

Serwis Społecznościowy – Dokument Projektowy

Opis Projektu

Serwis społecznościowy, umożliwiający:

- Rejestrację / Logowanie
- Upload zdjęć
- Dodawanie użytkowników do znajomych
- Dodawanie postów
- Ocenę postów
- Tworzenie grup użytkowników
- Dodawanie komentarzy do postów
- Czat z innymi użytkownikami

Motywacja

Serwis rozwiązuje problem komunikacji ze znajomymi, tworzenia grup o określonym przeznaczeniu oraz wymianie informacji oraz treści pomiędzy użytkownikami.

Technologie

W projekcie wykorzystywane będą następujące technologie:

- ASP.NET Core (C#) – backend
- Angular (TypeScript) – frontend
- MySQL - baza danych

Motywacja wyboru technologii – angular oraz ASP.NET Core są nowoczesnymi oraz wypróbowanymi frameworkami, które będą adekwatne do celu postawionego w tym projekcie.

Przedział czasowy

Część 1 (Planowanie) - 16/10/2021 – 31/01/2022

Część 2 (Implementacja) – 28/02/2022 – [wstawić czas końcowy]

Role członków zespołu

- Tomasz Celiński – Lider projektu, backend developer / tester
- Damian Tomczyszyn – backend developer / tester
- Artur Żelazko – frontend developer / tester
- Dima Nimchynskyi – frontend developer / tester

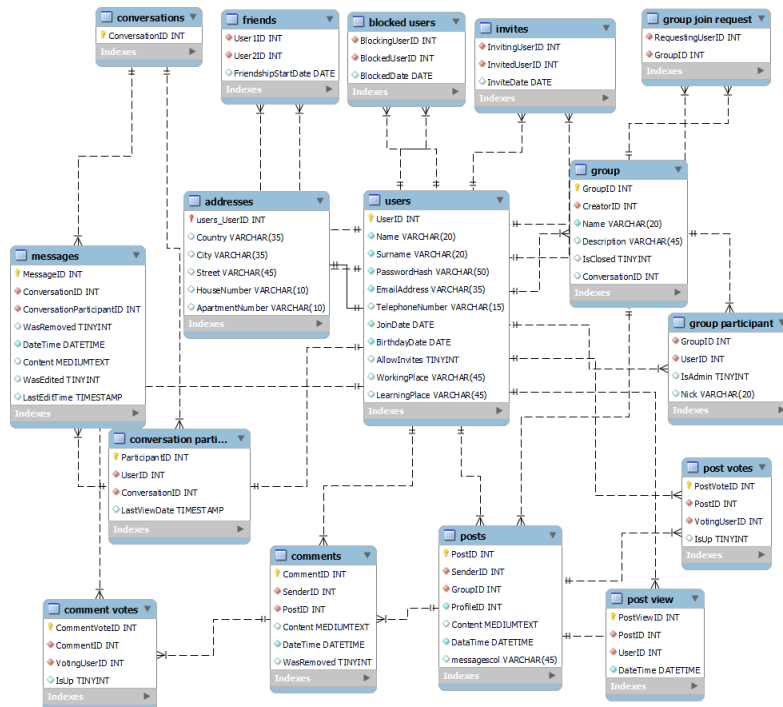
Milestoney, Epiki, Historyjki, Taski

- **Milestone 1**
 - Wybór technologii
 - Wybór stosu technologicznego
 - Wybór programów do tworzenia diagramów / schematów
 - Sprawy organizacyjne
 - Stworzenie szkieletu dokumentu projektowego
 - Określenie środowiska deweloperskiego
 - Utworzenie repozytorium
 - Wybór tematyki projektu
- **Milestone 2**
 - Przygotować opisy / wizualizacje routów
 - Opisać routy
 - Stworzyć mockupy
 - Przygotowanie schematów / diagramów cz1
 - Wstępny schemat bazy danych
 - Diagramy stanów
 - Zaktualizować dokument projektowy
 - Diagram klas – frontend cz 1
 - Analiza najczęściej wykonywanych zapytań do bazy danych
 - Diagram klas – backend

- Zaktualizować schemat bazy danych
- Przygotowanie schematów / diagramów cz2
 - Diagram klas – frontend cz 2
 - Normalizacja schematu bazy danych pod kątem najczęściej wykonywanych zapytań
 - Diagram sekwencji – frontend
 - Diagram sekwencji – backend
- **Milestone 3**
 - Przygotować scenariusze testowe cz1
 - Scenariusze testowe – frontend cz1
 - Scenariusze testowe – backend cz1
 - Scenariusze testowe – frontend cz2
 - Scenariusze testowe – backend cz2

Dokumentacja Techniczna

Schemat bazy danych:



Analiza najczęściej wykonywanych zapytań do bazy danych

1. Get api/conversations - Pobieranie konwersacji.
2. POST api/conversations/{id}/message - Wysłanie wiadomości.

```
SELECT * FROM messages
```

```
WHERE messages.ConversationParticipantID = '123456'
```

```
ORDER BY DateTime;
```

wyszukanie wszystkich wiadomości w konwersacji o id 123456 i wyświetlenie ich posortowanych po dacie malejaco.

W serwisie społecznościowym jednym z najczęściej wykonywanych działań jest chatowanie z innymi.

3. GET api/user-profile/feed?pageSize={pageSize}&page={page} - Pobranie zawartości strony.
4. GET api/posts/{id} - Wyświetlenie posta
5. SELECT DISTINCT ConversationID FROM conversation participant
JOIN users on conversation participant.UserID = users.UserID
WHERE UserID = '1234567';

Wyszukanie id konwersacji dla użytkownika o id 1234567.

Dużo czasu spędza się przy przeglądaniu relacji i wiadomości na stronie głównej.

6. Get api/user-profile/feed
7. Get api/user-profile/friends

```
SELECT FriendshipStartDate
```

```
FROM friends
```

```
JOIN users on friends.User1ID = users.UserID AND friends.User2ID = users.UserID
```

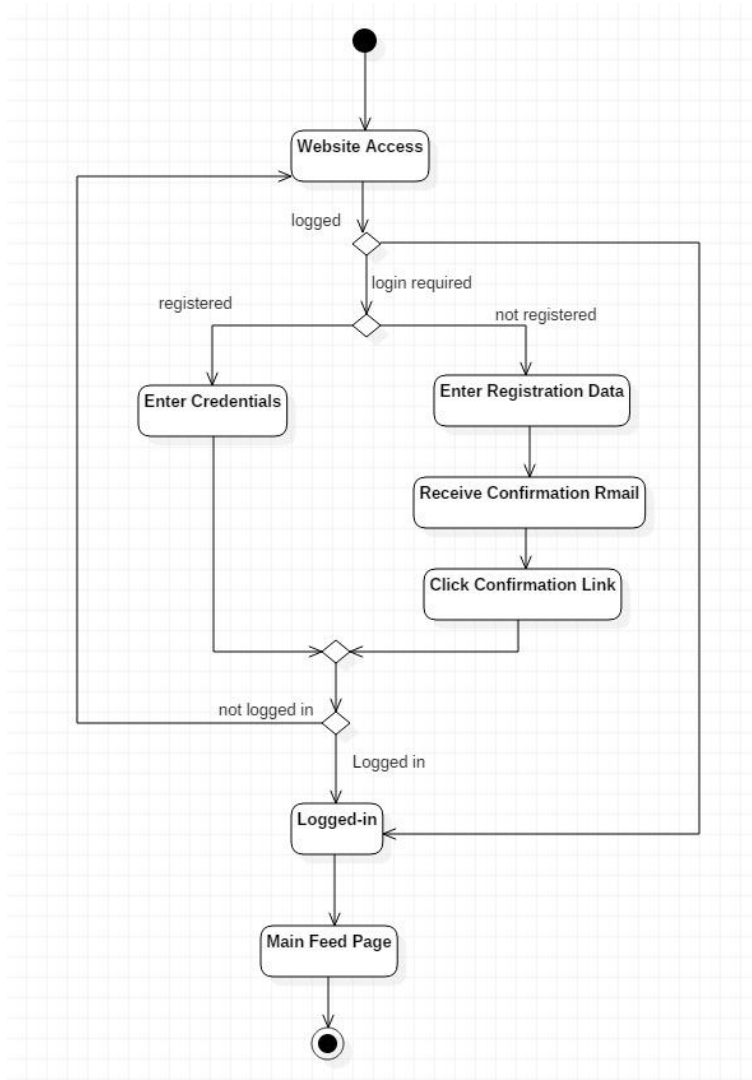
```
WHERE user.UserID = '1234567';
```

Wypisanie znajomych użytkownika o ID 1234567.

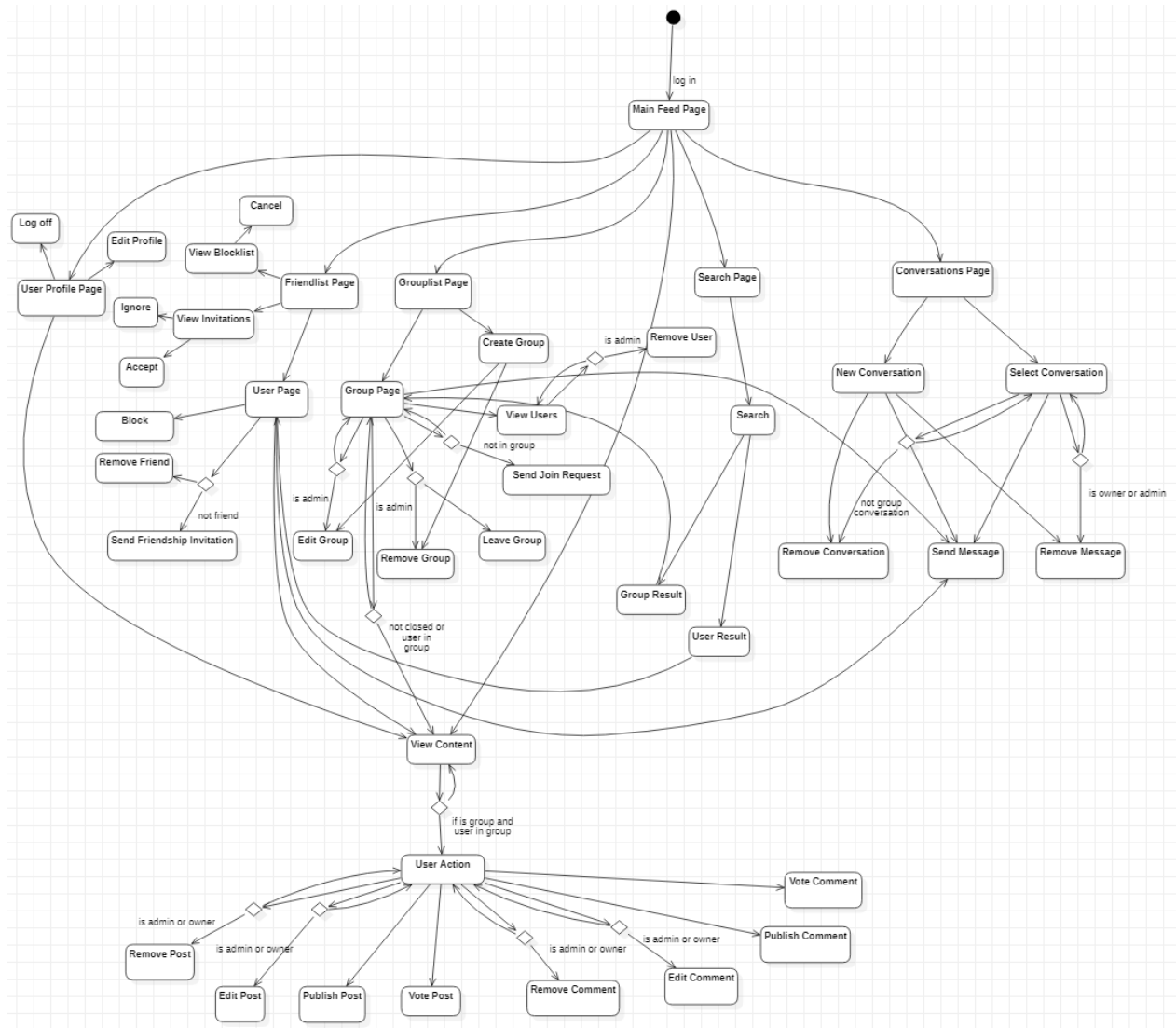
Informacje o użytkownikach są elementem który towarzyszy niemalże każdemu elementowi całym serwisie społecznościowym .

Diagramy stanów:

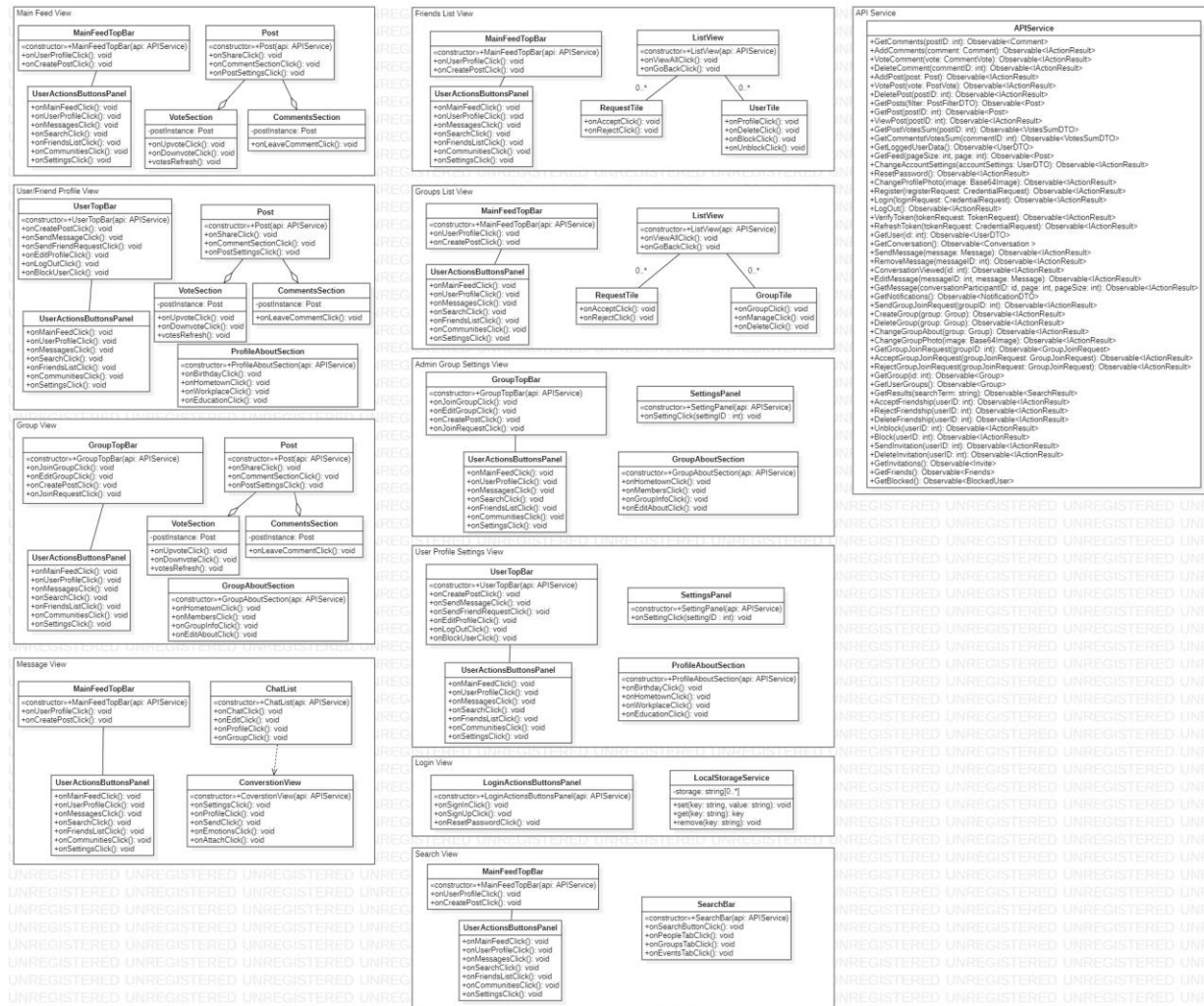
Logowanie:



Main feed:

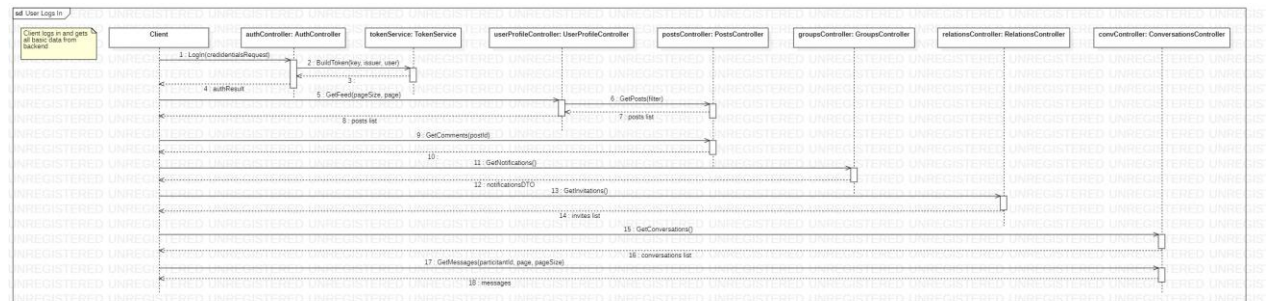


Frontend:

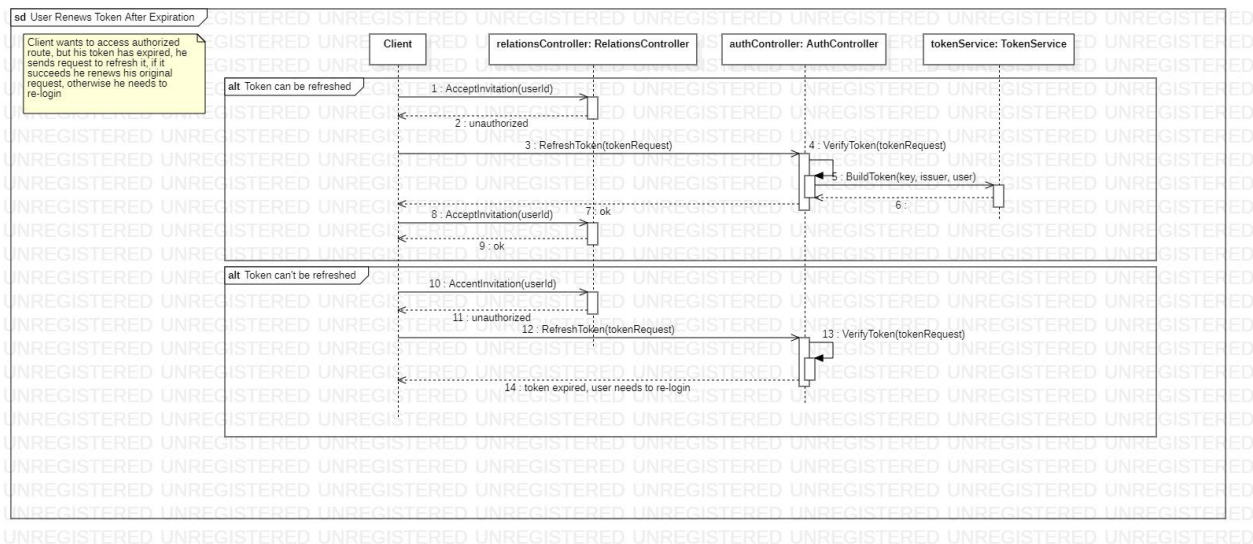


Diagramy sekwencji

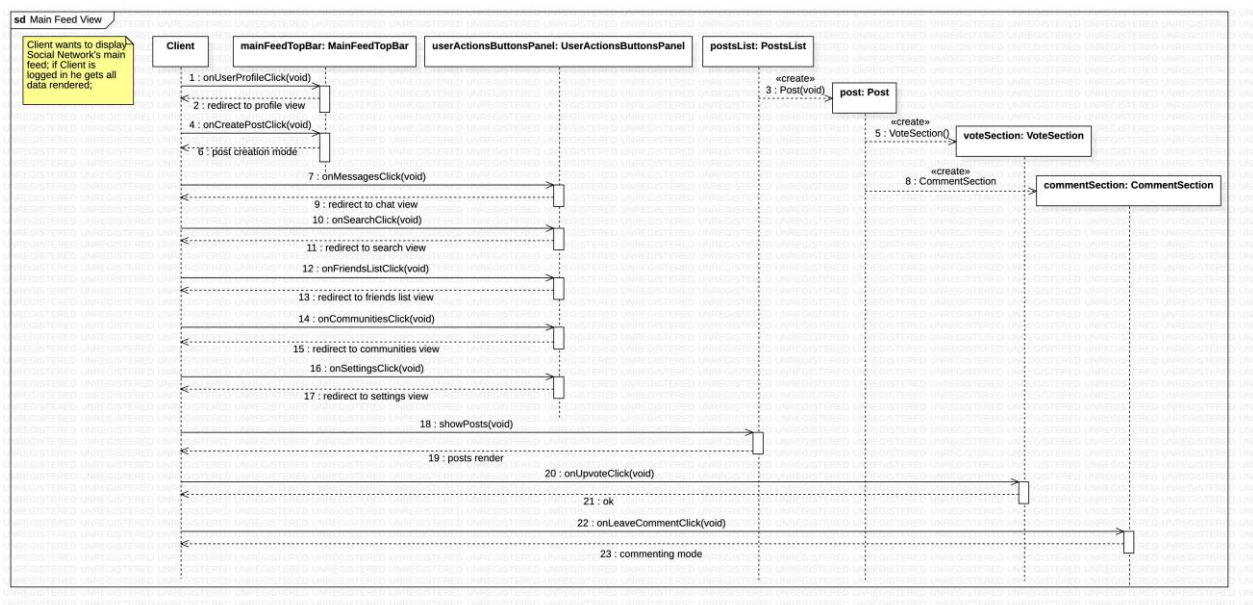
Logowanie i pobranie danych (backend)



Odnowienie tokenu (sukces oraz niepowodzenie, backend)



Main Feed View (frontend)



Infrastruktura sprzętowa / zasoby wykorzystywane w projekcie

Opis Środowisk

- Dev - Niestabilne środowisko do wprowadzania zmian mogących zagrozić stabilności systemu
- Test – Środowisko testerskie do przeprowadzania testów na funkcjonalnościach potwierdzonych w fazie dev
- Prod – Środowisko stabilne, na którym działa klient