

Dokumentacja OAuth2

Bezpieczeństwo systemów informatycznych

Bartłomiej Kręgielewski, Szymon Lepianka, Małgorzata Pinior

1. Cel

Celem projektu jest implementacja protokołu OAuth2, w postaci serwisu pozwalającego na logowanie się z wykorzystaniem funkcji SSO (Single Sign-On). W ramach projektu zostaną utworzone trzy aplikacje:

- Główna aplikacja implementująca OAuth2, pozwalająca na rejestrację użytkownika, przechowywująca jego dane oraz pozwalająca na udostępnianie jego danych dla innych aplikacji zarejestrowanych w systemie oraz pozwalająca na tzw. Single Sign-On.
- Dwie proste aplikacje klienckie, pozwalająca na przetestowanie działania.

Aby przećwiczyć aspekt integracji z zewnętrznym serwisem zostanie dodana opcja użycia własnego OAuth2 oraz Google Cloud API (gdzie nasze oprogramowanie stanowiłoby bramkę do google'a).

2. Terminologia

- 2.1. **Client** - aplikacja, która uzyskuje dostęp do chronionych zasobów w imieniu użytkownika (właściciela zasobów).
- 2.2. **Grant Type** - sposób autoryzacji w zależności od przypadku użycia.
- 2.3. **Resource server** - serwer nadzorujący chronione zasoby i odpowiadający na żądania dotyczące chronionych zasobów przy użyciu access token.
- 2.4. **Resource owner** - właściciel zasobów, autoryzuje aplikację, żeby posiadała dostęp do jego konta. Dostęp do aplikacji jest limitowany za pomocą scope.
- 2.5. **Scope** - permisja, pozwalająca na wykonanie danej akcji z chronionymi zasobami.

3. Architektura logiczna

- 3.1. **Users Management System**
- 3.2. **API** - wydawanie tokenów, przedłużania itp.
- 3.3. **Web Application** - rejestracja użytkownika w systemie i wyrażanie zgody na udostępnianie zasobów
- 3.4. **Token Generator System** - zarządzanie życiem tokenów:
 - generacja tokenu
 - odnawianie tokenu
 - odwoływanie tokenu
- 3.5. **Apps Management System** - system zarządzający aplikacjami korzystającymi z możliwości zalogowania się z użyciem OAuth2 service.

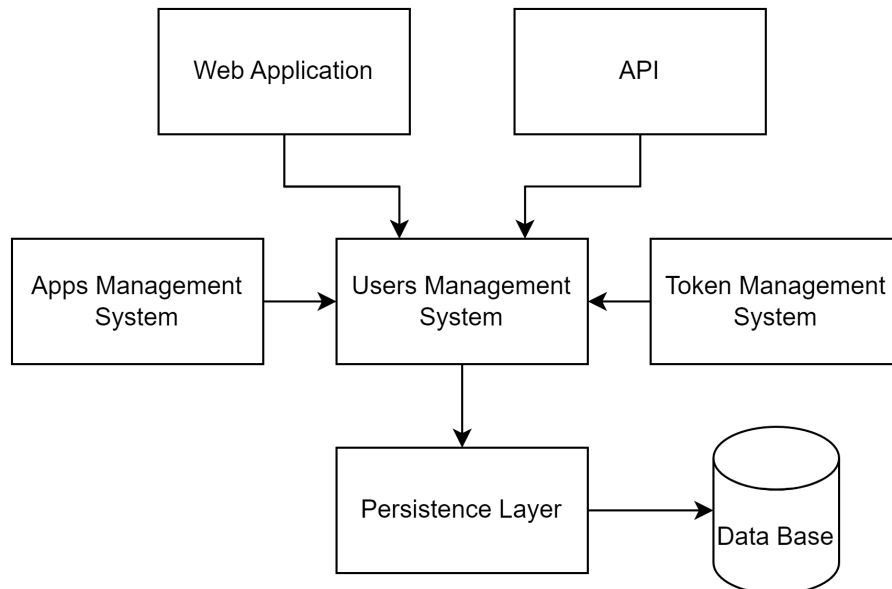
Jeżeli użytkownik chce jakąś aplikację dodać, żeby się autoryzowała przez nasz system, to musi ją zarejestrować.

3.6. **Persistence layer** - warstwa łącząca się z bazą danych (interfejs do niej)

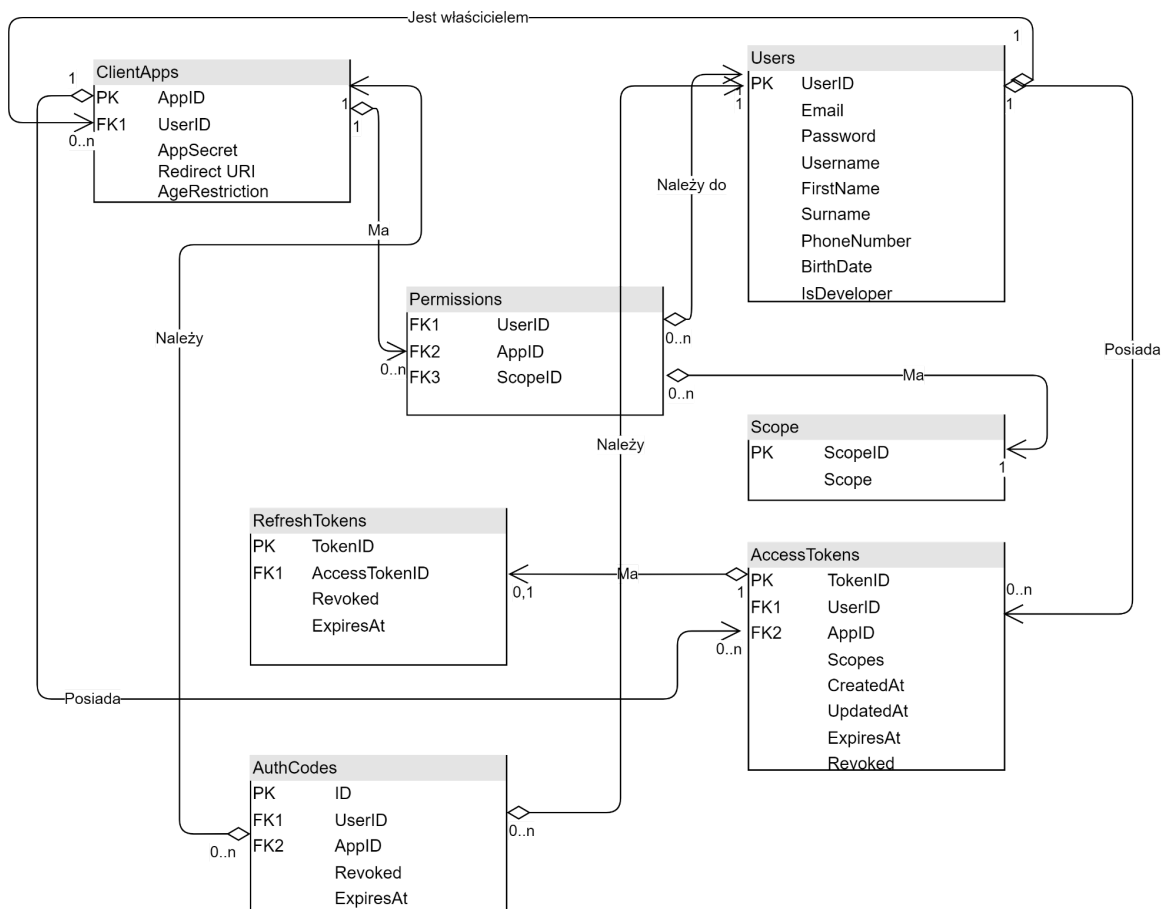
3.7. **Database** - przechowuje dane użytkowników

4. Architektura

4.1. Moduły w aplikacji



4.2. Model bazy danych



■ Role obiekty i ich reprezentacja w bazie danych

- Client App
- Redirect URI
- App ID
- User Secret
- AgeRestriction

■ User

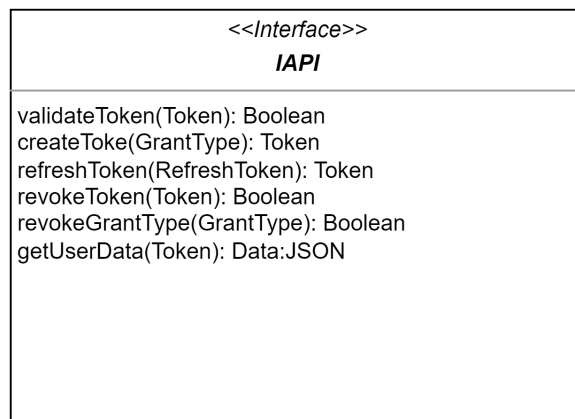
- UserID
- Email
- Password
- Username
- Imię
- Nazwisko
- Numer Telefonu
- Age

■ Permissions - Użytkownik musi wyrazić zgodę - pojawi mu się pop-up (Czy chcesz wyrazić zgodę, aby ta i ta aplikacja miała zgodę, aby zgodę do tych i do tych danych?).

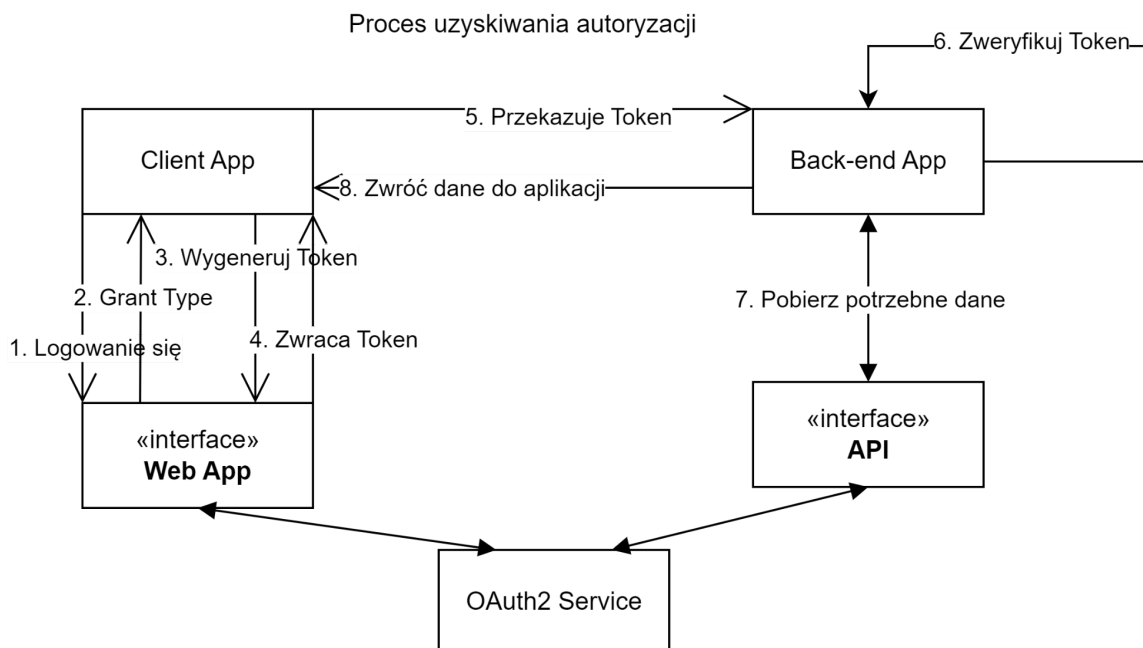
- UserID
- AppID
- Scope

- OAuth codes
 - ID
 - UserID
 - ClientID
 - Revoked
 - ExpiresAt

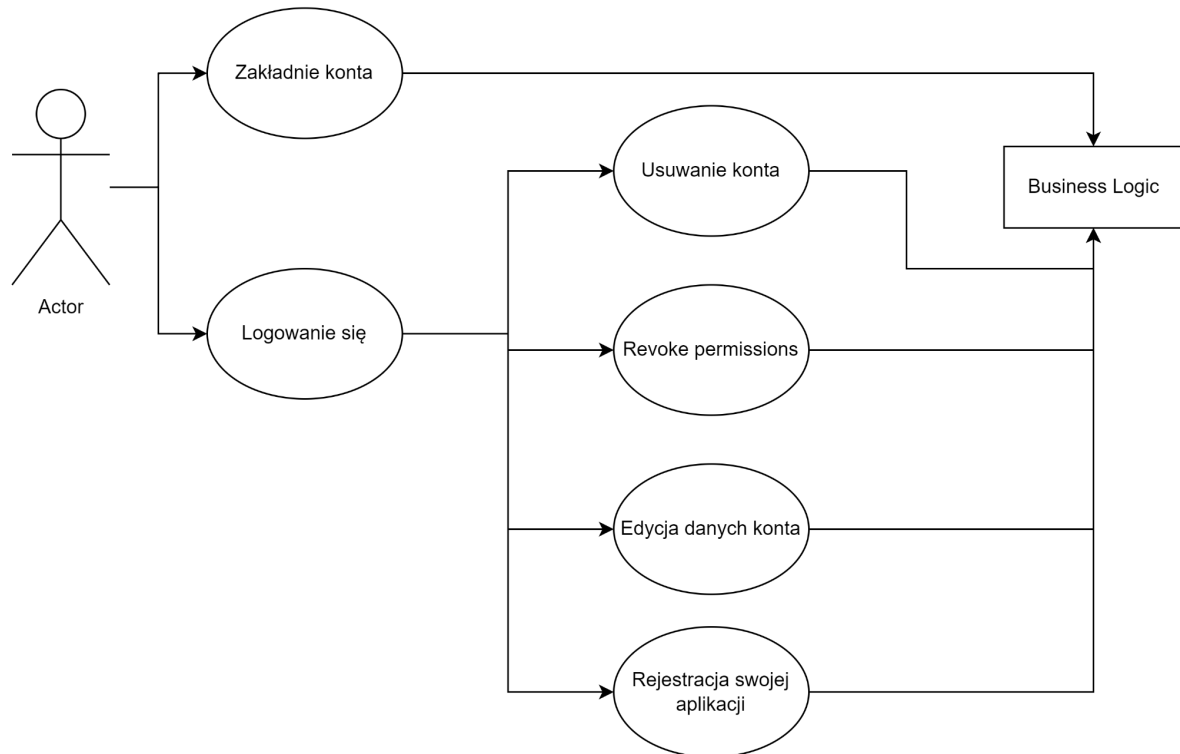
4.3. API Workflow



- Single Sign On - w aplikacji klienckiej zostanie pozostawiony Access Token, który będzie wysyłany z każdym requestem i na podstawie jego, aplikacja webowa nie będzie wymagała ponownego logowania tylko np. wyświetli ekran przyznania permissions aplikacji. Kiedy przez inną aplikację, zostanie kliknięta opcja zalogowania się, zostanie odesłany do Web Application. Do tego requestu zostanie dodany token. Wtedy serwer autoryzacyjny zweryfikuje, czy w bazie jest dany token. Sprawdzi, że dany użytkownik logował się do innej aplikacji, że wiadomo jakim jesteś użytkownikiem, i nie ma potrzeby logowania.

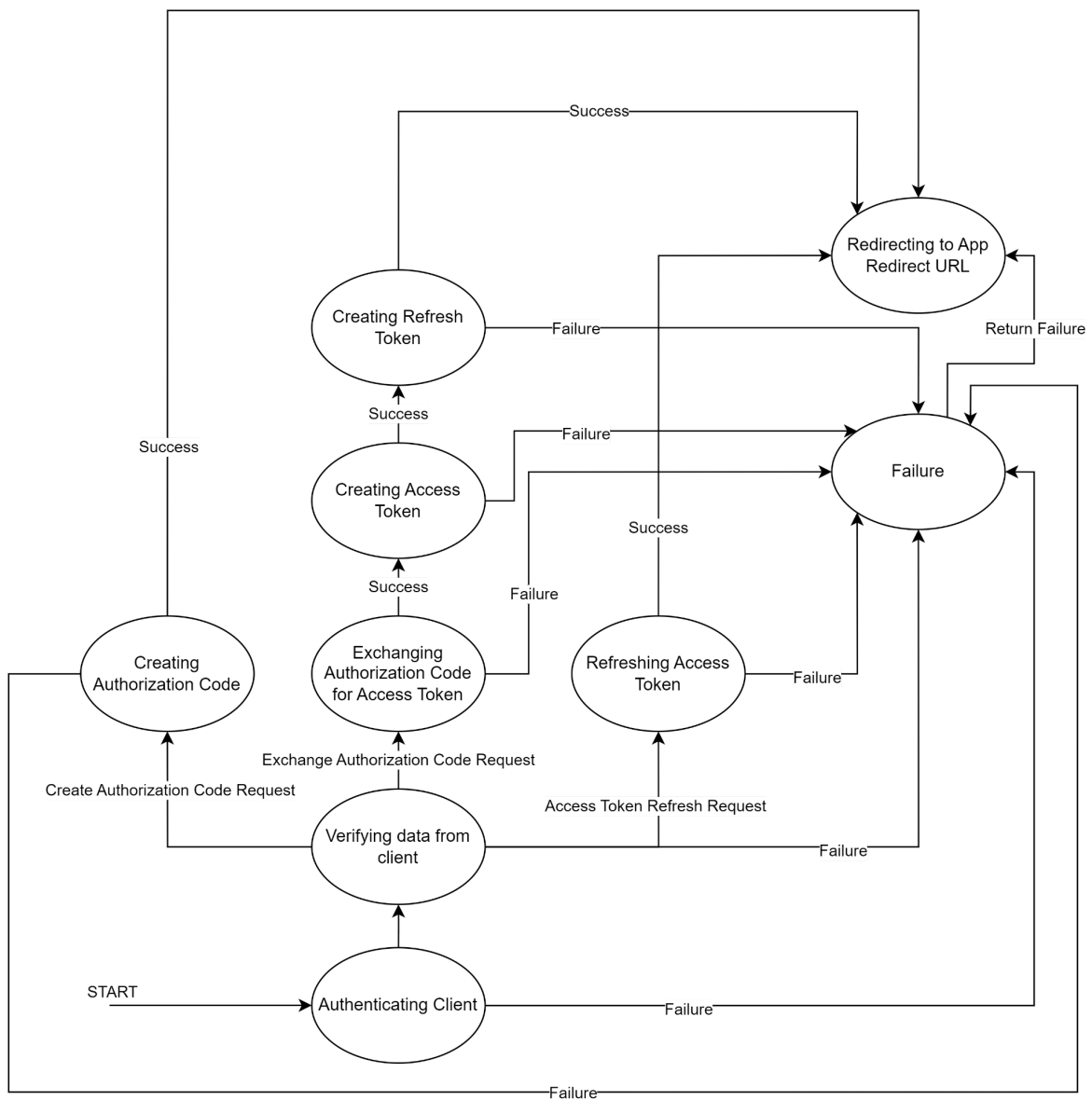


4.4. User Case - Web App



5. Stany

- Generowanie treści odpowiedzi na podstawie protokołu.
- Stworzenie nowych klas dla każdego z możliwych stanów i ekstrakcja zachowań zależnych od stanu do tych klas.
- Diagram stanów:

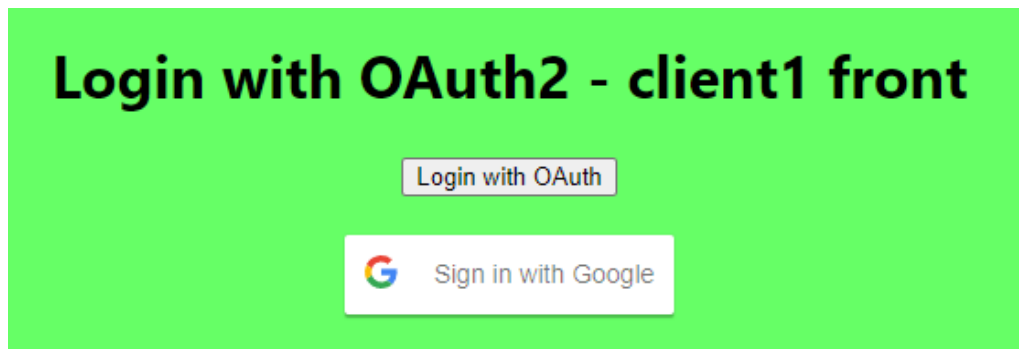


■ Możliwe stany:

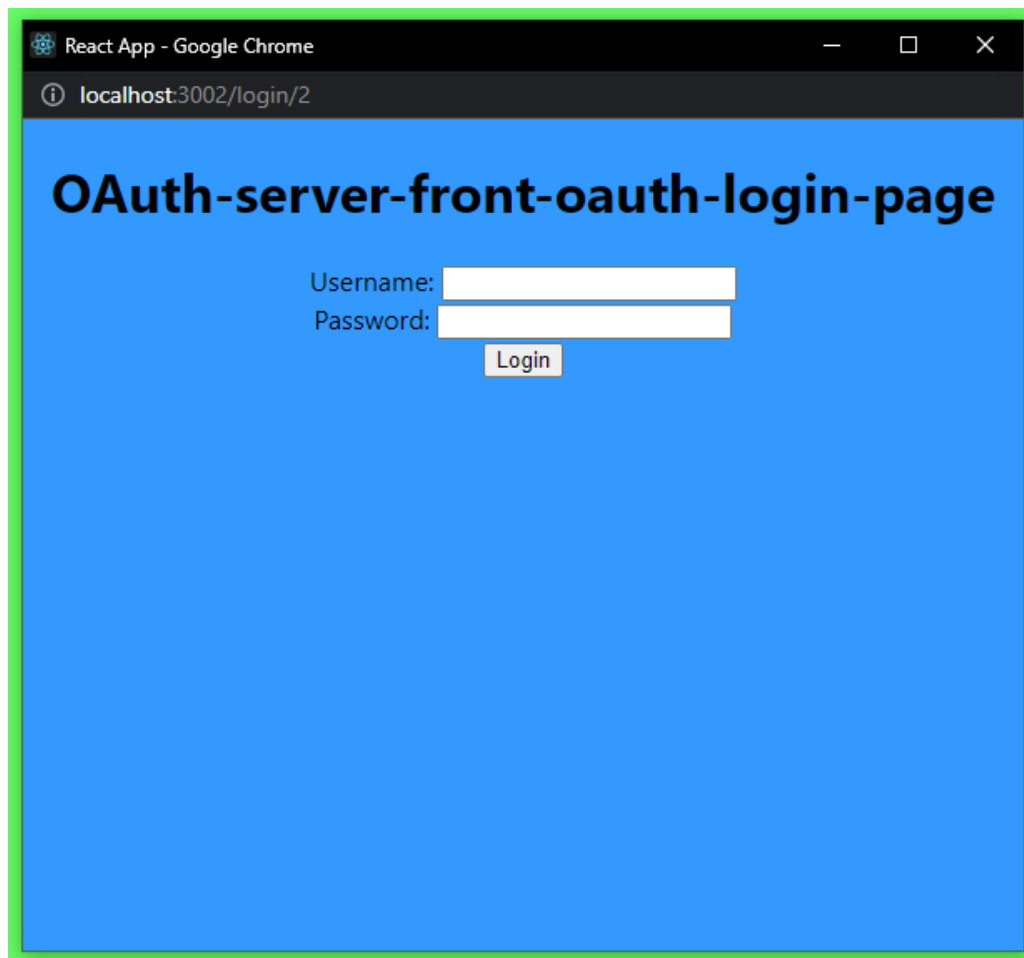
- Authenticating client
- Verifying data from client
- Creating Authorization Code
- Creating Access Token
- Creating Refresh Token
- Redirecting to App Redirect URL
- Exchanging Authorization Code for Access Token
- Refreshing Access Token
- Failure

6. Scenariusze przypadków użycia

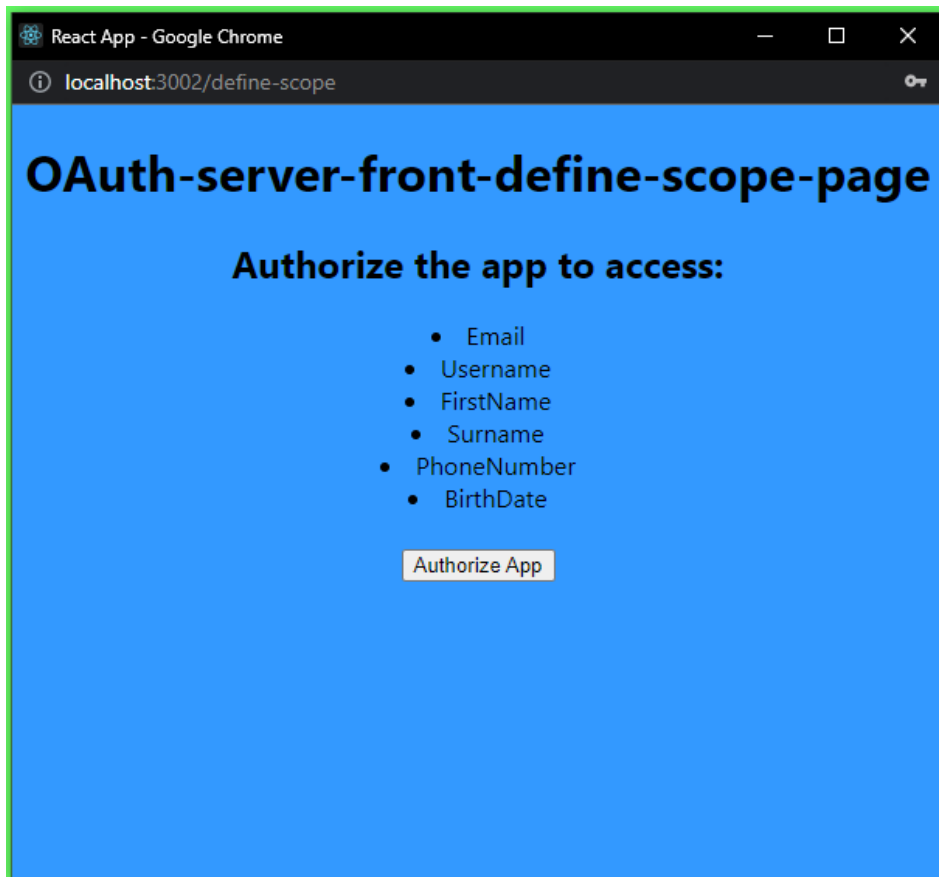
6.1. Logowanie z użyciem oauth_server



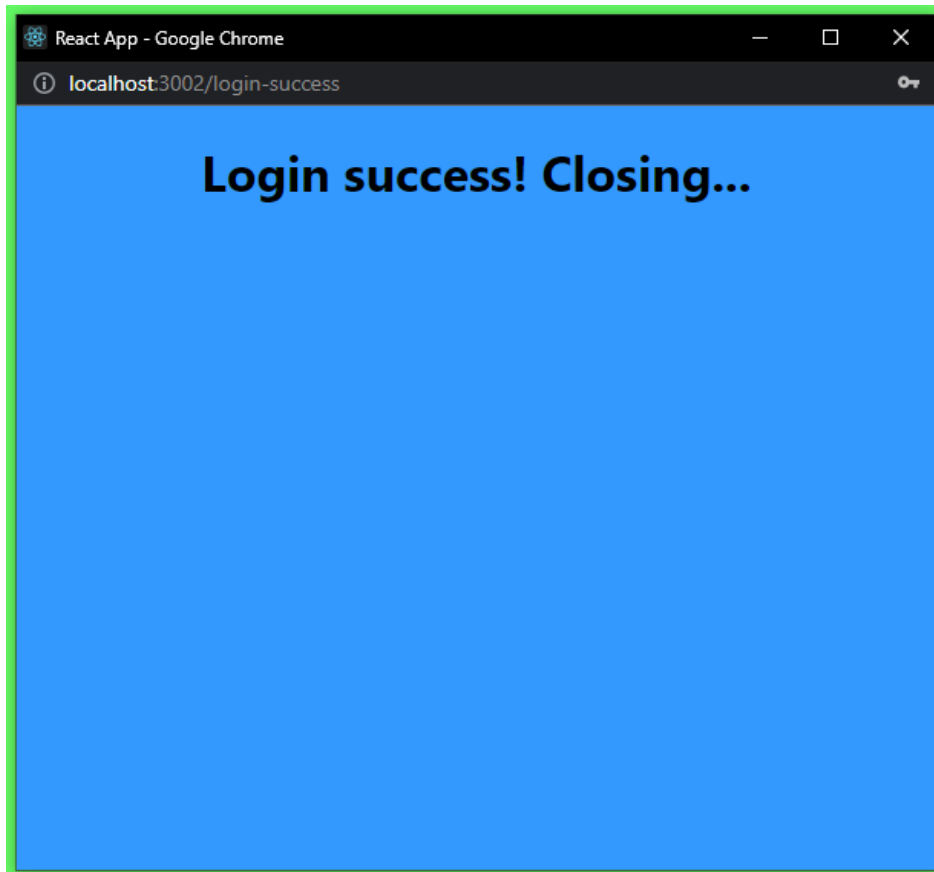
Wyskakuje nowe okno z prośbą o zalogowanie:



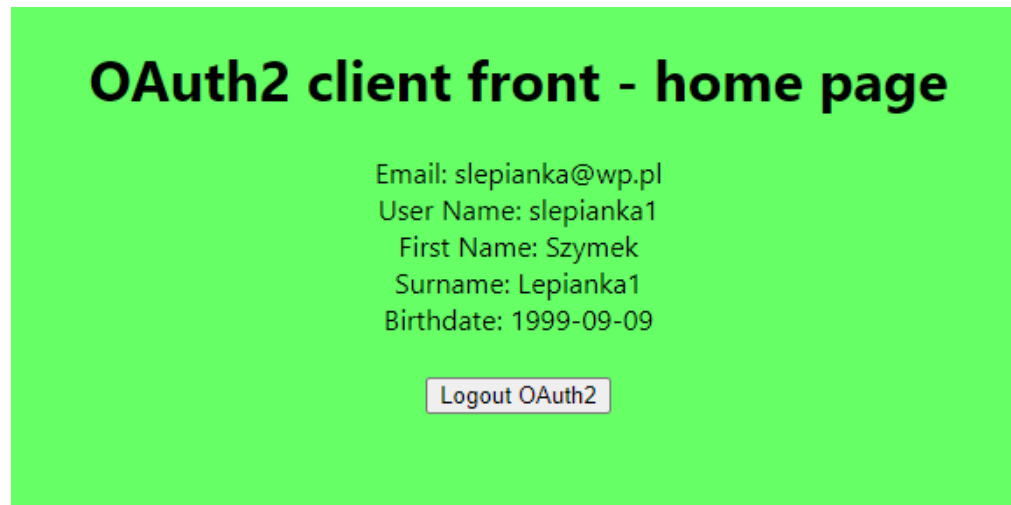
Po zalogowaniu, pojawia się komunikat o upoważnienie aplikacji do dostępu do danych:



Po zatwierdzeniu, pojawia się komunikat i okno jest zamykane:



A w aplikacji wyświetlane są dane użytkownika, pobrane z serwera autoryzującego, za pomocą Access Tokena:



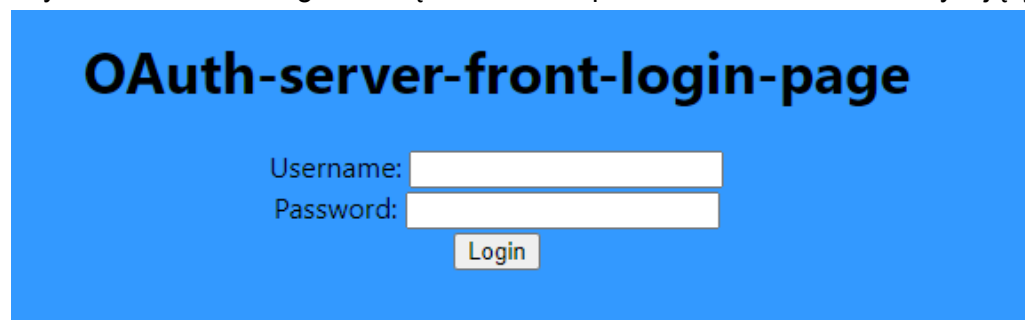
OAuth2 client front - home page

Email: slepianka@wp.pl
User Name: slepianka1
First Name: Szymek
Surname: Lepianka1
Birthdate: 1999-09-09

Logout OAuth2

6.2. Logowanie z użyciem oauth_server (w przypadku wcześniejszego zalogowania się w innej aplikacji)

Użytkownik może zalogować się również bezpośrednio w serwisie autoryzującym.

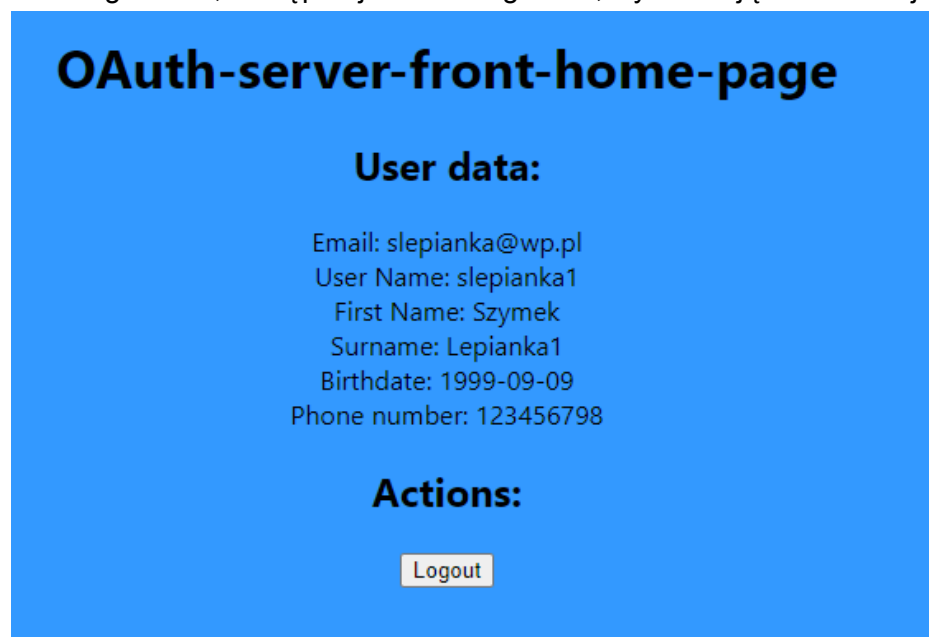


OAuth-server-front-login-page

Username:
Password:

Login

Po zalogowaniu, dostępna jest strona główna, wyświetlająca informacje o użytkowniku:



OAuth-server-front-home-page

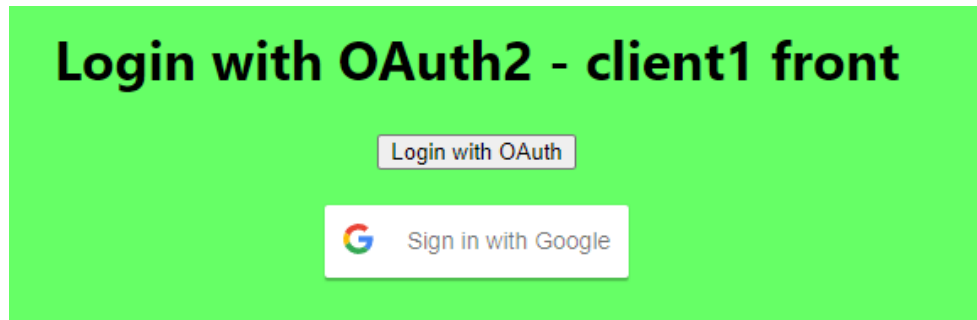
User data:

Email: slepianka@wp.pl
User Name: slepianka1
First Name: Szymek
Surname: Lepianka1
Birthdate: 1999-09-09
Phone number: 123456798

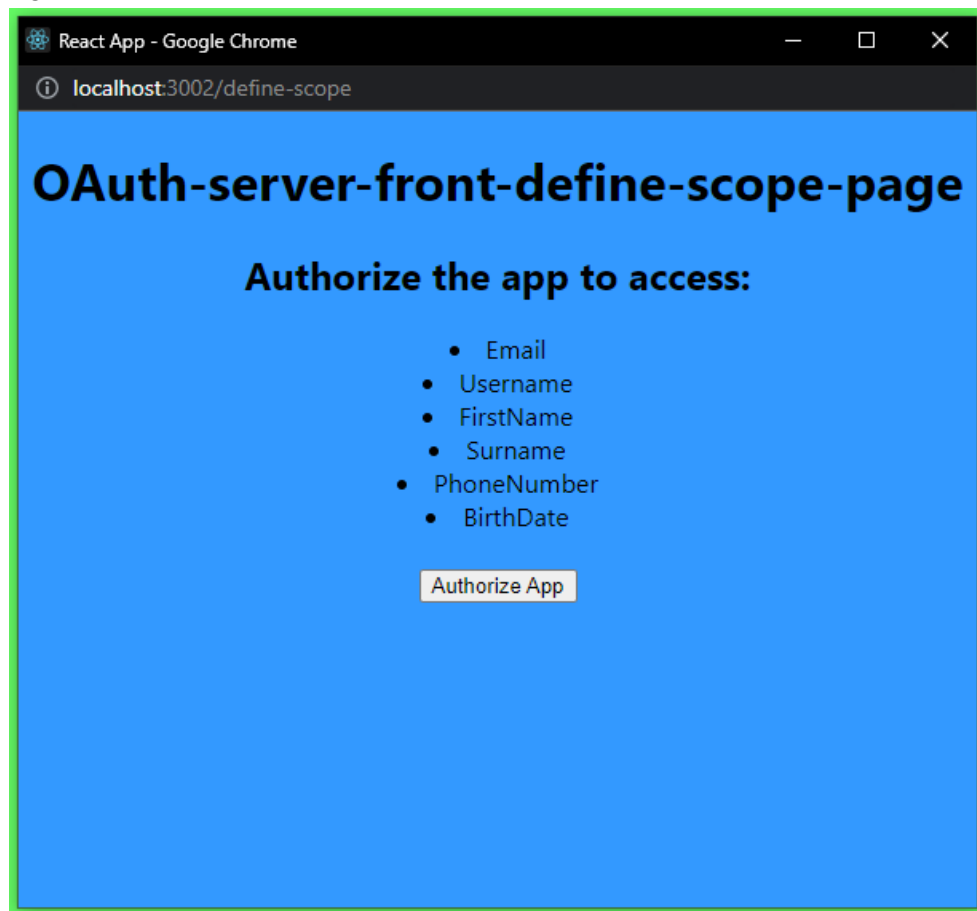
Actions:

Logout

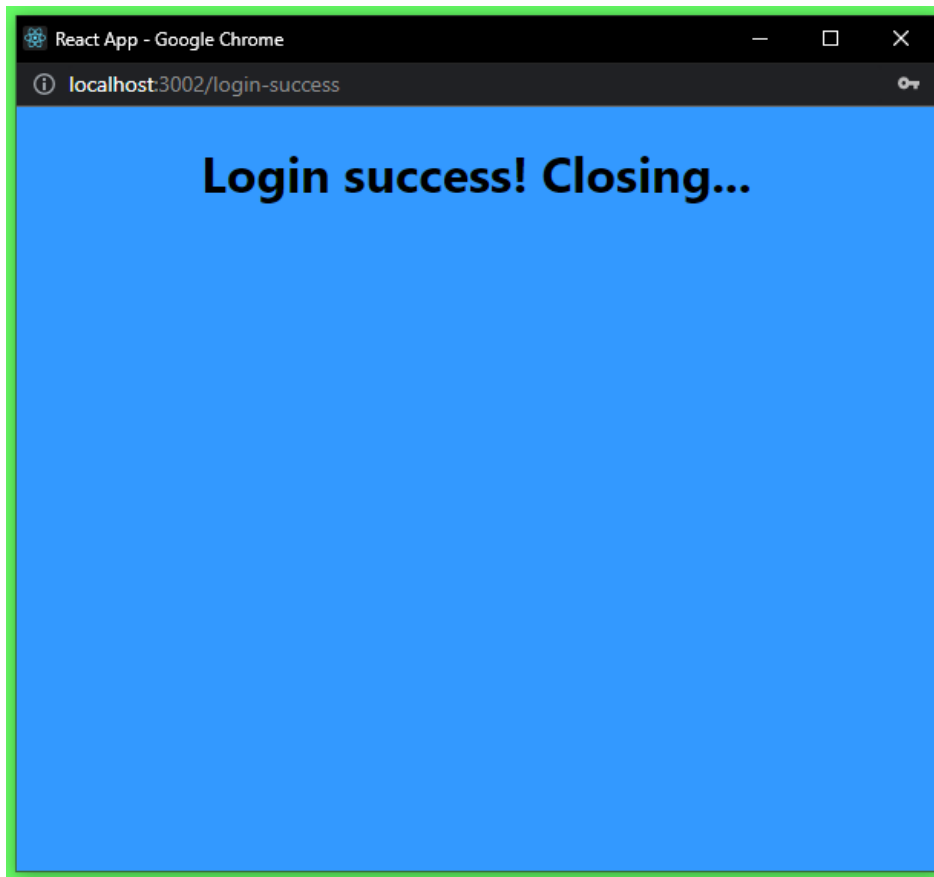
Przy próbie zalogowania się w klienckiej aplikacji:



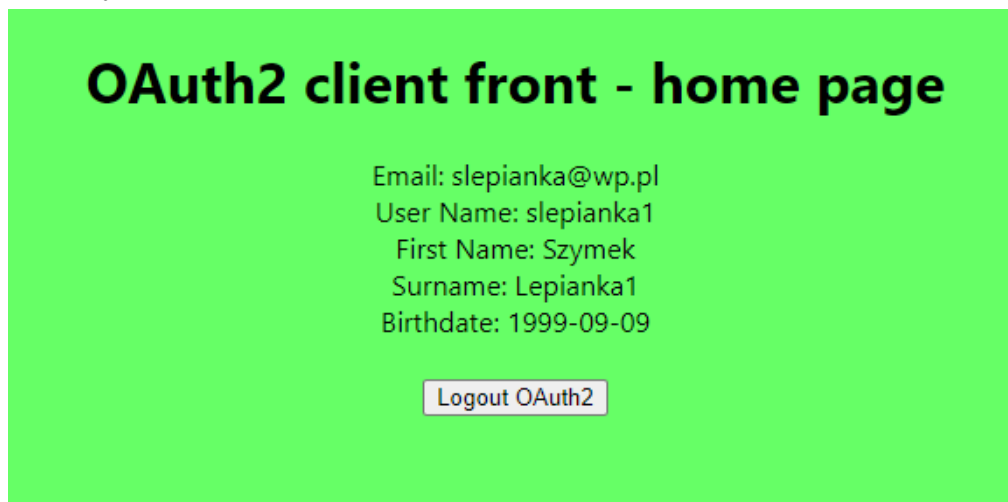
Użytkownik zostanie od razu poproszony o upoważnienie aplikacji, z pominięciem logowania:



Po zatwierdzeniu, pojawia się komunikat i okno jest zamykane:



A w aplikacji wyświetlane są dane użytkownika, pobrane z serwera autoryzującego, za pomocą Access Tokena:



6.3. Logowanie z użyciem Google OAuth2

W aplikacji klienckiej dostępna jest opcja zalogowania się z użyciem Google OAuth2:

Login with OAuth2 - client1 front

Login with OAuth



Sign in with Google

Pojawia się nowe okno z prośbą o zalogowanie się:

Logowanie – Konta Google - Google Chrome

accounts.google.com/o/oauth2/auth/identifier?redirect_uri=storagerelay%3A%2F%...

Zaloguj się przez Google

Zaloguj się

Przejdź do aplikacji **TestApp1**

Adres e-mail lub telefon

D

[Nie pamiętasz adresu?](#)

Aby można było przejść dalej, Google udostępni aplikacji TestApp1 Twoją nazwę użytkownika, adres e-mail, ustawienia języka i zdjęcie profilowe.

[Utwórz konto](#)

[Dalej](#)

polski ▼


[Pomoc](#) [Prywatność](#) [Warunki](#)

Logowanie – Konta Google - Google Chrome

accounts.google.com/signin/v2/challenge/pwd?redirect_uri=storagerelay%3A%2F%...

Zaloguj się przez Google

Szymon Lepianka

 slepianka@wp.pl

Wpisz hasło

☐ Pokaż hasło

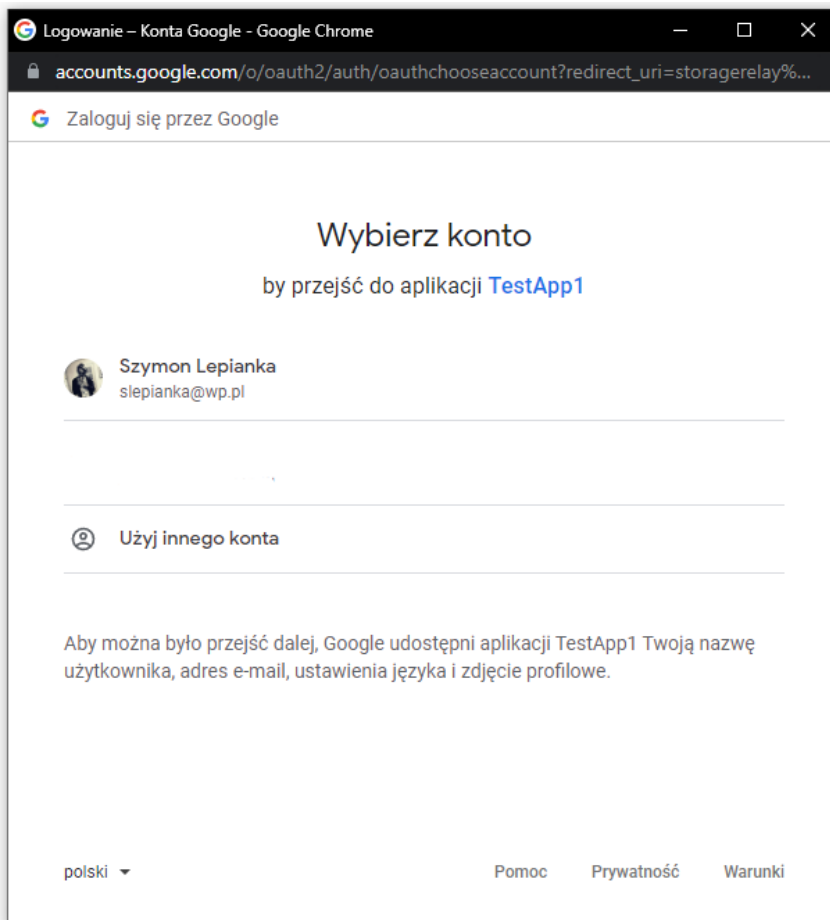
Aby można było przejść dalej, Google udostępni aplikacji TestApp1 Twoją nazwę użytkownika, adres e-mail, ustawienia języka i zdjęcie profilowe.

[Nie pamiętasz hasła?](#) [Dalej](#)

polski ▼ Pomoc Prywatność Warunki

Następnie ewentualna weryfikacja dwuetapowa.

W przypadku, gdy użytkownik jest zalogowany na urządzeniu kontem Google pojawia się komunikat z prośbą o wybór konta (zamiast logowania):



Następnie okno jest zamykane i w aplikacji wyświetlane są dane użytkownika:

