

[Home](#) / [Students](#)**Project details**

Progetto Esame

Docenti: Proff. Emanuel Di Nardo - Angelo Ciaramella

A. A. 2022/2023

Studente

- **Cognome:** Palladino
- **Nome:** Simone
- **Matricola:** 0124002316
- **Codice gruppo:** qi0b2gdv4i
- **Componenti gruppo:** 3

Traccia - Sistema Bancario 1

Si vuole simulare un *Sistema Bancario* per la gestione di *conti correnti*. Un correntista è identificato dal nome, cognome, data di nascita, residenza, tipo di conto, numero e tipologie di carte di credito associate. Il costo annuo del conto corrente viene deciso in base alla tipologia di contratto stipulato (prevedere almeno tre tipologie).

Scrivere un programma per la gestione del *Sistema Bancario*. Il sistema deve prevedere l'accesso in modalità amministratore e in modalità utente.

L'amministratore può effettuare le seguenti operazioni:

- inserire un nuovo correntista (ricerca per nome e cognome o per numero di conto)
- eliminare un correntista (ricerca per nome e cognome o per numero di conto)

L'utente può effettuare le seguenti operazioni:

- visualizzare le informazioni sulle proprie operazioni
- fare acquisti tramite una carta ad esso associata (*carta di credito, bancomat*). L'ammontare viene automaticamente addebitato sul suo conto
- richiedere l'annullamento dell'ultimo addebito
- aggiungere servizi alla sua tipologia di conto (e.g., prelievi illimitati, sconti per categorie di spesa, . . .)
- versare dei soldi sul conto

Note di sviluppo

La prova d'esame richiede la progettazione e lo sviluppo della traccia proposta. Lo studente può scegliere di sviluppare il progetto nelle due modalità: **Applicazione Web** o **programma standalone con supporto grafico**.

Il progetto deve essere sviluppato secondo le seguenti linee:

- usare almeno **due** pattern per persona (almeno **uno** per chi sceglie la modalità Web Application) tra i **design pattern** noti;
- attenersi ai principi della programmazione **SOLID**;
- usare il linguaggio **Java**;
- inserire sufficienti **commenti** (anche per Javadoc) e **annotazioni**;
- gestione delle **eccezioni**;
- usare i **file** o **database**;

E' possibile costruire l'applicazione standalone con supporto grafico tramite l'utilizzo di strumenti per la realizzazione di interfacce grafiche presenti in molti IDE (**GUI Designer** in IntelliJ e **WindowsBuilder** in Eclipse) oppure utilizzare tools compatibili con JavaFx come **Scene Builder** (compatibile con gli IDE).

Consegna progetto

Lo studente deve presentare una relazione sintetica (per chi usa *latex* è possibile scaricare un template dalla piattaforma e-learning). La relazione deve contenere:

- una breve descrizione dei **requisiti** del progetto;

- il diagramma UML delle classi;
- altri diagrammi se opportuni;
- parti rilevanti del **codice** sviluppato;

Per chi usa latex si consiglia di utilizzare la piattaforma Overleaf:

- <https://www.overleaf.com/>

Modalità di esame

La prima parte della prova di esame verterà sulla discussione del progetto. Lo studente deve preparare una **presentazione sintetica** (slide) per descrivere il progetto svolto. La seconda parte della prova verterà sulla discussione degli argomenti affrontati a lezione.