# 03\_ProgettoIncrocioSemaforico

## Task

* Incrocio di due strade con semafori

## Analyses

### Modello Incrocio

traffic light crossing

 [ Ostia ]

 \*

 TLRCB \* RCA (rail crossing)

[TLRailwayA]------------------------------- X ---------------------------------------[TLRailwayB]

[ButtonA] \* TLRCA [ButtonB]

 \*

 TLRB \* TLRC(GO)

[Quartiere XXXI] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [Municipio XI]

 TLRD(GO) \* TLRA

 \*

 \*

 [Roma]

TLOrchestrator

### Direttrice di traffico



### RoadTrafficDirector

La strada che viene abilitata al traffico dei vicoli tramite i semafori

**Comandi da inviare alle direttrici di traffico:**

* setOff, Spegni tutti i semafori
* setBeware, Metti i semafori a giallo lampeggiante
* setStop, Metti i semafori a ROSSO
* setGo, Metti i semafori a Verde
* setTryToStop, Metti a giallo fisso

#### Come funziona

##### Funzionamento standard

* Procedura di Avvio
	+ All’inizio semafori spenti
	+ Poi lampeggianti
	+ Poi gialli
	+ Poi stop/rossi
	+ Poi parte la routine
* Routine
	+ Si altenano rosso e verde/Giallo dulle due direttrici di traffico
* Procedura di Spegnimento
	+ Giallo/try to stop su tutte le direzioni
	+ Rosso/Stop su tutte le direzioni
	+ Lampeggiante/Beware su tutte le direzioni
	+ Spegnimento su tutte le direzioni

##### Caso: Arrivo il treno

* Bloccare la routine allo stop della direttrice A (Roma-Ostia)
	+ Giallo/try to stop sulla direttrice A
	+ Rosso/Stop sulla direttrice A

##### Caso: Incidente Stadale, Il vigile , tramite app, vuole bloccare tutto il traffico (STOP).

Procedura da eseguire immediatamente, l’eventuale verde deve essere interrotto subito.

* + Giallo/try to stop su tutte le direzioni
	+ Rosso/Stop su tutte le direzioni

##### Caso: Passa il Presidente della Repubblica, il vigile , tramite app, mettere in GO (verde) la direttrice A

Mettere il go (verde) forzato sulla direttrice A e quindi il rosso forzato sulla B

* Bloccare la routine nello stato desiderato

La routine verrà bloccata per un tempo MASSIMO di 15 minuti (configurabile)

### Mappatura stati della direttrice di traffico

### Funzionamento Traffic Light



* Classe semaforo : TrafficLight
	+ LightBulb lbRed = new LightBulb(TLcolour.RED)
	+ LightBulb lbYellow = new LightBulb(TLcolour.YELLOW)
	+ LightBulb lbGreen = new LightBulb(TLcolour.GREEN)
	+ EnamTLStates tlState (default, nel costruttore sarà impostato a BEWARE)
* Ci sarà un server principale che comanda (comunica) con i singoli semafori.

Stati del semaforo:

**Fare Enum (**EnamTLStates**)**

* Red , STOP
* Green , GO
* Yellow, TRYTOSTOP
* Flashing Yellow, BEWARE
* All Off, OFF

Rail Crossing





### Realizzazione Modello in scala

 



  HotSpot WiFi



### Funzionamento Crossroads (Incrocio Stradale)

* Classe direttrice di traffico : TrafficDirector
	+ Denominazione (ex: Direttrice Frosinone-Fiuggi) : name (Obbligatorio)
* Classe Estesa : TrafficDirectorTwoTL
	+ TLONE
	+ TLTWO
* Classe Incrocio : Crossroads
	+ Location (Obbligatorio)
* Classe Estesa : CrossroadsFourStreets

### Orchestrator



//sequenza di stop

setOffStateBySequence()

Se siamo in GO

GO -> TryToStop -> Stop -> Beware -> OFF

//sequenza

setBewareStateBySequence()

Se siamo in GO

GO -> TryToStop -> Stop -> Beware -> OFF

//sequenza

setStopStateBySequence()

GO -> TryToStop -> Stop -> Beware -> OFF

//sequenza

setGoStateForRTDABySequence()

Se siamo in OFF

OFF -> BEWARE -> TryToStop -> Stop -> GO

0 1 2 3 4

### CRRoutine

Si potrebbe gestire cin un thread

🡪 thread

### Base Use Case

* Funzionamento standard (I semafori agiscono in una sequenza in loop , in automatico)
* Le ferrovie dello Stato mandano il segnale che sta arrivando il treno.
	+ Mettere in STOP la direttrice A finchè non arriva il segnale che ci avverte che il treno è passato.
	+ Nei casi in cui NON è previsto un passaggio del treno, l’incrocio ferroviario deve restare SEMPRE in GO.
* Il vigile , tramite app, vuole bloccare tutto il traffico (STOP). Supponiamo che c’é un incidente stradale nei pressi dell’incrocio e si deve permettere le rimozione dei veicoli coivolti nell’incidente.
* Passa il Presidente della Repubblica, il vigile , tramite app, vuole mettere in GO (verde) la direttrice principale (Roma-Ostia) per permettere il passaggio dell’intero corteo. Finito il passaggio , il vigile, sempre tramite app, vuole ripristinare il normale funzionamento.

Advanced Use Case

* Passa il Presidente della Repubblica, il vigile , tramite app, vuole mettere in GO (verde) la direttrice pricipale (Roma-Ostia) per permettere il passaggio dell’intero corteo. Finito il passaggio , il vigile, sempre tramite app, vuole ripristinare il normale funzionamento.

Rispetto alla versione base potremmo gestire anche un timer che impedisce che il sitema semaforico resti in un certo stato (esempio in GO su una specifica direttrice) a tempo indeterminato.

**Queste le due proposte in breve:**

* Aggiunta di un timer con impostazione fissa (tempo massimo previsto 60 minuti) che metteremo ad valore di default a 15 minuti.
* Aggiunta di un timer con impostazione variabile (tempo massimo previsto 60 minuti) da gestire tramite app con valore di default a 15 minuti.

### Prossimi lavori

* Creare una classe “CrossRoads” che ha all’interno due direttrici di traffico
* Aggiunge alla classe tutto i provisioning delle direttrici copiandolo dal main dell’orchestrator

. Tale provisioning va messo nel contruttore di CrossRoads

* Fare un toString() che restituisca un’immagine a caratteri dell’incrocio (“CrossRoads”)

Simile a quella riportata di seguito

 [ Ostia ]

 \*

 TLRCB(BEWARE) \* RCA (rail crossing)

 --------------------------------- X ---------------------------------------

 \* TLRCA(BEWARE)

 \*

 TLRB(OFF) \* TLRC(OFF)

[Quartiere XXXI] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* [Municipio XI]

 TLRD(OFF) \* TLRA(OFF)

 \*

 \*

 [Roma]