SmartFlow

**Opis projektu**

**SmartFlow** - aplikacja internetowa stworzona w technologii ASP.NET Core 8.0 do planowania zarządzania budżetem domowym, która umożliwia użytkownikom zarządzanie finansami w prosty i przejrzysty sposób. Jej główne funkcje pozwalają użytkownikom na zarządzanie transakcjami, tworzenie kategorii wydatków/przychodów, definiowanie celów oszczędnościowych oraz przeglądanie statystyk finansowych. Aplikacja oferuje prosty i intuicyjny interfejs użytkownika, oparty na Bootstrapie i ASP.NET MVC.

Autorzy:  
  
Mikołaj Stachura 138585  
Kamil Wieczorek 138600

## **Specyfikacja technologiczna**

* Framework: ASP.NET Core 8.0
* Baza danych: MS SQL Server
* Frontend: Bootstrap 5, jQuery, .css
* System zarządzania użytkownikami: ASP.NET Identity Framework

**Główne funkcjonalności:**

**Monitorowanie przychodów i wydatków**

* Dodawanie i edytowanie transakcji, takich jak przychody i wydatki, z przypisaniem kategorii (np. żywność, transport, rachunki).

**Planowanie oszczędności**

* Ustalanie celów oszczędnościowych z możliwością śledzenia postępów.

**Historia transakcji**

* Przegląd szczegółowej historii przychodów i wydatków z przypisanymi do nich kategoriami oraz notatkami.

**Personalizacja kategorii finansowych**

* Dodawanie, edytowanie i usuwanie kategorii wydatków i przychodów w zależności od potrzeb użytkownika.

**Przeglądanie podsumowania finansów w Panelu**

* Przeglądanie wszystkich transakcji, celów oszczędnościowych z podziałem na sumę przychodów i wydatków w Panelu

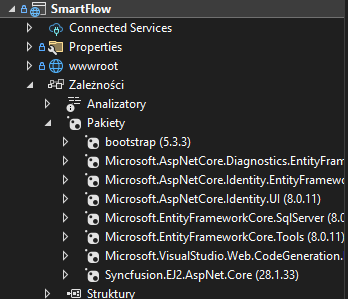
**Bezpieczeństwo i dostępność**

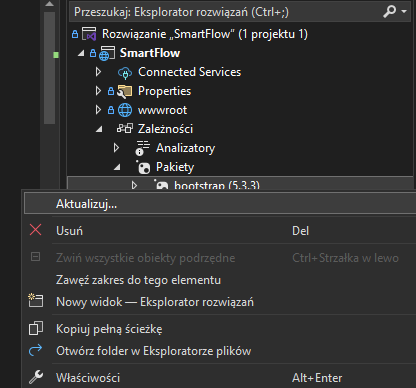
* Logowanie na indywidualne konto, przechowywanie danych w zabezpieczonej bazie.

## Instrukcje pierwszego uruchomienia projektu

1. Sklonuj repozytorium projektu za pomocą komendy   
‘git clone git@github.com:kaysior/SmartFlow.git’ użytej w git bash

2. Zainstaluj wymagane pakiety zależności (dependencies) w Visual Studio:



Prawym na pakiety oraz ”Aktualizuj”  
  


3. W terminalu uruchom polecenie:

Update-Database

W przypadku gdy Update-Database wyrzuca błędy, np. "There is already an object named 'AspNetRoles' in the database" albo inne, prosimy najpierw usunąć całą bazę danych, a następnie ją zaktualizować

Drop-Database

Update-Database

4. Skompiluj projekt i zyskaj dostęp do aplikacji.

## **Struktura projektu**

**Modele**

1. Category.cs

* Opis: Reprezentuje kategorię wydatków użytkownika.
* Pola:  
  + `Id` (int): Klucz główny.
  + `Name` (string): Nazwa kategorii (maks. 100 znaków, wymagane).

2. Transaction.cs

* Opis: Model opisujący pojedynczą transakcję finansową.
* Pola:
* `Id` (int): Klucz główny.
* `Amount` (decimal): Kwota transakcji (wymagane, > 0).
* `Date` (DateTime): Data wykonania transakcji.
* `CategoryId` (int): Klucz obcy powiązany z kategorią.
* `Note` (string): Opis transakcji.

3. SavingsGoal.cs

* Opis: Przechowuje informacje o celach oszczędnościowych użytkownika.
* Pola:
* `Id` (int): Klucz główny.
* `GoalName` (string): Nazwa celu.
* `TargetAmount` (decimal): Docelowa kwota oszczędności.
* `TargetDate` (decimal): Docelowa kwota oszczędności.
* `CurrentAmount` (decimal): Aktualny stan oszczędności (domyślnie 0).

**Kontrolery**

1. CategoriesController

* Metody:
* `Index` ([GET]): Wyświetla listę wszystkich kategorii.
* `Create` ([GET, POST]): Dodawanie nowych kategorii.
* `Edit` ([GET, POST]): Edycja istniejącej kategorii.
* `Delete` ([POST]): Usuwanie kategorii.

2. TransactionsController

* Metody:
* `Index` ([GET]): Lista transakcji użytkownika z możliwością filtrowania według
* `Create` ([GET, POST]): Dodawanie nowej transakcji z logiką dla kategorii “Oszczednosci”.
* `Edit` ([GET, POST]): Edycja szczegółów transakcji z logiką dla kategorii “Oszczednosci”.
* `Delete` ([POST]): Usuwanie transakcji.

3. SavingsGoalsController

* Metody:
* `Index` ([GET]): Lista celów oszczędnościowych.
* `Create` ([GET, POST]): Dodawanie nowego celu.
* `Edit` ([GET, POST]): Edycja istniejącego celu.
* `Delete` ([POST]): Usuwanie celu.

4. DashboardController

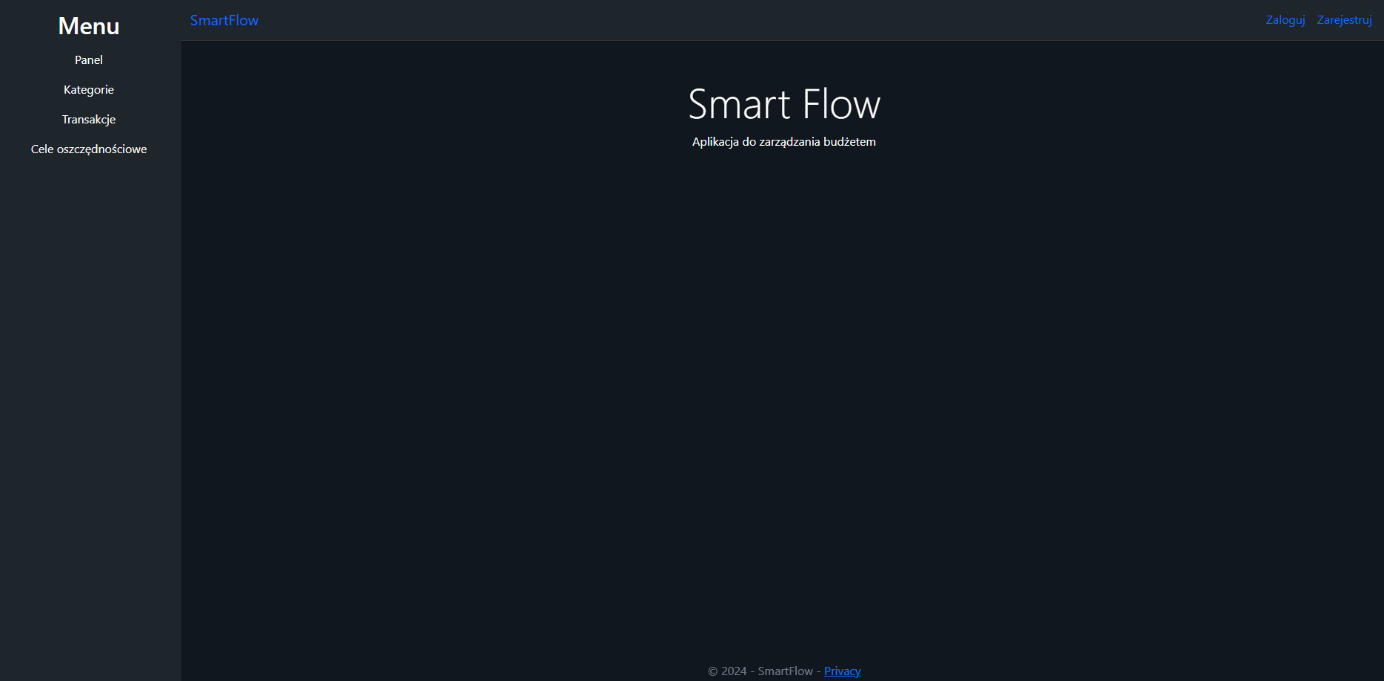
* Metody:
* `Index` ([GET]): Prezentacja podsumowania finansowego.

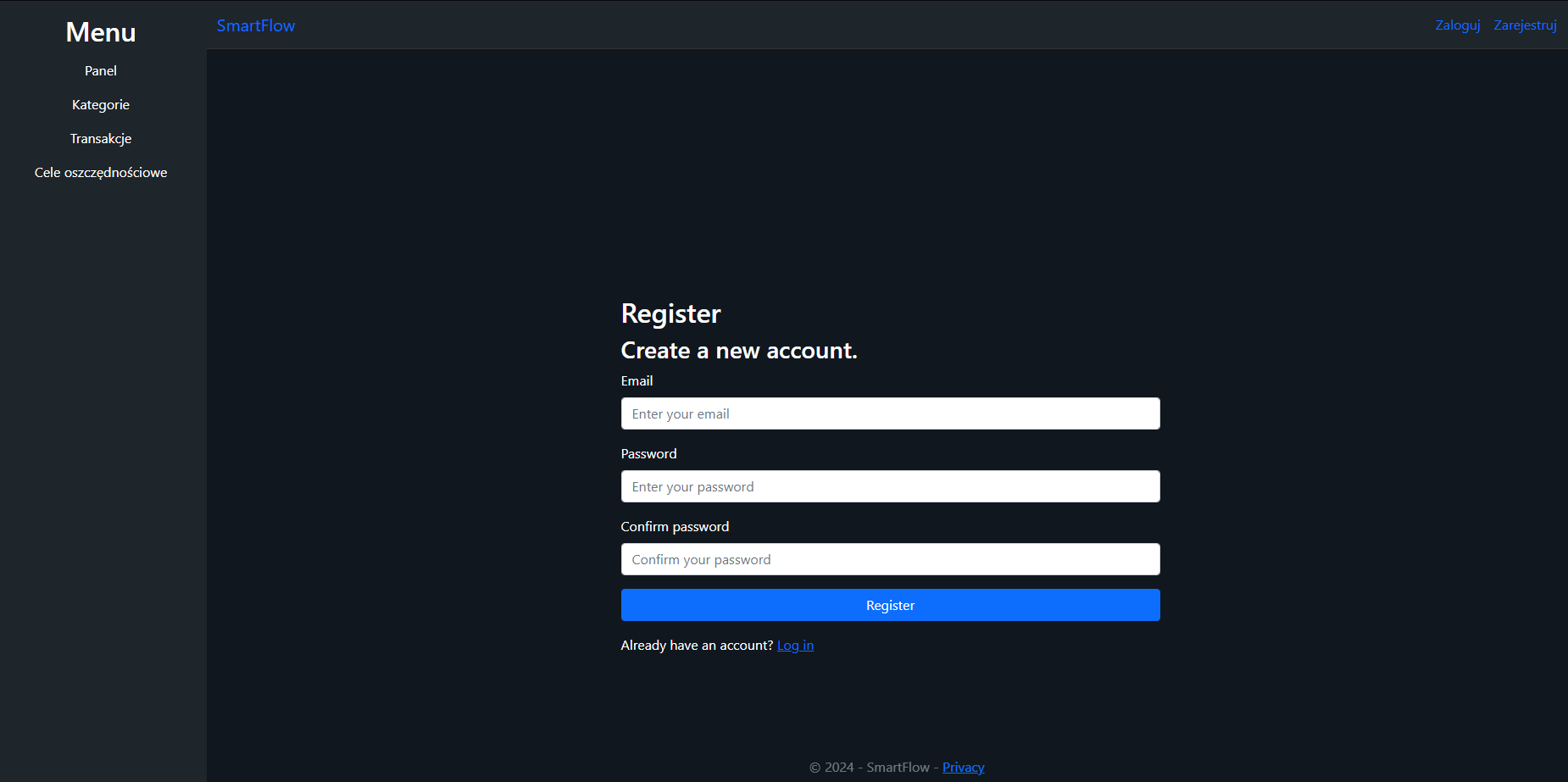
## **Widoki i UI**

* Strona główna: Strona wyświetlająca nazwę aplikacji oraz krótki opis czym ona jest. Prosty ekran logowania/rejestracji w prawym górnym rogu.
* Dashboard: Strona podsumowania, gdzie możemy wszystko w ładny i przejrzysty sposób monitorować.
* Widok transakcji: Historia transakcji z możliwością tworzenia, edytowania oraz ich usuwania.
* Widok kategorii: Lista kategorii z ikonami z możliwością tworzenia, edytowania oraz ich usuwania.
* Widok celów oszczędnościowych: Szczegółowa lista celów oszczędnościowych z możliwością tworzenia, edytowania oraz ich usuwania.

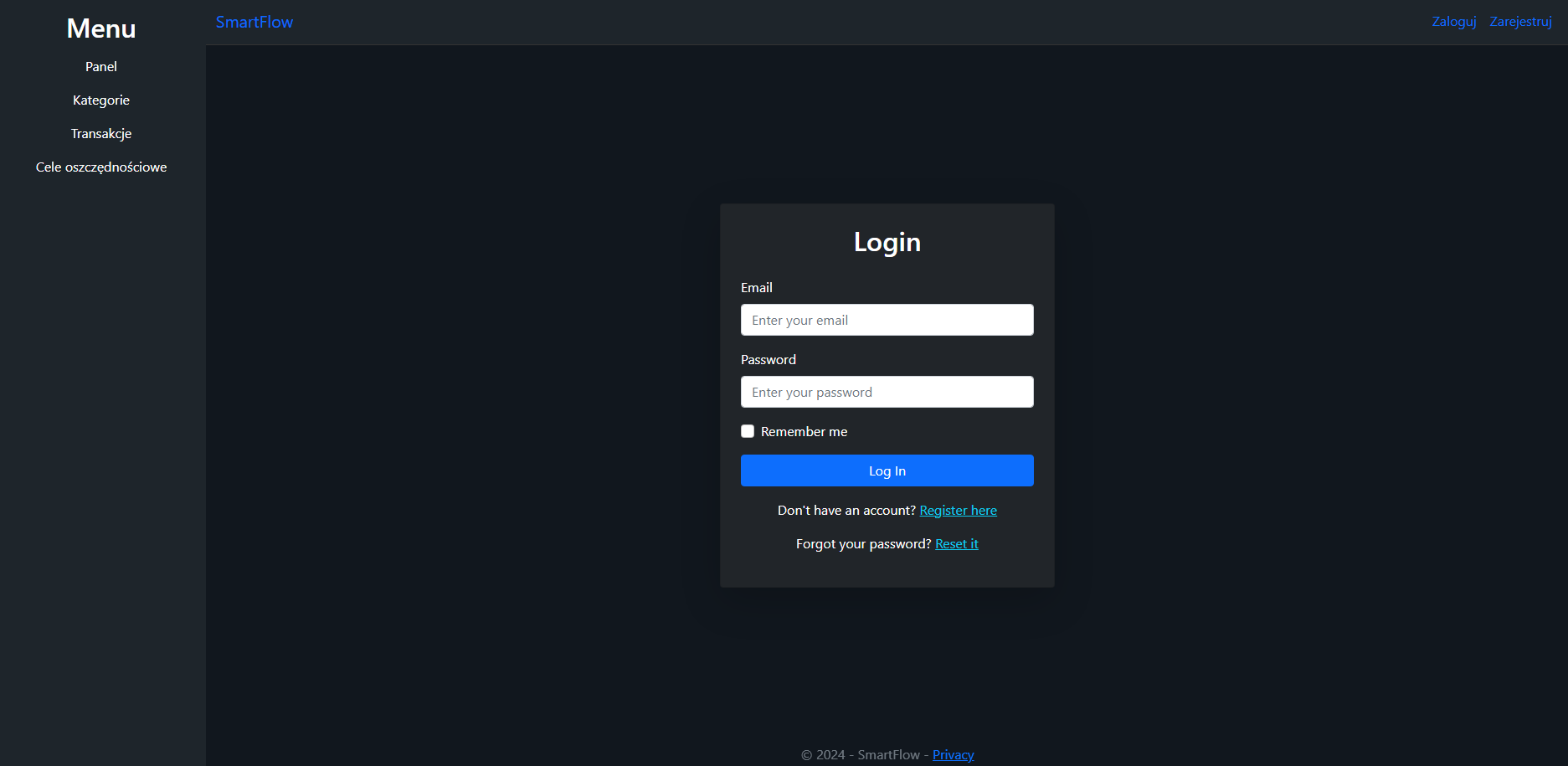
## Screeny z działania aplikacji

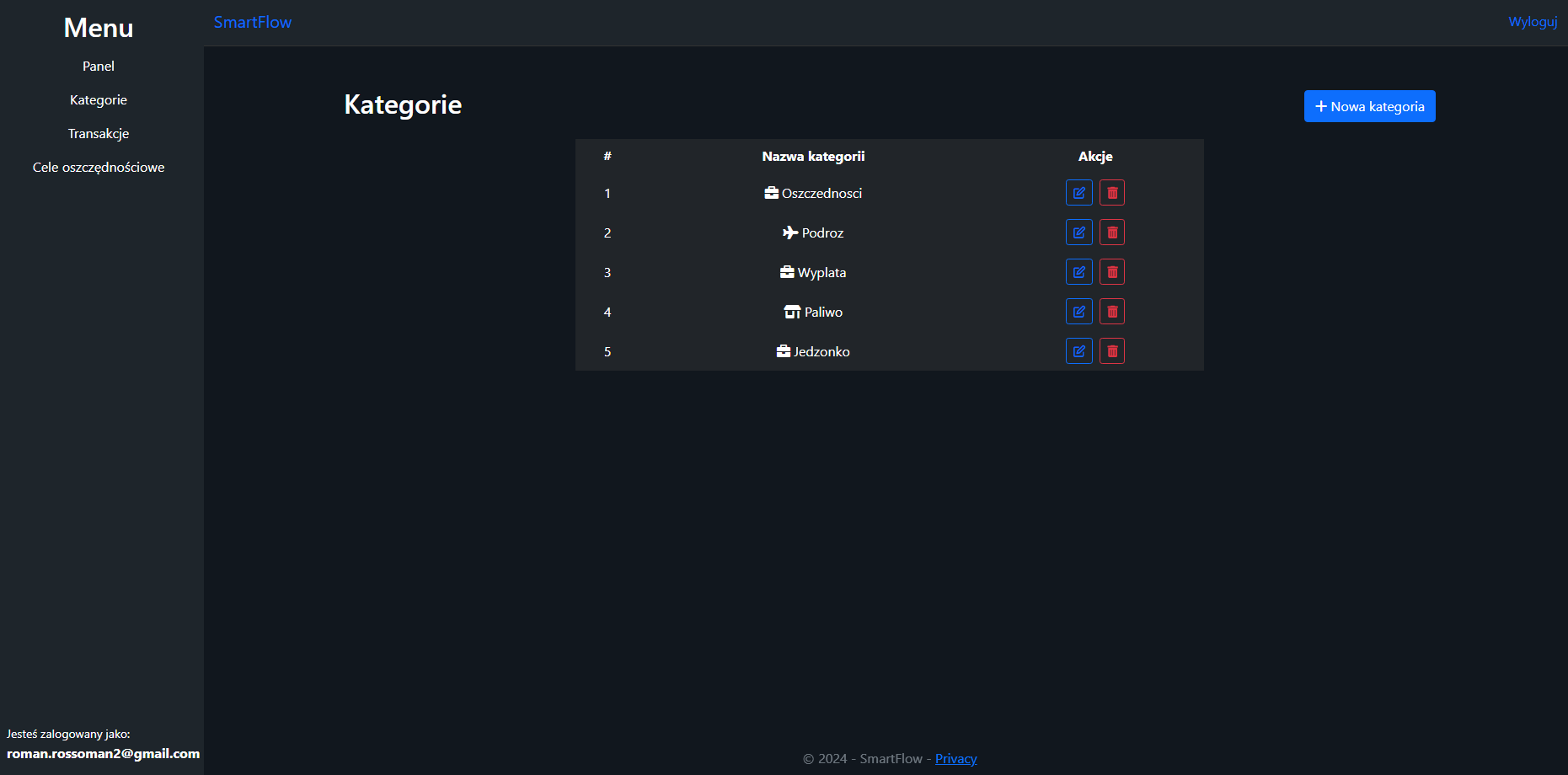
**Strona główna**



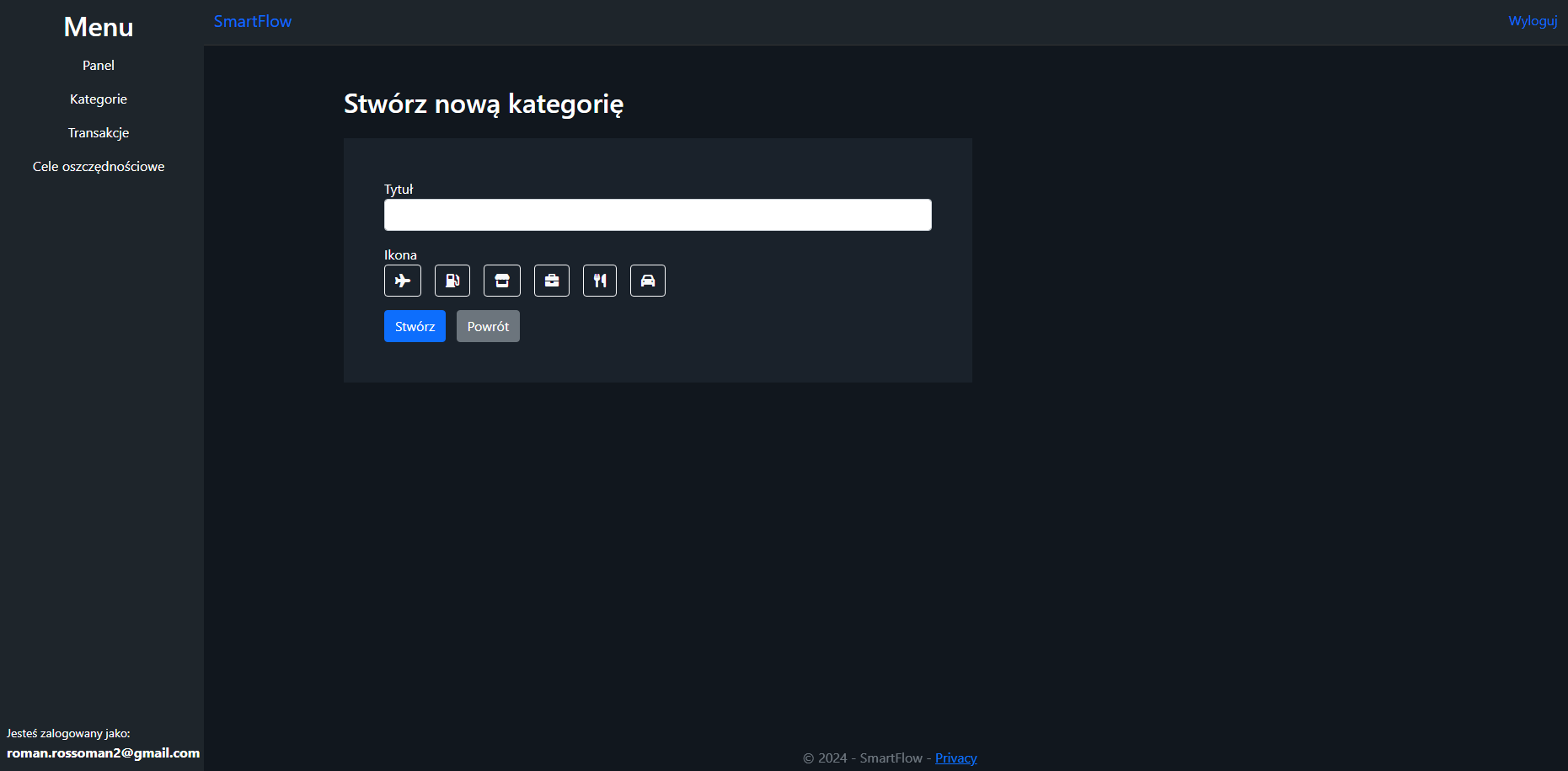
**Ekran rejestracji**  
  


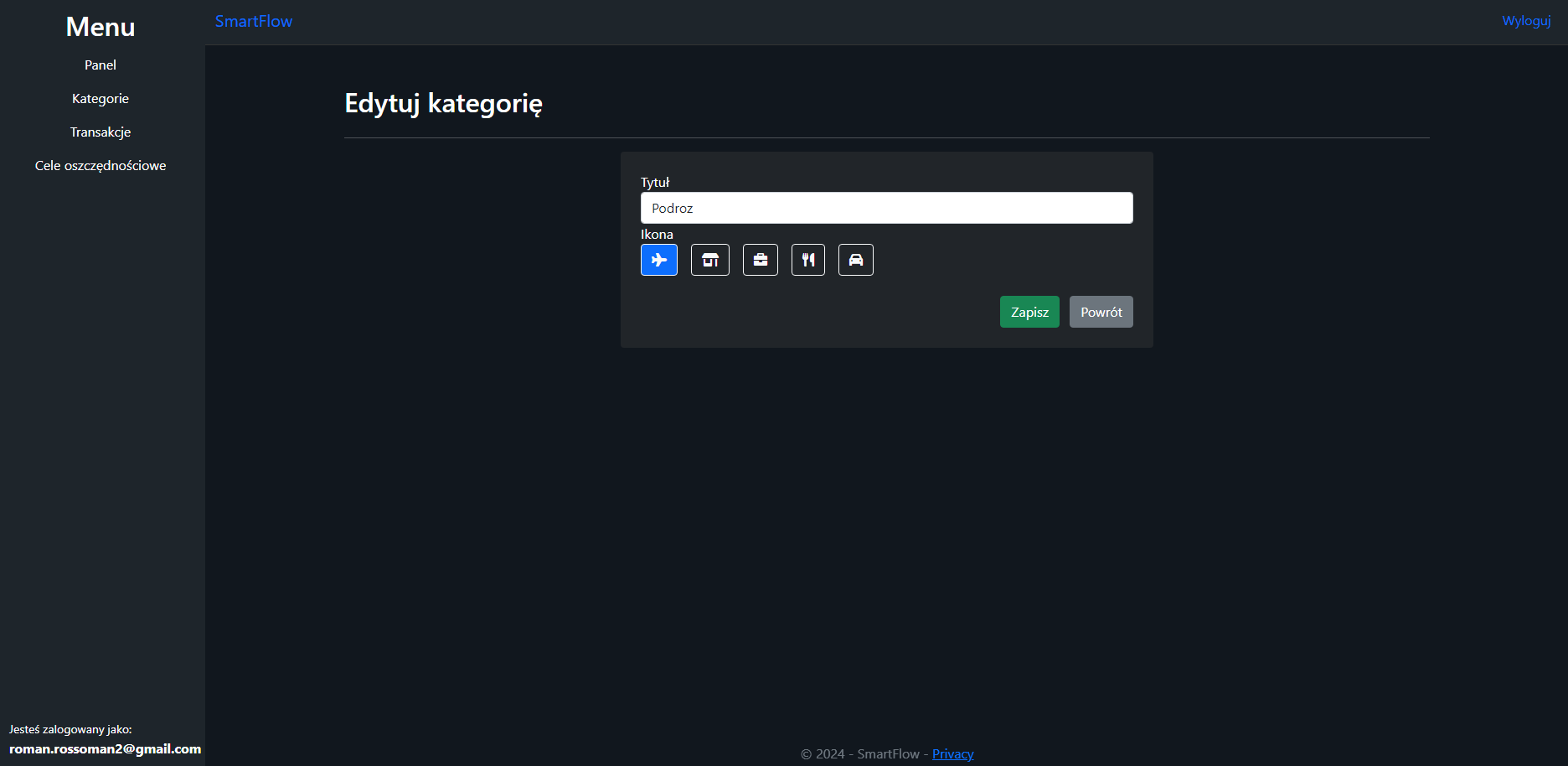
**Ekran logowania**



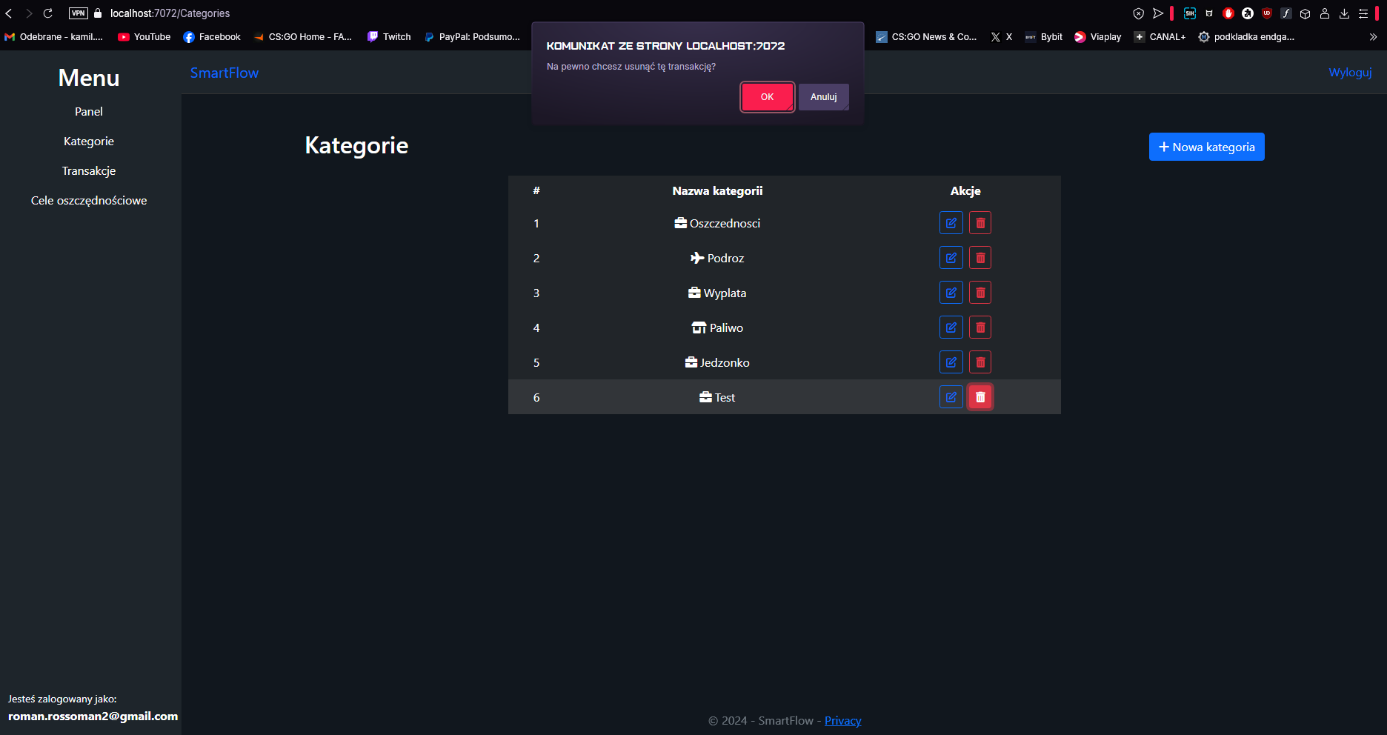
**Strona kategorii**  
  


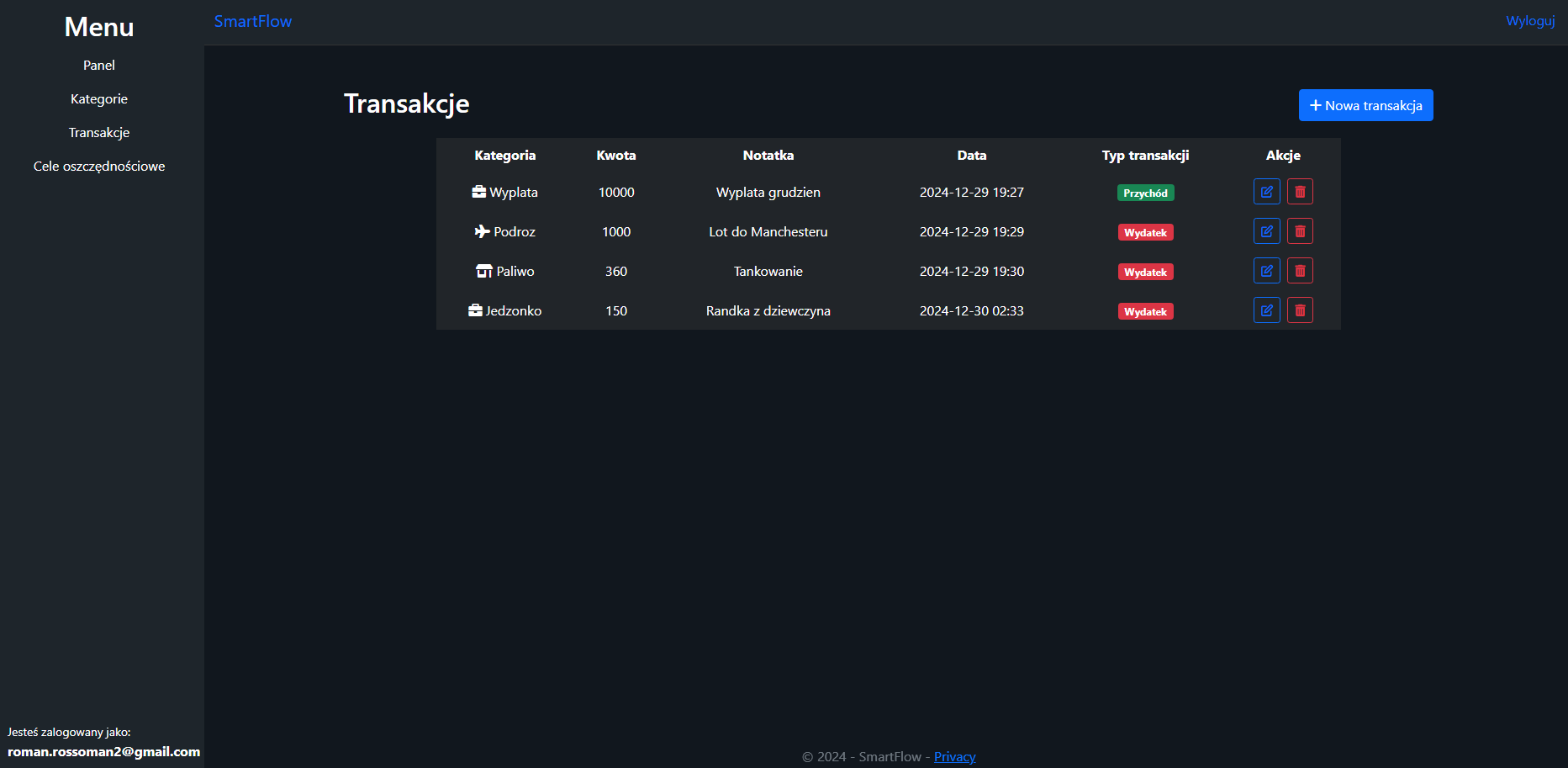
**Tworzenie kategorii**



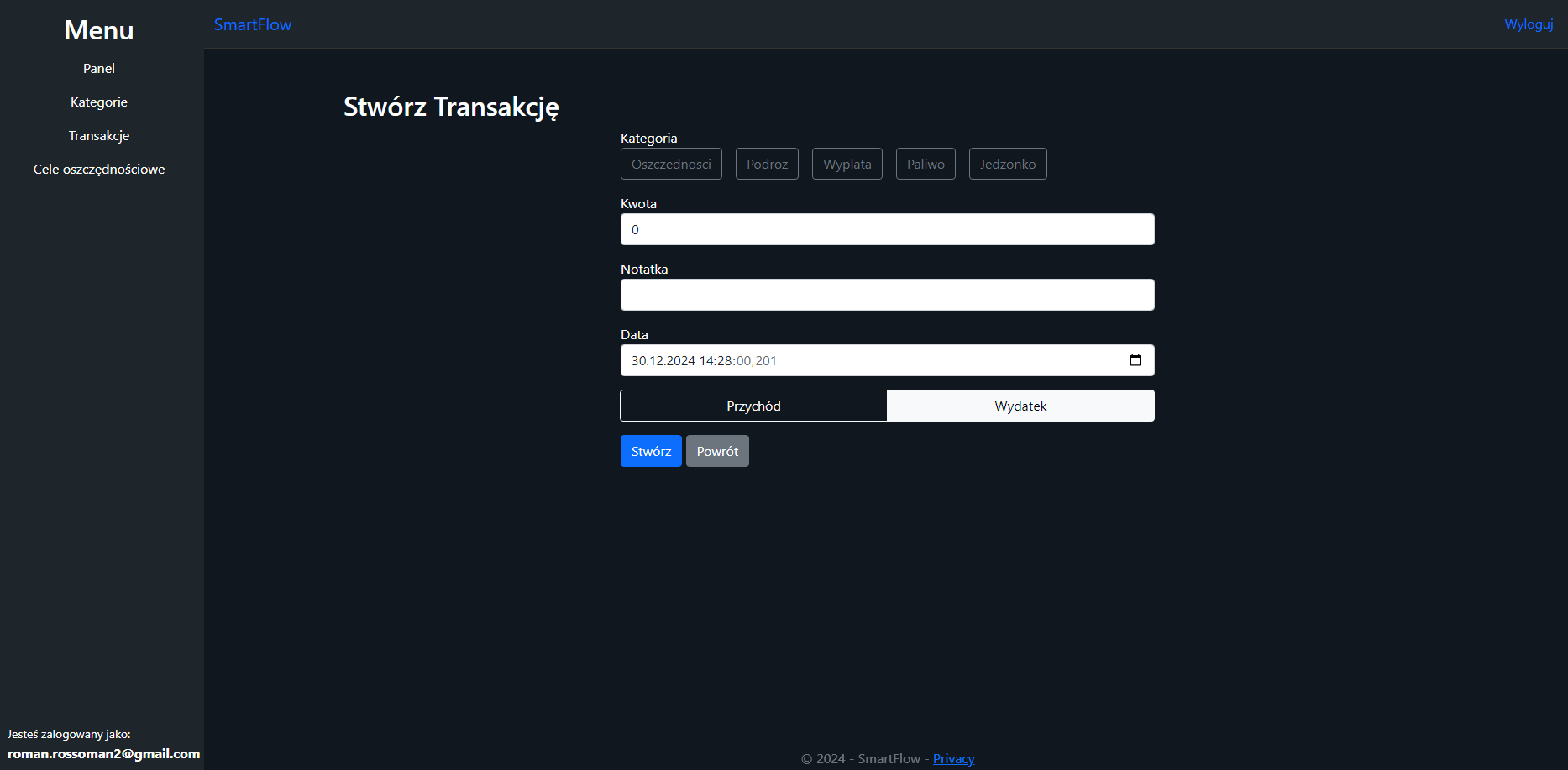
**Edytowanie kategorii**  
  


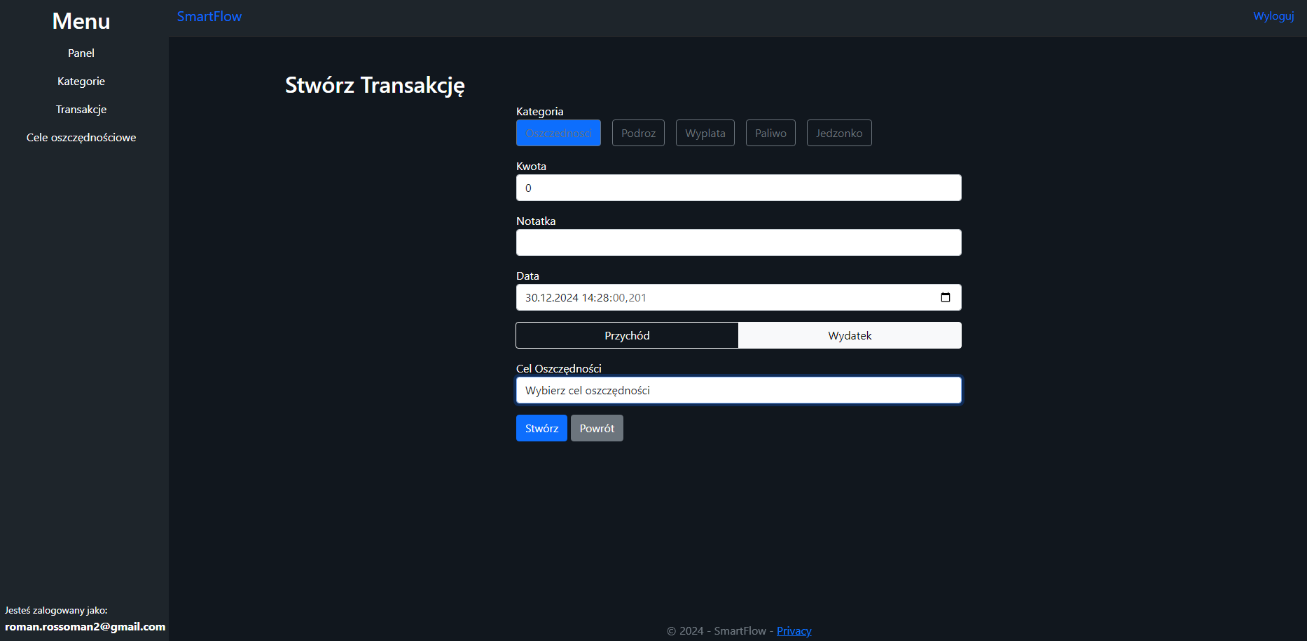
**Usuwanie kategorii**

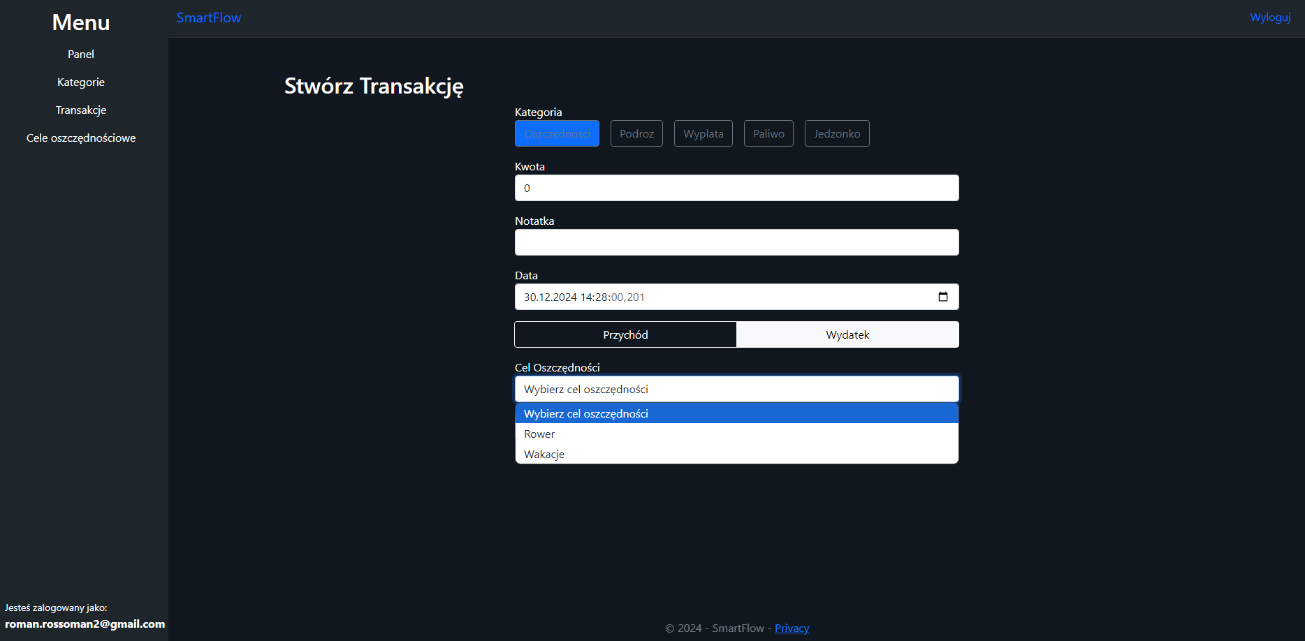


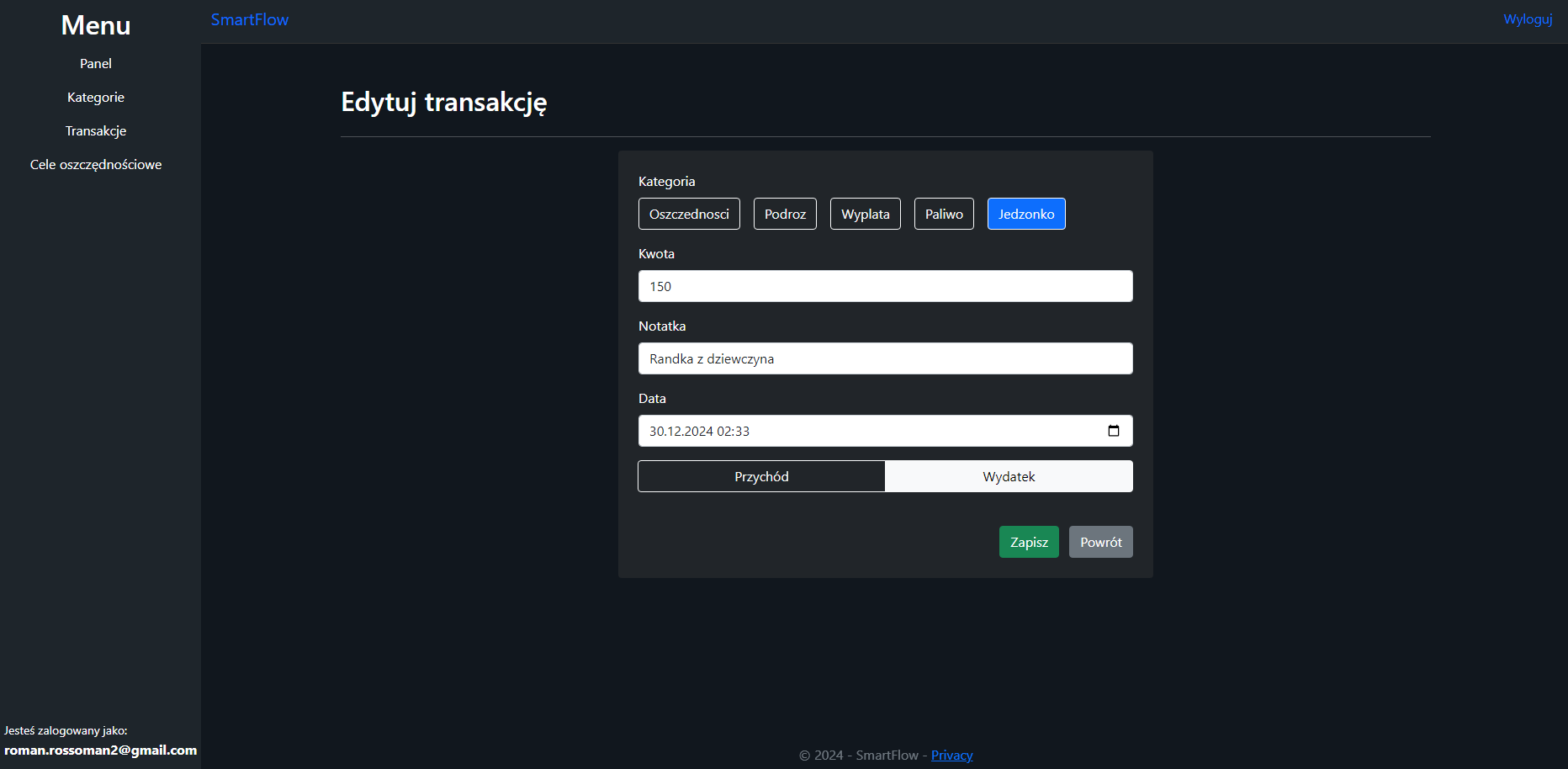
**Strona transakcji**  
  


**Dodawanie nowej transakcji** (kategorie do wyboru są tylko te, które wcześniej stworzyliśmy)

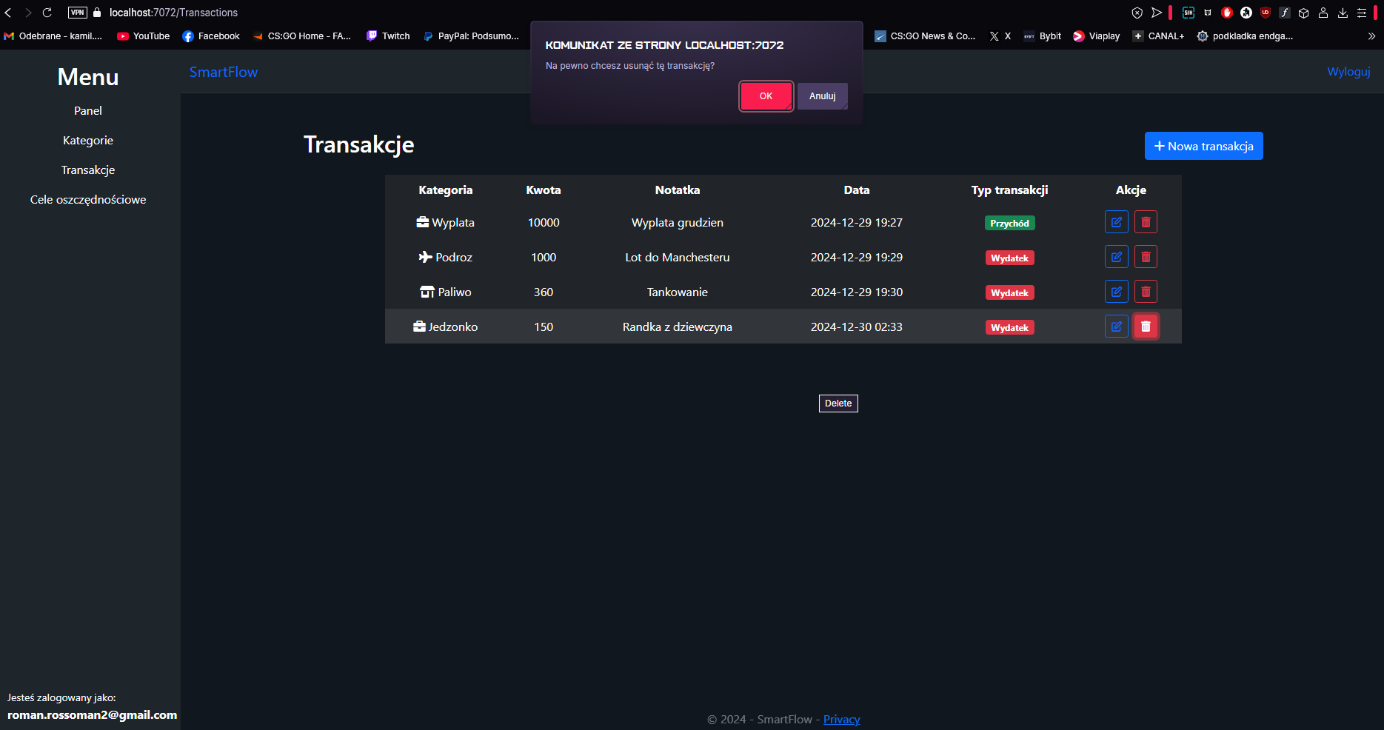


**Dodawanie transakcji oszczędnościowej wykorzystując kategorię ”Oszczednosci”**  
  


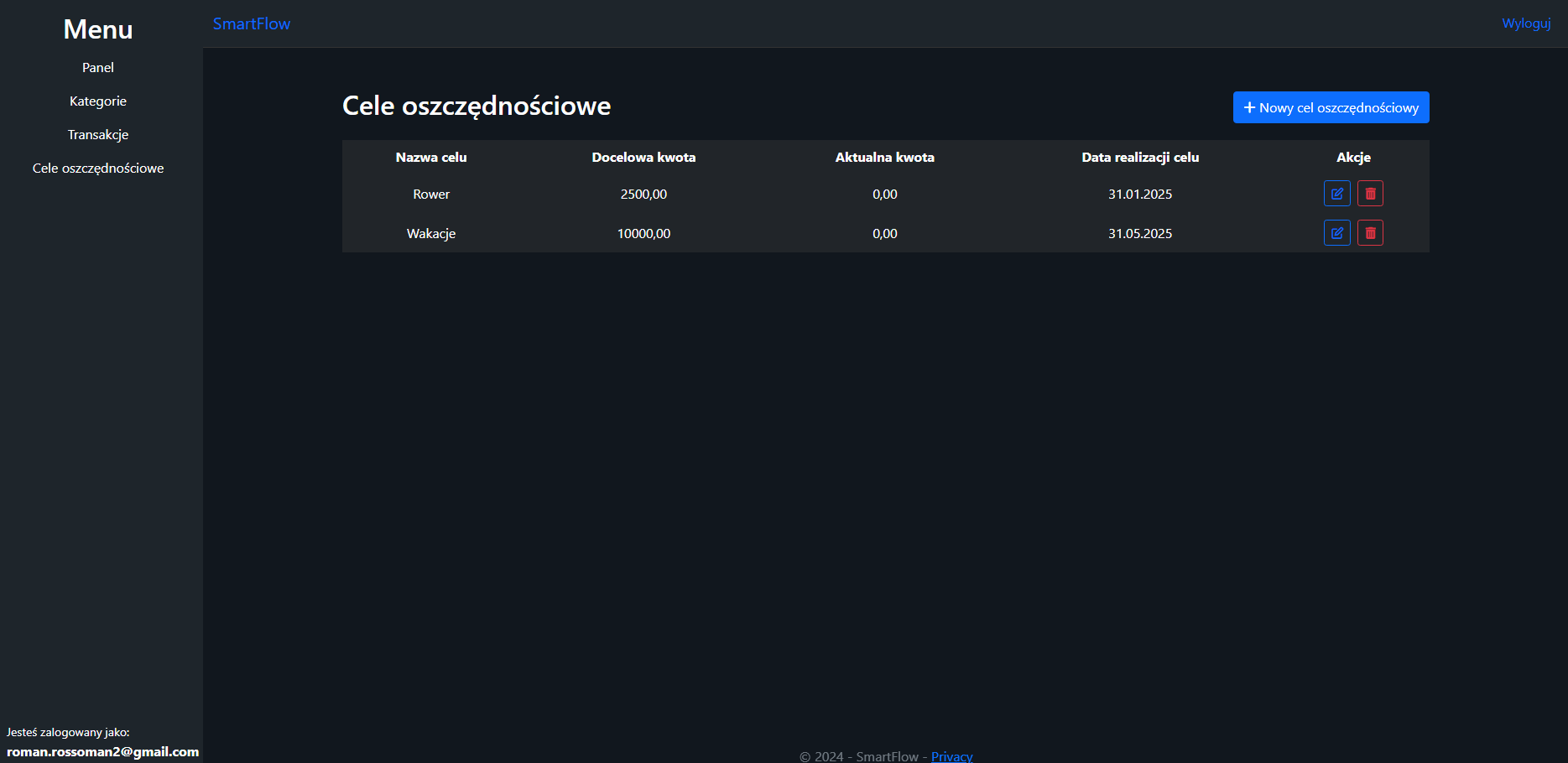


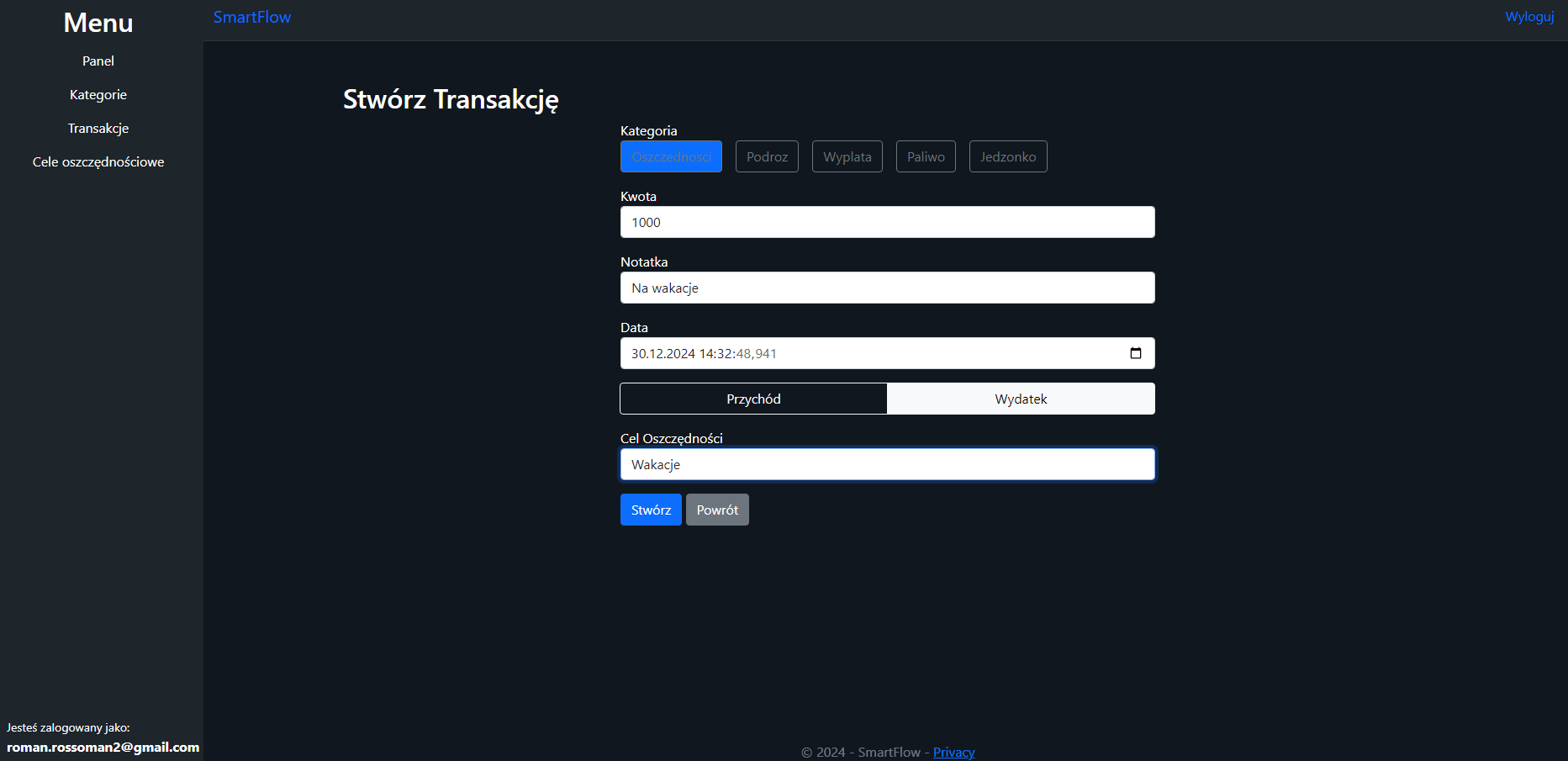
**Edytowanie transakcji**  
  


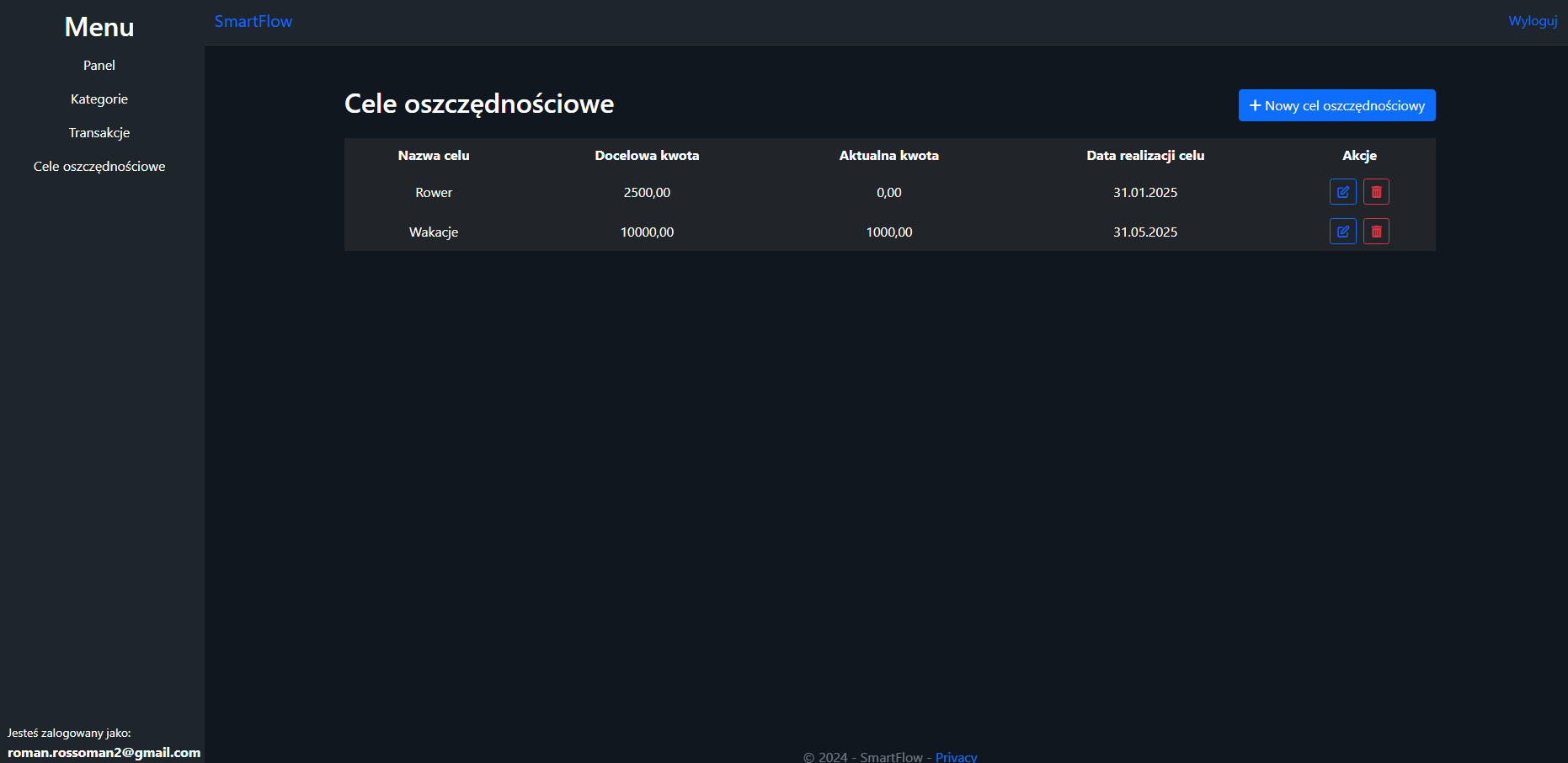
**Usuwanie transakcji**

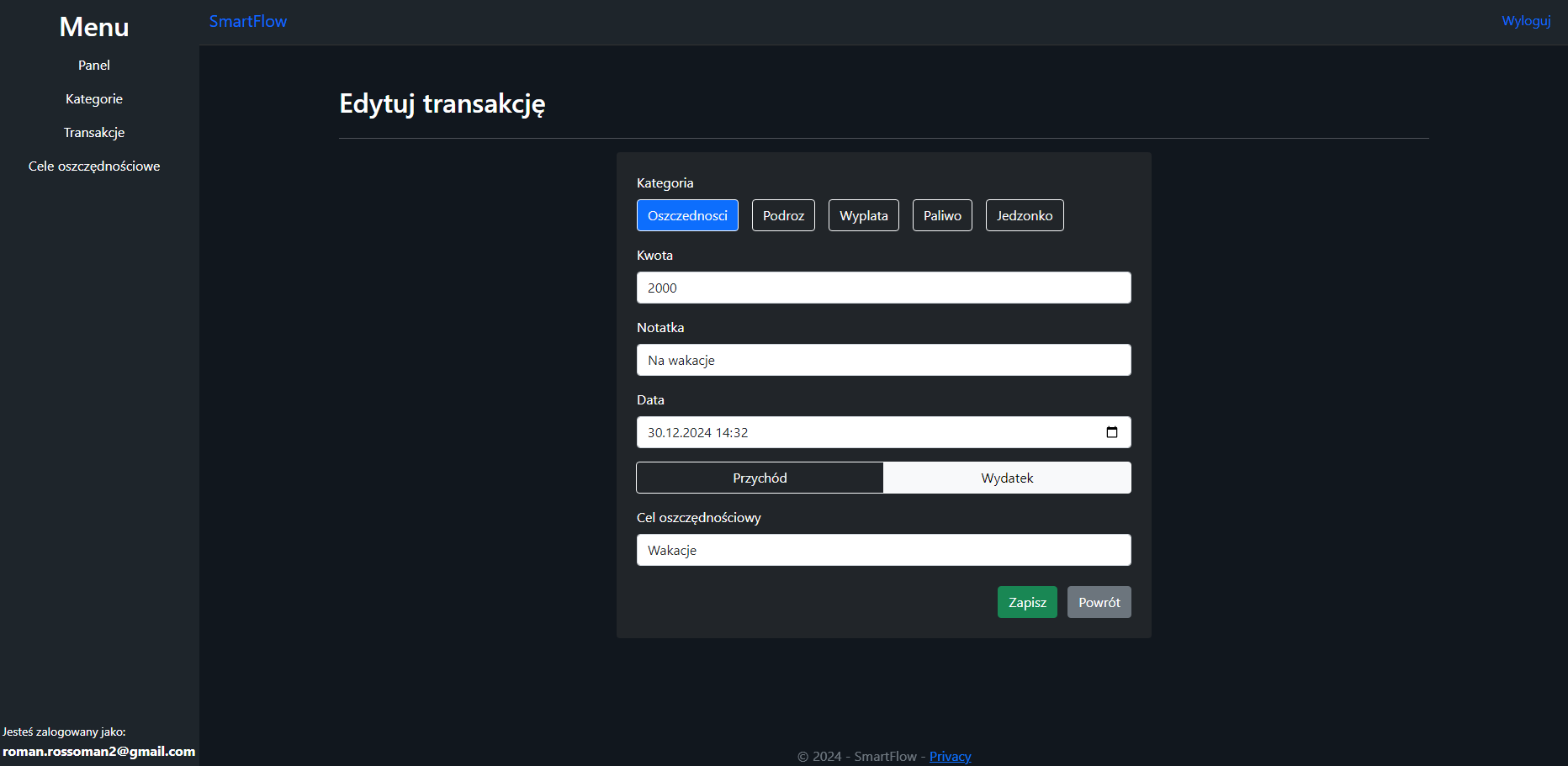


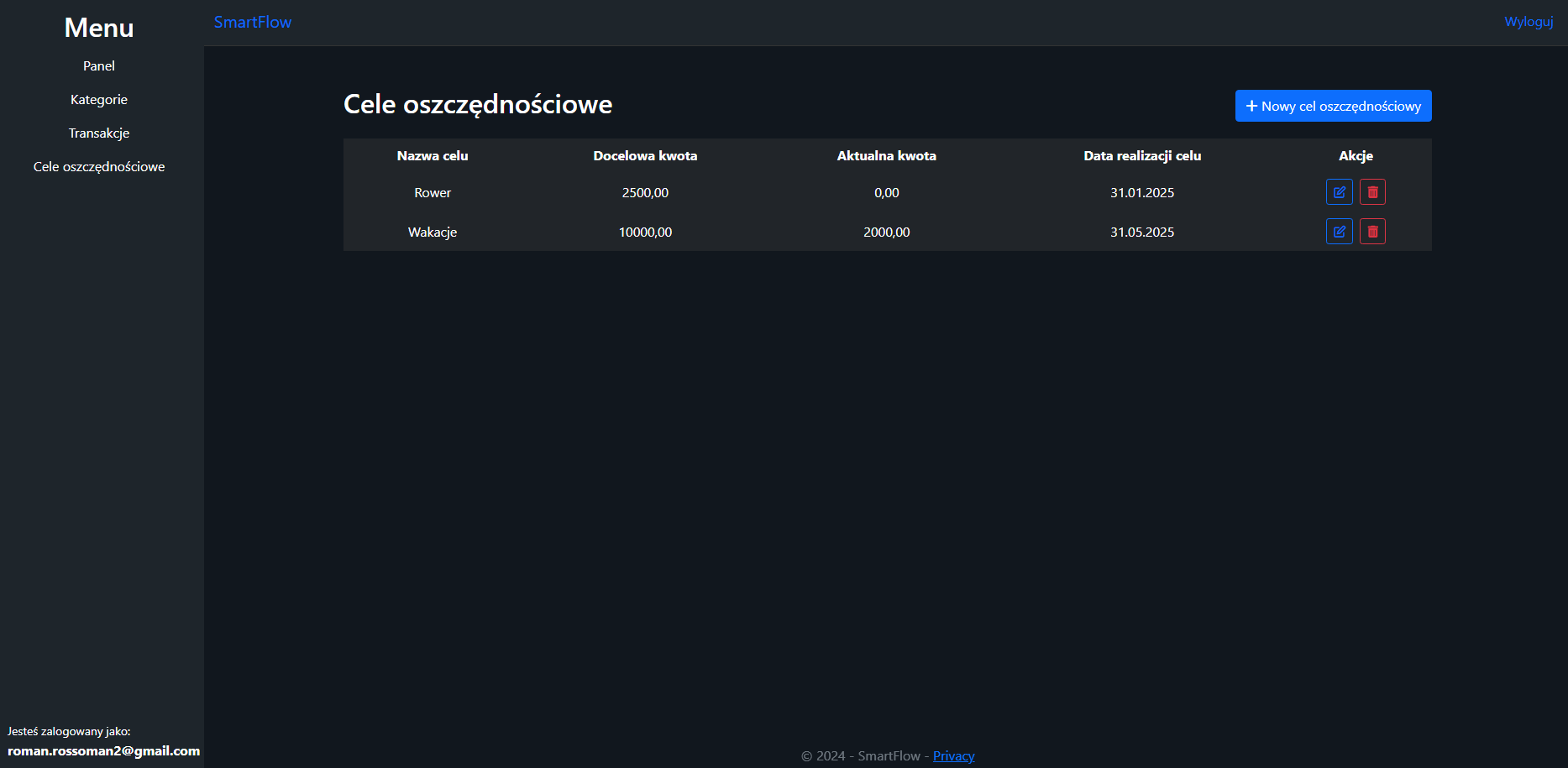
Strona celów oszczędnościowych



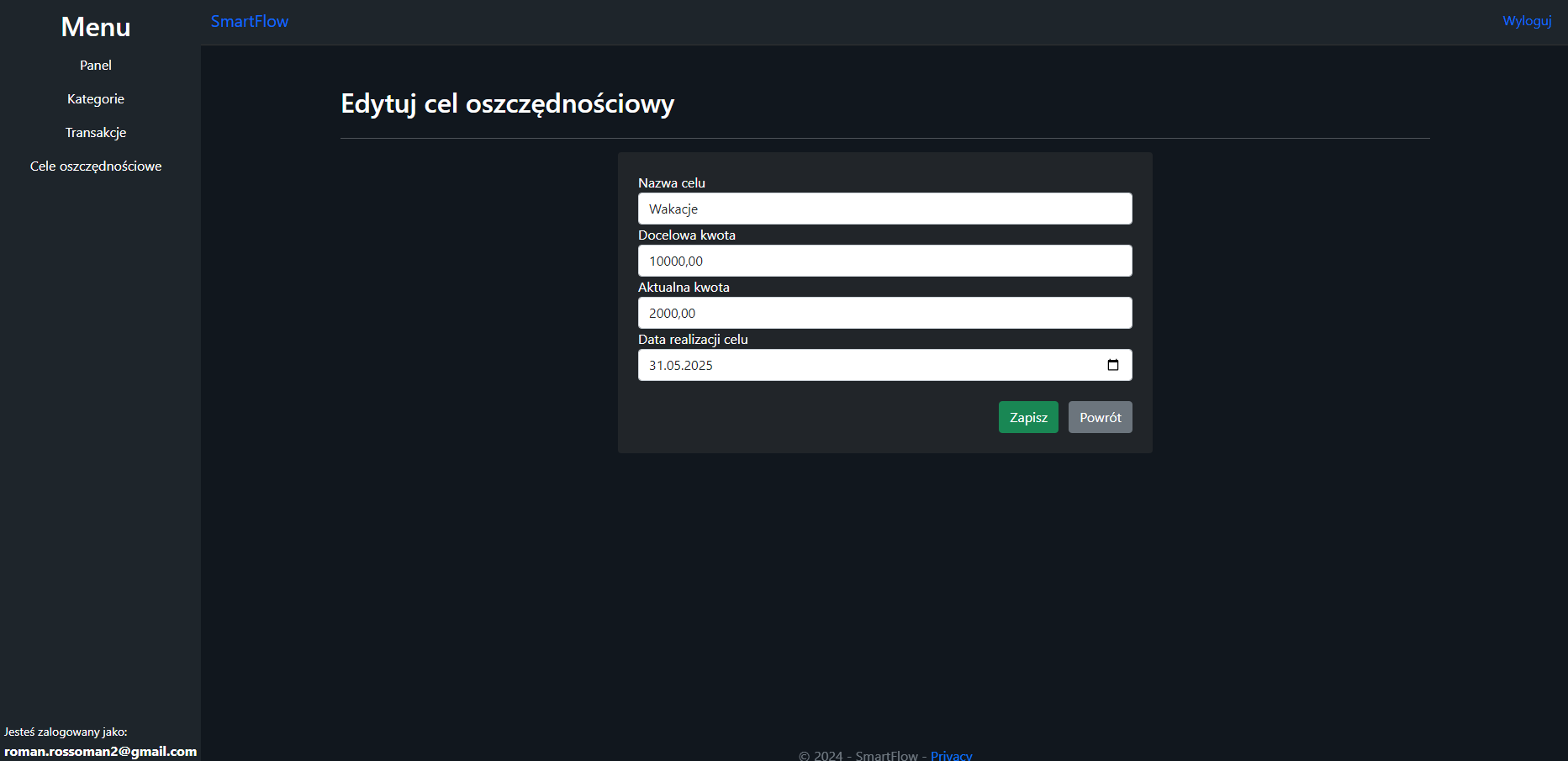
Dodanie nowej transakcji z kategorią “Oszczednosci” oraz wyborem celu oszczędnościowego “Wakacje”  
  




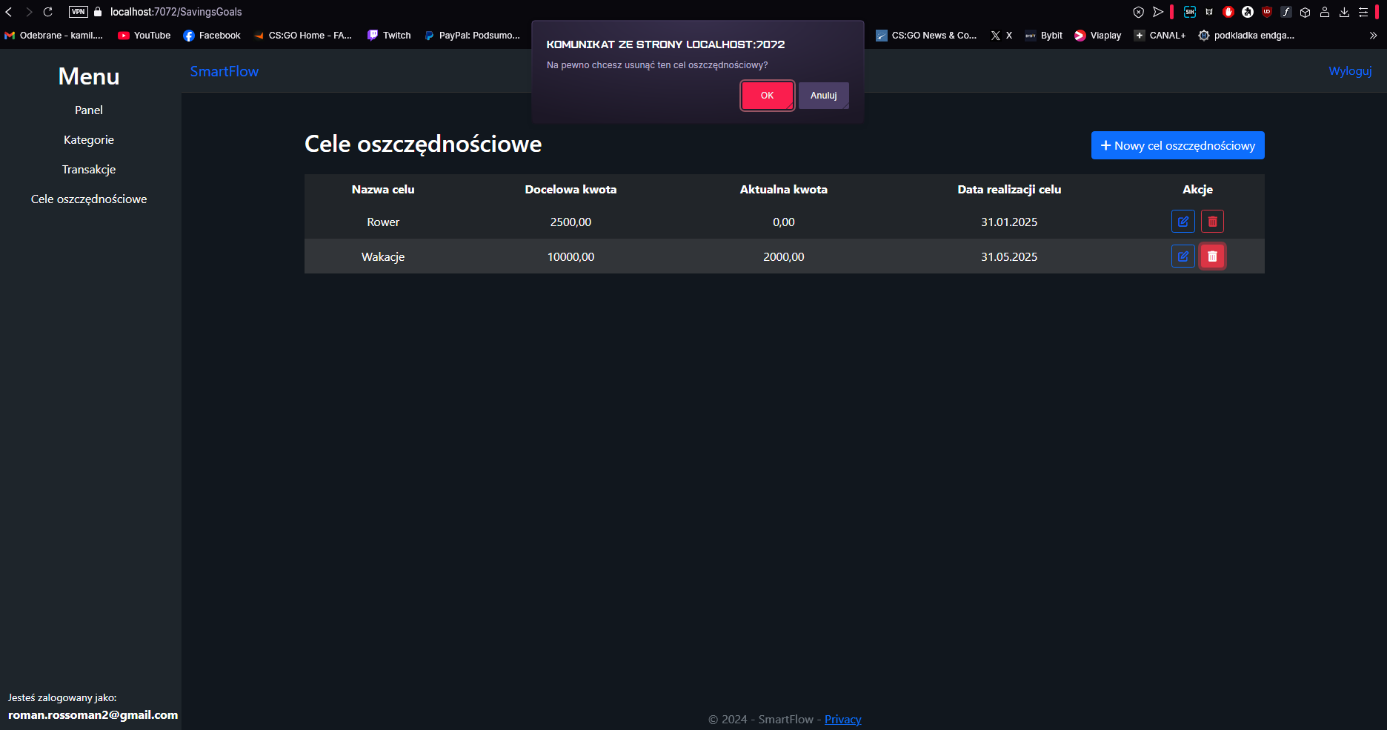
Edycja transakcji oszczędnościowej dla celu “Wakacje”  
  




Edytowanie celu oszczędnościowego



Usuwanie celu oszczędnościowego



Widok panelu

