



Traffic monitor

Ingegneria del software 2018/2019

Mazzoleni Sara
Megler Federico
Paone Angelo

Indice

Uml

- Use Case Diagram pag. 3
- Class Diagram pag. 4
- Activity Diagram – Accesso Sistema Utente pag. 5
- Activity Diagram – Accesso Sistema Admin pag. 6
- Activity Diagram – Inserimento Segnalazione pag. 7
- Activity Diagram – Invio Notifica pag. 8
- Collaboration Diagram – Interazione Notifica pag. 9
- Component Diagram pag. 10
- Deployment Diagram pag. 11
- Object Diagram – Aggiornamento Centralina pag. 12
- Object Diagram – Applicazione pag. 13
- Sequence Diagram – Accesso Sistema Utente pag. 14
- Sequence Diagram – Funzionamento Centralina pag. 15
- Sequence Diagram – Invio segnalazione pag. 16
- State Diagram – Centralina pag. 17
- State Diagram – Segnalazione pag. 18

Uml

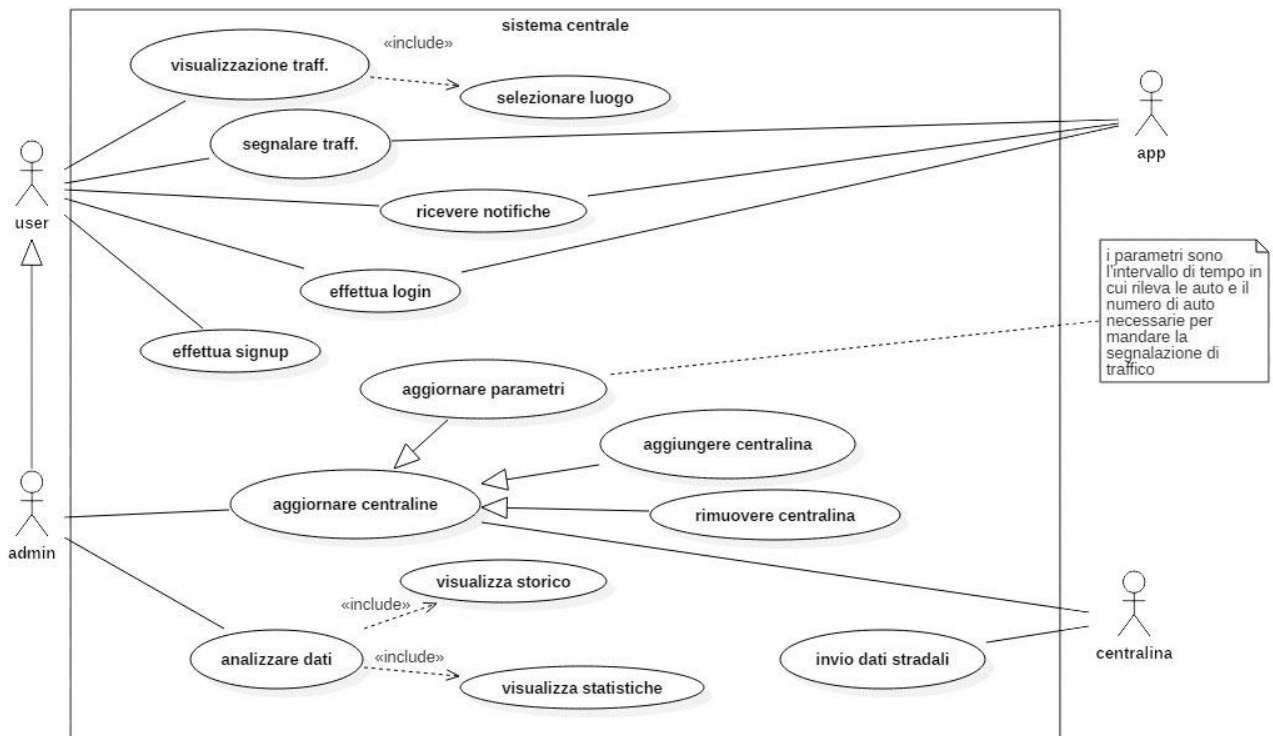
UseCase Diagram

Identificati i quattro attori abbiamo individuato tutte le attività ad essi collegate. L'**utente** deve visualizzare lo stato del traffico, fare segnalazione sul traffico e ricevere notifiche, dopo aver effettuato il login, possibile dopo essersi registrato.

L'**admin** ha inoltre il compito di aggiornare le centraline e di analizzare i dati del traffico, visualizzando storico e statistiche.

La **centralina** deve installare gli aggiornamenti ricevuti e inviare dati stradali al sistema centrale.

L'**app mobile** deve permettere di segnalare traffico e di ricevere notifiche.

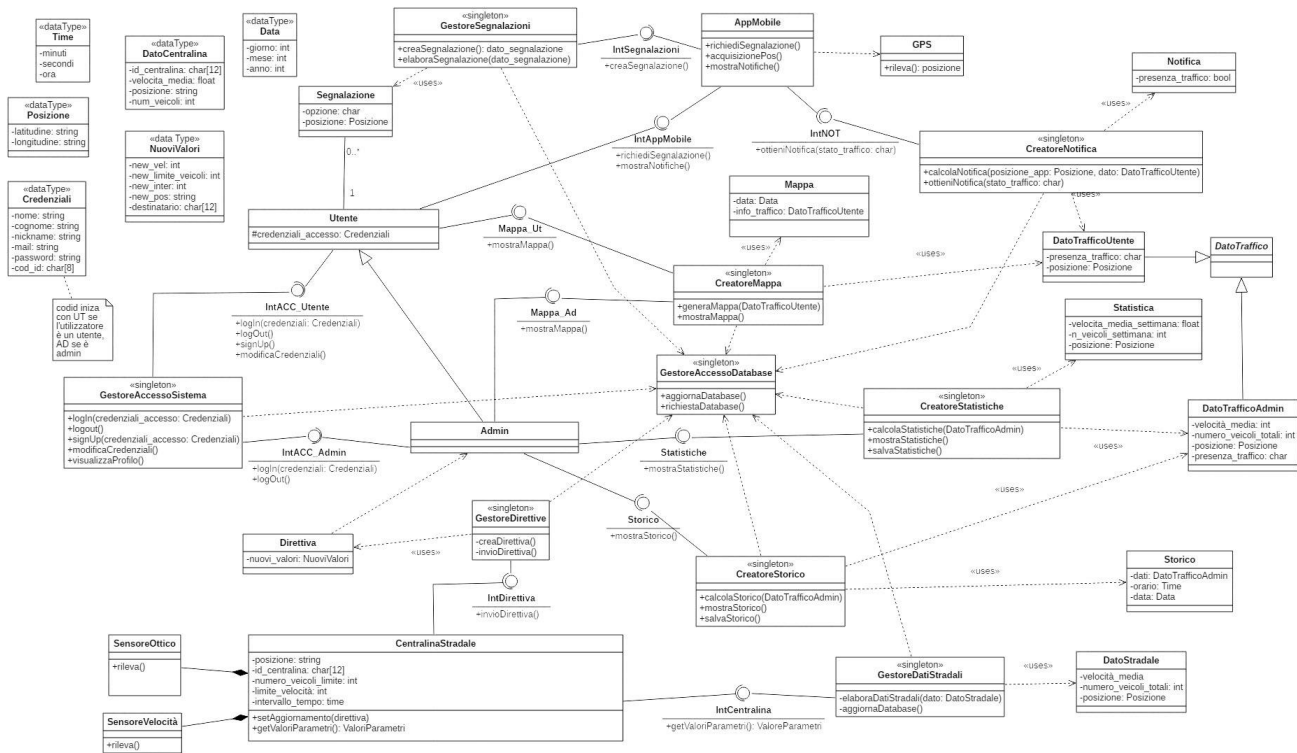


Class Diagram

Partendo dallo use case diagram abbiamo creato il class diagram generale del sistema.

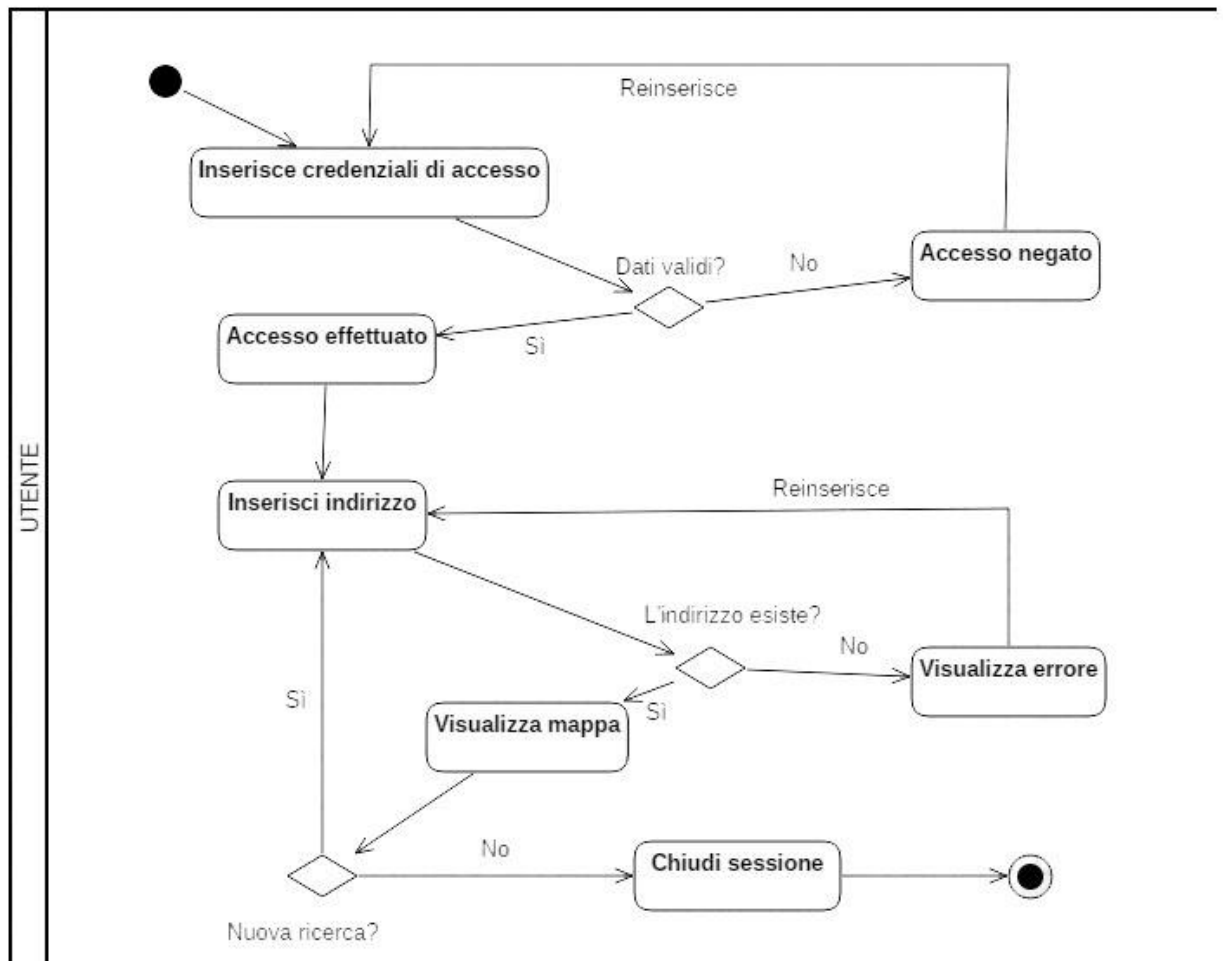
Sono state individuate le seguenti classi fondamentali: **Utente**, **Admin**, **Segnalazione**, **Notifica**, **AppMobile**, **Mappa**, **Dato traffico (DatoTrafficoUtente & DatoTrafficoAdmin)**, **Statistica**, **Direttiva**, **Storico**, **CentralinaStradale** e **DatoStradale**, correlate delle seguenti classi per gestirle: **GestoreAccessoSistema**, **GestoreSegnalazioni**, **CreatoreNotifica**, **CreatoreMappa**, **GestoreAccessoDatabase**, **CreatoreStatistiche**, **GestoreDirettiva**, **CreatoreStorico** e **GestoreDatiStradali**.

Abbiamo anche creato strutture di supporto per la descrizione del sistema, quali datatype: **Time**, **Data**, **DatoCentralina**, **Posizione**, **Credenziali** e **NuoviValori**.



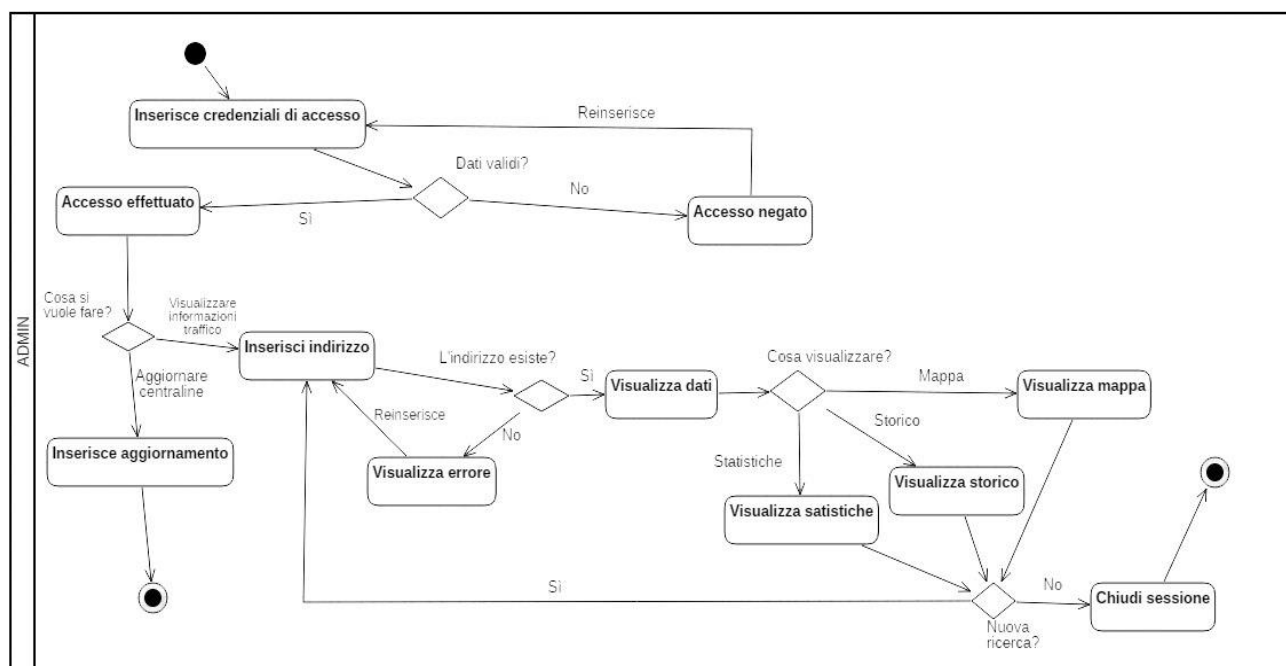
Activity Diagram – Accesso Sistema Utente

L'utente effettua l'accesso al sistema e inserendo un indirizzo può vedere la situazione del traffico in quella zona tramite una mappa, in seguito si disconnette.



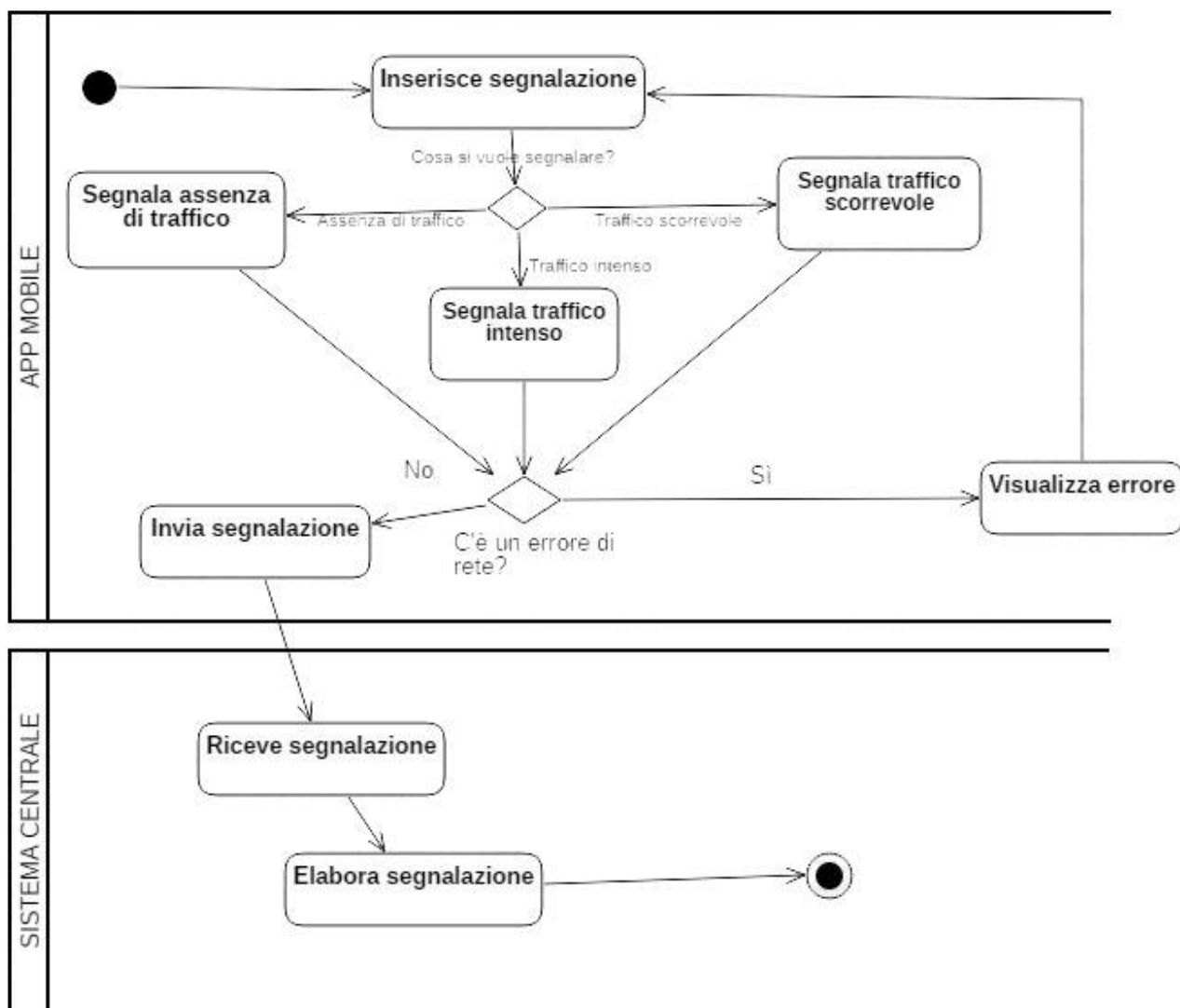
Activity Diagram – Accesso Sistema Admin

L'admin effettua l'accesso al sistema e può vedere i dati a lui relativi (statistiche, storico, mappa) oppure inserire un aggiornamento per le centralina, in seguito si disconnette.



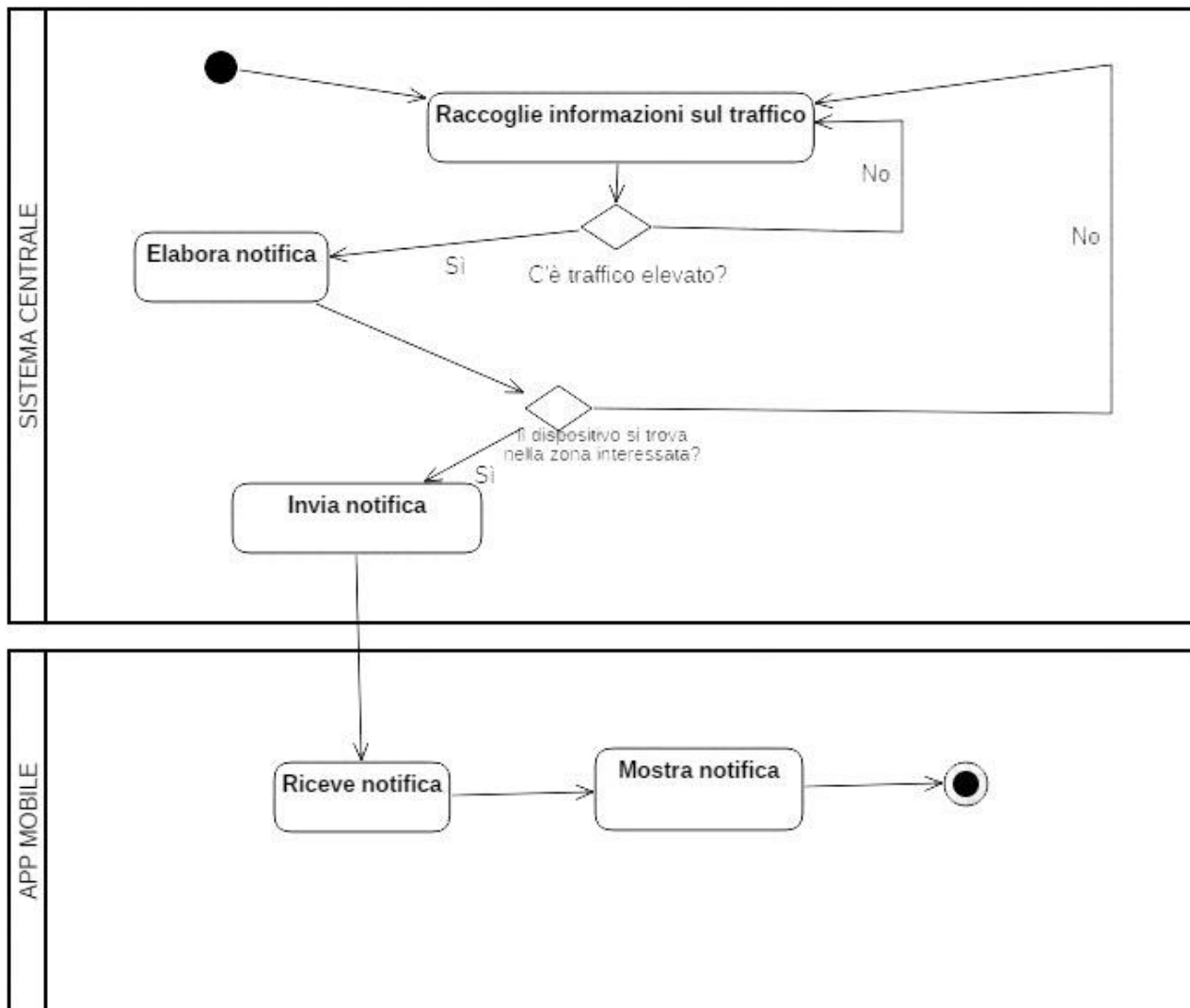
Activity Diagram – Inserimento segnalazione

L'applicazione riceve una segnalazione sulla situazione del traffico (assente, scorrevole o intenso) dall'utente e la invia al sistema centrale, che la riceve ed elabora.



Activity Diagram – Invio notifica

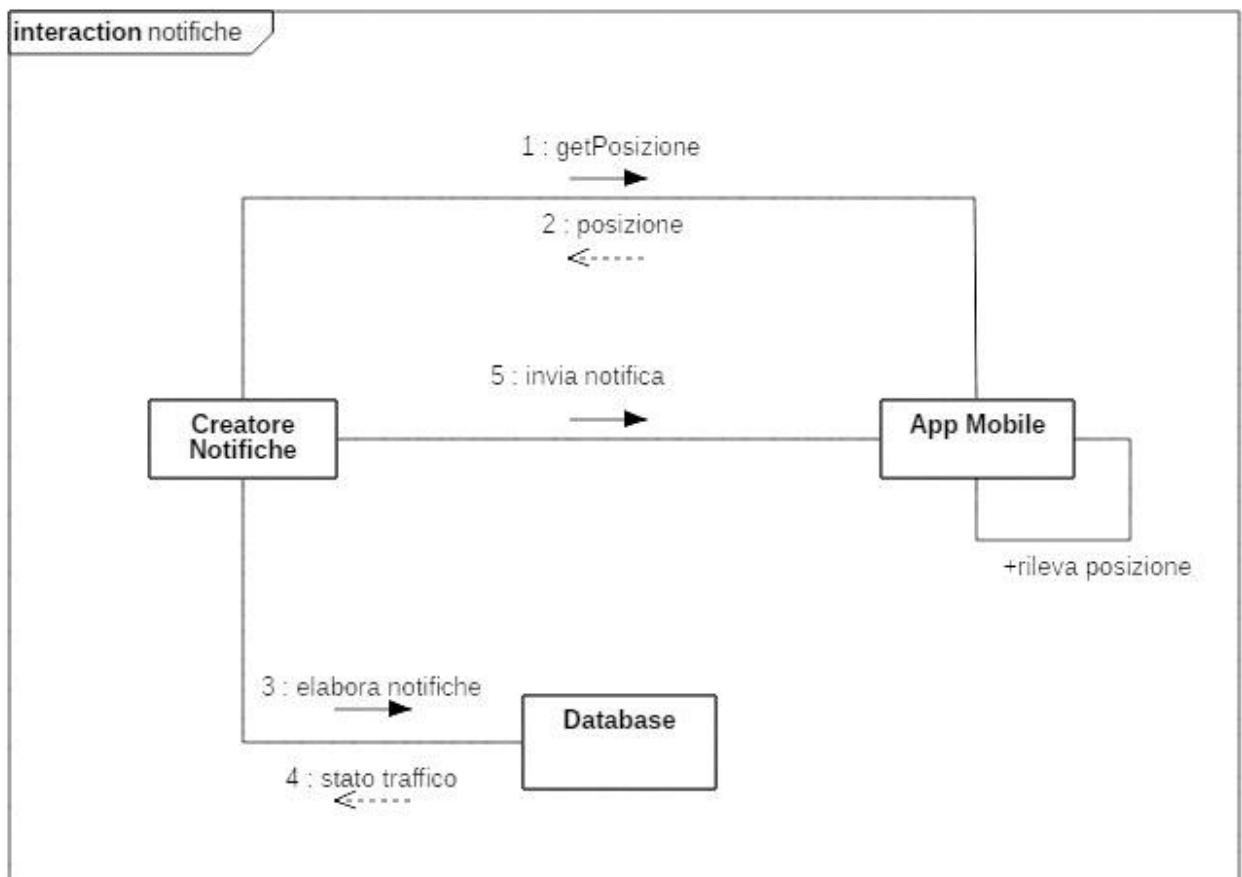
Ricevuti tutti i dati stradali, il sistema centrale li elabora per rilevare situazioni critiche di traffico e manda una notifica alle applicazioni mobile nella zona interessata, le quali la ricevono e la mostrano.



Collaboration Diagram – Interazione notifica

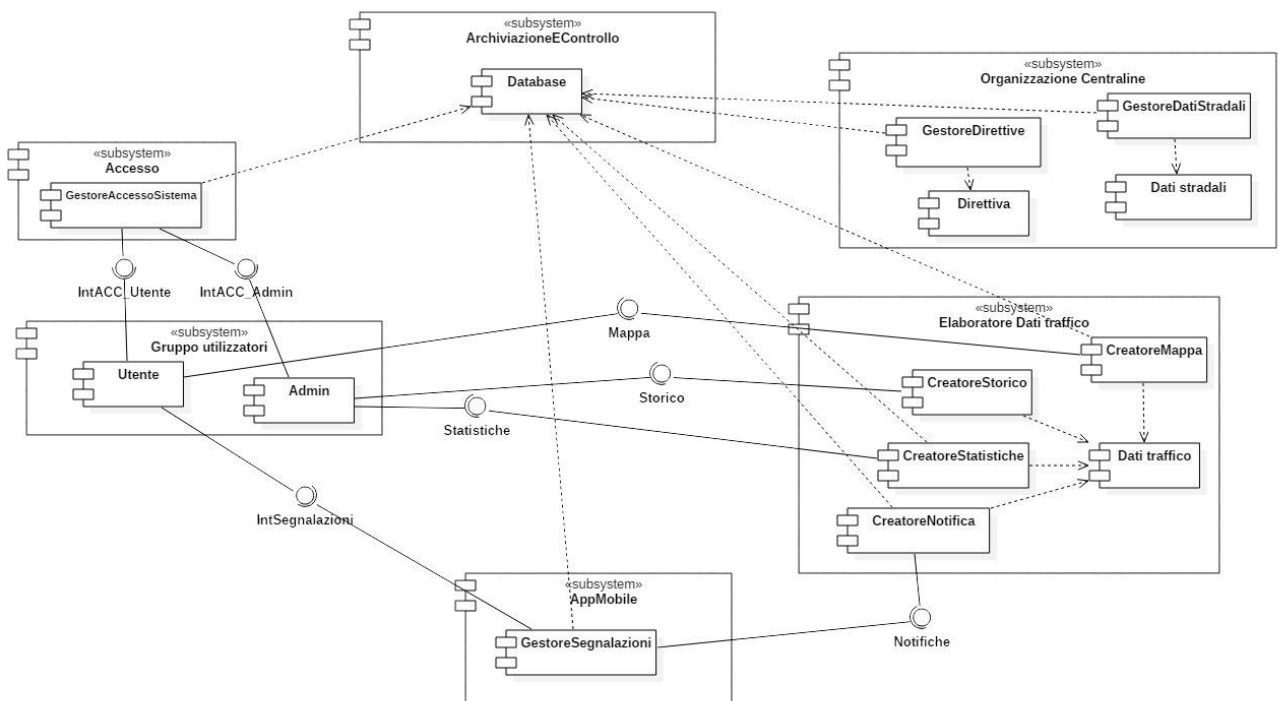
Nel diagramma sono riportate le varie cooperazioni del sistema al fine di creare una notifica.

In particolare, l'**AppMobile** rileva la propria posizione, la invia al **CreatoreNotifiche**, il quale elabora la notifica ricevendo le relative informazioni sul traffico dal **Database**, e la invia all'app.



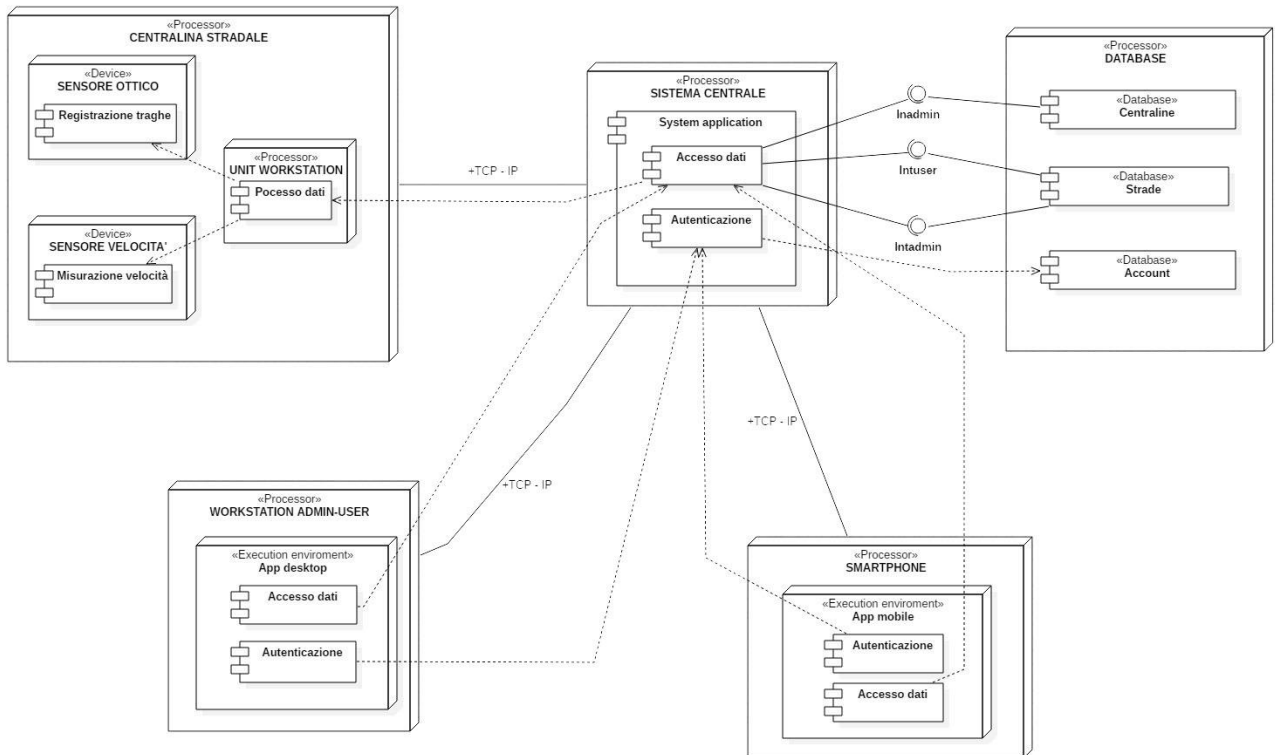
Component Diagram

Abbiamo diviso il sistema centrale in più sottosistemi che si dividono i compiti per la gestione dell'ecosistema. Tramite questo diagramma delle componenti descriviamo le relazioni tra i vari sottosistemi e le interfacce che le permettano.



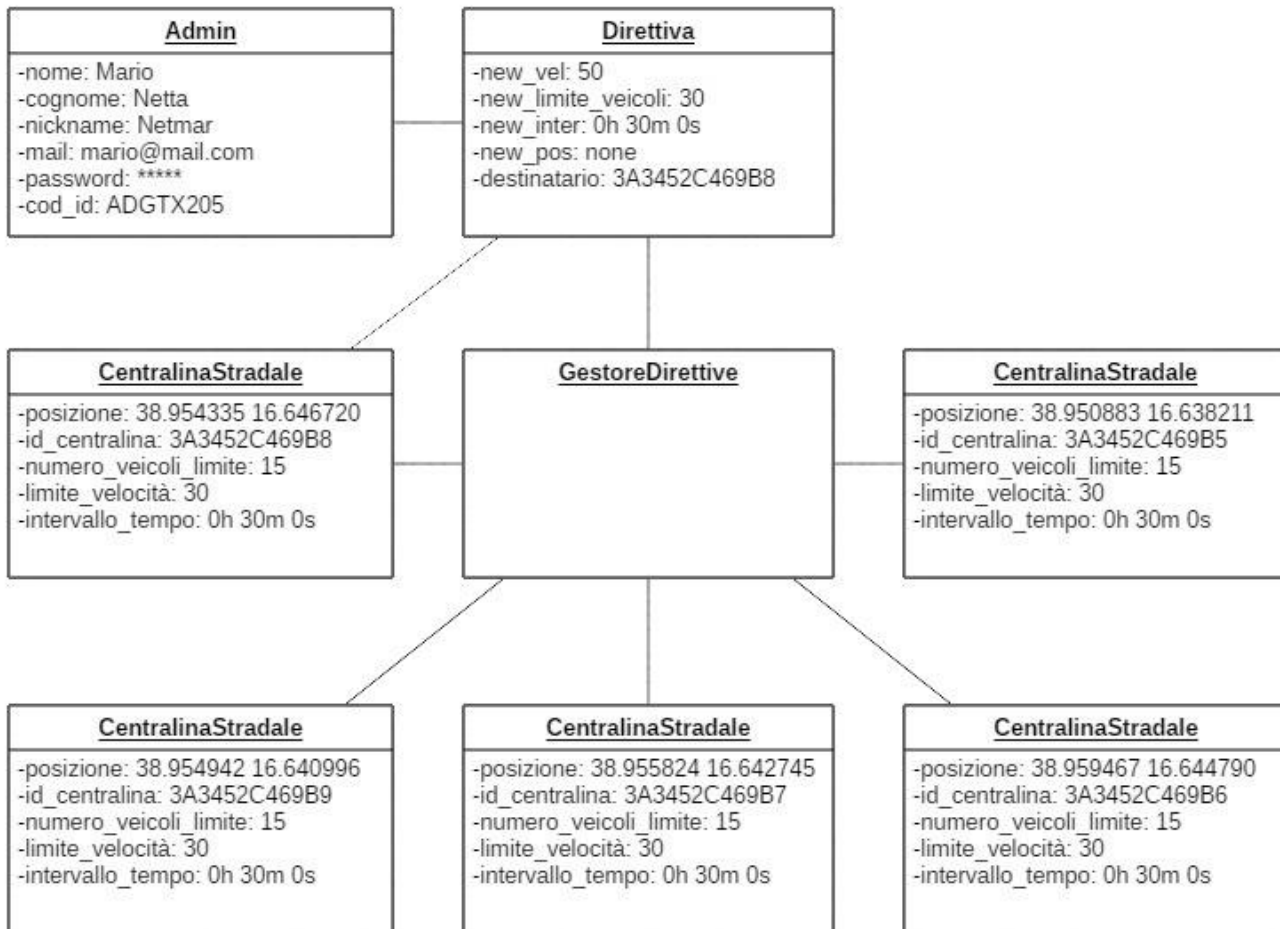
Deployment Diagram

Nel seguente diagramma mostriamo la distribuzione del programma nelle macchine fisiche. Tutti fanno riferimento al sistema centrale e effettuiamo connessioni TCP/IP.



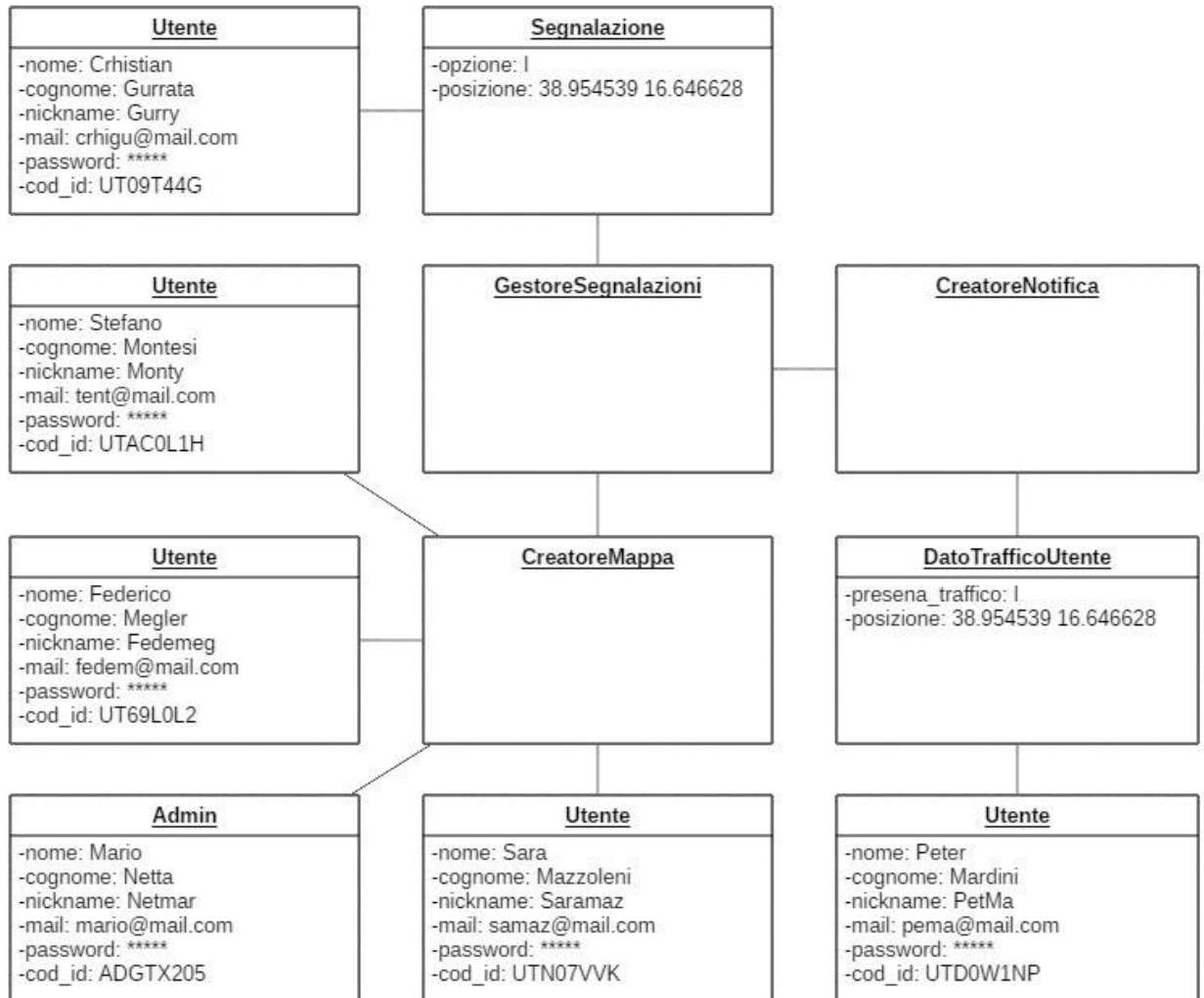
Object Diagram – Aggiornamento Centralina

Per quanto riguarda l'aggiornamento della centralina, sono stati identificati i seguenti oggetti: **Direttiva**, **Admin**, **Centralina**, **GestoreDirettiva**, **CentralinaStradale**.



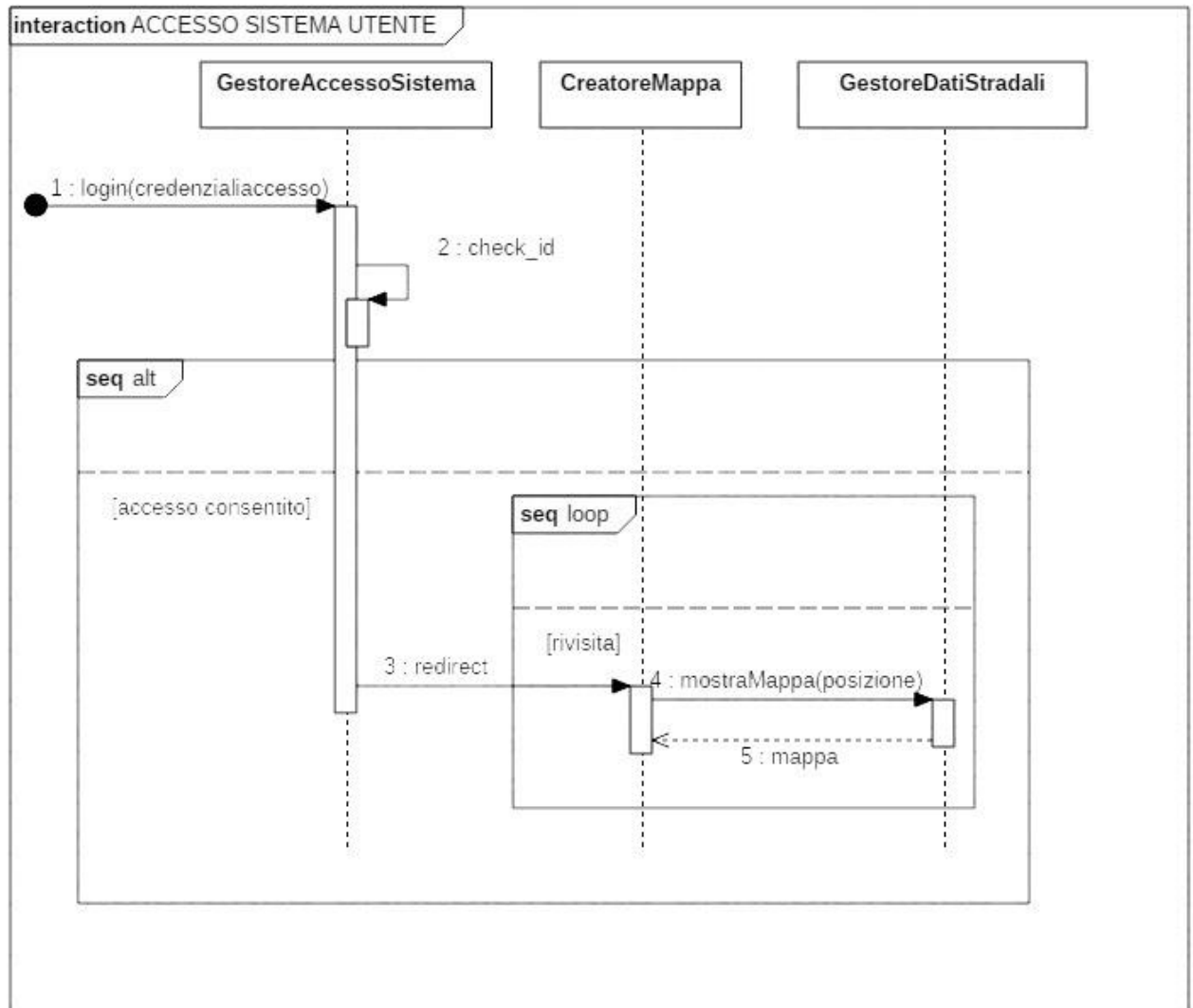
Object Diagram – Applicazione

Per quanto riguarda l'applicazione, sono stati identificati i seguenti oggetti: **Utente**, **Segnalazione**, **GestoreSegnalazioni**, **CreatoreNotifica**, **Utente**, **CreatoreMappa**, **DatoTrafficoUtente**, **Admin**.



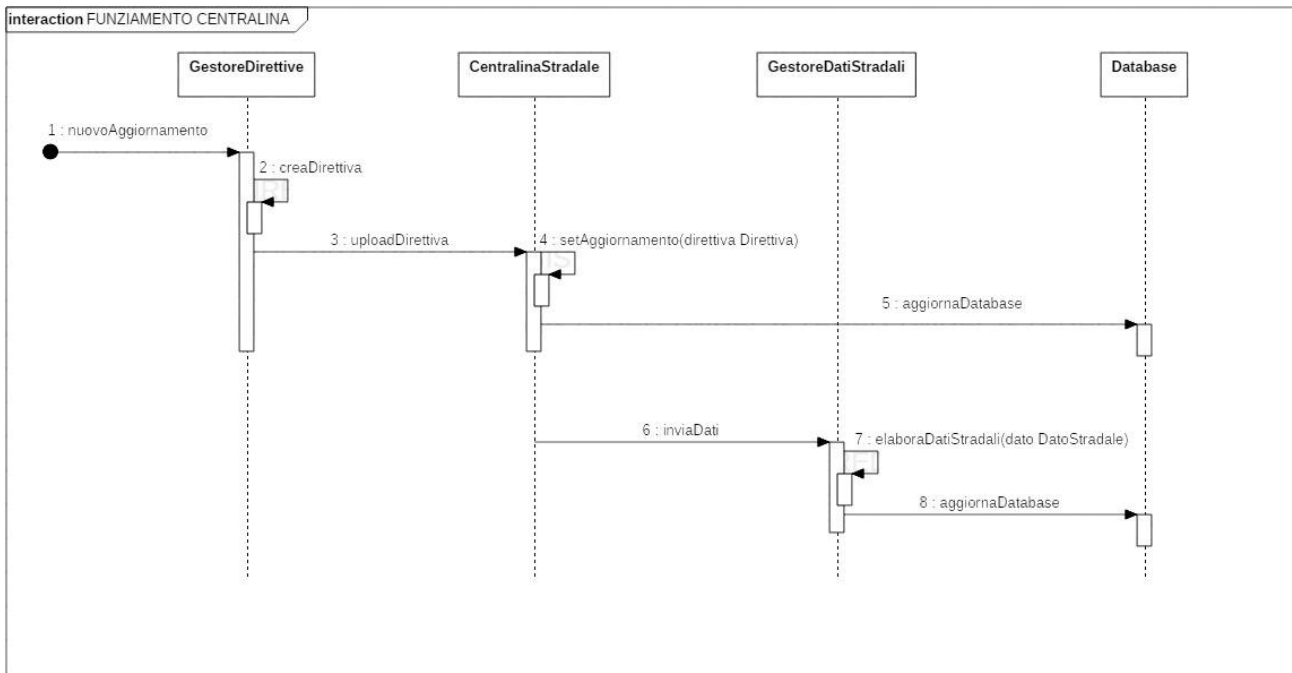
Sequence Diagram – Accesso Sistema Utente

In questo diagramma descriviamo il comportamento del sistema quando un utente si collega ad esso. In particolare, grazie al login, l'utente può effettuare più ricerche in un'unica sessione.



Sequence Diagram – Funzionamento Centralina

In questo caso vengono spiegati i passaggi dal rilevamento dei dati da parte della centralina fino al salvataggio delle informazioni raccolti.

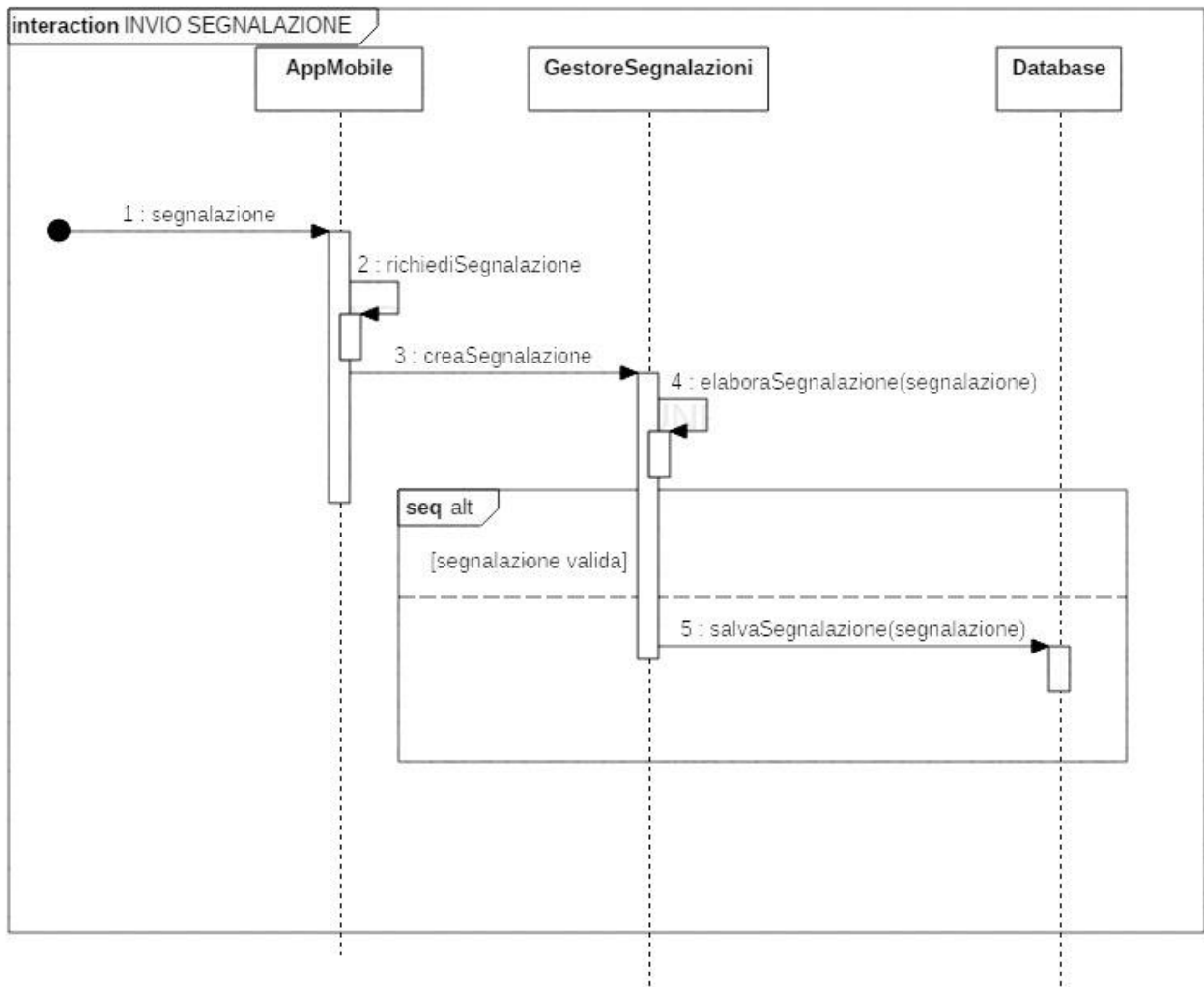


Sequence Diagram – Invio segnalazione

Gli oggetti coinvolti in questo processo sono:

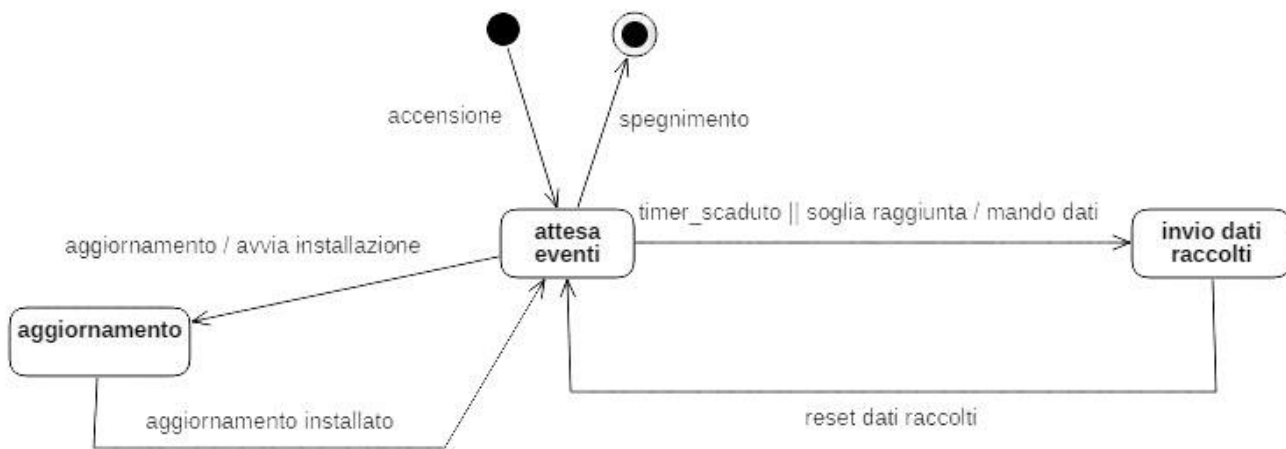
- AppMobile
- Gestore Segnalazioni
- Database

L'utente effettua una segnalazione che viene elaborata dal **Gestore segnalazioni** che si preoccupa infine di salvare tutti i dati nel **Database**.



State Diagram – Centralina

La centralina si può trovare inizialmente in stato di attesa, può poi o ricevere e installare un aggiornamento o inviare i dati del traffico al raggiungimento della soglia di tempo e di macchine critica.



State Diagram – Segnalazione

Alla creazione di una segnalazione, ci si trova inizialmente in stato di attesa, da cui si esce nel momento in cui essa viene effettuata, per poi essere inviata. Si può ora verificare un errore di connessione che comporta un errore dell'invio, o un invio dei dati con una connessione stabilita correttamente, che porta al salvataggio della segnalazione.

