

Wymagania

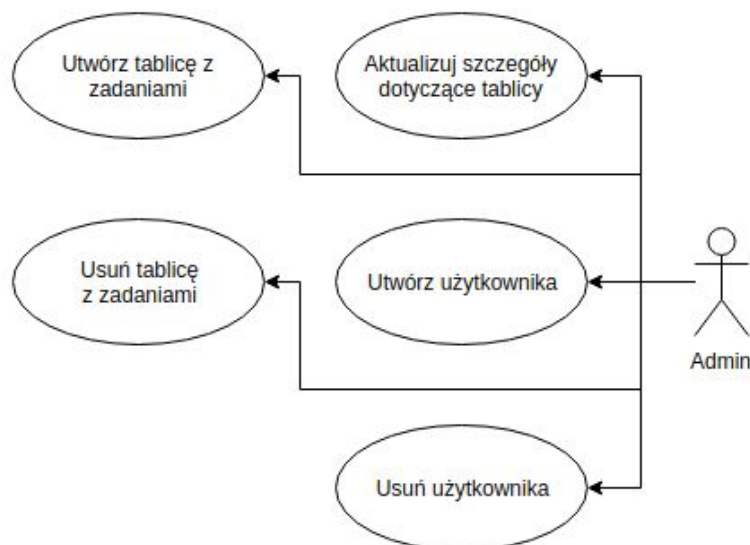
Wymagania użytkownika

1. Zarządzanie tablicami z zadaniami.
 - a. Przeglądanie istniejących tablic.
2. Zarządzanie zadaniami w tablicy.
 - a. Przeglądanie istniejących zadań.
 - b. Tworzenie nowych zadań.
 - c. Aktualizacja statusu zadania.
 - d. Przypisywanie użytkowników do zadań.
 - e. Dodawanie komentarzy do dyskusji pod zadaniem.
3. Zarządzanie kontem użytkownika.
 - a. Aktualizacja danych użytkownika.
 - b. Zmiana hasła dostępu do serwisu.

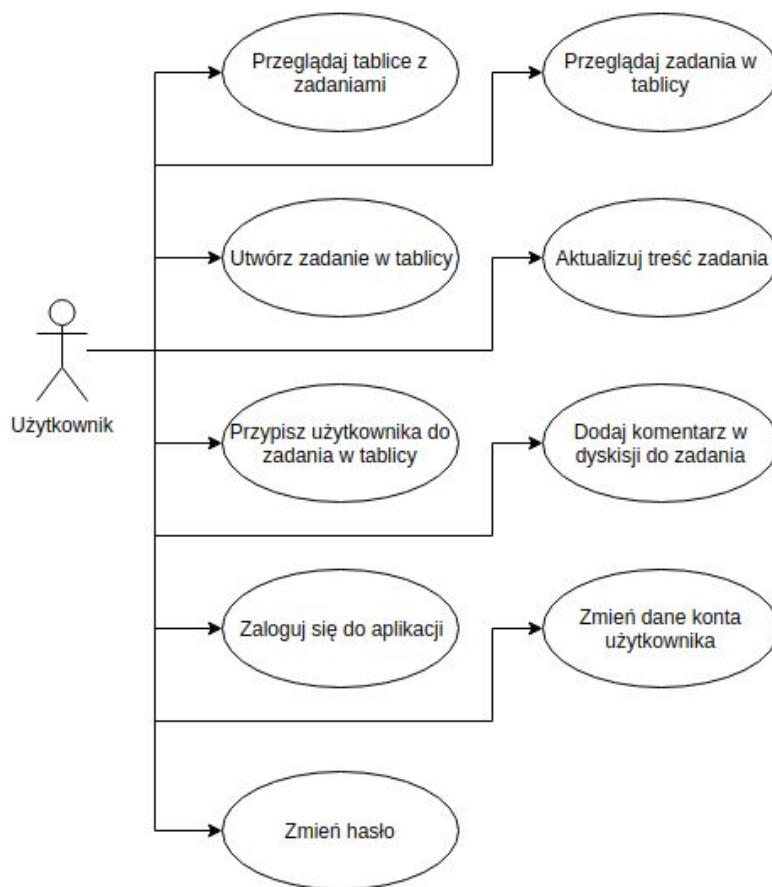
Wymagania administracyjne

1. Zarządzanie użytkownikami.
 - a. Tworzenie użytkowników.
 - b. Usuwanie użytkowników.
2. Zarządzanie tablicami z zadaniami.
 - a. Tworzenie nowych tablic.
 - b. Modyfikowanie istniejących tablic.
 - c. Usuwanie istniejących tablic.

Diagramy przypadków użycia



Rys. 1. Diagram przypadków użycia dla administratora.

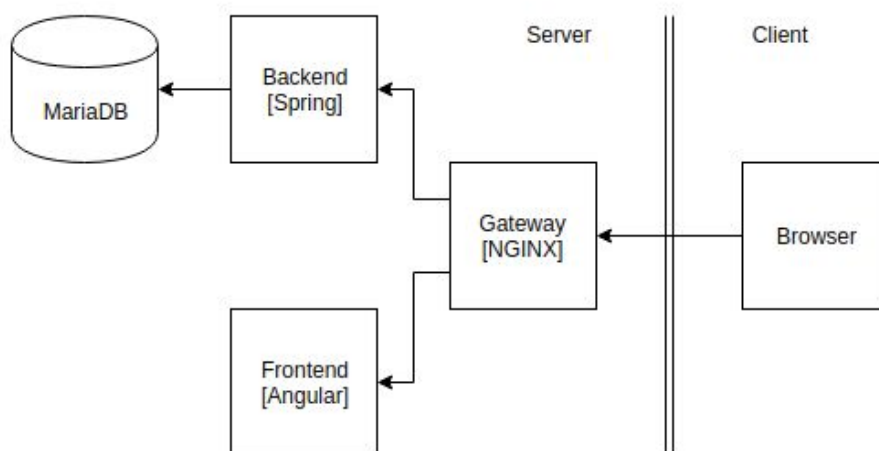


Rys. 2. Diagram przypadków użycia dla zwykłego użytkownika.

Architektura aplikacji

Głównymi elementami aplikacji są dwa serwisy HTTP.

1. Pierwszy współpracując z bazą danych serwuje użytkownikom dane w formie RESTowego API.
3. Drugi serwuje statyczne pliki aplikacji frontendowej, która będzie uruchamiana w przeglądarce użytkownika.



Rys. 3. Schemat architektury systemu.

Oba serwisy są zamknięte za gatewayem NGINX, realizującym routing do tych aplikacji.

1. Jeśli URL wysłany przez klienta rozpoczyna się od /api, żądanie HTTP jest przekazywane do backendu.
2. W przeciwnym razie żądanie jest przekazywane do frontendu.

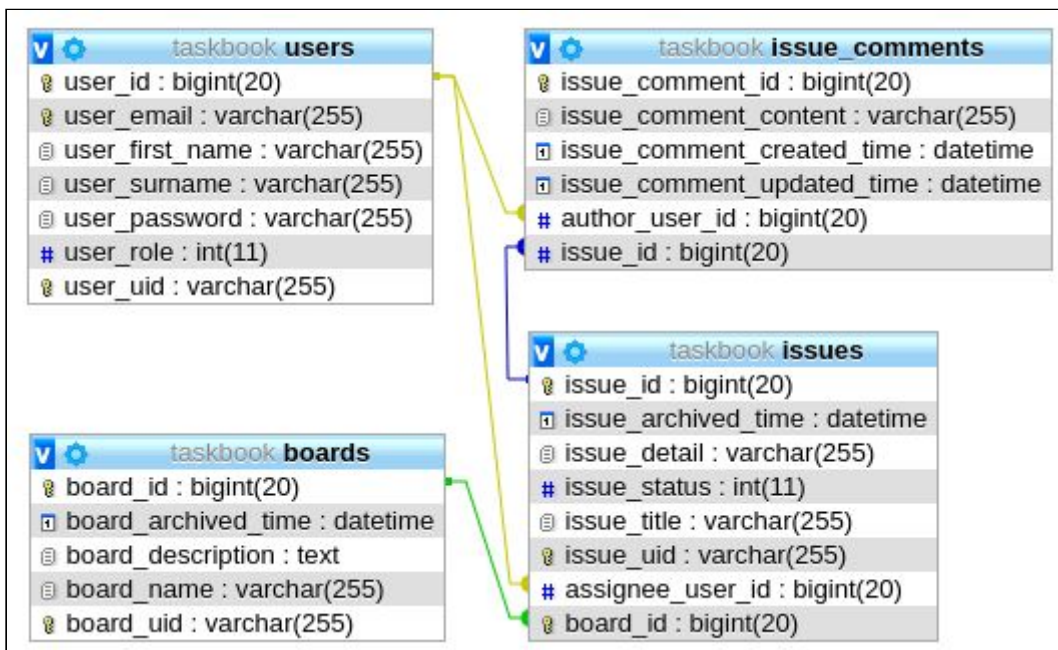
Klientem może być dowolna przeglądarka internetowa.

Aplikacja backendowa została zaimplementowana we frameworku [Spring](#). Aplikacja frontendowa została zaimplementowana we frameworku [Angular](#). Oba frameworki są dojrzałe i nadają się do implementacji oprogramowania biznesowego.

Do wdrożenia aplikacji zastosowano konteneryzację z wykorzystaniem technologii [Docker](#).

Baza danych

Aplikacja wykorzystuje bazę danych [MariaDB](#), z uwagi na swoją prostotę, wsparcie, kwestie licencyjne oraz kompatybilność z popularną bazą [MySQL](#).



Rys. 4. Model bazy danych wygenerowany został przy pomocy narzędzia [phpMyAdmin](#).

API RESTowe

/api/boards

- GET
- POST
- GET /{uid}
- PUT /{uid}
- DELETE /{uid}

/api/boards/{board}/issues

- GET
- POST
- GET /{uid}
- PUT /{uid}
- DELETE /{uid}

/api/boards/{board}/issues/{issue}/comments

- GET
- POST
- GET /{uid}

/api/issues

- GET

/api/users

- GET
- POST
- GET /{uid}
- PUT /{uid}
- DELETE /{uid}

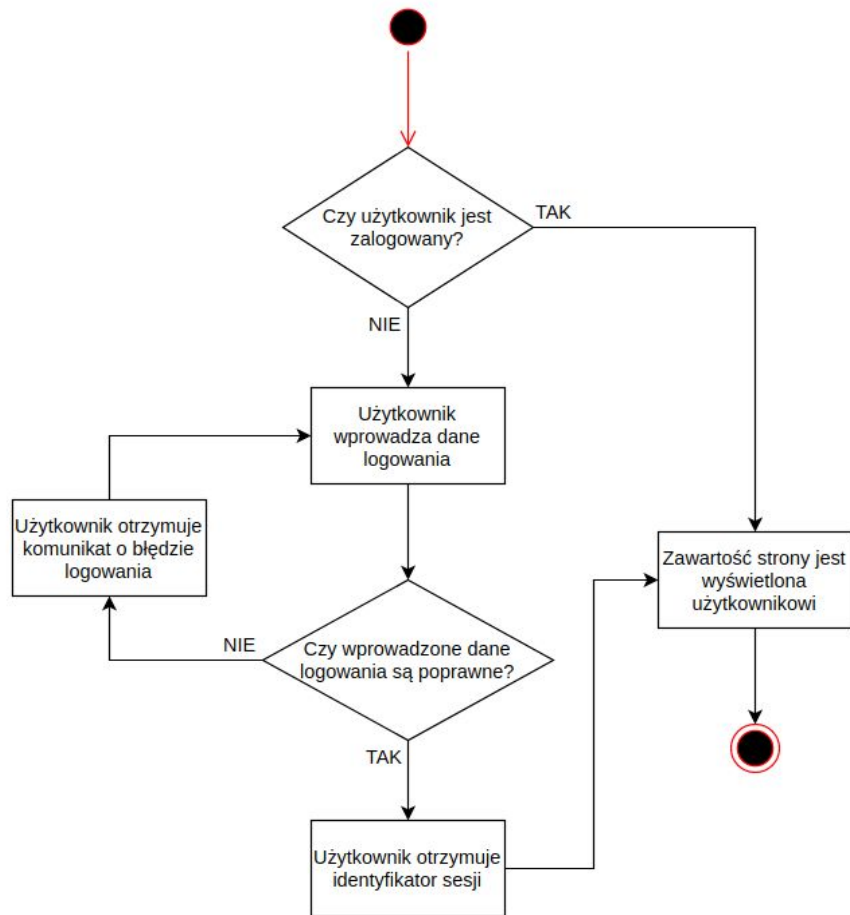
/api/account

- GET
- PUT
- PUT /password

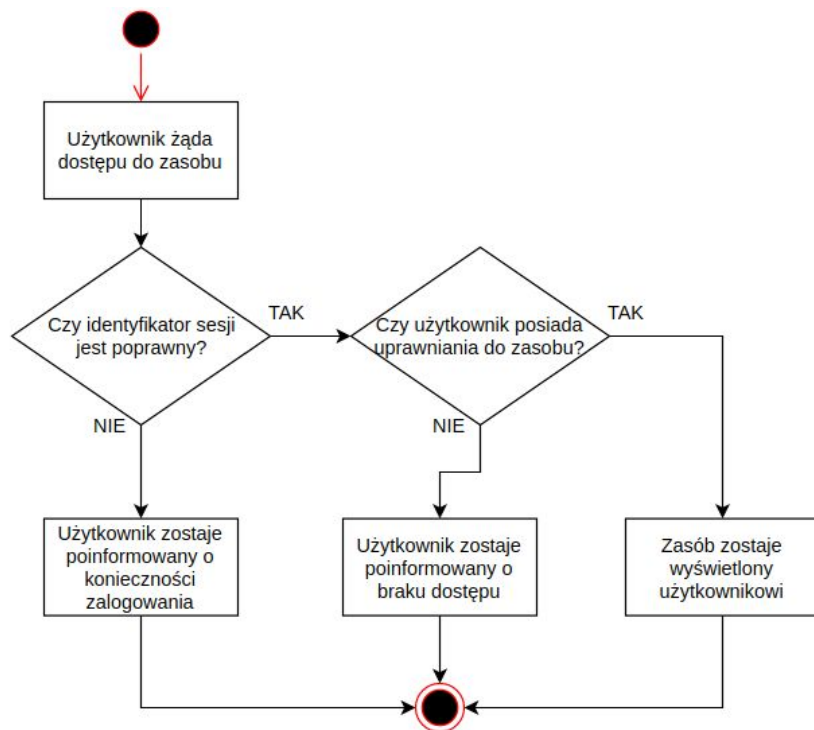
/api/login

- POST

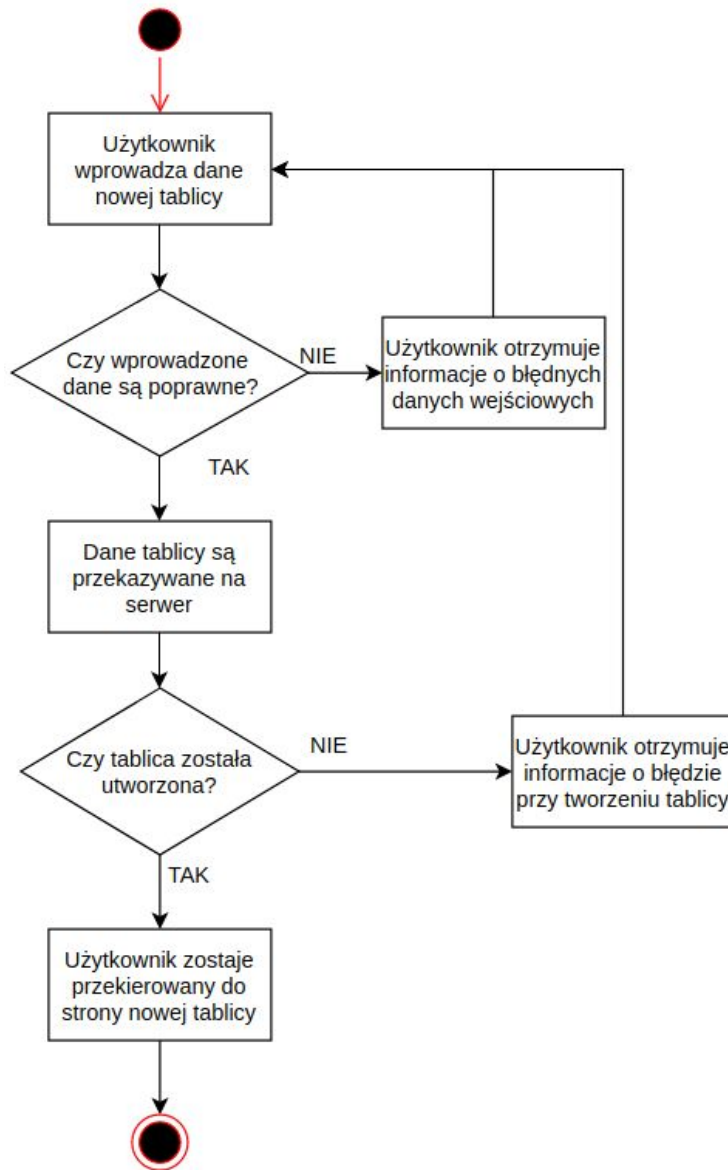
Podstawowe diagramy aktywności



Rys. 5. Diagram aktywności prezentujący logowanie użytkownika do systemu.



Rys. 6. Diagram aktywności prezentujący proces autoryzacji.

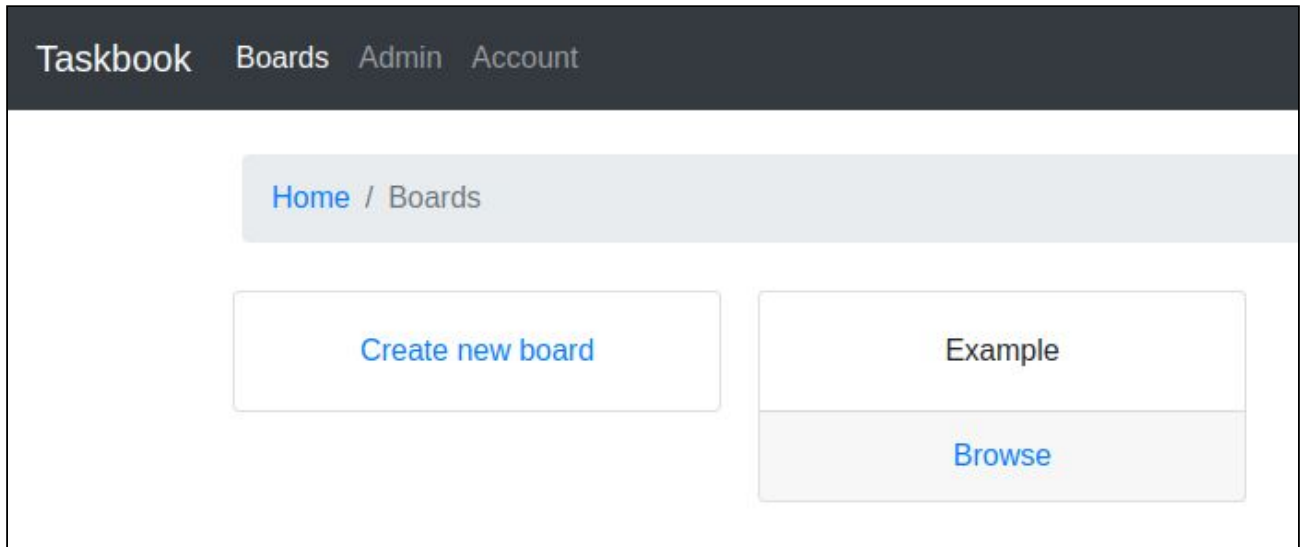


Rys. 7. Diagram przedstawiający tworzenie nowej tablicy.

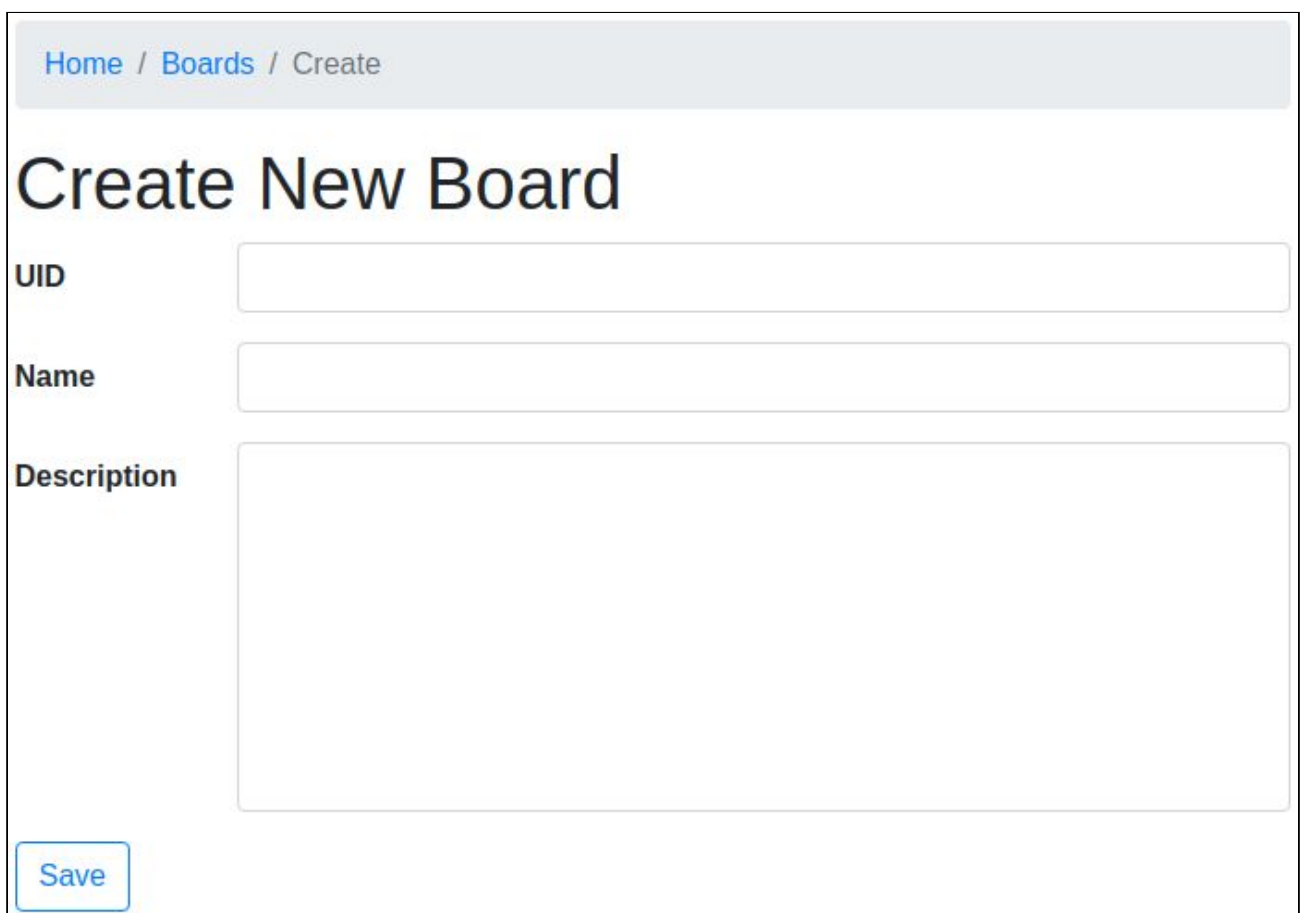
Procedura tworzenia dowolnego innego zasobu jest analogiczna.

Implementacja aplikacji

Poniżej zaprezentowano zrzuty ekranu prezentujące zaimplementowane moduły aplikacji.



Panel przeglądania tablic z zadaniami.

The screenshot displays the 'Create New Board' form. At the top, there is a breadcrumb trail 'Home / Boards / Create'. The title 'Create New Board' is prominently displayed. The form consists of three input fields: 'UID', 'Name', and 'Description'. The 'Description' field is a larger text area. A 'Save' button is located at the bottom left of the form.

Panel tworzenia nowej tablicy.

Home / Boards / EX / Issues

Issues Settings

« 1 »

Issue	Status	Assignee
[EX-1] First issue	in_progress	Admin

Panel przeglądania zadań w tablicach.

Home / Boards / EX / Issues / Create

Issues Settings

Title

Description

Tworzenie nowych zadań.

Home / Boards / EX / Settings

Issues Settings

UID EX

Name Example

Description Example board for demonstration purposes

Save

Panel aktualizacji danych na temat tablicy.

1 comments in total.

Add new comment

Comment

2020-01-08T21:19:47Z
I've commented this issue before.

~ Admin <admin@example.com>

« 1 »

Panel komentarzy pod zadaniem.

Home / Admin / Users

New User

« 1 »

#	E-mail	First name	Last name	Role	
1	admin@example.com	Admin		Admin ▼	Delete
2	user@example.com			User ▼	Delete

« 1 »

Panel zarządzania użytkownikami.

Home / Admin / Users / Create

E-mail address *

Role *

Password *

Confirm password *

First name

Last name

Create user

Panel tworzenia nowego użytkownika.

Account Settings

UID	45ca714e2d544a06b6ae3100c4bd5dff
E-mail address	admin@example.com
Role	admin
First name	Admin
Last name	

[Update settings](#)

Panel ustawień konta użytkownika.

Change password

Old password	<input type="password"/>
New password	<input type="password"/>
Confirm new password	<input type="password"/>

[Cancel](#) [Change password](#)

Panel zmiany hasła użytkownika.