**Kolegium Nauk Przyrodniczych  
Uniwersytet Rzeszowski**

**Przedmiot:**

**Bazy danych**

**Dokumentacja projektu**

**Zarządzanie budżetem domowym**

**Wykonali:**

Miłosz Józefczyk 125185

Paweł Grechuta 125184

Sebastian Domin 127773

**Prowadzący: Mgr inż. Aleksander Wojtowicz**

**Kierunek, rok, grupa: Informatyka i ekonometria, II, lab. 1**

**Rzeszów 2024**

**Wstęp:**

Projekt to aplikacja desktopowa, której zadaniem jest stworzenie przyjaznej i intuicyjnej platformy do zarządzania budżetem domowym. Użytkownik po zalogowaniu się, może monitorować oraz przeprowadzać operacje CRUD na swoich wydatkach, oraz przychodach.

**Technologie oraz narzędzia:**

W projekcie wykorzystany został język programowania C#. Aplikacja korzysta z bazy danych stworzonej przez system Microsoft SQL. Do obsługi C# wykorzystywane jest środowisko Visual Studio, zaś do zarządzania bazą danych wykorzystywany jest Microsoft SQL Server Management Studio.

**Linki do pobrania:**

<https://www.microsoft.com/pl-pl/sql-server/sql-server-downloads>

https://visualstudio.microsoft.com/pl/free-developer-offers/

**DIAGRAM ERD BAZY DANYCH:**

**Obraz zawierający tekst, zrzut ekranu, diagram, Równolegle

Opis wygenerowany automatycznie**

Rysunek 1 - Diagram ERD

Diagram ten ilustruje system zarządzania użytkownikami, transakcjami, celami oszczędnościowymi, przypomnieniami, raportami okresowymi i grupami użytkowników

Tabela "Users" zawiera informacje o użytkownikach, takie jak UserID, Username, Password, Email i Role. Jest centralną tabelą, powiązaną z wieloma innymi tabelami:

* "Transactions" przechowuje transakcje użytkowników (TransactionID, UserID, Amount, Date, Description, Type).
* "SavingGoals" zawiera cele oszczędnościowe (GoalID, UserID, Amount, GoalType, Description).
* "Reminders" przechowuje przypomnienia (ReminderID, UserID, Frequency, NextReminderDate, Description).
* "PeriodicReports" zawiera raporty okresowe (ReportID, UserID, ReportType, DateFrom, DateTo, ReportData).
* "UserGroupMemberships" przechowuje członkostwa w grupach użytkowników (UserID, GroupID).

Tabela "UserGroups" zawiera grupy użytkowników (GroupID, GroupName) i jest powiązana z "UserGroupMemberships".

Pozostałe tabele to "Items" (ItemID, Name, Price) oraz "EFMigrationsHistory" (MigrationId, ProductVersion), które nie mają bezpośrednich relacji z innymi tabelami w diagramie.

**RELACJE W BAZIE DANYCH:**

1. Users (Użytkownicy)
   * Relacje:
     + Jedna encja "Users" jest powiązana z wieloma encjami "Transactions". Każda transakcja ma przypisane pole UserID, które jest kluczem obcym odnoszącym się do UserID w tabeli "Users".
     + Jedna encja "Users" jest powiązana z wieloma encjami "SavingGoals". Każdy cel oszczędnościowy ma pole UserID, które jest kluczem obcym odnoszącym się do UserID w tabeli "Users".
     + Jedna encja "Users" jest powiązana z wieloma encjami "Reminders". Każde przypomnienie ma pole UserID, które jest kluczem obcym odnoszącym się do UserID w tabeli "Users".
     + Jedna encja "Users" jest powiązana z wieloma encjami "UserGroupMemberships". Każde członkostwo w grupie użytkowników ma pole UserID, które jest kluczem obcym odnoszącym się do UserID w tabeli "Users".
     + Jedna encja "Users" jest powiązana z wieloma encjami "PeriodicReports". Każdy raport okresowy ma pole UserID, które jest kluczem obcym odnoszącym się do UserID w tabeli "Users".
2. Transactions (Transakcje)
   * Relacje:
     + Każda transakcja jest powiązana z jednym użytkownikiem poprzez pole UserID, co tworzy relację wiele do jednego z tabelą "Users".
3. SavingGoals (Cele oszczędnościowe)
   * Relacje:
     + Każdy cel oszczędnościowy jest powiązany z jednym użytkownikiem poprzez pole UserID, co tworzy relację wiele do jednego z tabelą "Users".
4. Reminders (Przypomnienia)
   * Relacje:
     + Każde przypomnienie jest powiązane z jednym użytkownikiem poprzez pole UserID, co tworzy relację wiele do jednego z tabelą "Users".
5. PeriodicReports (Raporty okresowe)
   * Relacje:
     + Każdy raport okresowy jest powiązany z jednym użytkownikiem poprzez pole UserID, co tworzy relację wiele do jednego z tabelą "Users".
6. UserGroups (Grupy użytkowników)
   * Relacje:
     + Każda grupa użytkowników może mieć wiele członkostw w grupach użytkowników, co tworzy relację jeden do wielu z tabelą "UserGroupMemberships".
7. UserGroupMemberships (Członkostwa w grupach użytkowników)
   * Relacje:
     + Każde członkostwo w grupie użytkowników jest powiązane z jednym użytkownikiem poprzez pole UserID, co tworzy relację wiele do jednego z tabelą "Users".
     + Każde członkostwo w grupie użytkowników jest powiązane z jedną grupą użytkowników poprzez pole GroupID, co tworzy relację wiele do jednego z tabelą "UserGroups".

Pozostałe tabele, czyli "Items" oraz "EFMigrationsHistory", nie mają bezpośrednich relacji z innymi tabelami w tym diagramie.

**Uruchomienie aplikacji:**

* Uruchamiamy SQL Server Management Studio

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

Rysunek 2- Uruchamianie SQL Server Management Studio

* Łączymy się z lokalnym serwerem

Obraz zawierający tekst, zrzut ekranu, numer, Czcionka

Opis wygenerowany automatycznie

Rysunek 3 - łączenie się z lokalnym serwerem

Zaznaczamy tutaj opcje „Trust server certificate” i wciskamy Connect.

* Tworzymy nowe zapytanie wciskając opcje New Query

Obraz zawierający tekst, zrzut ekranu, wyświetlacz, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 4 - tworzenie zapytania

* Tworzymy nową bazę danych o nazwie FinancialDatabaseApp

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie

Rysunek 5 - tworzymy bazę danych

* Otwieramy projekt z aplikacją w Visual Studio:

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

Rysunek 6 - otwieramy projekt przy pomocy VS

* Otwieramy Eksplorator rozwiązań skrótem Ctrl + ; , a następnie uruchamiamy rozwiązanie o nazwie financialApp.sln:

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

Rysunek 7 - Uruchamiamy rozwiązanie

* Tworzymy połączenie z bazą danych:

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie

Rysunek 8 - Tworzymy łączenie z bazą danych

* Uzupełniamy połączenie z bazą danych o nazwe serwera, zaznaczenie certyfikatu serwera zaufania i wybrania odpowiedniej bazy danych:

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, numer

Opis wygenerowany automatycznie

Rysunek 9 - tworzymy połączenie projektu z bazą

* Uruchamiamy Narzędzia -> Menedżer Pakietów NuGet -> Konsola Menedżera Pakietów

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

Rysunek 10 - uruchamiamy konsole menedżera pakietów

* Wpisujemy w konsoli update-database aby uruchomić migracje:

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

Rysunek 11 - uruchamiamy migracje

* Z pliku insert.sql kopiujemy zawartość i tworzymy nowe zapytanie na naszej bazie danych wklejając zawartość pliku insert.sql

Obraz zawierający tekst, elektronika, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 12 - Uruchamiamy inserty

* Kompilujemy nasz program

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

Rysunek 13 - Kompilujemy program

* Uruchamiamy nasz program:

Obraz zawierający zrzut ekranu, Czcionka, Grafika, design

Opis wygenerowany automatycznie

Rysunek 14 - Uruchamiamy program

**GUI:**

Pierwszym okienkiem, jakie użytkownik zobaczy po uruchomieniu aplikacji, jest okienko logowania :

Obraz zawierający tekst, zrzut ekranu, Czcionka, Prostokąt

Opis wygenerowany automatycznie

Rysunek 15 - okienko logowania

Z poziomu tego okienka, użytkownik ma możliwość logowania, bądź wybrać opcje rejestracji:

Obraz zawierający tekst, zrzut ekranu, Czcionka, Prostokąt

Opis wygenerowany automatycznie

Rysunek 16 – okienko rejestracji

Jeśli użytkownik wybierze opcje rejestracji to ma możliwość powrotu do okna logowania po wciśnięciu przycisku Login.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, System operacyjny

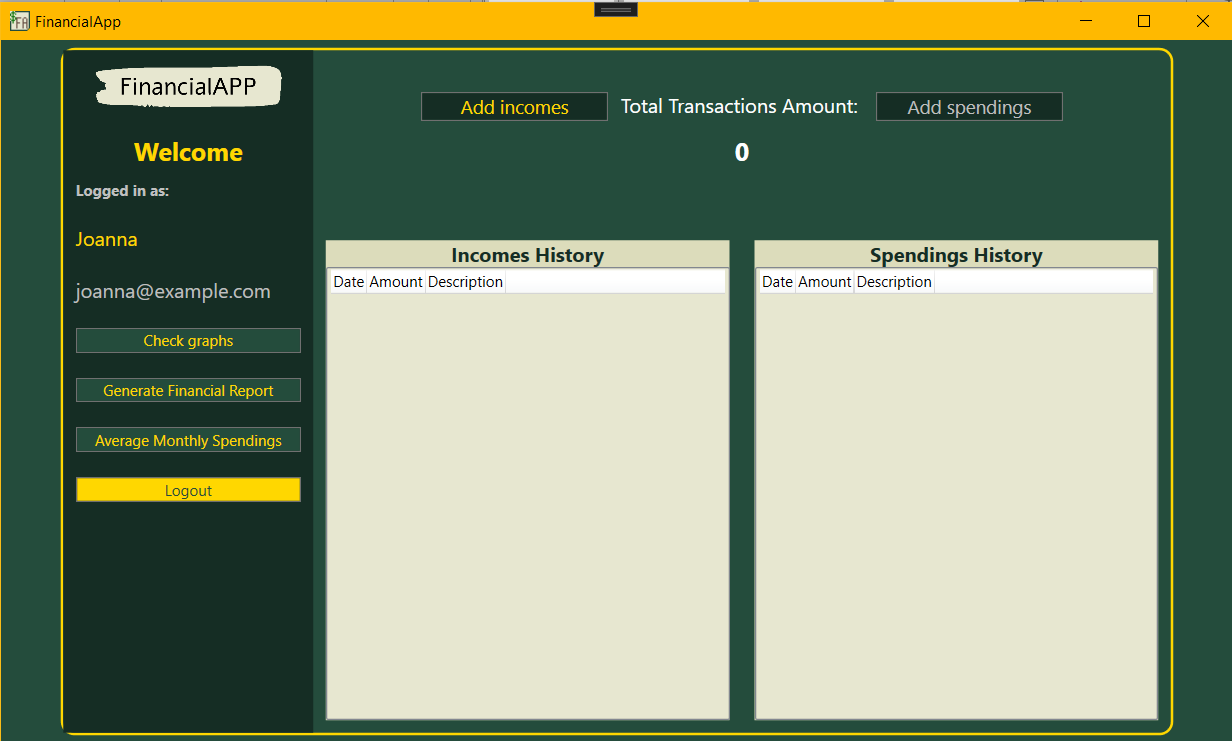
Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, oprogramowanie, Strona internetowa

Opis wygenerowany automatycznie

Rysunek 17- niepoprawna rejestracja Rysunek 18 -poprawna rejestracja

Jak widać na powyższych zdjęciach, jeżeli dane przy rejestracji będą takie same jak dane w bazie danych, rejestracja się nie uda – w przeciwnym wypadku rejestracja będzie udana

**Widok użytkownika:**

Po zalogowaniu się na konto, użytkownik widzi następujący widok:  


Rysunek 19 - widok UserDashboard

Z poziomu tego okienka możemy m.in. dodać przychód:  
Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Prostokąt

Opis wygenerowany automatycznie

Rysunek 20 - dodanie przychodu

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Ikona komputerowa

Opis wygenerowany automatycznie

Rysunek 21 - przychód dodany

Po dodaniu przychodu, jej wartość sumuję się w Total Transactions Amount, a jego opis, wartość oraz data dodania będzie wypisana w Incomes History.

Możemy również dodać wydatek:

Obraz zawierający tekst, zrzut ekranu, Prostokąt, oprogramowanie

Opis wygenerowany automatycznieRysunek 22- dodawanie wydatku

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Strona internetowa

Opis wygenerowany automatycznie

Rysunek 23 - wydatek dodany

Po dodaniu tego wydatku, zostanie on również zsumowany w Total Transactions Amount, i będzie wpisany w Spendings History.

Możemy również wygenerować wykres prezentujący stosunek wydatków do przychodów:

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, diagram

Opis wygenerowany automatycznie

Rysunek 24 - wykres stostunku wydatków do przychodu

Użytkownik ma również możliwość wygenerowania raportu swoich transakcji:

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Opis wygenerowany automatycznie

Rysunek 25 - Raport finansowy

Użytkownik może także sprawdzić średnie miesięczne wydatki:

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, numer

Opis wygenerowany automatycznie

Rysunek 26 - Raport średnich miesięcznych wydatków

**Widok administratora:**

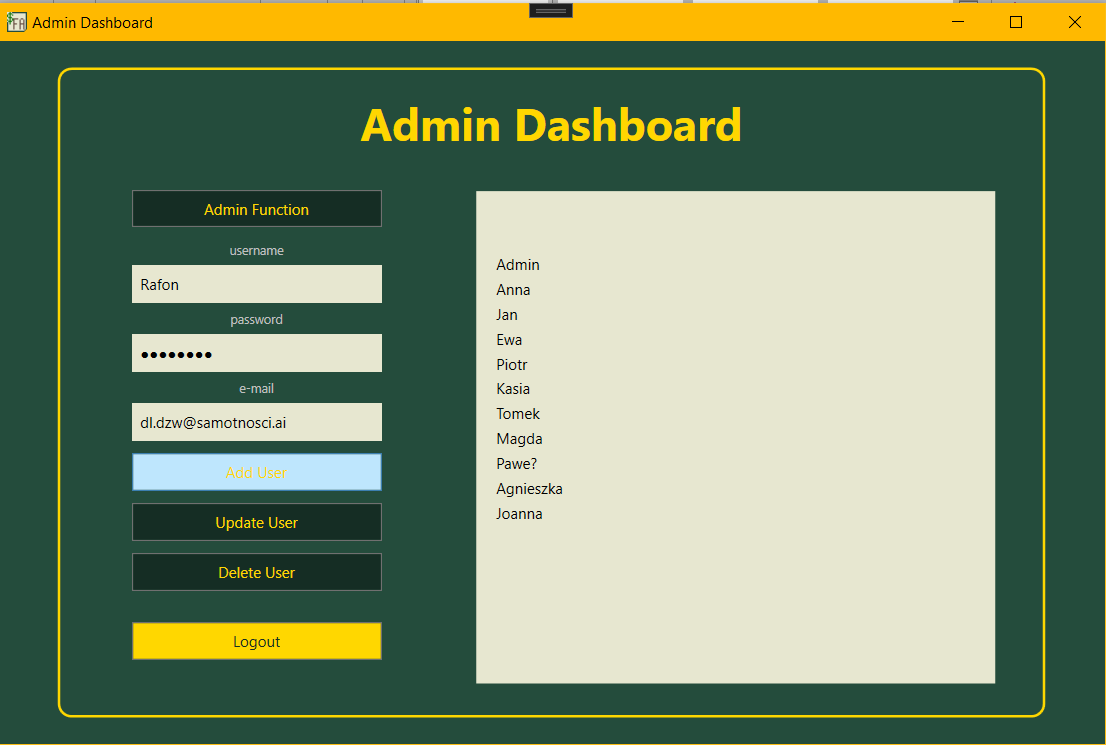
Jeśli użytkownik posiada rolę admin, to zamiast UsersDashboard, widzi AdminDashboard:

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Prostokąt

Opis wygenerowany automatycznie

Rysunek 27 - widok AdminDashboard

Z tego poziomu, administrator ma możliwość usunięcia użytkownika, edytowania jego danych bądź dodania użytkownika:



Rysunek 28 - dodanie użytkownika

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Ikona komputerowa

Opis wygenerowany automatycznie

Rysunek 29 - użytkownik został dodany

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Opis wygenerowany automatycznie

Rysunek 30 - usuwanie użytkownika

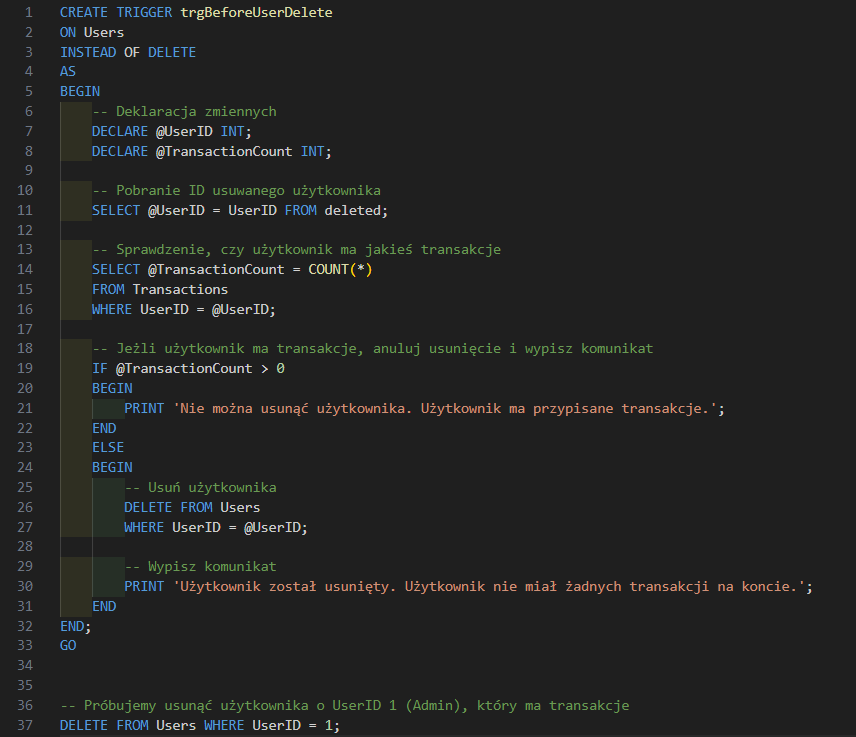
Obraz zawierający tekst, zrzut ekranu, oprogramowanie, numer

Opis wygenerowany automatycznie

Rysunek 31 - użytkownik został usunięty

**FUNKCJE BAZODANOWE:**

**Trigger 1 – trgBeforeUserDelete**

****

Rysunek 32- Wyzwalacz trgBeforeUserDelete

Ten wyzwalacz jest uruchamiany przed usunięciem użytkownika z tabeli Users. Zamiast natychmiastowego usunięcia użytkownika, wyzwalacz sprawdza, czy użytkownik ma przypisane transakcje w tabeli Transactions. Jeśli użytkownik ma transakcje, wyzwalacz anuluje usunięcie i wyświetla komunikat informujący, że nie można usunąć użytkownika z powodu przypisanych transakcji. W przeciwnym razie, użytkownik zostaje usunięty, a wyzwalacz informuje, że usunięcie powiodło się, ponieważ użytkownik nie miał żadnych transakcji**.**

**Trigger 2 - trgCheckNegativeBalanceObraz zawierający tekst, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie**

Rysunek 33- Wyzwalacz trgCheckNegativeBalance

Ten wyzwalacz jest uruchamiany po dodaniu nowej transakcji do tabeli Transactions. Dla każdej nowo wstawionej transakcji, wyzwalacz oblicza aktualne saldo użytkownika, biorąc pod uwagę wszystkie przychody i wydatki. Jeśli nowa transakcja powoduje, że saldo użytkownika stanie się ujemne, transakcja jest anulowana i wyświetlany jest komunikat informujący, że użytkownik nie może mieć ujemnego salda.

**Function 1 – CalculateUserBalance**

**Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie**

Rysunek 34 - Funkcja fnAverageMonthlySpendings

Ta funkcja służy do obliczania salda finansowego użytkownika na podstawie jego przychodów i wydatków w określonym okresie. Funkcja przyjmuje trzy parametry: UserID, StartDate i EndDate. Oblicza ona sumę przychodów i wydatków użytkownika w podanym okresie oraz zwraca wynik salda. Funkcja wyświetla również komunikat informujący, czy saldo użytkownika jest dodatnie, czy ujemne.

**Function 2 - GetTransactionSummaryObraz zawierający tekst, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie**

Rysunek 35 - Funkcja fnFinancialReport

Ta funkcja zwraca podsumowanie transakcji użytkownika w podanym okresie. Przyjmuje trzy parametry: UserID, StartDate i EndDate. Oblicza ona sumy przychodów i wydatków użytkownika w danym okresie oraz zwraca te wartości razem z opisem.

**Procedure 1 - CalculateUserBalanceObraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie**

Rysunek 36 - Procedura CalculateUserBalance

Procedura ta działa podobnie jak funkcja CalculateUserBalance, z tą różnicą, że jest to procedura przechowywana. Oblicza saldo finansowe użytkownika na podstawie jego przychodów i wydatków w określonym okresie, przyjmując parametry UserID, StartDate i EndDate. Wyświetla komunikaty informujące o stanie salda użytkownika**.**

**Procedure 2 - GenerateUserFinancialReportObraz zawierający tekst, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie**

Rysunek 37 - Procedura GenerateUserFinancialReport

Procedura ta generuje raport finansowy użytkownika za dany okres. Przyjmuje trzy parametry: UserID, StartDate i EndDate. Procedura oblicza sumę przychodów, wydatków oraz saldo użytkownika w podanym okresie. Następnie wyświetla szczegółowe podsumowanie finansowe oraz listę wszystkich transakcji użytkownika w danym okresie.

Wszystkie te funkcje, wyzwalacze i procedury są zaprojektowane w celu zarządzania i analizowania danych finansowych użytkowników, zapewniając integralność danych oraz dostarczając szczegółowych raportów finansowych