

Uniwersytet Przyrodniczo-Humanistyczny
W Siedlcach

Kierunek: Informatyka

Yahor Falkouski

Grupa 1, 88457

Temat projektu: Banking System

Praca wykonana pod kierunkiem
Anna Kołkowicz

Siedlce, 2023

Spis treści

<i>Temat projektu:</i>	- 2 -
<i>Specyfikacja problemu, założenia i ograniczenia:</i>	- 2 -
<i>Opis klas, algorytmów, zmiennych:</i>	- 3 -
<i>Diagram klas w UML:</i>	- 5 -
<i>Kod programu:</i>	- 5 -
<i>Ważne cechy programu:</i>	- 6 -
<i>Przykładowe dane i wyniki:</i>	- 7 -
<i>Instrukcja dla użytkownika:</i>	- 8 -

Temat projektu:

Prosty system bankowy umożliwiający: zakładanie kont, wpłaty i wypłaty oraz sprawdzanie salda.

- program umożliwia: założenie konta, sprawdzenie salda, wykonanie wpłaty/wypłaty, wzięcie kredytu.
- uwzględni klienta i pracownika firmy.
- system dopuszcza możliwość posiadania kont w różnych walutach, co najmniej dwóch.
- wypłaty wyższych kwot wymagają akceptacji pracownika banku.
- wzięcie kredytu wymaga akceptacji pracownika.

Specyfikacja problemu, założenia i ograniczenia:

Aplikacja jest modelem podstawowego systemu bankowego, który umożliwia użytkownikom wykonywanie podstawowych transakcji bankowych. System obsługuje dwa rodzaje walut: USD i PLN, co ogranicza jego międzynarodowe zastosowanie. Ponadto wszystkie dane są przechowywane w plikach tekstowych, co może powodować problemy z wydajnością i bezpieczeństwem podczas skalowania.

Struktura programu:

controllers:

- AuthenticationController
- BankException
- ClientController
- InitializableUI

dao:

- AccountDao
- UserDao
- DatabaseAccessObject

gui:

- AuthenticaionPanel
- ClienPanel

models:

- Account
- User

services:

- Currency

Main

Opis klas, algorytmów, zmiennych:

Klasa User: Ta klasa reprezentuje użytkownika w systemie bankowym. Posiada trzy atrybuty: username (String), password (String) i userType (UserType). UserType jest wyliczeniem z dwiema możliwymi wartościami: Client i Employee.

Klasa Account: Ta klasa reprezentuje konto bankowe. Posiada trzy atrybuty: accountNumber (String), currency (Currency) i balance (double). Waluta to enum z dwiema możliwymi wartościami: USD i PLN.

Currency enum: Ten enum reprezentuje typ waluty. Ma dwie możliwe wartości: USD i PLN.

Wyliczenie UserType: To wyliczenie reprezentuje typ użytkownika. Ma dwie możliwe wartości: Client i Employee.

Klasa ClientController: Ta klasa obsługuje funkcjonalność dostępną dla klientów. Zawiera metody do otwierania konta, przeglądania salda, dokonywania wpłat, wypłacania środków, ubiegania się o kredyt i wylogowywania się.

Klasa ClientPanel: Ta klasa reprezentuje panel klienta w GUI. Zawiera przyciski do otwierania konta, przeglądania salda, dokonywania wpłat,

wypłacania środków, wnioskowania o kredyt i wylogowywania się. Zawiera również obszar tekstowy do wyświetlania szczegółów konta i listy kont.

Klasa `AuthenticationController`: Ta klasa jest kontrolerem uwierzytelniania. Zawiera metody logowania i rejestracji nowego użytkownika. Po pomyślnym uwierzytelnieniu inicjalizuje interfejs klienta lub pracownika.

Klasa `AuthenticationPanel`: Ta klasa jest panelem uwierzytelniania w GUI. Zawiera pola do wprowadzania nazwy użytkownika i hasła oraz listę rozwijaną do wyboru typu użytkownika. Zawiera również przyciski do logowania i rejestracji nowego użytkownika.

Klasa `UserDao`: Ta klasa obsługuje ładowanie i zapisywanie danych użytkownika.

Klasa `AccountDao`: Ta klasa obsługuje ładowanie i zapisywanie danych konta.

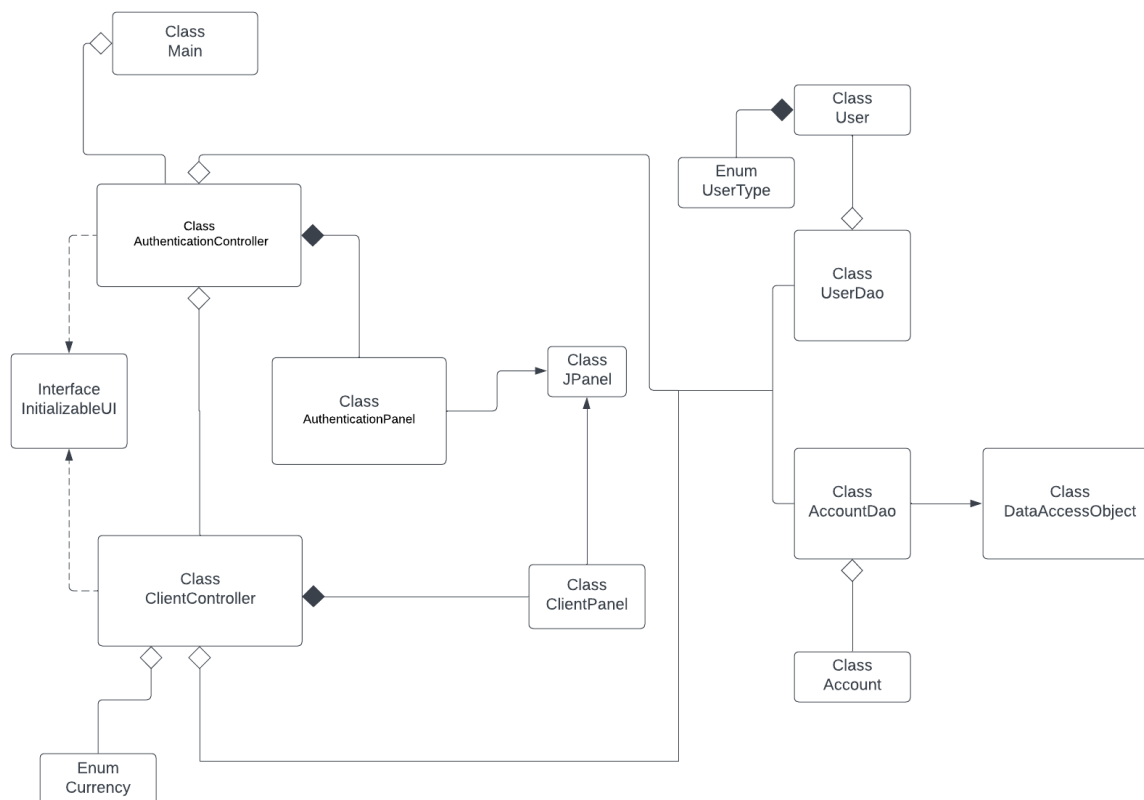
Klasa `Main`: Ta klasa jest punktem wejścia do aplikacji. Zawiera metodę `main`, która inicjalizuje główne komponenty aplikacji i rozpoczyna proces uwierzytelniania.

Klasa `BankException`: Ta klasa dziedziczy po klasie `IOException` i spełnia ją funkcję.

Klasa abstrakcyjna `DatabaseAccessObject`: Definiuje podstawowy interfejs dostępu do bazy danych. Posiada metody ładowania i zapisywania danych, a także konwertowania obiektów na ciągi znaków i odwrotnie. Klasy dziedziczące po tej klasie abstrakcyjnej muszą implementować te metody zgodnie z konkretną bazą danych, z którą pracują

Interfejs `InitializableUI`: definiuje metodę `startUserInterface()`. Metoda ta służy do inicjalizacji i uruchomienia interfejsu użytkownika.

Diagram klas w UML:



Kod programu:

Kod programu jest dostępny w udostępnionym repozytorium GitHub. Główne klasy i ich funkcjonalność zostały opisane powyżej.

[Kod programu tutaj.](https://github.com/Shwalz/BankingSystemApp)

<https://github.com/Shwalz/BankingSystemApp>

Ważne cechy programu:

public void startAuthentication()

```
public void startAuthentication() {  
    AuthenticationPanel panel = new AuthenticationPanel(this);  
    authenticationFrame.getContentPane().add(panel);  
    authenticationFrame.pack();  
    authenticationFrame.setVisible(true);  
}
```

Ta metoda rozpoczyna proces uwierzytelniania, otwierając panel uwierzytelniania. Jest to pierwsza metoda wywoływana podczas uruchamiania aplikacji i ma kluczowe znaczenie dla zabezpieczenia systemu.

public void saveUser()

```
public void saveUser(User user) throws IOException {  
    List<User> users = loadUsers();  
    users.add(user);  
    try (BufferedWriter writer = new BufferedWriter(  
        new OutputStreamWriter(  
            new FileOutputStream(USER_DATA_FILE),  
            FILE_ENCODING))) {  
        for (User u : users) {  
            String line = convertUserToLine(u);  
            writer.write(line);  
            writer.newLine();  
        }  
    } catch (IOException e) {  
        throw new IOException("Failed to save user data", e);  
    }  
}
```

Ta metoda zapisuje dane użytkownika do pliku danych użytkownika. Najpierw ładuje wszystkich istniejących użytkowników, następnie dodaje nowego użytkownika do listy, a na koniec zapisuje zaktualizowaną listę z powrotem do pliku. Jeśli wystąpi błąd zapisu do pliku, metoda generuje wyjątek `IOException`.

Przykładowe dane i wyniki:

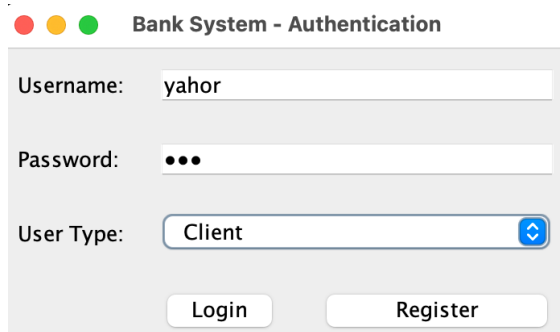
Aplikacja używa plików tekstowych do przechowywania danych użytkownika i konta. Na przykład dane użytkownika są przechowywane w formacie: username,password,userType.

```
yahor,123,Client  
admin,admin,Employee
```

Dane konta są przechowywane w formacie: accountNumber, currency,balance.

```
1,USD,5000.0  
2,PLN,2500.0
```

Interfejs logowania i rejestracji:



The image shows a web form titled "Bank System - Authentication". It contains three input fields: "Username:" with the value "yahor", "Password:" with three dots indicating a masked password, and "User Type:" with a dropdown menu showing "Client". Below these fields are two buttons: "Login" and "Register".

Bank System - Authentication

Username: yahor

Password: ●●●

User Type: Client

Login Register

Główny interfejs użytkownika:

Bank System - Client Interface

Open Account

1
2

View Balance Deposit

Withdraw Loan Application

Logout

Account Number: 2
Currency: PLN
Balance: 2500.0

Instrukcja dla użytkownika:

1. Uruchom aplikację: Uruchom plik Main.java, aby uruchomić aplikację. Po uruchomieniu otworzy się panel uwierzytelniania.
2. Rejestracja: Jeśli jesteś nowym użytkownikiem, wybierz "Zarejestruj się". Wprowadź nazwę użytkownika, hasło i typ użytkownika (Klient lub Pracownik), a następnie kliknij "Zarejestruj się". Po pomyślnym zarejestrowaniu zostaniesz automatycznie zalogowany.
3. Logowanie: Jeśli jesteś już zarejestrowany, wprowadź swoją nazwę użytkownika, hasło i wybierz typ użytkownika, a następnie kliknij "Zaloguj się". Jeśli uwierzytelnienie przebiegnie pomyślnie, zostaniesz przeniesiony do panelu użytkownika.
4. Otwieranie konta: W panelu użytkownika wybierz opcję "Otwórz konto". Wprowadź wymagane dane, a następnie kliknij "Wyślij". Twoje nowe konto zostanie utworzone.

5. Wyświetl saldo: Aby wyświetlić saldo konta, wybierz konto z listy i kliknij "Wyświetl saldo". Saldo zostanie wyświetlone w polu tekstowym.
6. Depozyt: Aby wpłacić pieniądze na swoje konto, wybierz konto z listy i naciśnij przycisk "Depozyt". Wprowadź kwotę, którą chcesz wpłacić i naciśnij "Wyślij". Kwota zostanie przelana na konto użytkownika.
7. Wypłata środków: Aby wypłacić środki z konta, wybierz konto z listy i kliknij przycisk "Wypłać". Wprowadź kwotę, którą chcesz wypłacić i naciśnij "Wyślij". Kwota zostanie pobrana z konta.
8. Wniosek o pożyczkę: Aby złożyć wniosek o pożyczkę, wybierz konto z listy i kliknij "Wniosek o pożyczkę". Wprowadź wymagane dane i kliknij "Wyślij". Wniosek zostanie przesłany do rozpatrzenia.
9. Wylogowanie: Aby wylogować się z systemu, naciśnij "Wyloguj". Nastąpi powrót do panelu uwierzytelniania.