

Primo 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 dei movimenti effettuati sui conti correnti di una banca. Ogni conto corrente ha un IBAN univoco (per semplicità di 6 caratteri), nome e cognome dell'intestatario e la data di sottoscrizione. Ad esempio, un possibile conto corrente è {"IT8001", "Mario Draghi", 1 febbraio 2021}.

Ad ogni conto corrente sono associati dei movimenti. Ogni movimento ha una data completa di ora e minuti, un importo, e la tipologia (che può essere bonifico, pagamento pos e prelievo bancomat). Ad esempio alcuni possibili movimenti sono:

- 2 febbraio 2021 9:00, +1000€, bonifico
- 2 febbraio 2021 10:30, -400€, pagamento pos
- 3 febbraio 2021 20:30, -500€, bonifico

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 conti correnti”

- l'utente sceglie un criterio di ordinamento, che può essere “data crescente” o “nome crescente”. *NOTA: per la scelta del criterio, utilizzare JRadioButton*
- il sistema mostra in una tabella la lista di conto correnti che sono stati sottoscritti prima della data odierna, ordinati secondo il criterio scelto dall'utente. Nella tabella bisogna mostrare iban, intestatario e data sottoscrizione

“Utente inserisce nuovo movimento”

- l'utente seleziona un conto corrente dai risultati
- il sistema mostra, in un nuovo pannello, i dati del conto e i movimenti attualmente registrati. Per ogni movimento bisogna mostrare data, importo e tipologia
- l'utente può decidere di inserire i dati di un nuovo movimento, specificando tutti i dati.

Scenario alternativo

- dati scorretti: il sistema mostra un messaggio di errore all'utente
- data non valida: il movimento non può essere registrato quando la banca è chiusa. **La banca è aperta solo dal lunedì al venerdì, dalle 8:30 alle 20:30.** Se l'utente specifica un orario non valido, il sistema mostra un messaggio di errore all'utente

“Utente verifica archivio”

- selezionando una voce di menu, l'utente verifica se ci sono più conto correnti con lo stesso nome e cognome sottoscritti nella stessa data
- il sistema mostra un messaggio con il risultato calcolato al passo precedente

“Utente ricerca mesi preferiti” – Per la prova di livello intermedio

- il sistema calcola, per ogni tipologia di spesa, qual è il mese dell'anno in cui ci sono più movimenti di quella tipologia
- i dati saranno mostrati in una tabella ordinata per tipologia crescente. Il mese dev'essere riportato in italiano. Un esempio di tabella è la seguente

Nota: è necessario sviluppare i relativi test di regressione

Tipologia	Mese	Numero di operazioni
Bonifico	Luglio	10 operazioni di tipo bonifico effettuati a luglio
Pagamento POS	Febbraio	4 operazioni di tipo pos effettuati a febbraio

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**