



Traccia:

Si vuole realizzare il sistema software EduRank, per supportare le attività di un corso universitario.

Il sistema consente al docente di aggiungere un nuovo task che dovrà essere svolto da tutti gli studenti. Ciascun task è caratterizzato da un codice identificativo di 6 cifre, un titolo, una descrizione ed una deadline nel formato gg-mm-aaaa. Ciascuno studente ha un'area personale in cui sono memorizzati nome, cognome, matricola e la lista dei task finora assegnati.

Dopo che gli studenti hanno caricato lo svolgimento di un task, il docente e i suoi assistenti possono valutarlo. Una valutazione consiste in un testo di massimo 1000 parole. Invece, il voto finale in trentesimi può essere assegnato solo dal docente. Lo stesso svolgimento del task dello stesso studente può ricevere più valutazioni ma uno ed un solo voto. Il voto finale può essere assegnato solo se il docente ha prodotto la propria valutazione. Pertanto, se il docente prova ad assegnare un voto in assenza di valutazioni, il sistema produce un messaggio di errore.

Alcuni task più complessi devono essere svolti in gruppo. Ciascun gruppo è caratterizzato da un identificativo numerico e deve essere formato da un minimo di 2 studenti e da un massimo di 4. Soltanto lo studente leader del gruppo può consegnare lo svolgimento di un task svolto dal gruppo.

In base ai voti finali, il sistema mostra la classifica degli studenti e la classifica dei gruppi. Le classifiche sono aggiornate dal sistema una volta al giorno a mezzanotte, se non è stata aggiornata dal docente nelle ultime 24 ore. Durante tale operazione, il sistema assegna automaticamente voto zero ad un task se lo studente non ha consegnato il suo svolgimento entro la deadline.

Quesiti:

1. Diagramma dei casi d'uso del sistema. [3 pt]
2. System domain model, tenendo conto delle responsabilità delle classi. [10 pt]
3. Activity diagram per modellare la dinamica della funzionalità di aggiornamento della classifica degli studenti [6 pt]
4. Progettare un insieme di casi di test black box (usando la tecnica delle Classi di Equivalenza) per provare la funzionalità di Assegnazione di un Voto ad un task svolto da uno studente. Si considerino anche i valori vicini ai confini (boundaries) delle classi di equivalenza [4 pt]
5. Descrivere le quattro principali tecniche di validazione dei requisiti [2 pt]
6. Descrivere il pattern architetturale Model-View-Controller e rappresentare mediante un sequence diagram come avviene l'aggiornamento dei dati in risposta ad un evento esterno dell'utente [7 pt]