

**Instytut Informatyki i Automatyki**

**Kierunek Informatyka**

**Projekt zespołowy 2**

***Temat: „Publiczny czat, aplikacja przeglądarkowa i okienkowa”***

**Prowadzący:** Dr inż Janusz Rafałko

Wykonujący projekt: Patryk Chiliński

Adrian Mocianko

Mateusz Pierzchała

Julian Skowroński

Łukasz Zalewski

Studia niestacjonarne I stopnia

Kierunek: **Informatyka**

Semestr: VI, grupa I

Spis treści

1. Opis projektu3
2. Wymagania funkcjonalne3
3. Wymagania niefunkcjonalne3
4. Przypadki użycia4

4.1 Diagram przypadków użycia5

1. Diagram sekwencji6
2. Technologia6
3. Dodatek A6

Podział pracy6

Harmonogram7

Instalacja i konfiguracja7

Opis elementów interfejsu graficznego aplikacji internetowej9

Indeks tabel i ilustracji

Tabela1. Lista wymagań funkcjonalnych aplikacji3

Tabela2. Lista wymagań niefunkcjonalnych aplikacji3

Tabela3. Przypadki użycia4

Rys1. Diagram przypadków użycia5

Rys2. Diagram sekwencji6

Rys3. Uruchomienie aplikacji internetowej8

Rys4. Strona logowania8

Rys5. Strona główna aplikacji internetowej9

Rys6. Interfejs graficzny aplikacji internetowej9

**1. Opis projektu**

Aplikacja ma za zadanie realizację funkcjonalności pozwalającej na tekstowe komunikowanie się między użytkownikami poprzez aplikację internetową oraz klienta na urządzenia PC. Bazować będzie na otwartym, publicznym czacie niewymagającym rejestracji, jedynie podania nazwy użytkownika. Aplikacja zawierać będzie możliwość przesyłania wiadomości tekstowych o różnych kolorach, rodzajach czcionki, "emotikon", grafik o niedużej rozdzielczości.

**2. Wymagania funkcjonalne**

Tworzona aplikacja powinna dać użytkownikowi możliwość dołączenia do publicznego czatu, wysyłania i odbierania wiadomości.

|  |  |
| --- | --- |
| Wymaganie funkcjonalne | Opis |
| Logowanie do aplikacji | Wybranie nazwy użytkownika |
| Wysyłanie wiadomości | Użytkownik może wysyłać wiadomości publiczne do innych użytkowników |
| Odbieranie wiadomości | Użytkownik może odbierać wiadomości od innych użytkowników |
| Zmiana stylów | Użytkownik może zmieniać styl aplikacji internetowej |

Tabela1: Lista wymagań funkcjonalnych aplikacji

**3. Wymagania niefunkcjonalne**

|  |  |  |
| --- | --- | --- |
| Obszar wymagań | Nr wymagania | Opis |
| Użyteczność | 1 | Aplikacja okienkowa i przeglądarkowa powinna być dostępna w języku polskim |
| Niezawodność | 2 | Program nie może kończyć się w sposób nieprzewidziany |
| 3 | Metadane użytkowników muszą być dokładnie zabezpieczone przed nieuprawnionym dostępem przez osoby trzecie |
| Wydajność | 4 | Czas potrzebny na wysłanie wiadomości nie dłuższy niż 30 sekund |
| 5 | Komunikator powinien mieć możliwość obsługi kilkunastu użytkowników jednocześnie. |
| Utrzymanie | 6 | Komunikator będzie utrzymywany przez projektantów. |
| 7 | Komentarze w kodzie w języku polskim |

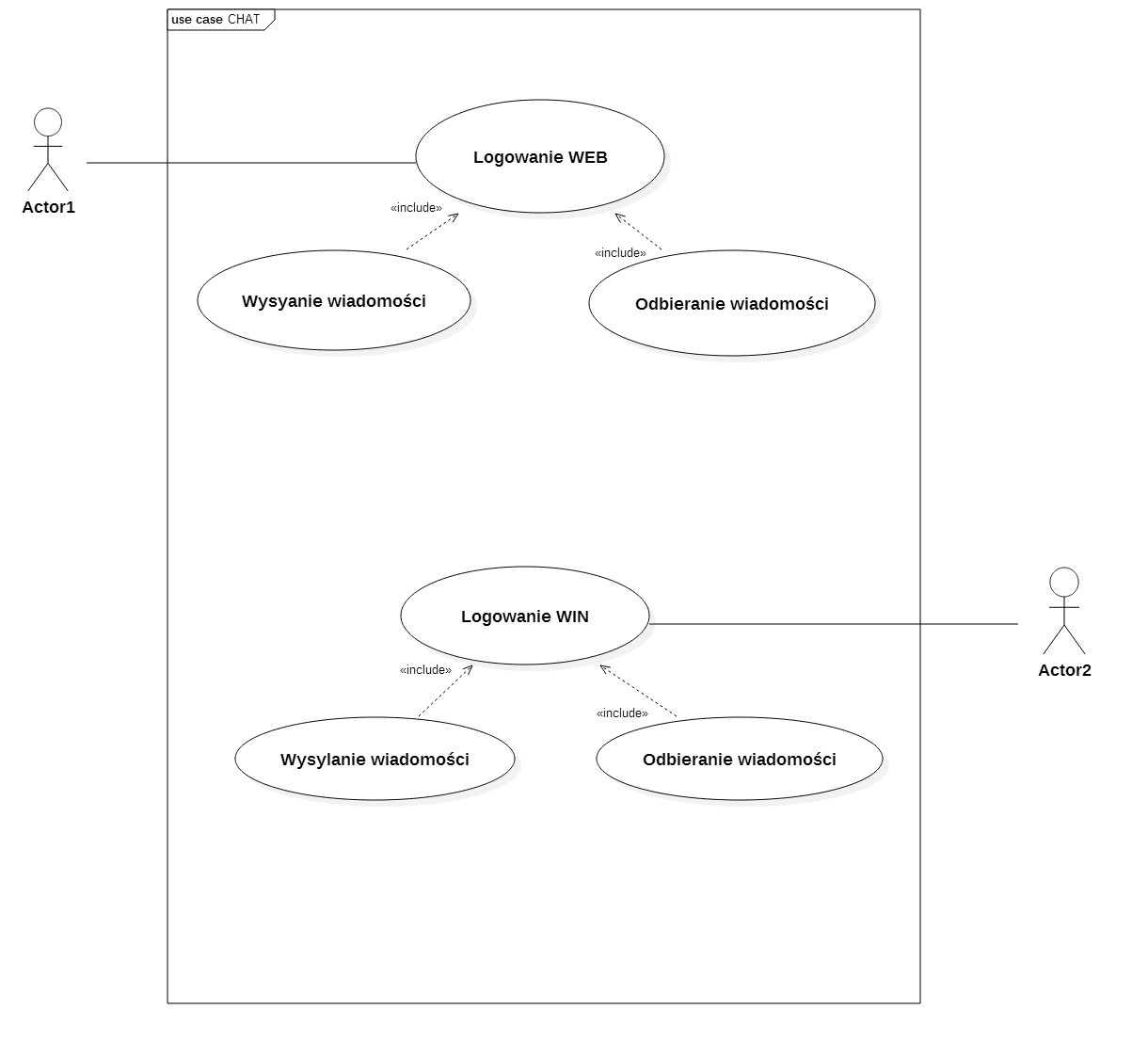
Tabela2: Lista wymagań niefunkcjonalnych aplikacji

**4. Przypadki użycia**

|  |  |
| --- | --- |
| Przypadek użycia | Scenariusz główny |
| 1.Zaloguj | - uruchomienie aplikacji  -podanie nazwy użytkownika |
| Scenariusz alternatywny |
| - błąd podczas logowania, nazwa użytkownika jest już zarezerwowana |
| 2.Połącz | Scenariusz główny |
| - wybranie opcji „połącz”  - po nawiązaniu połączenia użytkownik ma możliwość wysyłania wiadomości |
| Scenariusz alternatywny |
| - brak połączenia z Internetem |
| 3.Rozłącz | Scenariusz główny |
| - wybranie opcji rozłącz  - po rozłączeniu aplikacja blokuje elementy interfejsu dostępne jedynie po zalogowaniu |
| 4.Wyślij wiadomość | Scenariusz główny |
| - wysłanie wiadomości do innego użytkownika |
| Scenariusz alternatywny |
| -system wykrywa błąd podczas wysyłania wiadomości i wyświetla komunikat |
| 5.Odbierz wiadomość | Scenariusz główny |
| - otrzymanie wiadomości od innego użytkownika |
| Scenariusz alternatywny |
| -system wykrywa błąd podczas odbierania wiadomości i wyświetla komunikat |

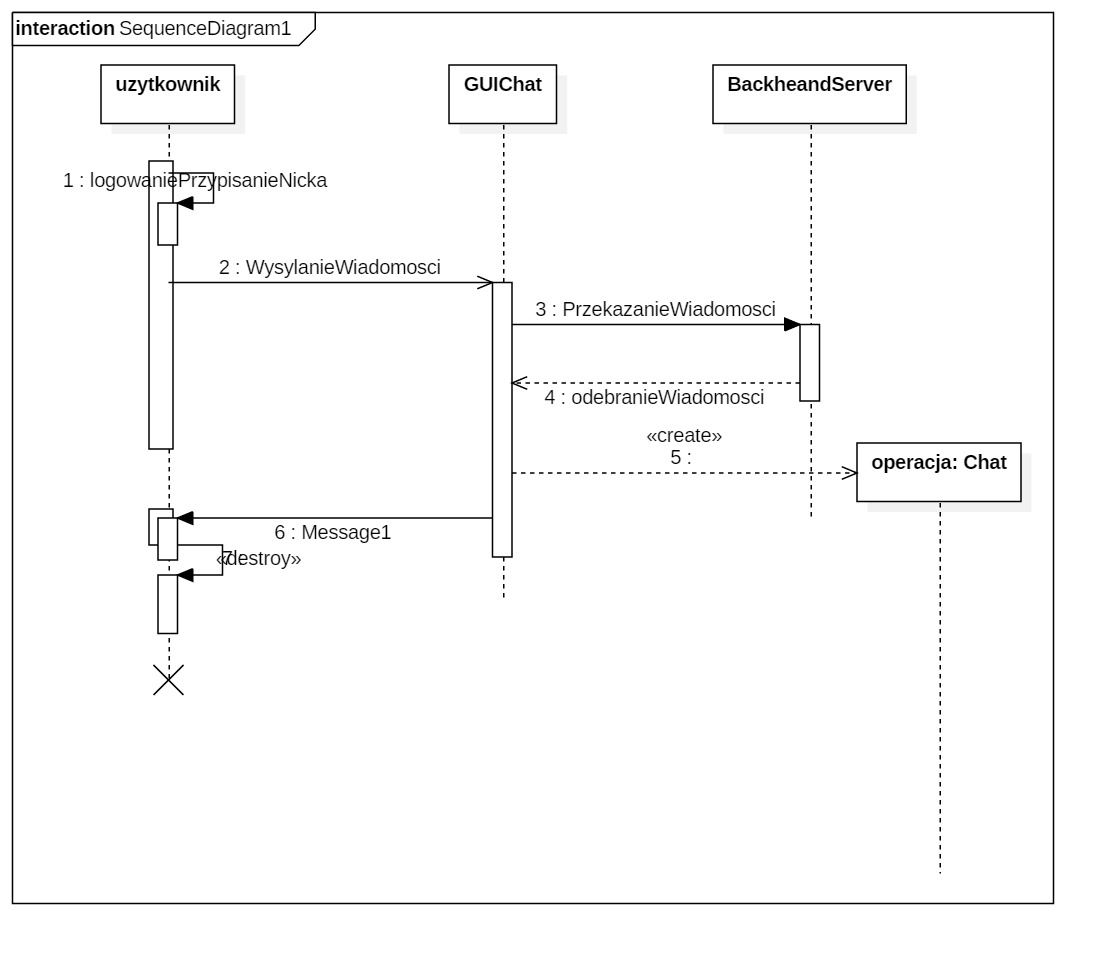
Tabela3: Przypadki użycia czatu

**4.1 Diagram przypadków użycia**



Rys1:Diagram przypadków użycia

**5. Diagram sekwencji**



Rys2: Diagram sekwencji

**6. Technologia**

Aplikacja okienkowa – będzie napisana w języku c# wykorzystując środowisko IDE Microsoft Visual Studio

Aplikacja internetowa – będzie stworzona z wykorzystaniem języka HTML5, CSS oraz JavaScript

Serwer- Node Js, Socket.Io

**7. Dodatek A**

**Podział pracy:**

Opracowanie HTML, CSS, JavaScript - Patryk Chiliński, Adrian Mocianko

Przesyłanie Wiadomości (GET, POST MESSAGES),implementacja ruchy sieciowego –Julian Skowroński, Mateusz Pierzchała

Obsługa serwera-Julian Skowroński

Interfejs graficzny aplikacji okienkowej- Łukasz Zalewski

Formatowanie Tekstu , emotikony – Patryk Chiliński, Adrian Mocianko , Łukasz Zalewski

Sporządzenie dokumentacji – Patryk Chiliński,

Diagramy UML- Adrian Mocianko, Mateusz Pierzchała

**Harmonogram:**

24.03-07.04

1. Sporządzenie dokumentacji ,diagramów przypadków użycia, czynności, sekwencji ,założenia funkcjonale i niefunkcjonalne
2. Uruchomienie i wstępna konfiguracja serwera,
3. Podłączenie i weryfikacja połączenia (Web client-server oraz widowsclent-server)

07.04-21.04

1. Utworzenie wstępnej wersji aplikacji internetowej
2. Implementacja logowania użytkowników
3. Utworzenie wstępnej wersji aplikacji okienkowej

21-04 -12.05

1. Zaawansowana rozbudowa aplikacji internetowej
2. Zaawansowana rozbudowa aplikacji okienkowej
3. Uzupełnienie przesyłanych przez serwer treści
4. Przechowywanie przez serwer zalogowanych użytkowników

12.05-26.05

1. Zespalanie całości projekt
2. Test działania aplikacji.

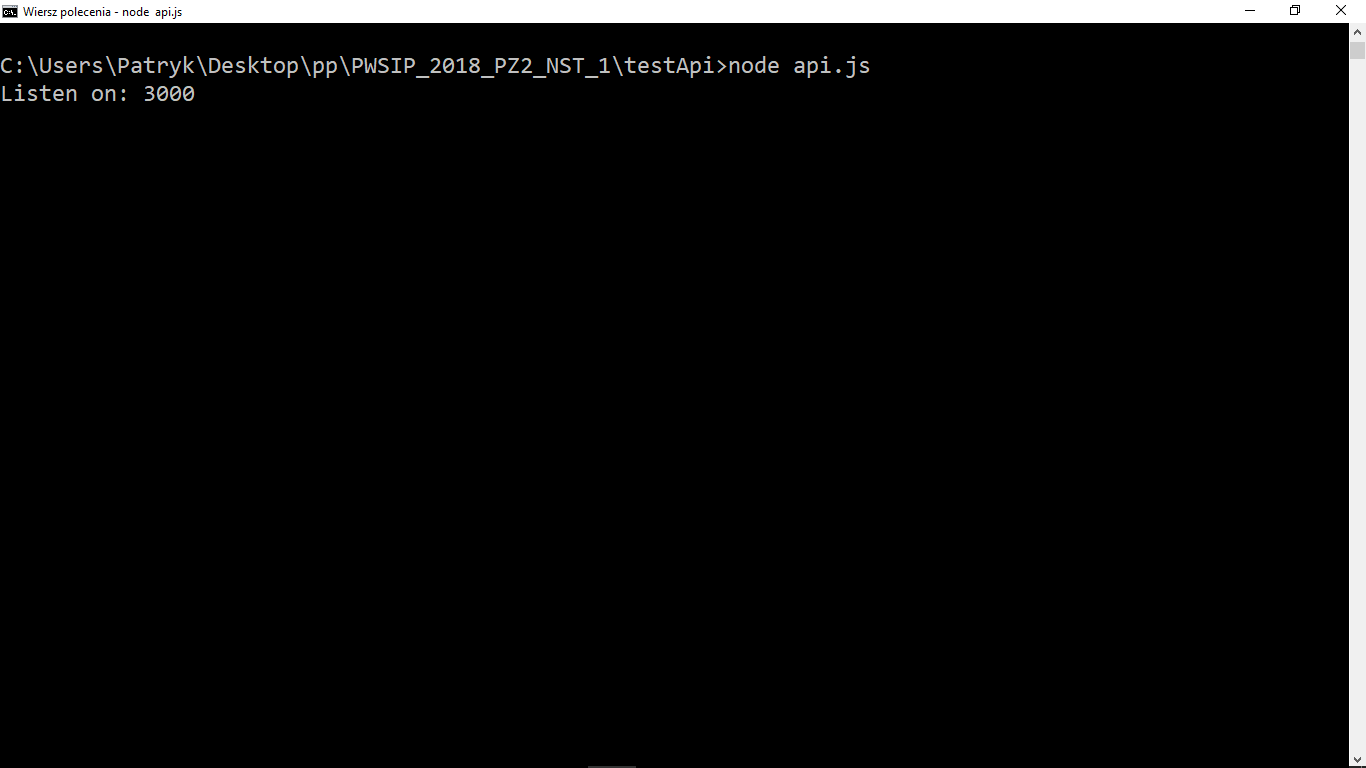
26.05

Finalizacja i Prezentacja Projektu

**Instalacja i konfiguracja:**

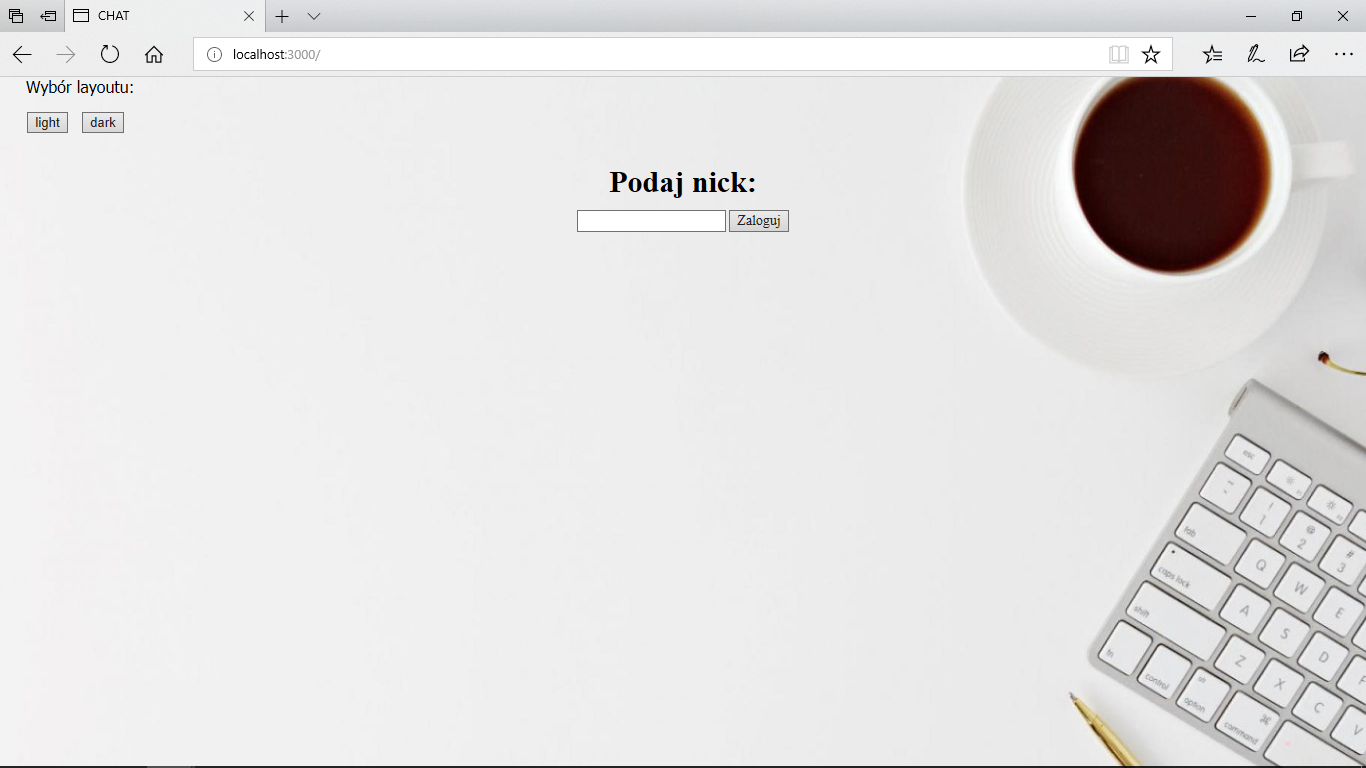
Do uruchomienia klienta aplikacji internetowej wymagany jest Node.js, który można pobrać z witryny nodejs.org.

W celu uruchomienia aplikacji należy uruchomić wiersz polecenia, przejść do katalogu, w którym znajduje się projekt i uruchomić go poleceniem node api.js(nazwa aplikacji)



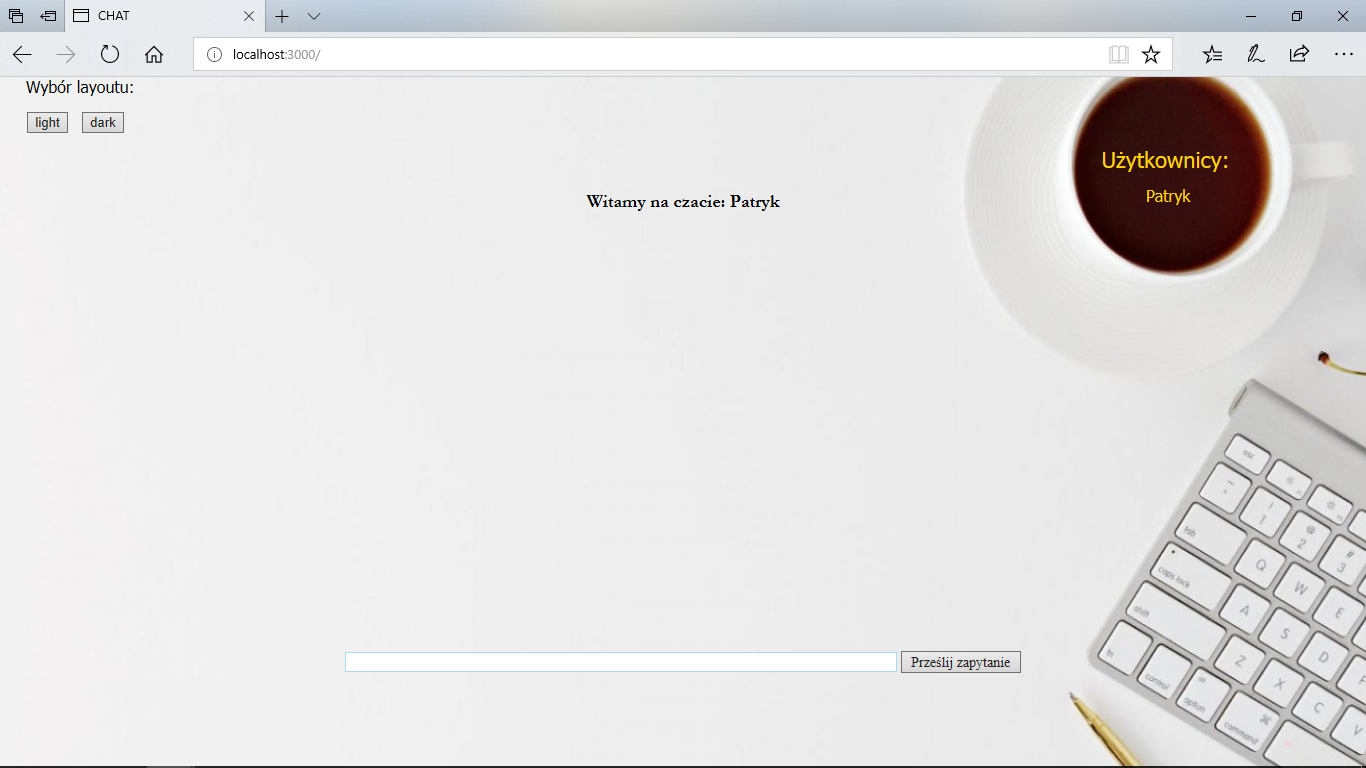
Rys3: Uruchomienie aplikacji internetowej

Następnym krokiem jest uruchomienie przeglądarki internetowej i wpisanie w polu adresowym localhost:3000



Rys4: Strona logowania

Aby zalogować się należy wpisać swój nick.



Rys5: Strona główna aplikacji internetowej

Zalogowany użytkownik może wysyłać wiadomości, wpisując je w pole tekstowe.

**Opis elementów interfejsu graficznego aplikacji internetowej:**



Rys6: Interfejs graficzny aplikacji internetowej