva centralina* che si occupa di riattivare la centralina dopo che l'operatore l'ha riattivata fisicamente e aggiorna l'attributo *funzionante* della classe *Centralina* nel datastore.



L'activity rappresenta il caso d'uso *Comunicazione fine intervento* che si occupa di creare il messaggio di fine intervento, inviarlo al servizio tecnico e aggiornare l'attributo *disponibilità* della classe *Operatore* nel datastore.



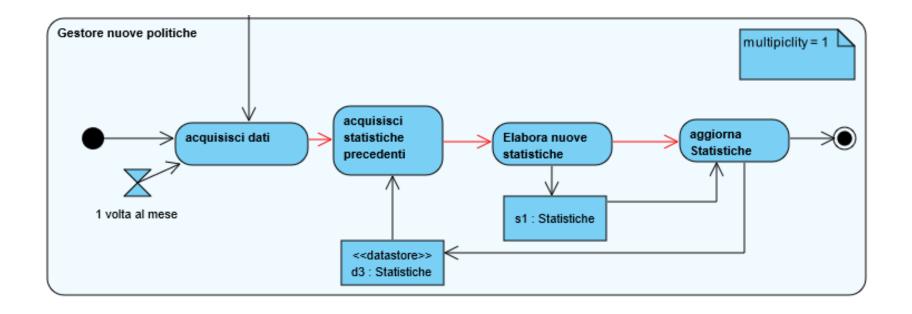




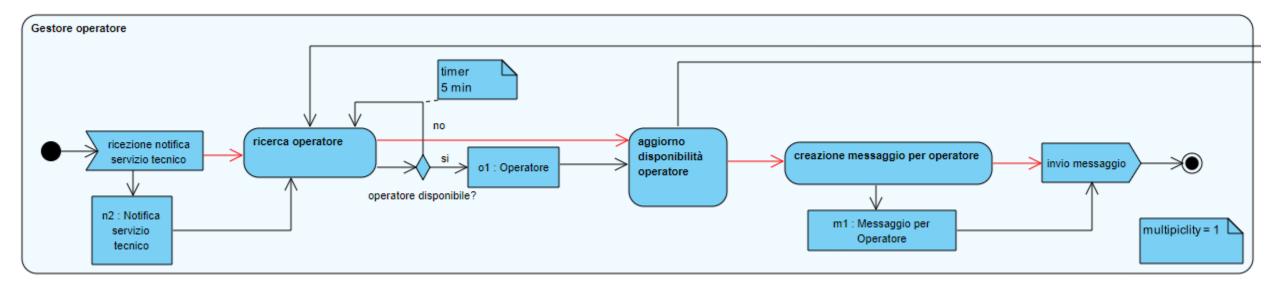
Vengono raggruppate le attività con lo scopo di ottenere componenti compatti e isolati. La componente *Gestore anomalie* è costituita dalle attività *Rileva potenza*, *Verifica anomalia*, *Disattiva centralina* e *Notifica servizio* tecnico.

Si occupa di rilevare la potenza del sensore, verificare se il dato è anomalo e nel caso disattivare la centralina e inviare la notifica al servizio tecnico.

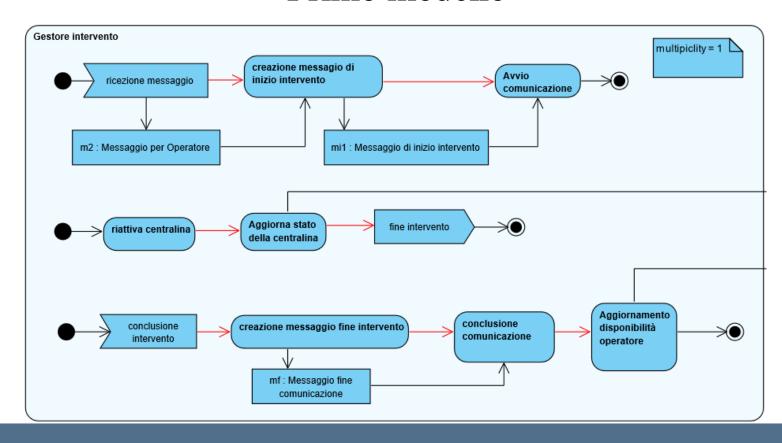
I datastore *DatoSensore+DatoCentralina* e *DatoCentralina* sono esterni al componente. La molteplicità di questa componente è 1.



La componente *Gestore nuove politiche* è costituita dall'attività *Definizione nuove politiche*. Si occupa di acquisire i dati *DatoSensore+DatoCentralina* e le *Statistiche* precedenti per elaborarne di nuove. Il datastore *DatoSensore+DatoCentralina* è esterno al componente. La molteplicità di questa componente è 1.



La componente *Gestore operatore* è costituita dalle attività I*dentifica operatore e Notifica operatore*. Si occupa di identificare l'operatore adeguato, utilizzando il datastore esterno *Operatore*, per poi contattarlo. La molteplicità di questa componente è 1.



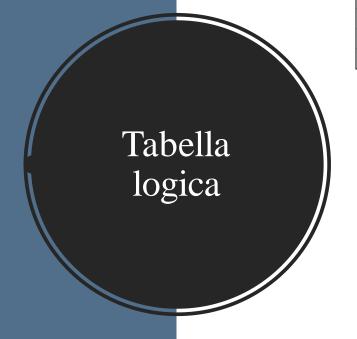
La componente Gestore intervento è costituita dalle attività Comunicazione inizio intervento, Riattiva centralina e Comunicazione fine intervento.

Queste attività sono state raggruppate in un unico componente poiché insieme rappresentano l'intero intervento dell'operatore sulla centralina.

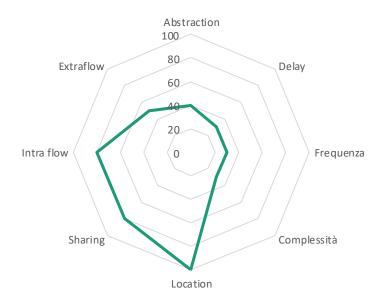
Si occupa di tutte quelle operazioni legate all'operatore come la comunicazione con il servizio tecnico e il rispristino della centralina.

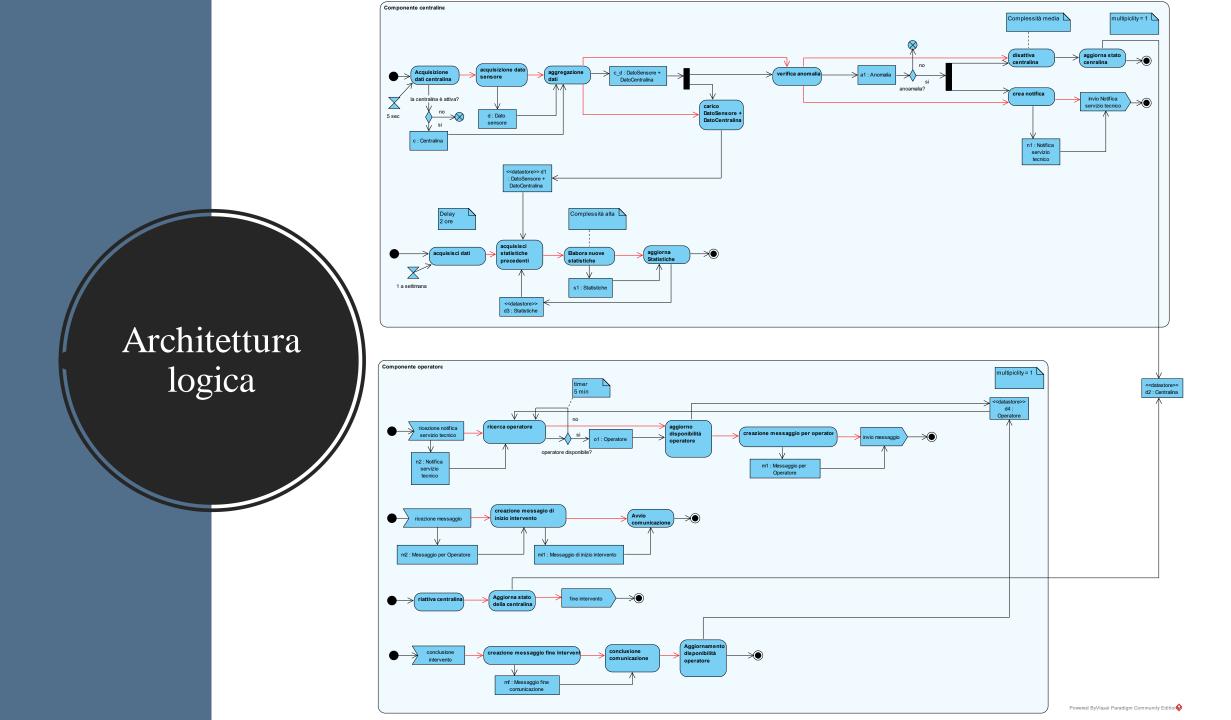
I datastore DatoCentralina e Operatore sono esterni al componente.

La molteplicità di questa componente è 1.

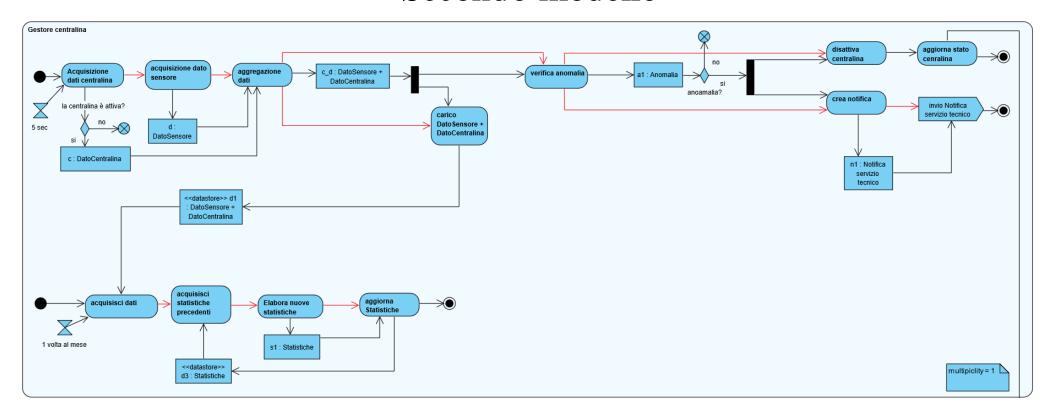


Abstraction	40	Medio-bassa, le componenti gestiscono differenti tipi di dati
Delay	30	Bassa, la maggior parte dei componenente ha livelli omogenei di delay
Frequenza	30	Bassa, ogni componente ha livelli omogenei di frequenza
Complessità	30	Media, ogni componente ha livelli omogenei di complessità
Location	100	Alta, gli attori sono sparsi in luoghi molto differenti in tutti i momenti
Sharing	80	Alta, poiché sono presenti molte condivisioni di dati con datastore esterni
Intra flow	80	Alta, ogni componenete interagisce con un altro componenente
Extraflow	50	Media, le componenti interagiscono con un solo tipo di attore, che è multi-istanziato, tranne servizio tecnico





## Secondo modello



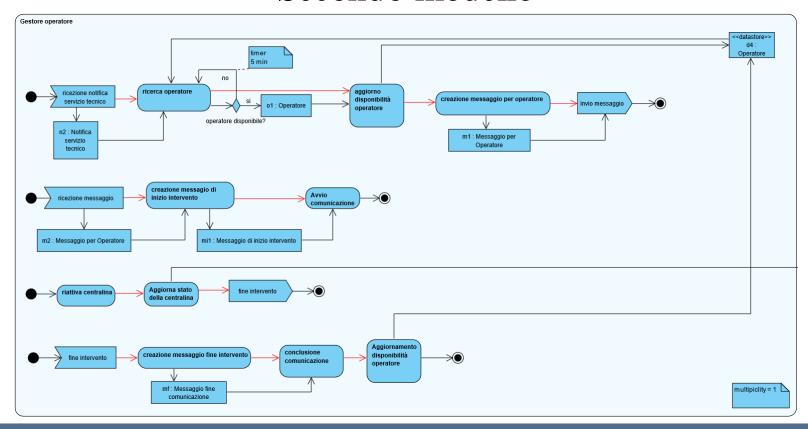
La componente Gestore Centralina è costituita dalle attività Rileva potenza sensore, Verifica anomalia, Disattiva centralina, Notifica servizio tecnico e Definizione nuove politiche.

Queste attività sono state raggruppate come unico componente per la condivisione del datastore *DatoSensore+DatoCentralina* per diminuire lo sharing dell'architettura.

Il datastore DatoCentralina è esterno al componente.

La molteplicità di questa componente è 1.

## Secondo modello

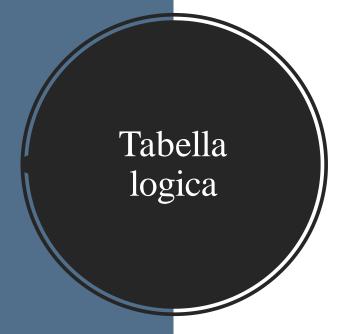


La componente Gestore operatore è costituita dalle attività Identifica operatore, Notifica operatore, Comunicazione inizio intervento, Riattiva centralina e Comunicazione fine intervento.

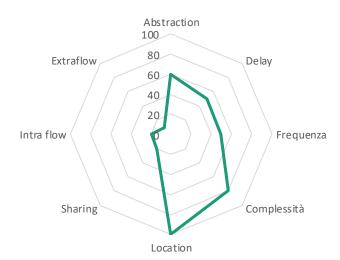
Queste attività sono state raggruppate come unico componente per la condivisione del datastore *Operatore* per diminuire lo sharing dell'architettura.

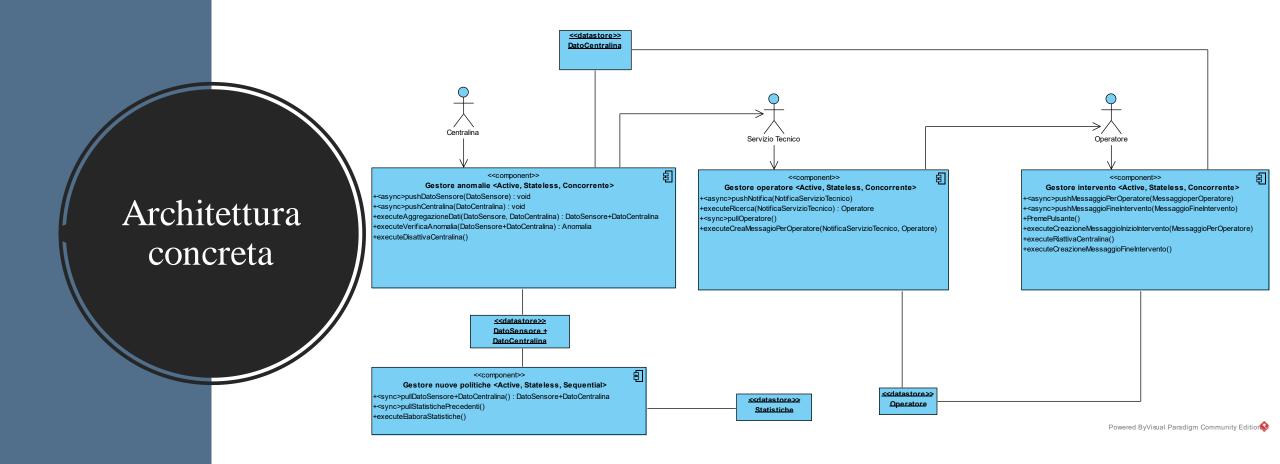
Il datastore *Dato Centralina* è esterno al componente.

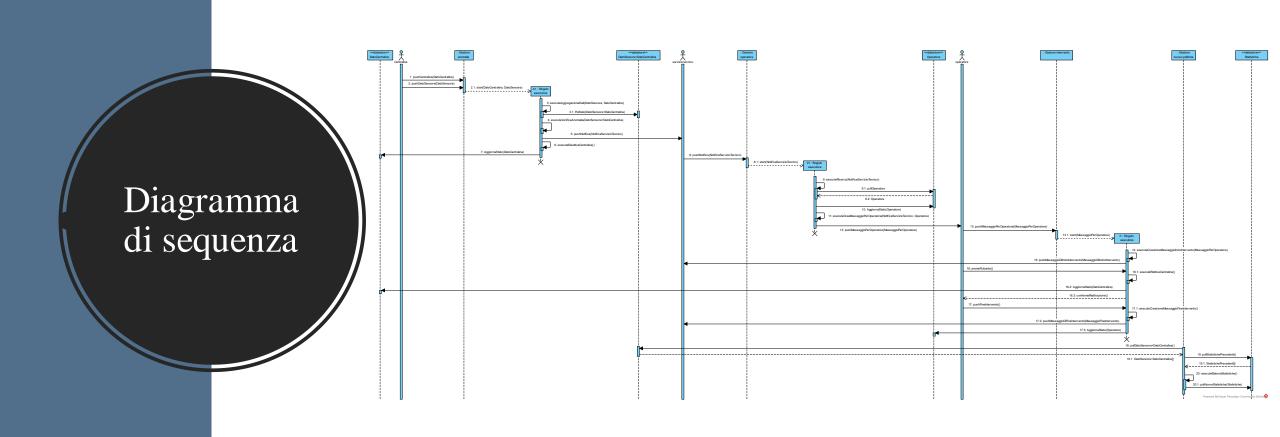
La molteplicità di questa componente è 1.

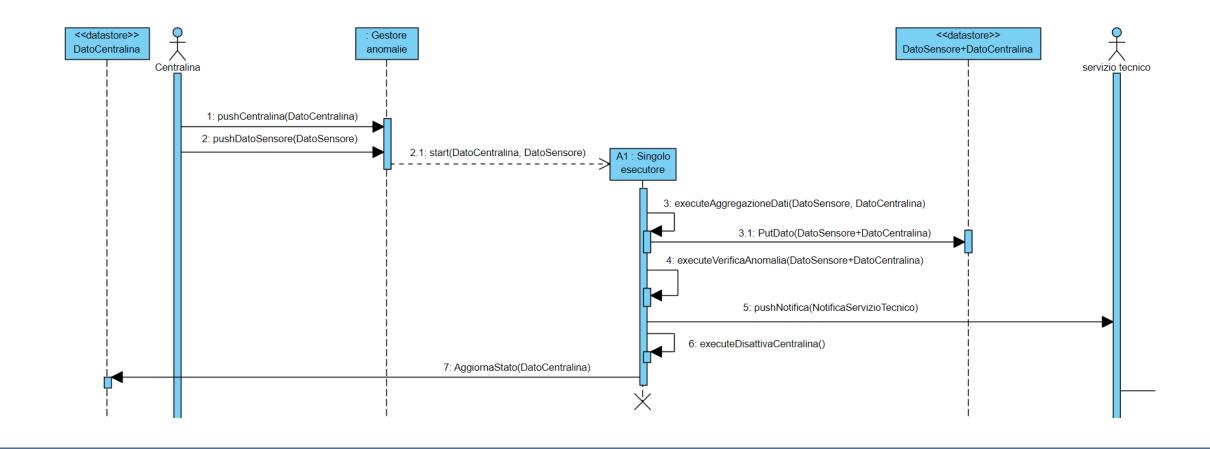


Abstraction	60	Medio-alta, le componenti gestiscono quasi tutti i tipi di dati
Delay	50	Medio, le componenti non hanno livelli omogenei di delay
Frequenza	50	Medio, le componenti non hanno livelli omogenei di frequenza
Complessità	80	Alta, ogni componente ha livelli differenti di complessità
Location	100	Alta, gli attori sono sparsi in luoghi molto differenti in tutti i momenti
Sharing	20	Bassa, vi è un solo datastore esterno alle componenti
Intra flow	20	Bassa, solo il componente 'Gestore centralina' interagisce con il componente 'Gestore operatore'
Extraflow	10	Basso, non interagiscono con attori esterni



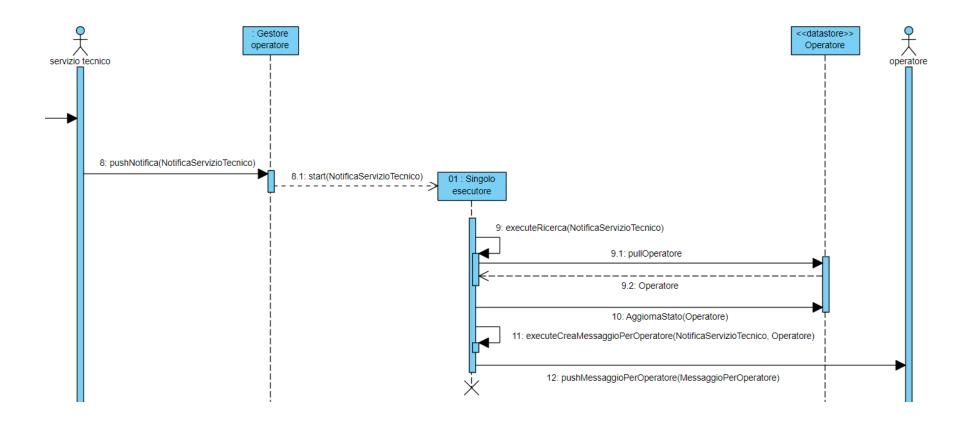




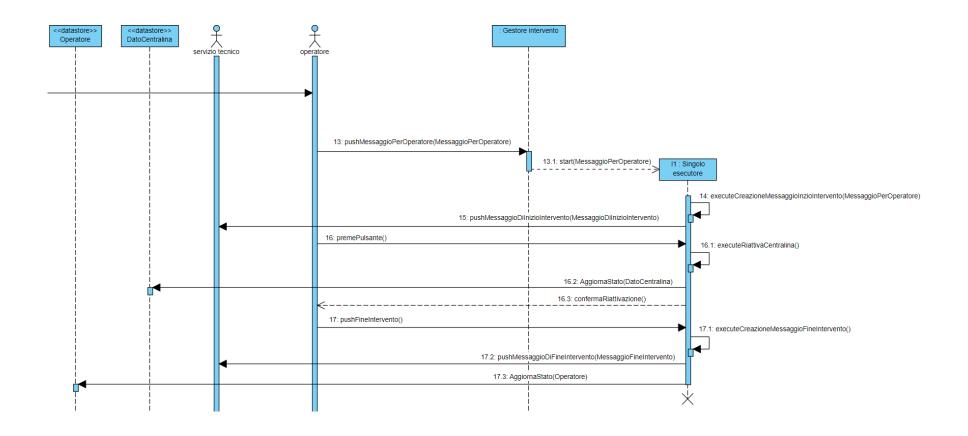


Questo diagramma di sequenza mostra le azioni della componente *Gestore anomalie*.

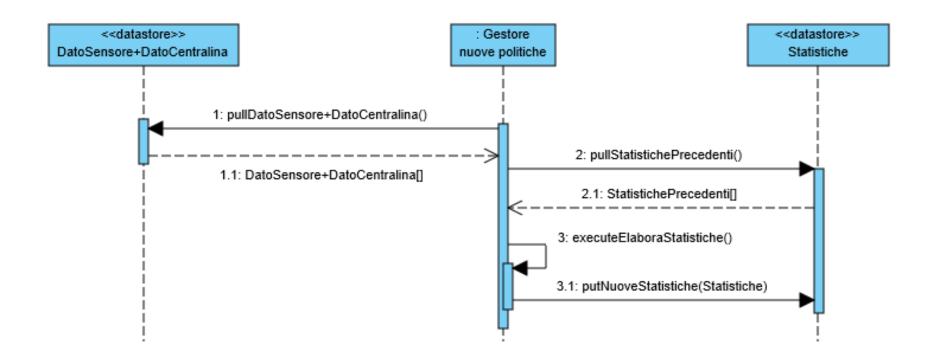
Questa componente è attiva, stateless e concorrenziale. Viene gestito in modo concorrenziale in quanto il *Gestore anomalie* dovrà gestire differenti dati provenienti da differenti centraline.



Questo diagramma di sequenza mostra le azioni della componente *Gestore operatore*. Questa componente è attiva, stateless e concorrenziale. Viene gestito in modo concorrenziale in quanto il *Gestore operatore* dovrà gestire differenti dati provenienti dal *Servizio tecnico*.



Questo diagramma di sequenza mostra le azioni della componente *Gestore intervento*. Questa componente è attiva, stateless e concorrenziale. Viene gestito in modo concorrenziale in quanto il *Gestore intervento* dovrà gestire differenti dati provenienti dal *Operatore*.



Questo diagramma di sequenza mostra le azioni della componente *Gestore nuove politiche*.

Questa componente è attiva, stateless e sequenziale. Viene gestito in modo sequenziale in quanto il *Gestore nuove politiche* viene creato una volta per tutte e richiede mensilmente i dati necessari per eseguire le sue azioni.