OOP PROJECT: FIRST BANK OF NERDS

SPC Viner, James —— SSG Smith, Wesley —— SSG Palomares, Joana

Abstract

A working model of a small bank.

1 PROJECT SUMMARY:

Introducing the ultimate solution for the tech-savvy, game-loving community - a fully functional small bank that makes saving for the ultimate beach vacation a breeze! Our project is designed to cater to the needs of nerds who work hard and play harder. With our innovative banking system, you can manage your hard-earned programmer's salary with ease and enjoy the pleasures of life without any financial stress.

Our project features a state-of-the-art banking system that includes accounts, customers, and seamless transactions such as depositing or withdrawing. Our menu interface ensures that you can navigate through the application with ease, and without any confusion or hassle. Whether you're saving for your next Dota 2 game or just trying to enjoy a stress-free life, our small bank has got you covered!

Our project has been designed with utmost attention to detail, ensuring that every aspect of the application is optimized for the ultimate user experience. With our application, you can effortlessly manage your accounts and transactions, while having complete peace of mind knowing that your money is in safe hands.

So what are you waiting for? Take control of your finances, save for your dream beach vacation, and indulge in the pleasures of life with our cutting-edge small bank application. With our application, there is no question about how to use it. It's simple, intuitive, and built for the modern-day nerd. Try it out today and experience the ultimate banking solution for yourself!

2 FEATURES TARGETED:

- `[ MAN PAGE ]` → Program documentation that has the name, description, synopsis, options, bugs, and author of the program.
- `[ UML DIAGRAM ]` → Provide a UML diagram for the classes in the project.
- `[ MONEY MARKET FUND ACCOUNT ]` → An account type that must not allow more than two transactions on a single MMF account when the program is running.
- `[ ALTERNATE CURRENCY ]` → Allowing the customer to withdraw or deposit in an alternate currency.
- `[ OVERDRAFT PROTECTION ]` → Providing the customers an overdraft protection. A small overdraft fee of $35.00 will incur for each overdrawn transaction.

3 ARCHITECTURE:

1. DATA

```
The PARENT class
(Abstract class for use as a generic banking account.)
    class Account(ABC):
        balance : float

        withdraw()
        deposit()

SUB-CLASSES of Account
    class Savings:
    class Checking:
    class RetirementAccount:
    class MoneyMarket:

Others
    class Customer:
        ID : int
        first_name : str
        last_name : str
        age : int
        user_ID : int
        accounts : dict

    class Menu:
      selection : str
```

2. SIGNIFICANT FUNCTIONS and METHODS

- def get_all_balances():
- def list_accounts_of_type()
- def add_account()
- def deposit_into()
- def withdraw_from()

4 ORDER OF PROJECT:

- The order of working with the first bank of nerds project:

  1. Spending a good portion of the afternoon dissecting, and breaking down different aspects of the project to create the pre-project design documentation.

  2. Building the base requirement of the program starting with the packages.

  3. Create the modules that consist of the classes and concurrently build the unit test development.

  4. Create different branches on the targeted features that the team is attempting.

  5. Writing the write-up individually as the last phase of the project.

5 USER INTERFACE:

- The program will rely on the use of the command line interface for navigating through the use of our banking system.