EsameService.java

**🧾 Metodo aggiungiEsame (String materiaEsame, String matricola, int votoEsame, LocalDate dataEsame)**

Questo metodo si occupa di aggiungere un nuovo esame al singolo studente, l’esame può avere esito positivo o negativo, verrà aggiunto lo stesso. Altre peculiarità del metodo sono l’aggiornamento delle medie grazie al metodo di supporto implementato ad hoc e gestione di possibili errori durante l’inserimento da parte dell’utente.

**🔷 1. Ricerca dello studente tramite matricola**

Lo studente viene cercato nel database tramite la matricola. Se non viene trovato, viene lanciata una

ResourceNotFoundException con un messaggio specifico.

**🔷 2. Ricerca della materia tramite nome**

Similmente allo studente, anche la materia viene cercata per nome. Se non viene trovata, si lancia una ResourceNotFoundException.

**🔷 3. Controllo coerenza dipartimento**

Si verifica che studente e materia appartengano allo stesso dipartimento con una query custom. Se non è così, viene sollevata un'IllegalStateException.

**🔷 4. Controllo esistenza dell'esame**

Si verifica se lo studente ha già sostenuto l'esame nella materia selezionata. Se esiste, si controlla lo stato dell'esame per evitare duplicati o sovrascritture indebite.

**🔷 5. Gestione esame già esistente**

- Se l'esame è già stato superato (voto >= 18), si blocca l'inserimento.

- Se la nuova data coincide con una già registrata, si blocca per evitare duplicazioni.

- Se l'esame è stato fallito, si aggiorna con il nuovo voto e data e si incrementano le bocciature se necessario.

**🔷 6. Creazione nuovo esame**

Se non esiste nessun esame per quella materia, ne viene creato uno nuovo. Se il voto è insufficiente (<18), si registra una prima bocciatura.

**🔷 7. Salvataggio e aggiornamento medie**

L'esame viene salvato nel database e subito dopo viene aggiornata la media ponderata e i CFU totali dello studente.

**🧾 Metodo aggiornaMedie (String matricola)**

Questo metodo si occupa di **ricalcolare e aggiornare tutte le medie accademiche** di uno studente dopo che ha sostenuto o modificato un esame. È essenziale per mantenere **i dati aggregati** coerenti nel sistema. Questo metodo è di supporto solamente per il metodo aggiungiEsame, è stato creato di proposito.

🔷 **1. Recupero dello studente**

🔷 **2. Recupero degli esami superati**

**🔷 3. Controllo lista vuota per evitare divisioni per 0 + log info**

**🔷 4. Inizializzazione dei contatori**

**🔷 5. Ciclo sugli esami e calcolo delle somme (calcoli standard)**

* Calcolo la somma pesata: voto × CFU (per la media ponderata)
* Sommo i CFU totali
* Sommo i voti (per la media aritmetica)

**🔷 6. Calcolo e assegnazione delle medie con casting**

* Media aritmetica: somma voti / numero esami
* Media ponderata: somma pesata / CFU totali
* Media di conseguimento: conversione in centodecimi

🔷 7**. Aggiornamento dei CFU e salvataggio**

**🧾 Metodo rimuoviTuttiGliEsami(String matricola)**

Questo metodo effettua una eliminazione completa di tutti gli esami dello studente, inoltre azzera le sue statistiche.

🔷 1. Recupero lo studente, se lo studente non esiste lancio un’eccezione

🔷 2. Recupero la lista degli esami dello studente

🔷 3. Se la lista è vuota lancio un’eccezione

🔷 4. Se lo studente esiste e la lista è > 1 allora procedo con l’eliminazione di tutti gli esami