

UNI-chat - dokumentacja

Przetwarzanie danych w chmurach obliczeniowych

Kamil Jagodziński (nr. Id. 402309)

Działanie aplikacji

Strona aplikacji jest hostowana przy użyciu narzędzia render (<https://render.com>).
[\[Link do hostowanej strony\]](#).

Ogólny opis

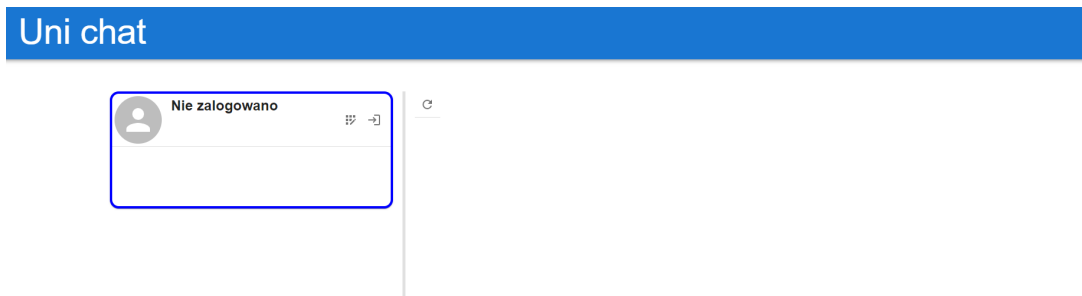
Projektem jest prosta aplikacja webowa typu SPA, inspirowana podstawowymi funkcjami serwisów takich jak messenger czy twitter. Pozwala ona na prowadzenie czatu zalogowanym, studentom oraz przeglądanie wiadomości, nawet nie będą zalogowani.

Po wejściu na stronę, użytkownik może wykonać kilka czynności:

- Przeglądać umieszczone wpisy,
- Zalogować się,
- Założyć konto,
- Dodać wpis (tylko zalogowani),
- Edytować konto (tylko zalogowani),
- Wylogować się.

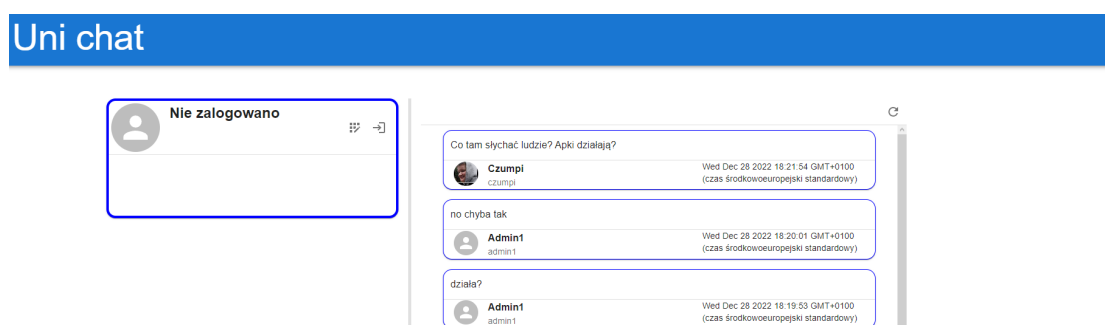
1. Przeglądanie wpisów:

Po wejściu na stronę jest ona prawie pusta. Renderowany jest tylko panel użytkownika. Na panelu widnieje informacja, że obecny użytkownik nie jest zalogowany.



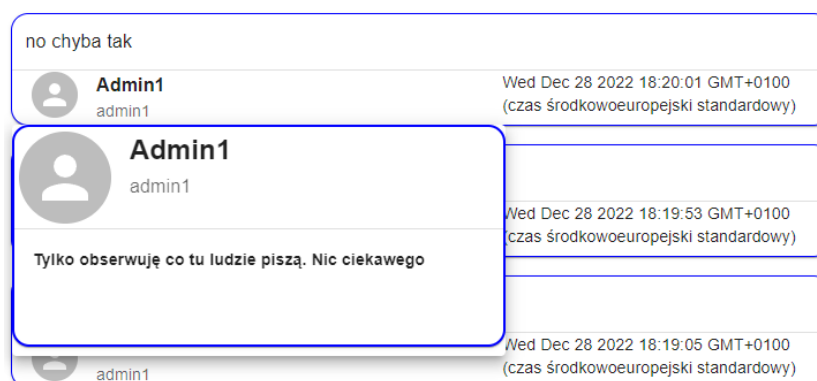
(Rys. 1) Ekran początkowy aplikacji

Aby wyświetlić umieszczone post należy kliknąć ikonę 'Odśwież', znajdującą się na prawo od panelu użytkownika – na ścianie (*wall'u*).



(Rys. 2) Strona po załadowaniu *wall'a*

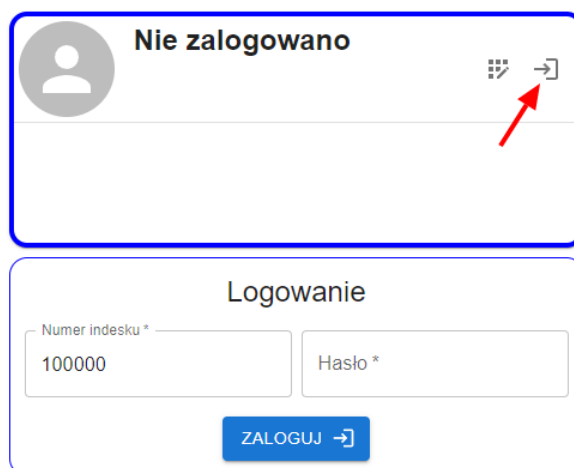
Ściana postów może być przewijana w dół i do góry. Po najechaniu kursorem na awatar lub nazwę użytkownika pojawi się popup z panelem jego profilu.



(Rys. 3) Popup użytkownika

2. Logowanie:

Aby się zalogować należy kliknąć ikonę logowania a następnie w panelu podać numer indeksu i hasło. Jeśli dane są poprawne pusty panel użytkownika zmieni się w panel aktualnie zalogowanego studenta. Do celów testów i prezentacji utworzono konto admin1 z numerem indeksu '100000' i hasłem '12'.



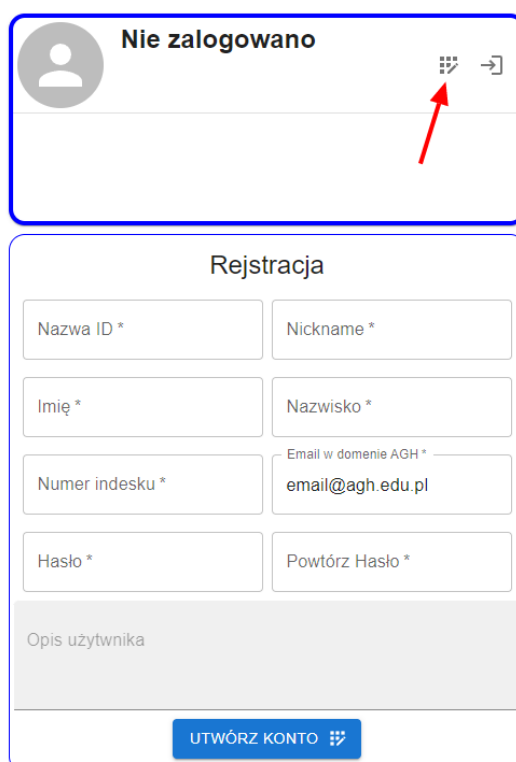
The image shows a user interface with two main sections. The top section, titled 'Nie zalogowano' (Not logged in), features a user icon and a red arrow pointing to a registration icon (a grid of dots with an arrow). The bottom section, titled 'Logowanie' (Login), contains a form with two input fields: 'Numer indeksu *' (Index number) with the value '100000' and 'Hasło *' (Password). Below these fields is a blue button labeled 'ZALOGUJ →' (Login).

(Rys. 4) Panel użytkownika i panel logowania.
Zaznaczono ikonę logowania

3. Tworzenie konta:

Aby utworzyć konto należy kliknąć ikonę rejestracji i wypełnić cały podany formularz (za wyjątkiem opisu użytkownika). Jeśli użytkownik spróbuje wykorzystać numer indeksu lub nazwę ID, która jest już w użyciu utworzenie konta nie powiedzie się. W przypadku błędnego powtórzenia hasła lub braku czy złego uzupełnienia któregoś z parametrów wyświetlony zostanie alert o błędzie.

Jeśli konto utworzono poprawnie, nastąpi automatyczne zalogowanie na nowe konto.



The image shows a user interface with two main sections. The top section, titled 'Nie zalogowano' (Not logged in), features a user icon and a red arrow pointing to a registration icon (a grid of dots with an arrow). The bottom section, titled 'Rejstracja' (Registration), contains a form with several input fields: 'Nazwa ID *' (ID name), 'Nickname *', 'Imię *' (First name), 'Nazwisko *' (Last name), 'Numer indeksu *' (Index number), 'Email w domenie AGH *' (Email in AGH domain) with the value 'email@agh.edu.pl', 'Hasło *' (Password), and 'Powtórz Hasło *' (Repeat password). Below these fields is a large text area for 'Opis użytkownika' (User description). At the bottom is a blue button labeled 'UTWÓRZ KONTO →' (Create account).

(Rys. 5) Panel użytkownika i panel tworzenia konta.
Zaznaczono ikonę rejestracji

The screenshot shows a web browser window with the address bar displaying 'uni-chat-kamil-jagodzinski-semestr1.onrender.com'. The page has a blue header with the word 'chat' partially visible. A dark notification box in the top right corner contains the text '...strony uni-chat-kamil-jagodzinski-semestr1.onrender.com' and 'Brak nazwyID', with an 'OK' button. Below the header, a user profile card is shown with a grey person icon and the text 'Nie zalogowano'. To the right of this card is a small icon of a document with a checkmark. Below the profile card is a registration form titled 'Rejstracja'. The form contains several input fields: 'Nazwa ID *', 'Nickname *', 'Imię *', 'Nazwisko *', 'Numer indeksu *', 'Email w domenie AGH *' (with the example 'email@agh.edu.pl'), 'Hasło *', and 'Powtórz Hasło *'. There is also a text area for 'Opis użytkownika'. At the bottom of the form is a blue button labeled 'UTWÓRZ KONTO' with a small icon of a document with a checkmark.

(Rys. 6) Przykład alertu informującego o błędzie

4. Dodawanie wpisu:

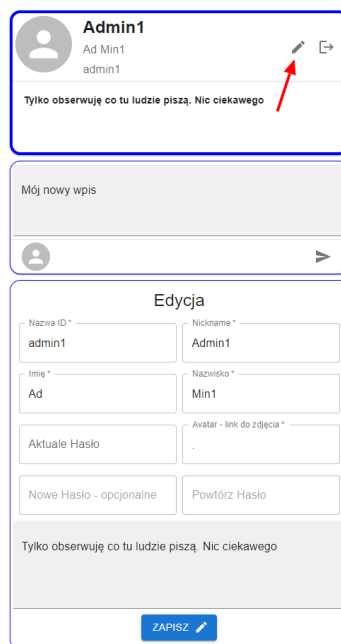
Aby dodać wpis należy być zalogowany. Pod panelem użytkownika znajduje się okno pozwalające na umieszczenie na *wall'u* nowego postu. Po wprowadzeniu treści wpisu należy kliknąć przycisk wyślij (w prawym dolnym rogu okna). Jeśli wiadomość została wysłana obszar wprowadzania tekstu zostanie wyczyszczony.

The screenshot shows a user profile card for 'Admin1'. The card has a grey person icon, the name 'Admin1', and the email 'Ad Min1' and 'admin1'. To the right of the email are icons for editing and sharing. Below the profile card is a text area for creating a new post, labeled 'Mój nowy wpis'. At the bottom of the text area is a small grey person icon and a blue arrow pointing right, indicating the 'send' button.

(Rys. 7) Panel dodawania nowego postu

5. Edycja konta:

Aby edytować konto należy wybrać z panelu użytkownika ikonę edycji i wprowadzić nowe dane w oknie. Początkowo znajdują się w nim aktualne wartości. Zmiany zatwierdza się wpisując hasło. Dodatkowo możliwe jest dodanie awatara użytkownika.

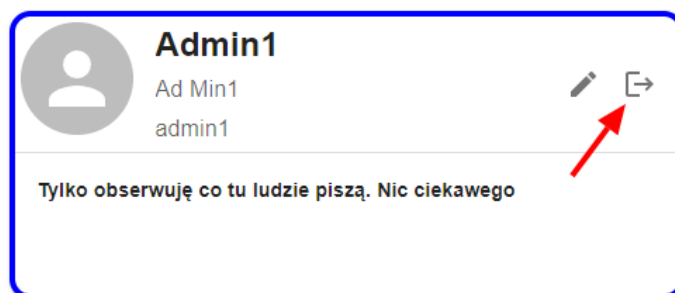


(Rys. 8) Okno edycji konta.

Na panelu użytkownika zaznaczono ikonę edycji

6. Wylogowanie:

W celu wylogowania się należy kliknąć ikonę 'wyloguj'. Użytkownik natychmiast zostanie wylogowany.



(Rys. 9) Panel użytkownika z zaznaczoną ikoną 'wyloguj'

Baza danych

Baza danych aplikacji składa się zbudowana jest z dwóch rodzajów węzłów – USER i POST oraz relacji jednostronnej relacji POSTED. Najbardziej rozbudowane są węzły USER – przechowują one wszystkie pobierane informacje o użytkowniku. Relacje POSTED oraz węzły POSTED zawierają tylko podstawowe informacje na temat zamieszczonego wpisu – treść, id (węzeł USER) oraz dokładny czas publikacji (relacja POSTED).



(Rys. 9) Diagram bazy danych wykorzystanej w projekcie

Zapytania do bazy danych:

Do wykonania zapytania każdorazowo otwiera i zamyka się połączenie(sesje) z bazą przy pomocy sterownika 'neo4j-driver'. Oprogramowanie pozwala na wykonywanie parametryzowanych zapytań z kodu napisanego w języku JavaScript.

1. Logowanie:

```
const result = await session.run(  
  'MATCH (a:USER {idx: $idx, pass: $pass}) RETURN a',  
  { idx: logID, pass: pass }  
)
```

2. Rejestracja:

```
const result = await session.run(
  'CREATE (a:USER { userid: $userID, nickname: $nickname, '+
  'name: $name, surname: $surname, '+
  'idx: $idx, email: $email, pass: $pass, '+
  'bio: $bio, avatar: " " }) RETURN a',
  {
    userID: userID,
    nickname: nickname,
    name: name,
    surname: surname,
    bio: bio,
    email: email,
    idx: logID,
    pass: pass }
)
```

3. Edycja konta:

```
const result = await session.run(
  'MATCH (n:USER {idx: $idx, pass: $oldPassword}) '+
  ' SET n.userid = $userID '+
  ' SET n.name = $name '+
  ' SET n.surname = $surname '+
  ' SET n.nickname = $nickname '+
  ' SET n.pass = $newPassword '+
  ' SET n.avatar = $avatar '+
  ' SET n.bio = $bio '+
  ' RETURN n',
  { idx: idx, userID: userID, name: name,
    surname: surname, nickname: nickname,
    oldPassword: oldPassword, newPassword: newPassword,
    avatar: avatar, bio: bio }
)
```

4. Dodawanie wpisu:

```
await session.run(
  " CREATE (a:POST {content: ' "+content+"'}) "+
  " WITH a "+
  " MATCH (n:USER {idx: $idx}) "+
  " CREATE (n)-[:POSTED{send: ' "+time+"'}]->(a) ",
  { idx: idx }
)
```

5. Zebranie postów do wyświetlenia:

```
const result = await session.run(
  'Match (u:USER) -[pd:POSTED]->(p:POST) ' +
  ' RETURN u.nickname, u.userid, u.avatar, ' +
  ' u.bio, p.content, pd.send, id(pd) ' +
  ' ORDER BY id(pd) DESC; ',
)
```

Wykorzystane technologie

Baza danych: [Neo4j](#)

Frontend: [JavaScript](#), [React.js](#), [Material UI](#)

Backend: [JavaScript](#), [React Redux](#), [Node.js](#), [Neo4j-driver](#)

Host: [Render.com](#)