

Terzo Appello – Prova di Livello Base-Intermedio

Cognome e Nome: _____ Matricola: _____

Tempo a disposizione: 3 ore

È necessario scrivere un'applicazione che simula il frammento di un sistema informativo per la gestione di annotazioni inserite su pagine web. Ogni pagina web è descritta da (a) un indirizzo univoco (b) un titolo (c) la data e ora di ultimo aggiornamento (d) il numero di parole contenute nella pagina. Una possibile pagina è {“https://wikipedia.org/Elisabetta”, “Regina Elisabetta”, 16/09/2022 12:00, 450 parole}. Su una pagina è possibile inserire varie annotazioni. Ogni annotazione è rappresentata da (a) l'indice della parola di inizio (b) l'indice della parola di fine (c) il colore utilizzato (i valori possibili sono solo Giallo, Rosso e Verde), (d) la data e ora dell'annotazione. Una possibile annotazione è {inizio: parola numero 3, fine: parola numero 6, colore: verde, inserita il 12/09/2022 11:30}.

L'applicazione deve consentire di svolgere i seguenti casi d'uso:

“Utente carica un archivio”

- l'utente carica l'archivio *NOTA: per velocizzare le operazioni, si suggerisce di utilizzare un "mock object" per simulare il funzionamento del DAO che carica i dati dal disco*

“Utente cerca pagine”

- l'utente inserisce un indirizzo, anche parziale (es. “wikipedia.org”)
- il sistema mostra in una tabella la lista delle pagine il cui indirizzo contiene il valore scelto dall'utente, ordinate per data di aggiornamento decrescente.

“Utente inserisce nuova annotazione”

- l'utente seleziona una pagina dai risultati
- il sistema mostra, in un nuovo pannello, i dati della pagina e le annotazioni attualmente registrate. Per ogni annotazione bisogna mostrare colore, parola inizio, fine e data
- l'utente può decidere di inserire i dati di una nuova annotazione specificando tutti i dati.
- Il sistema aggiunge l'annotazione alla pagina ed aggiorna i dati

Scenario alternativo

- dati scorretti: l'indice delle parole inserite dev'essere compatibile con il numero di parole presenti nel sistema. Inoltre, la parola iniziale non può essere successiva alla finale. In caso di errore il sistema mostra un messaggio di errore all'utente
- data ultima aggiornamento: se la pagina è stata aggiornata più di 45 giorni prima della data di inserimento dell'annotazione, il sistema mostra un messaggio di errore

“Utente verifica archivio”

- selezionando una voce di menu, il sistema cerca tutte le pagine che hanno annotazioni non aggiornate (ovvero annotazioni inserite prima della data di ultimo aggiornamento della pagina). Tra queste pagine viene calcolata quella con il maggior numero di annotazioni “Rosse”
- il sistema mostra un messaggio con il risultato calcolato al passo precedente

“Utente analizza archivio” – Per la prova di livello intermedio

- il sistema calcola, per ogni colore di annotazione, quante annotazioni sono presenti di quel colore e qual è la lunghezza media delle annotazioni
- i dati saranno mostrati in una tabella ordinata per tipologia crescente. Un esempio di tabella è la seguente

Nota: è necessario sviluppare i relativi test di regressione

Colore	# Annotazioni	Lunghezza media
Verde	3	4,5 parole

Sviluppare l'applicazione che implementa i casi d'uso elencati, seguendo il processo di sviluppo descritto a lezione, e in particolare le seguenti operazioni:

- costruire e documentare il modello concettuale dell'applicazione
- sviluppare il frammento di applicazione che implementa i casi d'uso descritti
- sviluppare l'interfaccia grafica in **Java Swing** utilizzando l'**architettura MVC** presentata a lezione, rispettando i requisiti per l'accessibilità e l'usabilità.
- sviluppare i test di regressione utilizzando la libreria JUnit
- per effettuare operazioni di logging utilizzare le librerie SLF4J e LogBack
- come sistema di costruzione del codice usare **NetBeans standard**

E' possibile trovare il materiale necessario nelle seguenti cartelle

- librerie jar: **c:\lib**
- materiale didattico: **c:\documentazione\POO1 e POO2**
- file di esempio di logback.xml: **c:\documentazione\POO1 - materiale software aggiuntivo\java**