



Projekt

Programowanie aplikacji internetowych w JEE

**Opis**

**Imię I nazwisko: Radosław Kłonica**

**Grupa laboratoryjna: 2.2**

# Zależności

Do projektu dołączono następujące zależności:

- Spring Data JPA

- Spring Web

- Spring Security

- Spring Validation

- Lombok

- Thymeleaf

- H2 Database  
- Thymeleaf Extras Spring Security6

- Jackson Datatype jsr310

# Opis

Aplikacja jest prostym systemem CRM, który umożliwia pełną obsługę operacji CRUD (tworzenie, odczyt, aktualizacja, usuwanie) dla klientów i pracowników (użytkowników aplikacji). Wszystkie funkcje są dostępne po zalogowaniu się do systemu, z wykorzystaniem Spring Security i modelu użytkownika rozszerzającego wbudowany użytkownik Springa. Dodatkowo, dostęp do poszczególnych działań jest ograniczany przez przypisane role użytkowników (np. użytkownik zwykły może tylko dodawać klientów, ale nie innych użytkowników).

Frontend aplikacji został stworzony z użyciem Thymeleaf, a także rozszerzenia Thymeleaf Extras Spring Security6. Dzięki temu rozszerzeniu można korzystać z nowych atrybutów w tagach HTML, takich jak „sec:authentication”. W tym przypadku, przykład użycia to sprawdzanie uprawnień użytkownika do wykonania danej akcji, np. „sec:authorize="hasRole('ROLE\_ADMIN')"” – co oznacza, że dana akcja jest dostępna tylko dla administratorów.

Dane użytkowników i klientów są przechowywane w bazie H2. Klienci są początkowo dodawani przy pomocy pliku import.sql, który zawiera 1000 losowych rekordów wygenerowanych za pomocą narzędzia Mockaroo (https://mockaroo.com). Natomiast użytkownicy są tworzeni na początku aplikacji, a odpowiedzialny za to jest plik Bootstrap.java, który zawiera Listener do tworzenia kont po uruchomieniu aplikacji. Dane dla kont użytkowników:

Administrator: login – admin, hasło – admin123

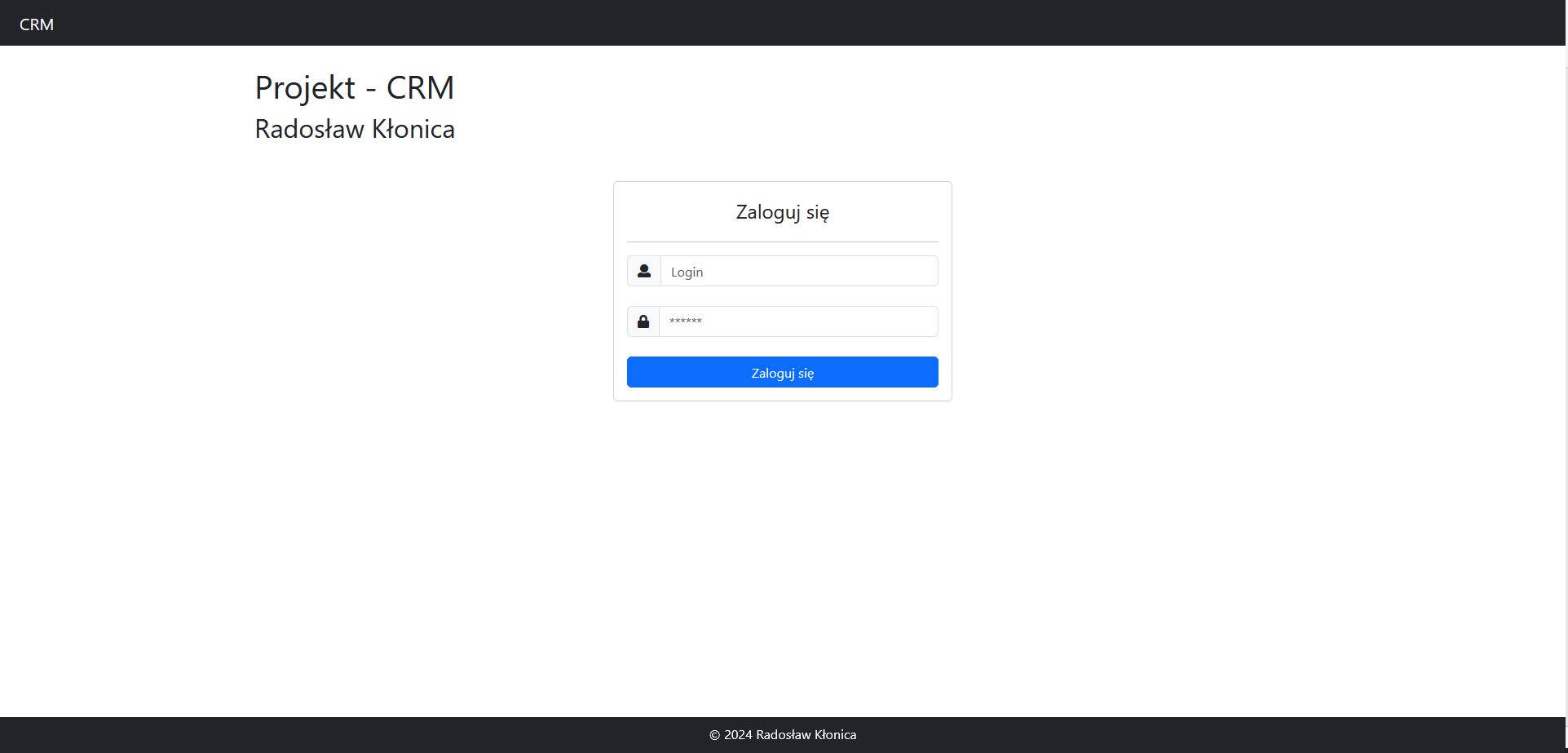
Użytkownik: login – user, hasło – user123

Ograniczony użytkownik: login – lowUser, hasło – lowuser123

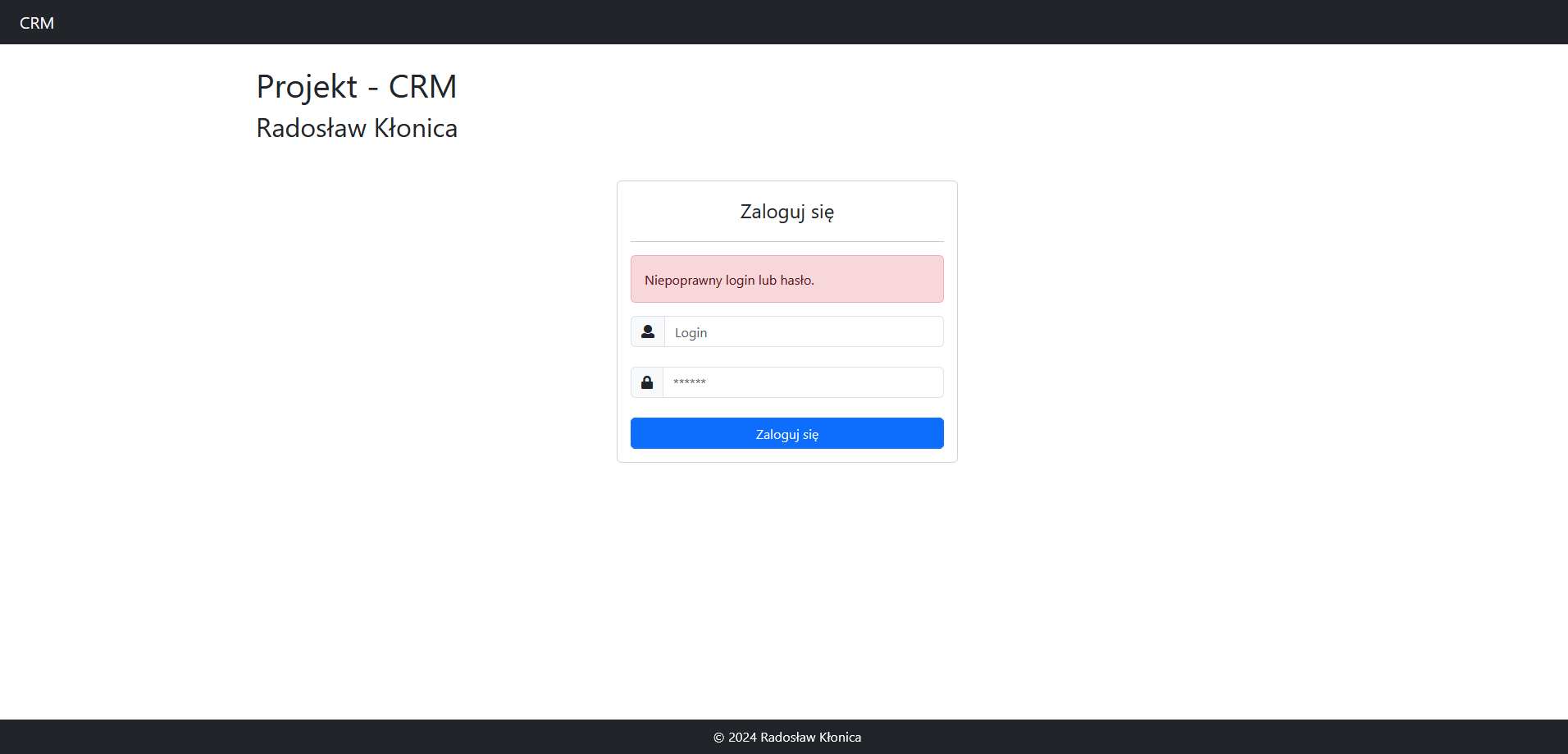
Zarządzanie użytkownikami umożliwia wykonywanie operacji CRUD oraz dezaktywację kont bez ich usuwania. W tym celu nadpisano metodę isEnabled() w dziedziczonym modelu użytkownika.

W aplikacji użyto również zależności Jackson Datatype jsr310, która upraszcza renderowanie struktur, w tym zawiera m.in. objectMapper.

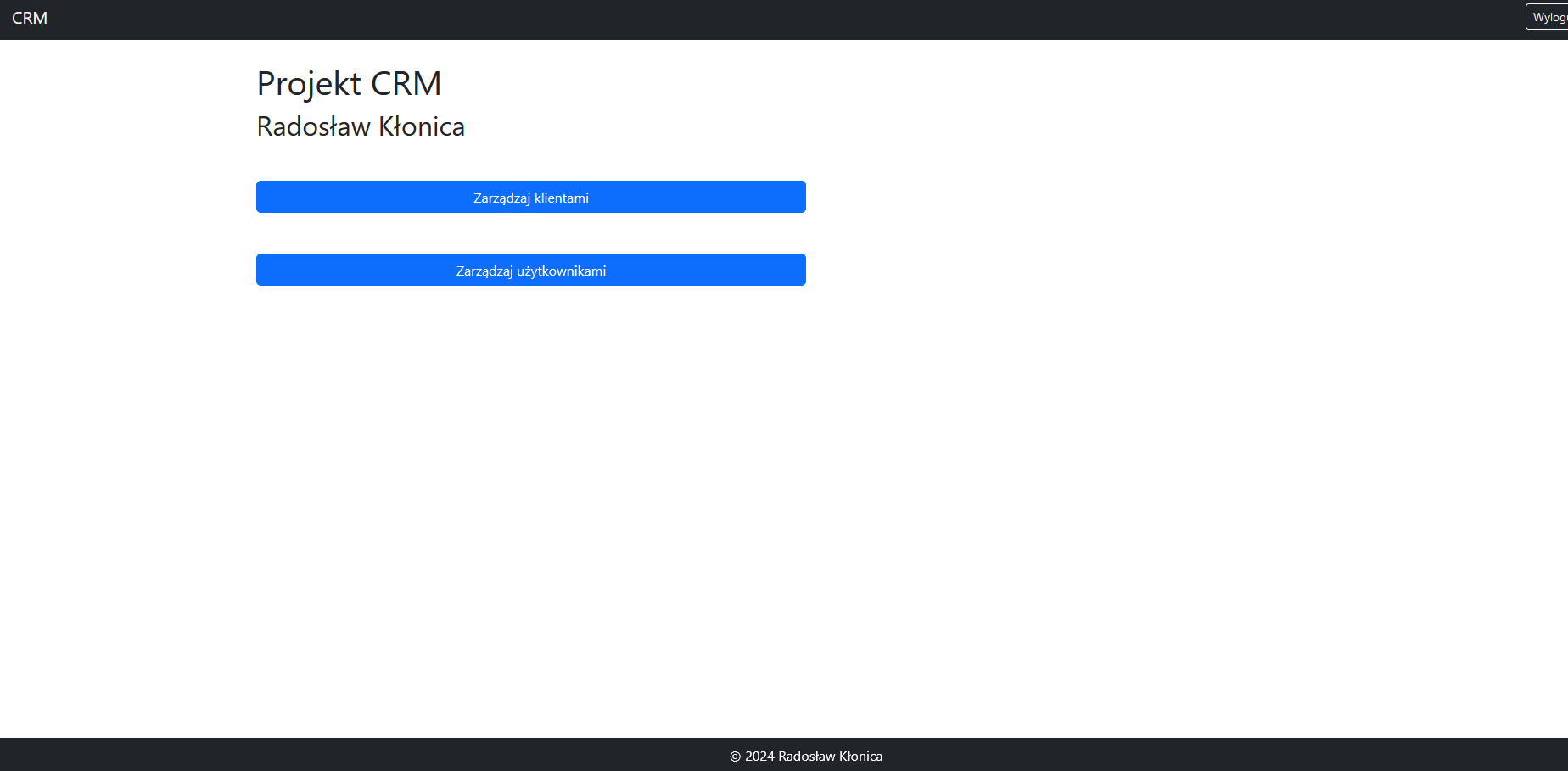
# Działanie aplikacji



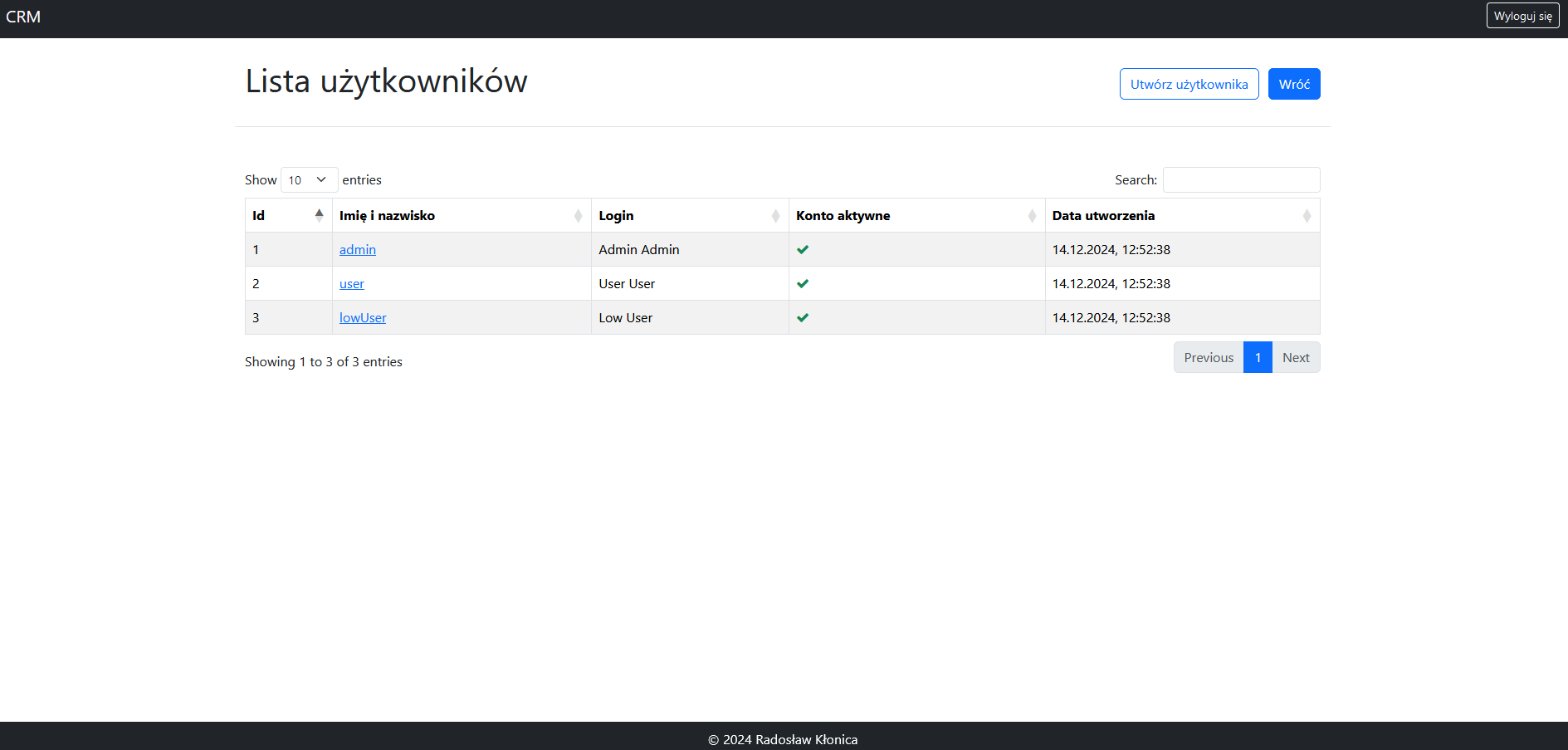
Rysunek 1 Logowanie



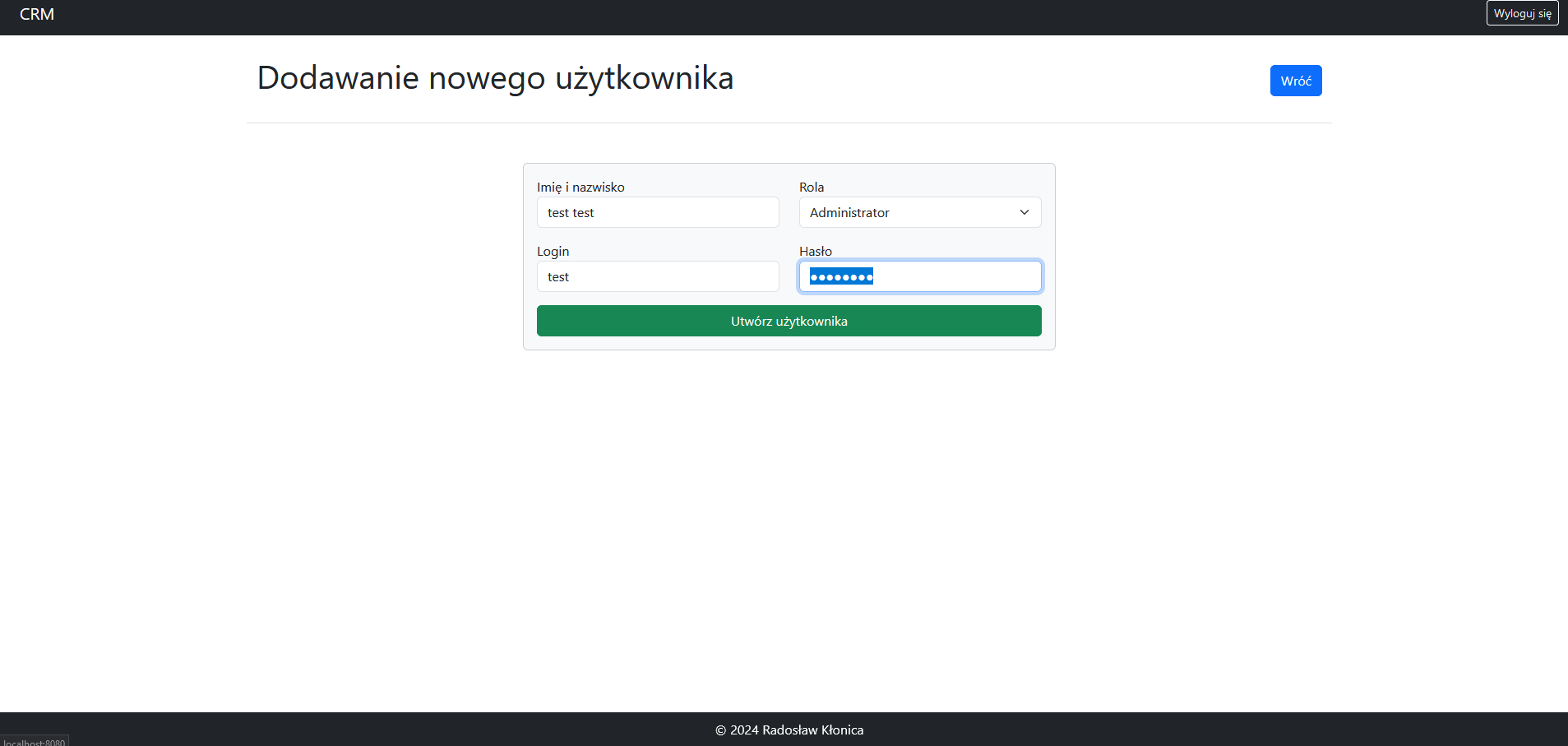
Rysunek 2 Walidacja - błędne dane logowania



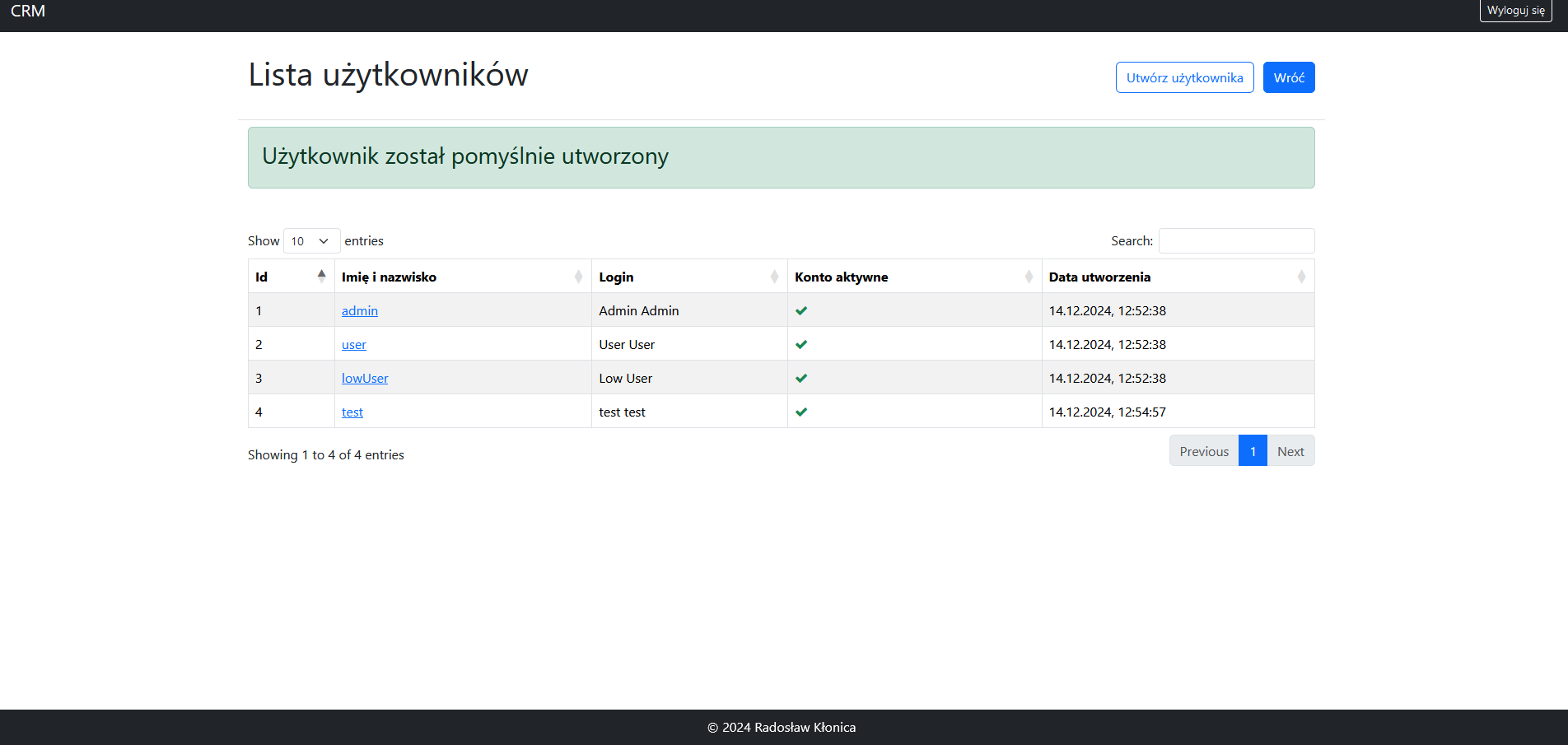
Rysunek 3 Panel - widok administratora



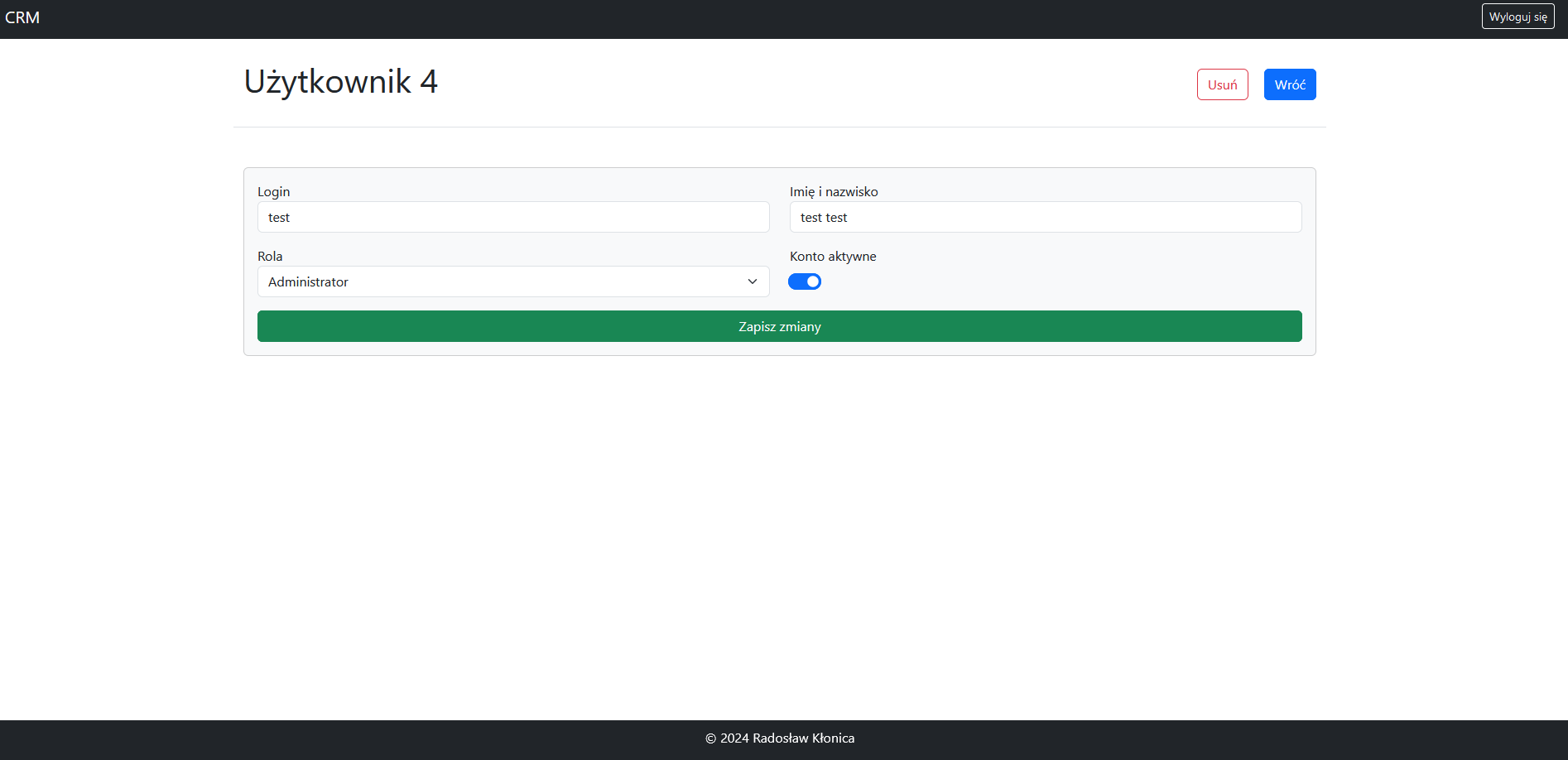
Rysunek 4 Lista użytkowników



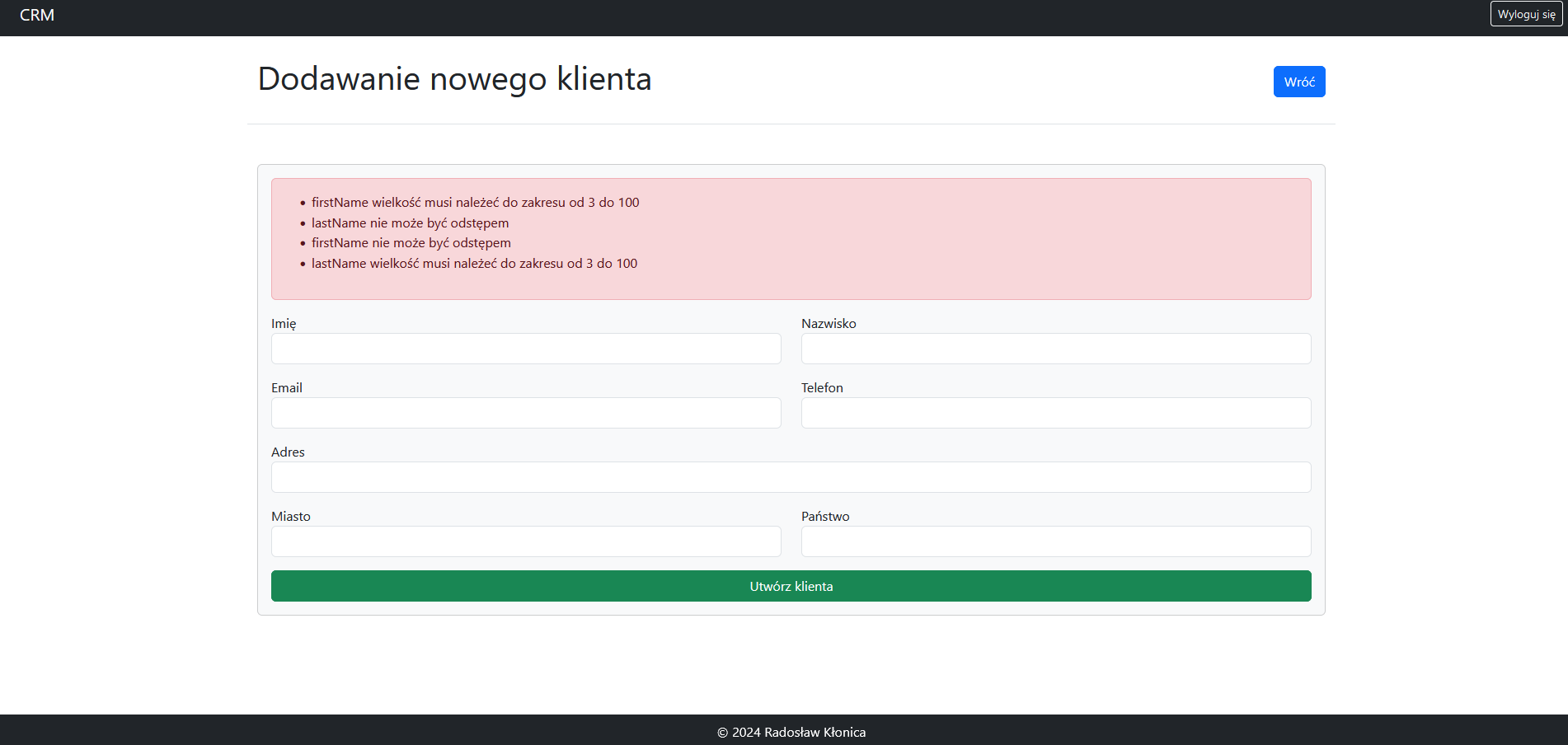
Rysunek 5 Dodanie nowego użytkownika



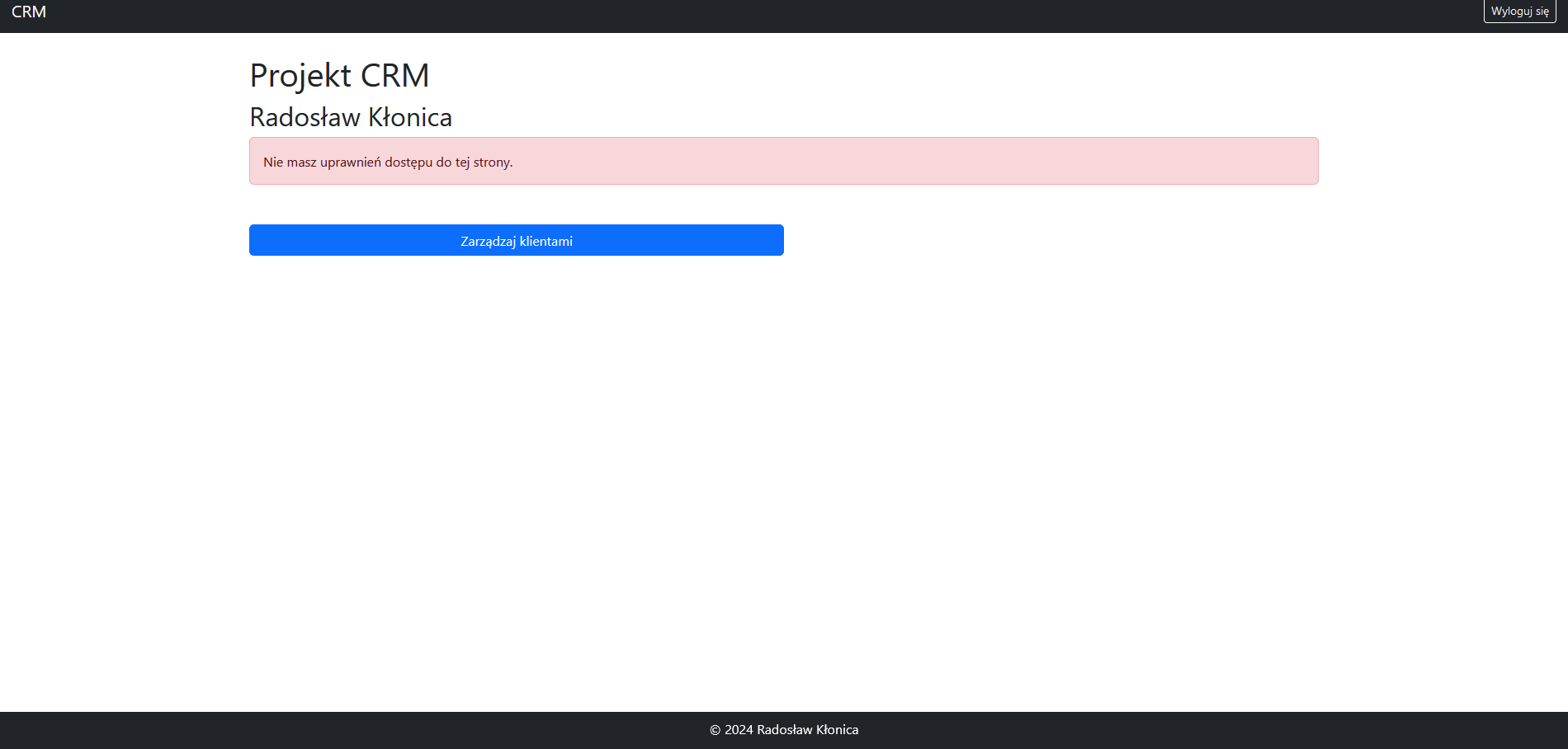
Rysunek 6 Informacja zwrotna po dodaniu użytkownika



Rysunek 7 Strona edycji użytkownika (możliwość wyłączenia konta użytkownika)



Rysunek 8 Walidacja - dodanie nowego klienta



Rysunek 9 Walidacja (konto użytkownika) - po próbie wejścia w zarządzanie użytkownikami poprzez odres strony