

ESAME DI FONDAMENTI DI INFORMATICA T-2 del 1°/07/2015

Proff. E. Denti – G. Zannoni

Tempo a disposizione: 4 ore MAX

NB: il candidato troverà nell'archivio ZIP scaricato da Esamix anche il software "Start Kit"

NOME PROGETTO ECLIPSE: CognomeNome-matricola (es. RossiMario-0000123456)

L'istituto bancario *ZetaBank* ha richiesto lo sviluppo di un'applicazione per simulare la concessione di finanziamenti.

DESCRIZIONE DEL DOMINIO DEL PROBLEMA.

Un finanziamento prevede l'erogazione di un importo, da restituire poi in rate periodiche, comprensivo di interessi, secondo un certo schema. In questa sede si considera solo il caso di tasso fisso, in cui il tasso d'interesse non varia per tutta la durata dell'operazione. Si considerano due schemi di restituzione: **a rate decrescenti** e **a rate costanti**.

Nello schema **a rate decrescenti**, il cliente rimborsa ogni volta un importo diverso, calante nel tempo, costituito da:

- una parte, costante, del capitale finanziato (quota fissa capitale)
- una quota, variabile, relativa agli interessi, calcolata sul capitale ancora da rimborsare e che perciò cala nel tempo.

Nello schema **a rate costanti**, invece, l'importo da pagare è fisso perché costituito da due parti che si compensano:

- una quota, che aumenta nel tempo, del capitale finanziato
- una quota, che invece diminuisce nel tempo, relativa agli interessi sul capitale ancora da rimborsare.

Il valore della rata si ottiene tramite una formula di matematica finanziaria, sotto riportata. Questo schema è più pratico, ma comporta un esborso globale per interessi un po' superiore, perché il capitale è rimborsato più lentamente.

DEFINIZIONI

In entrambi i casi occorre preliminarmente fissare tre elementi:

- la durata D del finanziamento (in anni)
- la periodicità dei versamenti (ossia, il numero P di rate da pagare nell'anno)
- il tasso d'interesse annuo nominale i_N

Dalla durata D e dalla periodicità P è immediato calcolare il *numero totale di rate* N , che risulta $N = D \times P$.

ESEMPI: se $D = 10$ anni e la periodicità è mensile ($P=12$), le rate saranno 120 ($N = 10 \times 12$);

se $D = 5$ anni e la periodicità è trimestrale ($P=12/3=4$), le rate da pagare saranno 20 ($N = 5 \times 4$).

Il tasso di interesse *nominale* i_N è riferito convenzionalmente a un anno: poiché però la periodicità dei pagamenti è solitamente diversa (mensile, trimestrale, etc.), nei calcoli occorre usare un valore proporzionato al periodo scelto (mese, trimestre, etc.) chiamato *tasso effettivo*, i , ottenuto dividendo il tasso annuale per la periodicità: $i = i_N / P$.

ESEMPI: se $i_N = 5\%$, il *tasso effettivo mensile* è $5/12 = 0,417\%$, il *tasso effettivo trimestrale* è $5/4 = 1,25\%$, ecc.

CALCOLO DEL FINANZIAMENTO

Detto C il capitale complessivo da finanziare, nello schema **a rate decrescenti**:

- la quota fissa capitale vale, per tutte le rate, $Q_C = C / N$
- la quota variabile relativa agli interessi vale invece $Q_k = i * (C - (k-1) Q_C)$, con $k = 1..N$ ($k =$ numero della rata).

All'inizio, $Q_1 = i * C$, perché non si è ancora rimborsato nulla e quindi gli interessi si pagano sull'intero capitale ricevuto; nelle rate successive la quota interessi inizia a calare, perché il capitale residuo da remunerare via via si assottiglia; dopo N rate, il cliente avrà restituito tutto il capitale ($N * Q_C = C$) e pagato interessi pari a $Q_{I-TOT} = i * C * (N+1)/2$.

ESEMPI: se $D = 10$ anni, $i_N = 4\%$, un capitale $C=60.000$ € da rimborsare con rate mensili comporta $N=120$ rate con tasso effettivo mensile $i = 0,333\% = 0.00333$, da cui quota fissa $Q_C = € 500$ e quota interessi decrescente da $Q_1 = i * C = € 200$ fino a $Q_{120} = € 1,66$. Alla fine, il cliente avrà pagato complessivamente interessi per $Q_{I-TOT} = 12.100$ euro.

Nello schema **a rate costanti**, invece, la rata fissa r si ottiene dalla formula:

$$r = \left(1 + \frac{1}{(1+i)^n - 1} \right) \cdot i \cdot C$$

Nel caso sopra, la formula produce una rata fissa di importo $r = € 607,47$ € per 120 rate: al termine si saranno quindi pagati € 72.896,50, di cui € 60.000 di rimborso capitale e € 12.896,50 di interessi (superiore ai 12.100 precedenti).

Il file di testo [Finanziamenti.txt](#) contiene la descrizione dei finanziamenti offerti da *ZetaBank*.

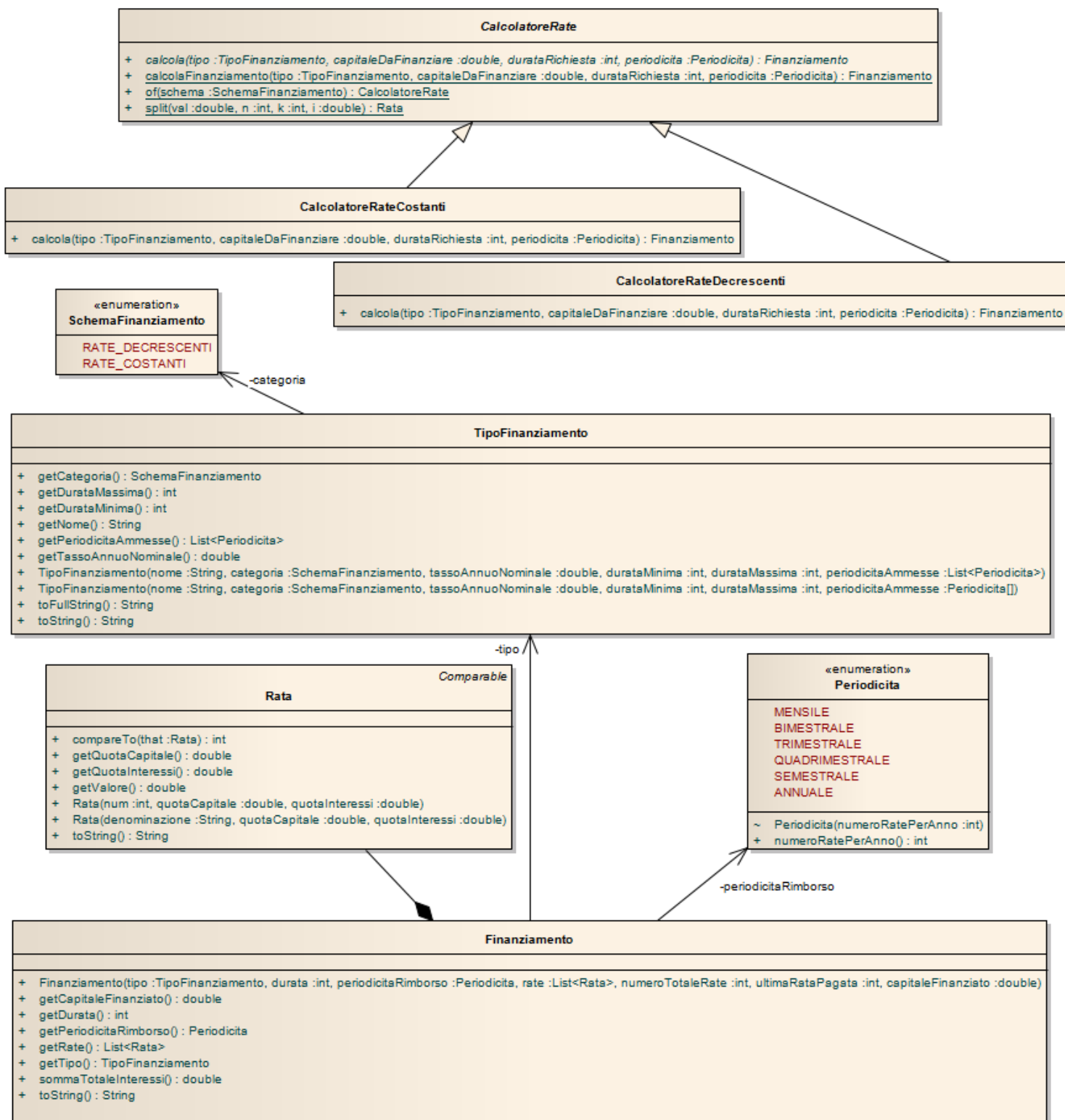
Parte 1

(punti: 19)

Dati (namespace zetabank.model)

(punti: 10)

Il modello dei dati deve essere organizzato secondo il diagramma UML più sotto riportato.



SEMANTICA:

- l'enumerativo **SchemaFinanziamento** (fornito) definisce i due schemi base di finanziamento (a rate decrescenti o a rate costanti);
- l'enumerativo **Periodicità** (fornito) definisce le tipiche periodicità di pagamento delle rate (mensile, bimestrale, trimestrale, quadrimestrale, semestrale, annuale), assegnando a ciascuna costante un valore pari al numero di rate annuali da essa implicitamente sotteso (quindi 12 per la periodicità mensile, 6 per la bimestrale, ecc.);

- c) la classe **Rata** (fornita) rappresenta una rata, composta delle due quote (rimborso capitale e interessi) accessibili tramite appositi metodi accessor. Il costruttore prevede tre argomenti: una denominazione, che può essere una stringa arbitraria o un intero, la quota capitali e la quota interessi.
- d) la classe **TipoFinanziamento** (fornita) rappresenta un tipo di finanziamento, con le sue caratteristiche: nome, categoria (rata decrescente o costante), tasso annuo nominale, durata minima e massima, periodicità ammesse;
- e) la classe **Finanziamento (da realizzare)** rappresenta uno specifico finanziamento, con le sue caratteristiche: il **TipoFinanziamento** assunto come riferimento, il capitale finanziato, la durata in anni, la periodicità di rimborso, la lista delle singole rate da pagare. Oltre a tutti gli accessor, il metodo **sommaTotaleInteressi** deve restituire la somma S di tutti gli interessi pagati nel corso dell'intero finanziamento. Il costruttore impone opportune precondizioni su tutti i parametri che vanno verificate in modo puntuale e non aggregato: il tipo dev' essere non nullo, la durata deve essere positiva, la periodicità del rimborso deve essere non nulla e contenuta nelle periodicità ammesse, la collezione delle rate deve essere non nulla e non vuota, il capitale finanziato deve essere positivo.
- f) la classe astratta **CalcolatoreRate** (fornita) incapsula il calcolo delle rate indipendentemente dallo specifico schema adottato. Il metodo statico **calcolaFinanziamento** prende in ingresso un **TipoFinanziamento** e il capitale da finanziare, indi verifica che la durata del finanziamento rientri nei limiti di durata previsti, lanciando **IllegalArgumentException** in caso contrario; altrimenti, genera e restituisce il **Finanziamento** corrispondente, invocando sul calcolatore opportuno (che provvede a creare) il metodo astratto **calcola**, con gli stessi argomenti ricevuti. Il metodo statico **of** istanzia e restituisce una specifica istanza di **CalcolatoreRate** in base allo **SchemaFinanziamento** passato come argomento. La classe fornisce un metodo factory **split** specializzato per il caso di rate costanti: esso costruisce una **Rata** suddividendo automaticamente il valore costante in quota capitale e quota interessi, note che siano il numero totale di rate N, l'indice della rata corrente $K = 1..N$, e il tasso d'interesse i .
- g) la classe **CalcolatoreRateDecrescenti (da realizzare)** concretizza **CalcolatoreRate** nel caso delle rate decrescenti; nel metodo **calcolaFinanziamento** occorre verificare le opportune precondizioni in modo puntuale.
- h) la classe **CalcolatoreRateCostanti (da realizzare)** concretizza **CalcolatoreRate** nel caso delle rate costanti; nel metodo **calcolaFinanziamento** occorre verificare le opportune precondizioni in modo puntuale.

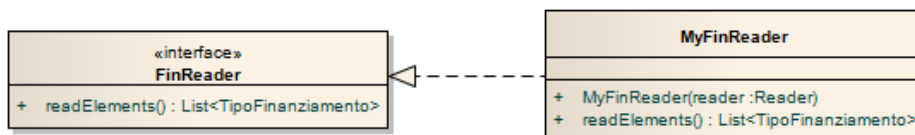
Lo Start Kit contiene i test per verificare il funzionamento di queste classi.

Persistenza (package zetabank.persistence)

(punti 9)

Come già anticipato, il file di testo **Finanziamenti.txt** contiene la descrizione dei finanziamenti offerti da *ZetaBank*, uno per riga. Ogni riga contiene una serie di campi separati da tabulazioni e precisamente: il nome commerciale del tipo di finanziamento, la sua categoria ("RATE DECRESCENTI" o "RATE COSTANTI"), tasso annuo nominale (un numero reale seguito da "%"), la durata minima e massima (due interi), le periodicità ammesse (una o più parole fra quelle definite dall'enumerativo **Periodicità**, separate fra loro da virgole e/o spazi).

L'architettura software è illustrata nel diagramma UML che segue:



SEMANTICA:

- a) l'interfaccia **FinReader** (fornita) dichiara il metodo **readElements** che restituisce una lista di **TipoFinanziamento**, lanciando, oltre alla "naturale" **IOException**, una **BadFileFormatException** nel caso di errore nel formato del file;

b) la classe **MyFinReader** (da realizzare) concretizza **FinReader** implementando **readElements** secondo il formato del file sopra descritto – in particolare, verificando:

- che il campo percentuale abbia effettivamente la forma di numero reale seguito da “%”;
- che i campi numerici siano effettivamente numerici e del giusto tipo (reale o intero);
- che le periodicità elencate siano tutte lecite (indipendentemente da maiuscole/minuscole).

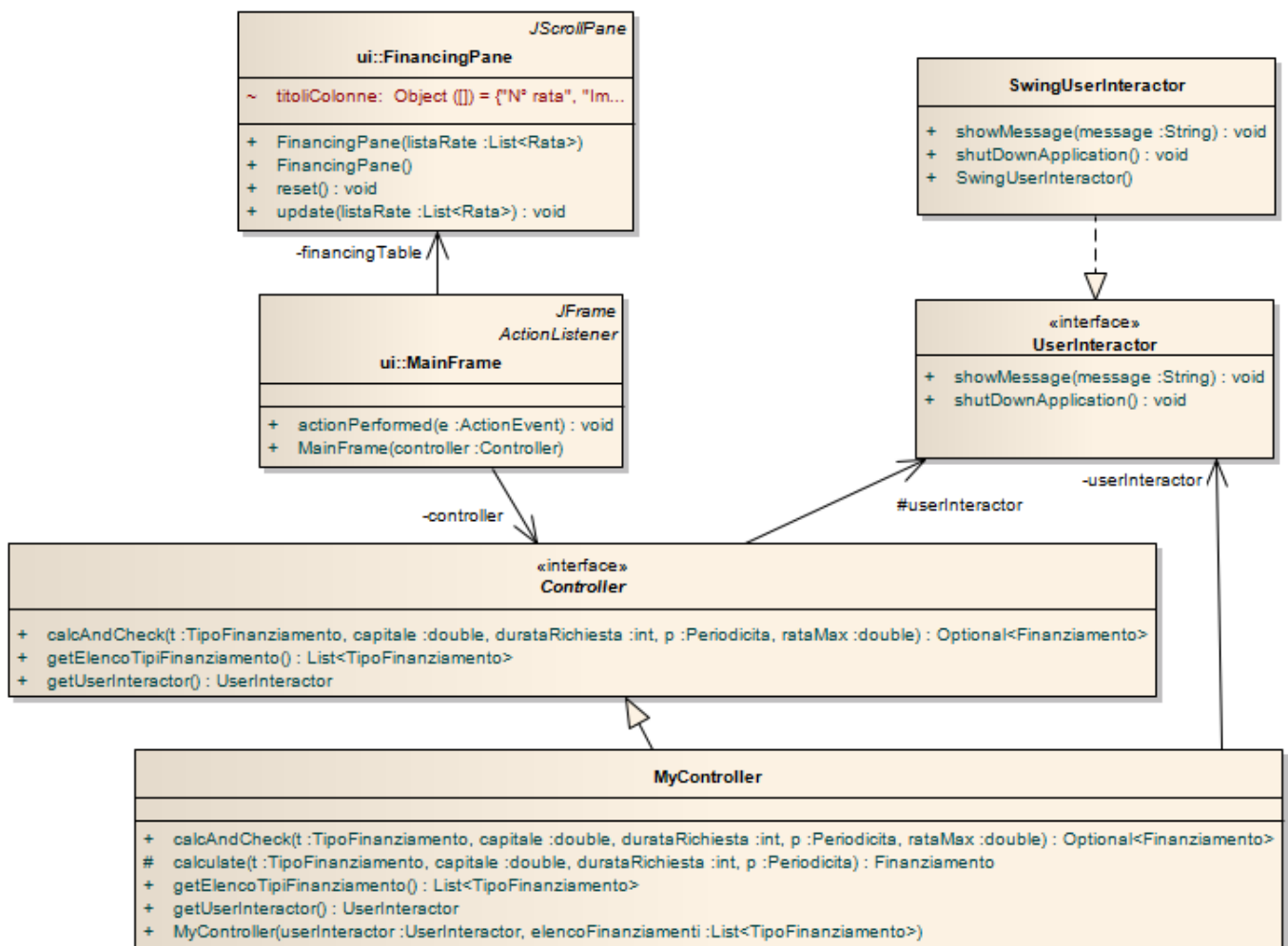
Lo Start Kit contiene i test per verificare il funzionamento di queste classi.

Parte 2

(punti: 11)

L'interfaccia grafica deve permettere all'utente di scegliere i tipi di finanziamento disponibili da una lista a discesa, inserendo poi i dati specifici di proprio interesse (capitale da finanziare, durata finanziamento, periodicità, **rata massima ammissibile**) ed ottenere nella tabella sottostante il calcolo completo delle rate; premendo il pulsante **CALCOLA**, si scatena il calcolo del finanziamento. Se la durata o la periodicità prescelta è incompatibile col tipo di finanziamento o **l'importo di una rata supera la rata massima ammissibile**, l'applicazione lo segnala tramite opportuni dialoghi (Figg. 5, 6).

La classe **Program** (fornita nello start kit) contiene il *main* di partenza dell'intera applicazione.



Controller (package `zetabank.ui.controller`) [fornito già implementato]

(punti 0)

L'interfaccia **UserInteractor** (fornita), implementata dalla classe **SwingUserInteractor** (fornita), consente di mostrare un messaggio all'utente (metodo **showMessage**) o di terminare l'applicazione (metodo **shutDownApplication**).

L'interfaccia **Controller** (fornita) dichiara i seguenti metodi, ognuno riportato con la sua semantica:

- **getUserInteractor** deve restituire uno **UserInteractor** in grado di mostrare messaggi all'utente;

- **getElencoTipiFinanziamento** deve restituire la lista dei **TipoFinanziamento** disponibili;
- **calcAndCheck** prende come argomento un **TipoFinanziamento**, il capitale da finanziare, la durata richiesta, la **Periodicità** desiderata e l'importo massimo accettabile della rata e deve restituire un **Optional** contenente il **Finanziamento** calcolato o vuoto in caso d'errore; se l'importo massimo accettabile della rata è superato, deve mostrare un opportuno messaggio tramite lo **UserInteractor**.

La classe **MyController** (fornita) implementa l'interfaccia **Controller** e presenta un costruttore con un argomento di tipo **UserInteractor** e uno di tipo **List<TipoFinanziamento>**.

Interfaccia utente (package zetabank.ui.controller)

(punti 11)

La classe **FinancingPane** (fornita) incapsula una tabella specializzata per mostrare finanziamenti: ha un costruttore di default, senza argomenti, che costruisce una tabella vuota (Fig. 1); è disponibile anche un costruttore che accetta una lista di **Rata**, usate per popolare la tabella. I dati visualizzati possono essere modificati invocando il metodo **update**, a cui va fornita una nuova lista di **Rata** aggiornata; il metodo **reset** invece cancella tutto il contenuto.

La classe **MainFrame** (da realizzare) rappresenta la GUI dell'applicazione: la combo in alto mostra i tipi di finanziamenti disponibili, le cui descrizioni riassumono le caratteristiche salienti dei singoli finanziamenti; i campi successivi permettono di inserire i dati numerici richiesti e di selezionare la periodicità desiderata.

Tutti i campi sono obbligatori, tranne la rata massima: se quest'ultima è lasciata vuota, si deve ritenere accettabile qualunque rata. La pressione del bottone **CALCOLA** fa scattare il calcolo, (ri)popolando la tabella (Figg 2-3): il precedente valore della tabella deve comunque essere cancellato, così da non lasciare dati spuri in caso di errori.

Da notare che, in caso di rata eccessiva o durata illegale, il controller notifica già autonomamente il problema all'utente (Fig. 4), quindi **MainFrame** non deve preoccuparsene – deve solo prestare attenzione al fatto che sarà restituito un **Finanziamento vuoto (Optional)**. Viceversa, competono direttamente a **MainFrame** le verifiche relative ai campi numerici e alla periodicità, che, se negative o non valide, devono dar luogo a opportuni messaggi d'errore (Figg. 5-6).

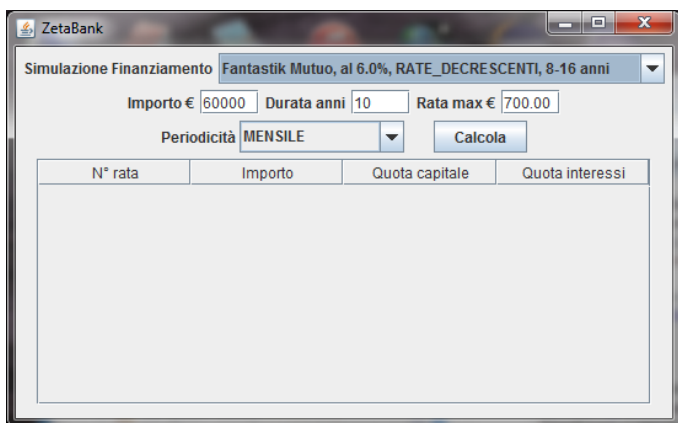


Figura 1



Figura 2

ZetaBank

Simulazione Finanziamento **PagaMoltoALungo, al 4.0%, RATE_DECRESCENTI, 10-25 anni**

Importo € Durata anni Rata max €

Periodicità **MENSILE**

N° rata	Importo	Quota capitale	Quota interessi
1	€ 450,00	€ 250,00	€ 200,00
2	€ 449,17	€ 250,00	€ 199,17
3	€ 448,33	€ 250,00	€ 198,33
4	€ 447,50	€ 250,00	€ 197,50
5	€ 446,67	€ 250,00	€ 196,67
6	€ 445,83	€ 250,00	€ 195,83
7	€ 445,00	€ 250,00	€ 195,00
8	€ 444,17	€ 250,00	€ 194,17
9	€ 443,33	€ 250,00	€ 193,33
10	€ 442,50	€ 250,00	€ 192,50
11	€ 441,67	€ 250,00	€ 191,67

Figura 3

Messaggio

Rata eccessiva: € 800,00 (max € 700,00)

Messaggio

Durata richiesta illegale: anni 10 (possibile [15-30])

Figura 4

Messaggio

Capitale, rata max e anni devono essere numeri

Figura 5

Messaggio

Periodicità non ammessa per questo tipo di finanziamento

Figura 6