

Il Treno, compito B

Nome _____ Cognome _____			cattedra						
Matricola _____ Aula _____ Posizione nell'aula _____				A	B	C	D	E	...
			1	A1	B1	C1	D1	E1	...
			2	A2	B2	C2	D2	E2	...
		

(come da schema, senza contare file e colonne vuote)

La prova si svolge usando il solo libro UML@Classroom

Domanda 1b (Rispondere su questo foglio) la chat di Bing, alla domanda “Mi trovi: attori, principali e secondari, e casi d'uso per il seguente progetto? <testo progetto>” ha riposto come di seguito.

- Correggere eventuali allucinazioni o errori di Bing barrando le parti da cancellare.
- Aggiungere gli attori che eventualmente mancano, aggiungere eventuali casi d'uso mancanti.
- Per i casi d'uso: dire quali dovrebbero essere in relazione includes/extend con un altro caso d'uso (es. dire UC3 includes UC9) e quali dovrebbero essere un passo della sequenza degli eventi di un altro caso d'uso (in questo caso dire ad es. UC3 in UC9).

Gli attori principali sono: Utente, Admin, Capostazione,

Gli attori secondari sono: Sistema, Stazione.....

I casi d'uso sono:

- UC 1) **Conoscere orario e binario:** permette all'utente di conoscere orario e binario di un treno in una stazione
- UC 2) **Inserire orario stagionale dei treni:** permette all'admin di inserire nel sistema l'orario stagionale dei treni.
- UC 3) **Controllare capienza delle stazioni:** permette all'admin di controllare che tutte le stazioni abbiano la capienza necessaria ad ospitare tutti i treni in sosta, in base all'orario stagionale dei treni.
- UC 4) **Aggiungere informazioni temporanee:** permette all'admin di aggiungere informazioni temporanee sui treni.
- UC 5) **Segnalare arrivo e partenza dei treni:** permette al capostazione di segnalare al sistema l'arrivo e la partenza dei treni da una stazione, tramite opportuni sensori sui binari.
- UC 6) **Trasmettere informazioni alle stazioni successive:** permette al sistema di trasmettere le informazioni relative al ritardo e al binario di un treno alle stazioni successive lungo il suo percorso.
- UC 7) **Valutare cambio di binario:** permette al sistema di valutare se cambiare il binario di arrivo di un treno.
- UC 8) **Visualizzare dati sullo schermo:** permette alla stazione di visualizzare sullo schermo i dati relativi al ritardo e al binario di un treno in arrivo, dopo aver ricevuto le informazioni dal sistema.
- UC 9) **Informare applicazioni client:** permette al sistema di informare le applicazioni client sottoscritte.
- UC 10) **Memorizzare indennità:** permette al sistema di memorizzare nel sistema un'indennità.

UC 11) **Sottoscrizione** Un Utente si può sottoscrivere al servizio notifica ritardi

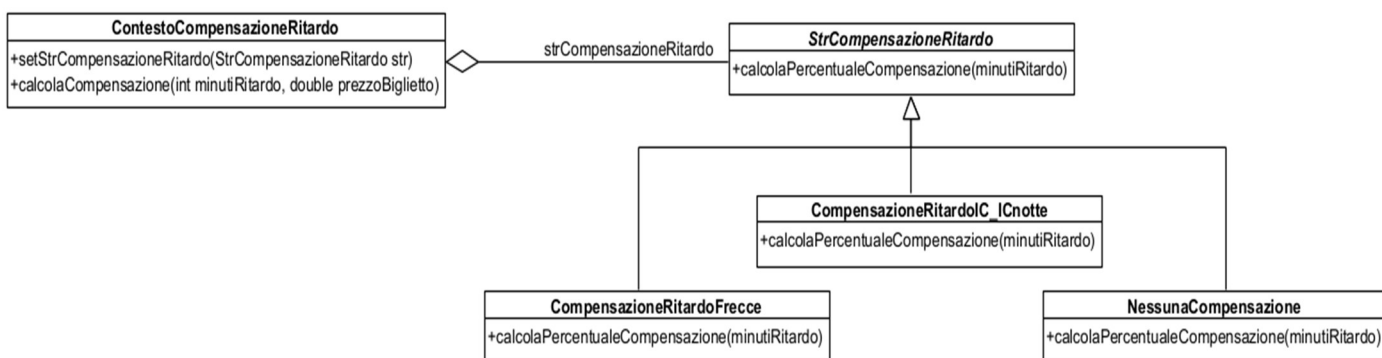
UC 12) **Richiesta indennità ritardi** Un utente può chiedere un'indennità per un ritardo (da discutere)

Eventuali “in”, includes o extends UC2 includes UC3; UC6,7,8,9 in UC5

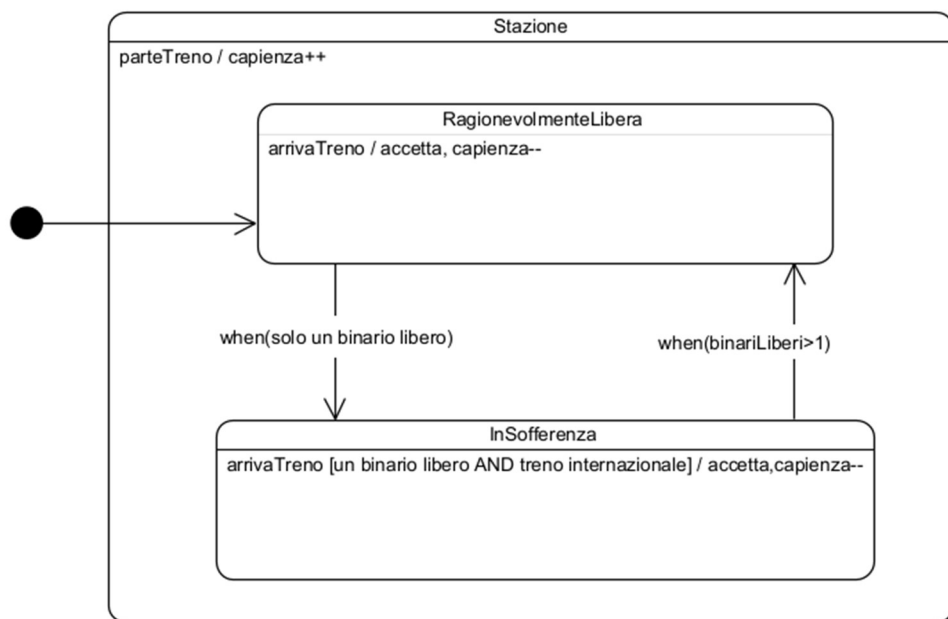
Domanda 2.b Seguendo il design pattern opportuno, dare il diagramma delle classi che comprende `ContestoCompensazioneRitardo`, la classe astratta `StrCompensazioneRitardo` e le classi che la specializzano.

```
public class ContestoCompensazioneRitardo {
    private StrCompensazioneRitardo strCompensazioneRitardo;
    public ContestoCompensazioneRitardo(StrCompensazioneRitardo str) { this.strCompensazioneRitardo = str; }
    public void setStrCompensazioneRitardo(StrCompensazioneRitardo str) { this.strCompensazioneRitardo = str; }
    public double calcolaCompensazione(int minutiRitardo, double prezzoBiglietto) {
        double percentualeBonus = strCompensazioneRitardo.calcolaPercentualeCompensazione(minutiRitardo);
        return percentualeBonus * prezzoBiglietto;
    }
}
```

Strategy

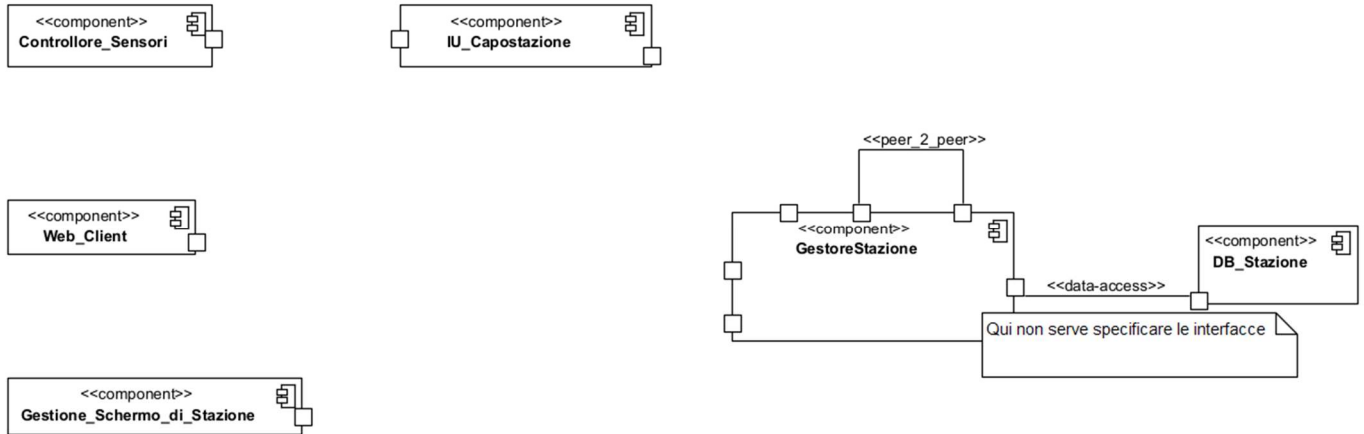


Domanda 4.b Dare un diagramma di macchina a stati che rappresenti gli stati in cui si trova una stazione ferroviaria relativamente all'occupazione dei binari per gestire eventuali conflitti dovuti a treni in ritardo (quindi non durante la definizione dell'orario stagionale). Nella valutazione dell'accettazione di un treno in arrivo, è richiesto che almeno uno dei binari della stazione sia sempre mantenuto libero: questo vincolo però non vale se il treno in arrivo è internazionale

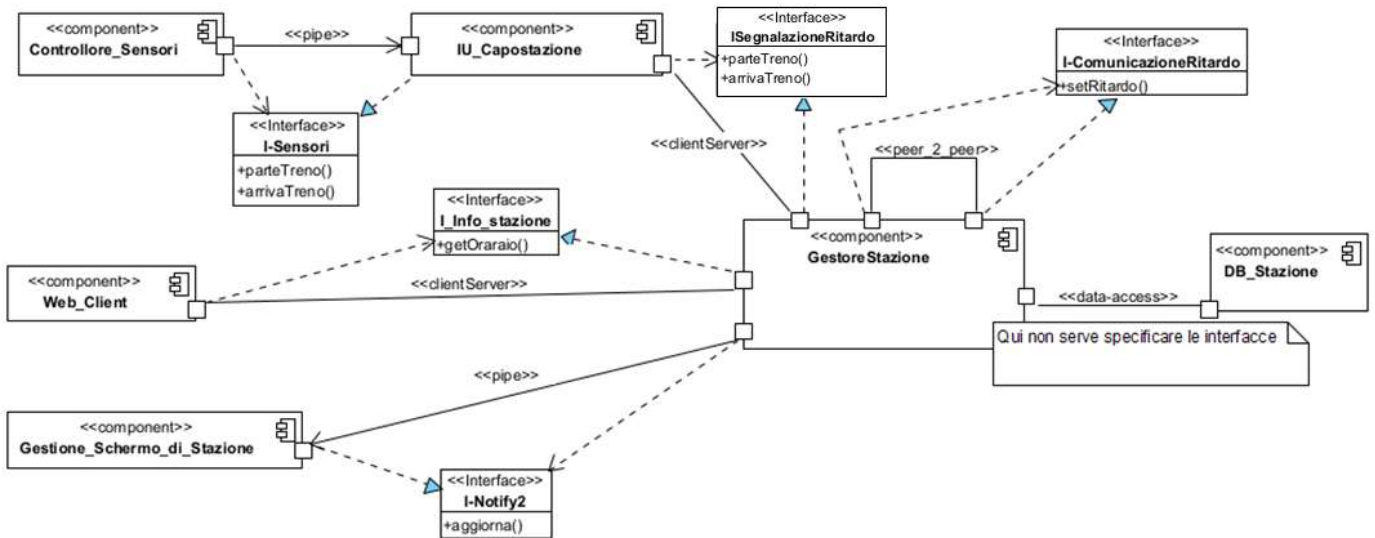


Domanda 3.b Completare la vista C&C data (sul sottosistema che realizza i seguenti requisiti--attenzione alle modifiche rispetto al progetto). Indicare tutte le interfacce delle componenti (tranne con DB).

L'arrivo e la partenza di un treno da una stazione sono segnalati opportuni sensori sui binari e confermati dal capostazione. Se il sistema identifica un ritardo nella partenza di un treno da una stazione (il controllo è fatto localmente in stazione), trasmette l'informazione alle stazioni successive. Una volta che la stazione successiva riceve questi dati, valuta se cambiare binario, seleziona automaticamente i dati associati a quella fermata di quel treno e li visualizza sullo schermo (mega-schermo di stazione) per gli utenti. Inoltre memorizza ritardo ed eventuale nuovo binario per interrogazioni via web.



(Nel diagramma mancano le molteplicità, ma sono ovvie)



Domanda 5.b Le indennità per ritardi sono calcolate come segue: 50% del biglietto per ritardi tra 30 e 59 minuti per le Frece e del 70% per ritardo maggiore di 59 minuti per Frece, Intercity e Intercity Notte. Basandosi su criteri a scatola chiusa, dare una proof obligation per il metodo: public double
 calcolaCompensazione(int minutiRitardo, double prezzoBiglietto) dell'esercizio 2.

	Input	Output	Ambiente	Criterio usato
1	Ritardo superiore a un'ora	0	no Frece, no IC, no IC_Notte	Partizione in classi
2	Ritardo superiore a un'ora	70% del prezzo biglietto	Frece AND IC AND IC_Notte	Partizione in classi
3	Ritardo di un'ora	70% del prezzo biglietto	Frece AND IC AND IC_Notte	Frontiera
4	Ritardo superiore a mezz'ora e inferiore a un'ora	50% del prezzo biglietto	Frece	Partizione in classi
5	Ritardo superiore a mezz'ora e inferiore a un'ora	0	IC AND IC_Notte	Partizione in classi
6	Ritardo inferiore a mezz'ora	0	Tutti I treni	Partizione in classi