

SRS norma IEEE830

Dla projektu Komunikator Internetowy

Przygotowany przez:

Jan Łęski

Jakub Baliński

Marcin Galas

Piotr Mucha

Patryk Cebrat

11.04.2019

1. Wprowadzenie

1.1. Cel projektu

Aplikacja „Komunikator Internetowy” jest tworzona na potrzeby uzyskania zaliczenia z przedmiotu Inżynieria Oprogramowania.

1.2 Docelowi odbiorcy i zawartość SRS

Główną grupą docelową stanowi zespół tworzący aplikację oraz osoba nadzorująca i oceniająca cały projekt.

Plan SRS:

- rozdział drugi to opis ogólny aplikacji.
- w rozdziale trzecim opisane są sytuacje związane z interfejsem użytkownika.
- rozdział czwarty zawiera informacje dotyczące funkcjonalności i przypadków ich użycia.
- rozdział piąty dotyczy planu pracy oraz informacje o zespole tworzącym aplikację.

1.3 Zakres projektu

„Komunikator Internetowy” będzie aplikacją przeznaczoną na komputery dzięki której, za pośrednictwem łącza internetowego, użytkownicy będą mogli się ze sobą komunikować przy użyciu wiadomości tekstowych.

2. Opis ogólny

2.1. Opis aplikacji

Program będzie pozwalał na prowadzenie konwersacji kilkunastu osobom w tym samym momencie. Zostanie zaimplementowany w formie aplikacji okienkowej z graficznym interfejsem. Przed rozpoczęciem rozmowy użytkownik będzie poproszony o podanie nicku którym będzie się identyfikował podczas konwersacji. Aplikacja będzie dawała możliwość przeglądania historii aktualnej rozmowy oraz pozwalała użytkownikowi wysłanie wiadomości określonej długości. Po wyłączeniu programu konwersacja nie będzie zapisywana. Dodatkowo wszystkie wiadomości będą odpowiednio zaszyfrowane, co uniemożliwi podsłuchanie rozmowy osobom niewskazanym. Łączność pomiędzy osobami korzystającymi z komunikatora będzie ustawiona poprzez sieć i gdy użytkownik straci połączenie aplikacja wyświetli odpowiedni komunikat, a wysyłanie wiadomości nie będzie możliwe dopóki połączenie nie zostanie przywrócone. Komunikator nie przewiduje możliwości zmiany pierwotnie wybranej przez użytkownika nazwy. W oknie rozmowy większą część ekranu aplikacji będzie zajmował obszar wyświetlający konwersację. Będzie on odpowiednio się powiększał gdy wiadomości przestaną się mieścić. Zaraz pod tym obszarem umieszczone zostanie miejsce do wpisywania wiadomości. Bezpośrednio obok pola tekstowego umieszczony zostanie przycisk „Wyślij” służący do wysłania wpisanej wiadomości oraz przycisk „😊” służący do dodania odpowiedniej emotikony.

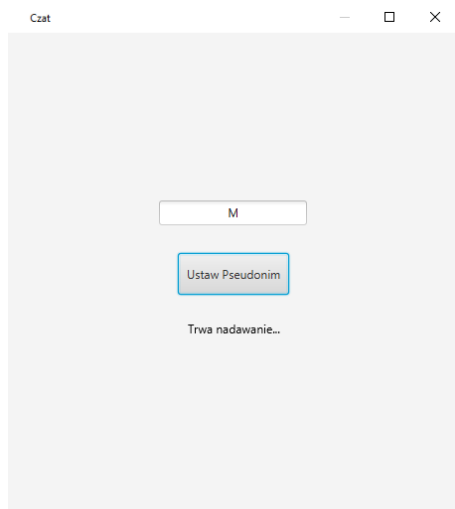
2.2. Wymagania sprzętowe

Do prawidłowego korzystania z aplikacji niezbędna będzie łączność komputera z Internetem oraz system operacyjny Windows.

3. Interfejs użytkownika

3.1. Interfejs użytkownika

Po włączeniu aplikacji użytkownik zostanie poproszony o podanie nicku i pojawi się następujące okienko.



Rys. 1

// Zmiana do Rys. 1 -> Ponieważ aplikacja ładuje się bardzo szybko, bezcelowe jest wyświetlanie „Trwa nadawanie”, więc ta funkcjonalność została pominięta.

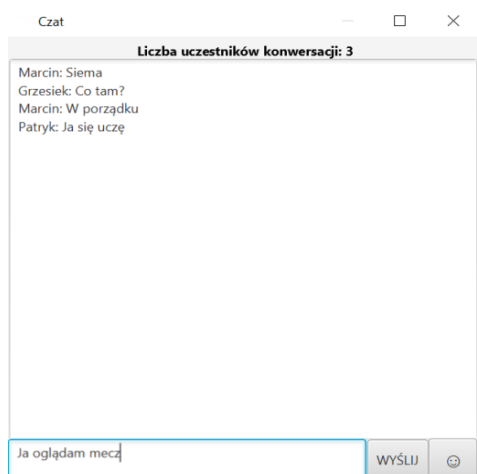
W razie niepowodzenia ukaże się następujący komunikat :



Rys. 2

I użytkownik będzie mógł powrócić do poprzedniego okna (Rys. 1).

Po podaniu odpowiedniego nicku (szczegóły w sekcji drugiej), jeżeli w chatroomie będą znajdowali się inni użytkownicy, użytkownikowi powinno ukazać się następujące okno.

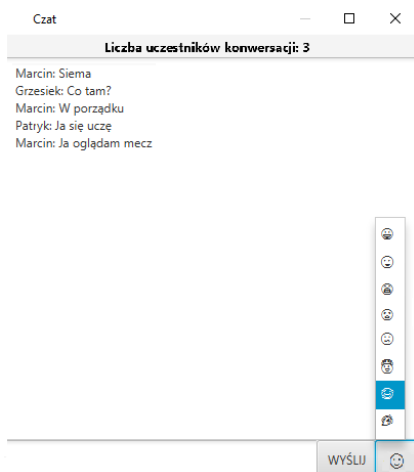


Rys. 3

Natomiast, jeżeli użytkownik będzie pierwszym uczestnikiem konwersacji, licznik uczestników będzie ustawiony na zero, a okno dialogowe będzie puste.

// Zmiana do powyższego -> Jeżeli uczestnik będzie pierwszym uczestnikiem konwersacji licznik zostanie ustawiony na 1 a nie na 0.

Jeżeli użytkownik będzie chciał dodać emotikonę ukaze mu się następujące menu :



Rys. 4

3.2 „To be determined List”

Ponieważ na tym etapie aplikacja jest dopiero tworzona, nie dysponujemy jeszcze wizualizacją wszystkich możliwych sytuacji, tego typu obrazki pojawią się w przyszłości.

4. Funkcjonalności i przypadki ich użycia

4.1. Lista funkcjonalności

1. Obsługa aplikacji przy użyciu graficznego interfejsu użytkownika.¹
2. Wyświetlenie informacji o nawiązaniu połączenia z serwerem² po uruchomieniu aplikacji.
3. Utworzenie chatroomu³ na potrzeby konwersacji.
4. Możliwość ustawienia unikalnego pseudonimu.⁴
5. Możliwość wysyłania wiadomości do innych uczestników konwersacji.
6. Możliwość dodania do wiadomości emotikon tekstowych ze zbioru predefiniowanych.⁵
7. Możliwość dodania do wiadomości hiperlinków.⁶
8. Limit znaków wiadomości.⁷
9. Szyfrowanie wiadomości za pomocą szyfru Cezara.
10. Możliwość odbierania i wyświetlania wiadomości od innych uczestników konwersacji.
11. Dźwięk powiadomienia⁸ informujący o wiadomości przychodzącej.
12. Możliwość otwierania hiperlinków za pomocą domyślnej przeglądarki internetowej.
13. Wyświetlanie licznika aktywnych uczestników⁹ konwersacji.

¹ poglądowy zrzut ekranu został umieszczony na drugiej stronie.

²informacja o nawiązaniu połączenia z serwerem – w przypadku nieudanej próby nawiązania połączenia spowodowanej błędem połączenia sieciowego zostanie wyświetlona informacja o niepowodzeniu.

³chatroom – unikalny zapis stanu konwersacji aktualizowany wraz z wymianą kolejnych wiadomości; tworzony w momencie uruchomienia aplikacji przez pierwszego użytkownika i usuwany z serwera automatycznie po zamknięciu aplikacji przez ostatniego użytkownika; maksymalnie 10 osób jednocześnie może brać udział w konwersacji w ramach chatroomu.

⁴unikalny pseudonim – pseudonim, którego nie posiada w danym momencie żaden użytkownik czatu; na pseudonim musi składać się minimalnie 2 i nie więcej niż 12 znaków.

⁵zbiór predefiniowanych emotikon tekstowych – będzie zawierał co najmniej 8 unikalnych elementów.

⁶hiperlink – w implementacji przyjmujemy, że jest to każdy ciąg znaków dopuszczony przez metodę *isValid(String)* klasy *UrlValidator* pochodzącej z pakietu Java *org.apache.commons.validator.routines*.

⁷limit znaków wiadomości – maksymalna liczba znaków składających się na wysyłaną wiadomość wynosi 280 (włącznie z białymi znakami).

⁸dźwięk powiadomienia – dźwięk można znaleźć pod adresem <https://bit.ly/2UmtlKa>.

⁹licznik aktywnych uczestników konwersacji – wyświetlany nad oknem konwersacji i aktualizowany cyklicznie co 60 sekund; informuje o liczbie osób aktualnie połączonych z serwerem czatu.

4.2. Scenariusze przypadków użycia

(aktorzy: Użytkownik [specyficzni: Użytkownik-Odbiorca, Użytkownik-Nadawca], Serwer; granice systemu wyznacza Aplikacja)

PU1. Obsługa aplikacji za pomocą graficznego interfejsu użytkownika (włączane przez wszystkie PU zawierające opis interakcji Użytkownika z Aplikacją oraz PU13)

Scenariusz główny

1. Użytkownik uruchamia Aplikację.
2. Aplikacja wyświetla na ekranie interfejs graficzny przeznaczony do jej obsługi.
3. Użytkownik przy pomocy klawiatury i myszy, za pośrednictwem interfejsu graficznego, prowadzi interakcję z Aplikacją.

PU2. Wyświetlenie informacji o poprawnym nawiązaniu połączenia z serwerem

Scenariusz główny

1. Aplikacja (po uruchomieniu) nawiązuje połączenie z Serwerem.
2. Aplikacja wyświetla informację o poprawnym nawiązaniu połączenia w postaci okienka dialogowego z informacją „*Pomyślnie nawiązano połączenie z serwerem.*”

// Zmiana do 2. -> Dla zwiększenia czytelności wiadomość „*Pomyślnie nawiązano połączenie z serwerem.*” Jest wyświetlana w tym samym okienku, a nie w nowym.

Scenariusz alternatywny: Nie udało się nawiązać połączenia z serwerem

1. Aplikacja (po uruchomieniu) dokonuje nieudanej próby nawiązania połączenia z Serwerem.

2. Aplikacja wyświetla informację o niepowodzeniu próby nawiązania połączenia w postaci okienka dialogowego z informacją „*Nie udało się nawiązać połączenia z serwerem. Sprawdź swoje połączenie internetowe.*”

3. Użytkownik zatwierdza informację poprzez wciśnięcie przycisku „OK” lub zamknięcie okienka dialogowego.

4. Aplikacja wyłącza się.

PU3. Utworzenie chatroomu na potrzeby konwersacji

Scenariusz główny

1. Serwer (po pierwszej udanej próbie nawiązania połączenia przez Aplikację) sprawdza, że chatroom nie został jeszcze utworzony.

2. Serwer tworzy chatroom.

3. Serwer przydziela Aplikację do utworzonego chatroomu.

Scenariusz alternatywny 1: Chatroom już istnieje i limit aktywnych uczestników konwersacji nie został osiągnięty

1. Serwer (po pierwszej udanej próbie nawiązania połączenia przez Aplikację) sprawdza, że chatroom został już utworzony.

2. Serwer sprawdza, że limit aktywnych uczestników konwersacji nie został osiągnięty.

3. Serwer przydziela Aplikację do istniejącego chatroomu.

Scenariusz alternatywny 2: Chatroom już istnieje i limit aktywnych uczestników konwersacji został osiągnięty

1. Serwer (po pierwszej udanej próbie nawiązania połączenia przez Aplikację) sprawdza, że chatroom został już utworzony.

2. Serwer sprawdza, że limit aktywnych uczestników konwersacji został osiągnięty.

3. Serwer informuje Aplikację o wypełnionym limicie uczestników konwersacji.

4. Aplikacja wyświetla informację o niepowodzeniu próby dołączenia do chatroomu w postaci okienka dialogowego z informacją „*Nie udało się dołączyć do chatroomu – limit uczestników konwersacji został osiągnięty. Spróbuj ponownie później.*”

5. Użytkownik zatwierdza informację poprzez wciśnięcie przycisku „OK” lub zamknięcie okienka dialogowego.

6. Aplikacja wyłącza się.

//Zmiana w punkcie 6. -> Aplikacja się nie wyłącza, tylko powraca do okienka umożliwiającego ponowne podanie pseudonimu.

PU4. Ustawienie pseudonimu

Scenariusz główny

1. Serwer (po przydzieleniu Aplikacji do chatroomu) wysyła do Aplikacji zapytanie o pseudonim
2. Aplikacja wyświetla okienko dialogowe z prośbą o podanie pseudonimu.
3. Użytkownik wpisuje pseudonim w odpowiednie pole tekstowe.
4. Aplikacja sprawdza, że pseudonim zawiera prawidłową liczbę znaków.
5. Aplikacja wysyła informację o wybranym przez Użytkownika pseudonimie do Serwera.
6. Serwer sprawdza, że pseudonim nie jest aktualnie używany w konwersacji.
7. Serwer przypisuje pseudonim do Aplikacji.
8. Serwer powiadamia Aplikację o udanym przypisaniu pseudonimu.

Scenariusz alternatywny 1: Nieprawidłowa liczba znaków w pseudonimie

- 1-3. Jak w scenariuszu głównym.
4. Aplikacja sprawdza, że pseudonim zawiera nieprawidłową liczbę znaków.
5. Aplikacja wyświetla informację o nieprawidłowej liczbie znaków w pseudonimie w postaci okienka dialogowego z informacją „*Nie udało się ustawić pseudonimu. Liczba znaków musi być nie mniejsza od 2 i nie większa od 12.*”
6. Użytkownik zatwierdza informację poprzez wciśnięcie przycisku „OK” lub zamknięcie okienka dialogowego i następuje powrót do kroku 2. scenariusza głównego.

Scenariusz alternatywny 2: Pseudonim jest aktualnie używany

- 1-5. Jak w scenariuszu głównym.
6. Serwer sprawdza, że pseudonim jest aktualnie używany w konwersacji.
7. Serwer powiadamia Aplikację, że podany pseudonim jest już zajęty.
8. Aplikacja wyświetla informację o niedostępności wybranego pseudonimu w postaci okienka dialogowego z informacją „*Nie udało się ustawić pseudonimu. Wybrany pseudonim jest już zajęty.*”
9. Użytkownik zatwierdza informację poprzez wciśnięcie przycisku „OK” lub zamknięcie okienka dialogowego i następuje powrót do kroku 2. scenariusza głównego.

PU5. Wysłanie wiadomości do innych uczestników konwersacji (zawiera PU8, PU9 – kroki 3., 4.; możliwe rozszerzenie przez PU6, PU7, PU11 – odpowiednio kroki 1., 1., 7.)

Scenariusz główny

1. Użytkownik-Nadawca wpisuje treść wiadomości w pole tekstowe przeznaczone do tworzenia wiadomości.
2. Użytkownik-Nadawca akceptuje treść wiadomości klikając przycisk „WYŚLIJ”
3. Aplikacja sprawdza, że limit długości wiadomości nie został przekroczony.
4. Aplikacja szyfruje kopię wiadomości.
5. Aplikacja wysyła zaszyfrowaną kopię wiadomości do Serwera.
6. Serwer odbiera zaszyfrowaną wiadomość i rozsyła ją do Użytkowników-Odbiorców.
7. Aplikacja sprawdza oryginalną wersję wiadomości pod kątem zawartości hiperlinków.
8. Aplikacja wyświetla wiadomość w oknie konwersacji Użytkownika-Nadawcy w postaci jego pseudonimu, dwukropka, spacji i następującej po tej sekwencji (sformatowanej, jeśli zawierała hiperlinki) treści wysłanej wiadomości.

// Zmiana do 2. -> Nadawca może zaakceptować treść wiadomości nie tylko klikając przycisk „WYŚLIJ”, ale również za pomocą przycisku enter.

Scenariusz alternatywny: Zerwanie połączenia internetowego

- 1-4. Jak w scenariuszu głównym.
5. Aplikacja wyświetla w okienku dialogowym komunikat „*Utracono połączenie z serwerem. Sprawdź swoje połączenie internetowe.*” i anuluje wysyłanie wiadomości.
6. Aplikacja pozostawia dotychczas wpisaną w polu tekstowym treść wiadomości w niezmienionej formie.

PU6. Dodanie do wiadomości emotikony tekstowej ze zbioru predefiniowanych (rozszerza krok 1. PU5)

Scenariusz główny

1. Użytkownik-Nadawca naciska przycisk opatrzony znakiem „😊”.
2. Aplikacja wyświetla menu kontekstowe z listą dostępnych emotikon tekstowych.
3. Użytkownik-Nadawca wybiera za pomocą kliknięcia myszki interesującą go emotikonę.
4. Aplikacja dodaje do aktualnej treści wiadomości ciąg znaków składający się na emotikonę.

Scenariusz alternatywny: Użytkownik-Nadawca rezygnuje z dodania emotikony

1-2. Jak w scenariuszu głównym.

3. Użytkownik-Nadawca naciska za pomocą myszki w dowolnym miejscu okna programu poza menu kontekstowym.

4. Aplikacja chowa menu kontekstowe z wyborem emotikony.

PU7. Dodanie do wiadomości hiperlinku (rozszerza krok 1. PU5)

Scenariusz główny

1. Użytkownik-Nadawca wpisuje lub wkleja ze schowka hiperlink w pole tekstowe przeznaczone do tworzenia wiadomości.

PU8. Sprawdzenie limitu znaków w wiadomości (włączane przez PU5)

Scenariusz główny

1. Aplikacja (przed zaszyfrowaniem) sprawdza, że długość wiadomości nie przekroczyła 280 znaków.

Scenariusz alternatywny: Przekroczony limit znaków

1. Aplikacja (przed zaszyfrowaniem) sprawdza, że długość wiadomości przekroczyła limit 280 znaków.
2. Aplikacja wyświetla w okienku dialogowym komunikat „*Nie udało się wysłać wiadomości. Przekroczono limit znaków (280).*” i anuluje wysyłanie wiadomości.
3. Aplikacja pozostawia dotychczas wpisaną w polu tekstowym treść wiadomości w niezmienionej formie.

PU9. Szyfrowanie wiadomości (włączane przez PU5)

Scenariusz główny

1. Aplikacja zamienia ciąg znaków składający się na wysyłaną wiadomość na inny za pomocą szyfru Cezara.

PU10. Odebranie i wyświetlenie wiadomości od innego uczestnika konwersacji (zawiera PU12 – krok 5.; możliwe rozszerzenie kroku 4. przez PU11)

Scenariusz główny

1. Serwer wysyła do Aplikacji informację o nowej wiadomości wysłanej przez pewnego Użytkownika-Nadawcę wraz z jej zaszyfrowaną treścią.

2. Aplikacja odszyfrowuje wiadomość.
3. Aplikacja przeszukuje wiadomość pod kątem zawartości hiperlinków.
4. Aplikacja wyświetla wiadomość na ekranie konwersacji w postaci pseudonimu Użytkownika-Nadawcy, dwukropka, spacji i następującej po tej sekwencji treści otrzymanej wiadomości.
5. Aplikacja odtwarza dźwięk powiadomienia.

PU11. Wyświetlenie i otwarcie hiperlinku (rozszerza PU5, PU10 – odpowiednio kroki 7., 4.)

Scenariusz główny

1. Aplikacja formatuje wszystkie zawarte w wyświetlanej wiadomości hiperlinki poprzez nadanie im koloru niebieskiego (#0066CC) i podkreślenie ich.
2. Aplikacja wyświetla sformatowaną treść wiadomości.
3. Użytkownik za pomocą myszki klika na hiperlink.
4. Aplikacja wywołuje domyślną przeglądarkę Użytkownika przekazując jej hiperlink do otwarcia.

Scenariusz alternatywny: Użytkownik-Odbiorca nie ma zainstalowanej przeglądarki internetowej lub żadna z zainstalowanych na jego komputerze przeglądarek internetowych nie jest przeglądarką domyślną

1-3. Jak w scenariuszu głównym.

4. Aplikacja wyświetla w okienku dialogowym komunikat *„Nie udało otworzyć hiperlinku. Sprawdź czy masz zainstalowaną przeglądarkę internetową. Sprawdź czy jedna z zainstalowanych przeglądarek jest przeglądarką domyślną.”*

PU12. Odtworzenie dźwięku informującego o wiadomości przychodzącej (włączane przez PU10)

Scenariusz główny

1. Aplikacja po otrzymaniu i wyświetleniu wiadomości odtwarza dźwięk powiadomienia.

PU13. Wyświetlanie licznika uczestników konwersacji

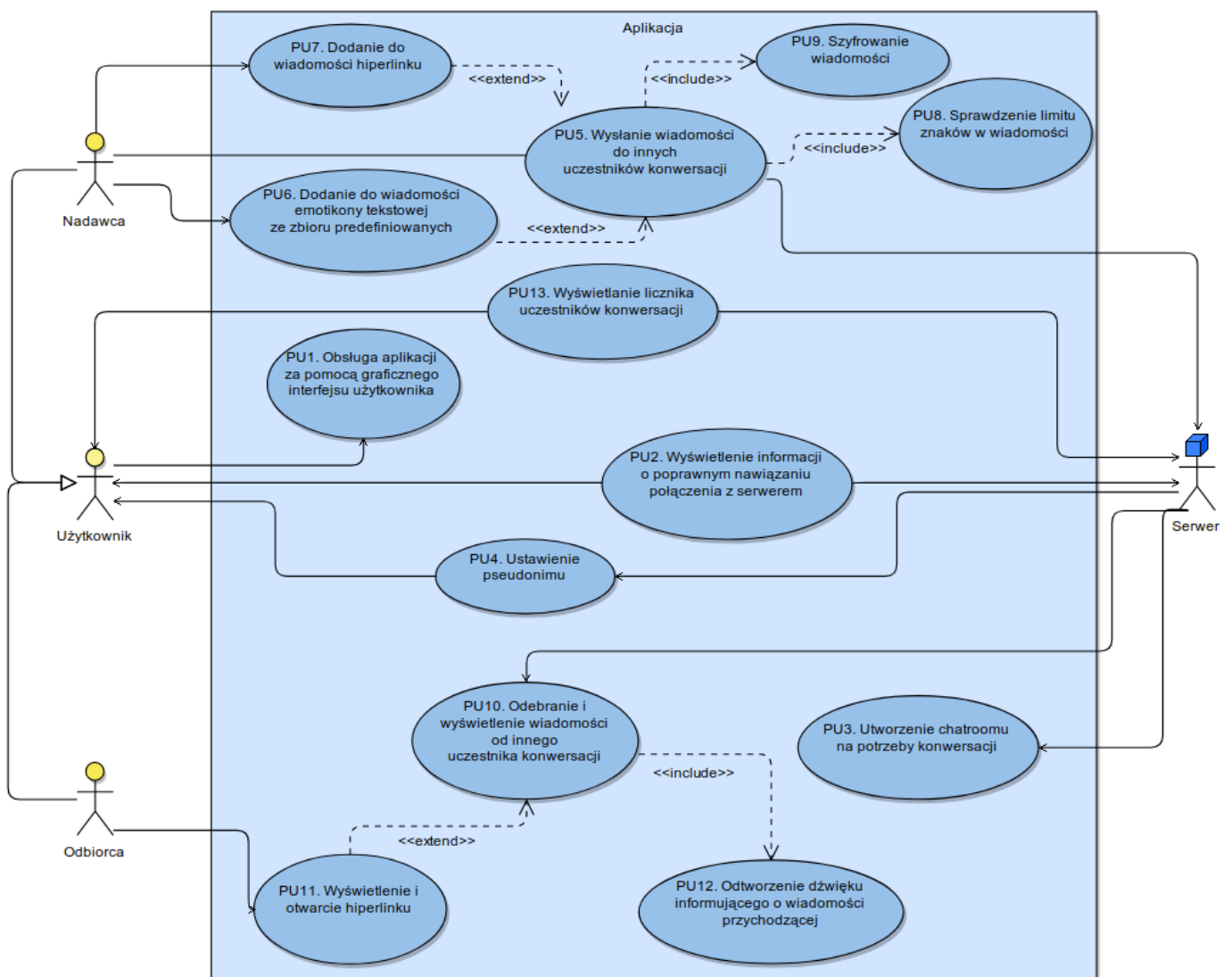
Scenariusz główny

1. Aplikacja odpytuje Serwer o liczbę połączonych Użytkowników.
2. Serwer wysyła do Aplikacji informację o liczbie aktualnie połączonych Użytkowników.

3. Aplikacja odświeża wyświetlaną nad oknem konwersacji informację w postaci napisu „*Liczba uczestników konwersacji:*” i następującej po nim liczbie aktualnie połączonych z Serwerem Użytkowników.

4. Po 60 sekundach następuje powrót do kroku 1.

Wszystkie opisane powyżej scenariusze przypadków są zobrazowane na poniższym wykresie:



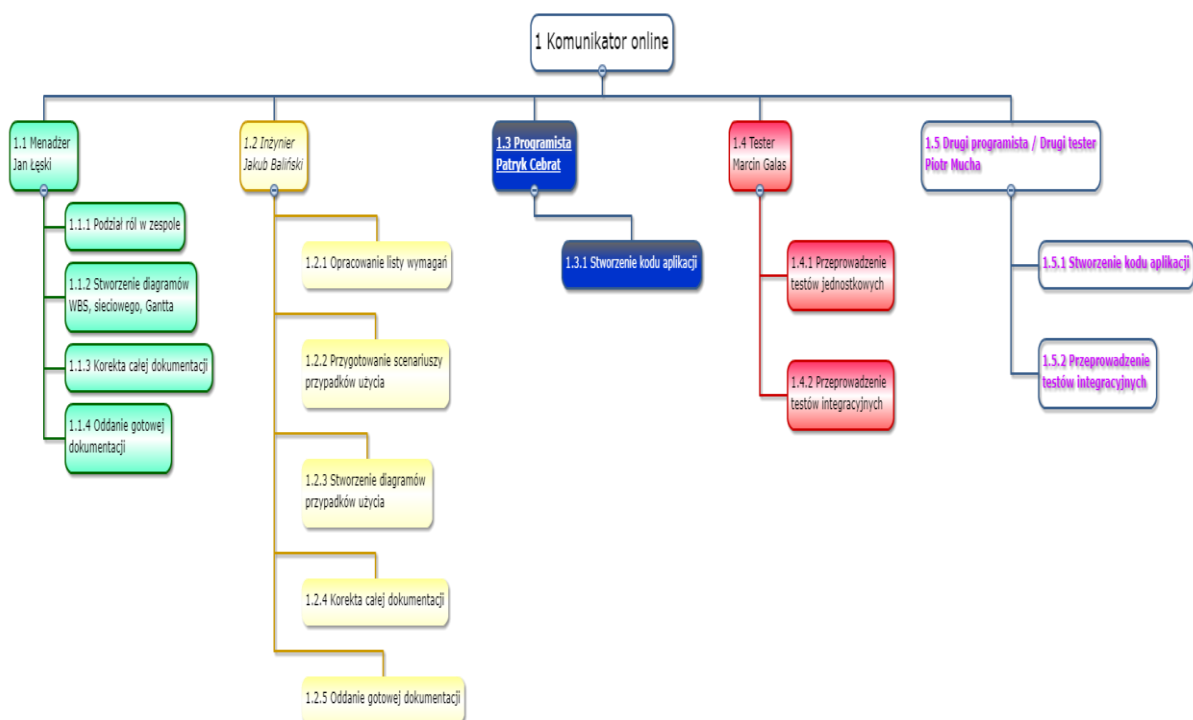
5. Plan realizacji aplikacji

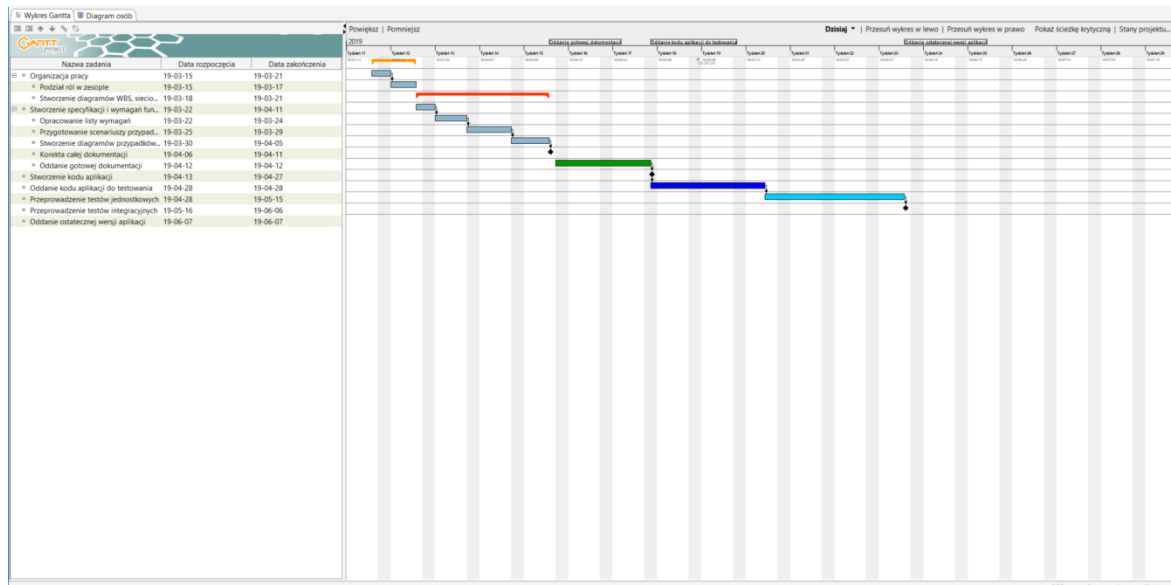
5.1. Skład zespołu:

- Jan Łęski – menadżer projektu
- Jakub Baliński – inżynier
- Patryk Cebirat - programista
- Marcin Galas – tester
- Piotr Mucha – drugi programista / drugi tester

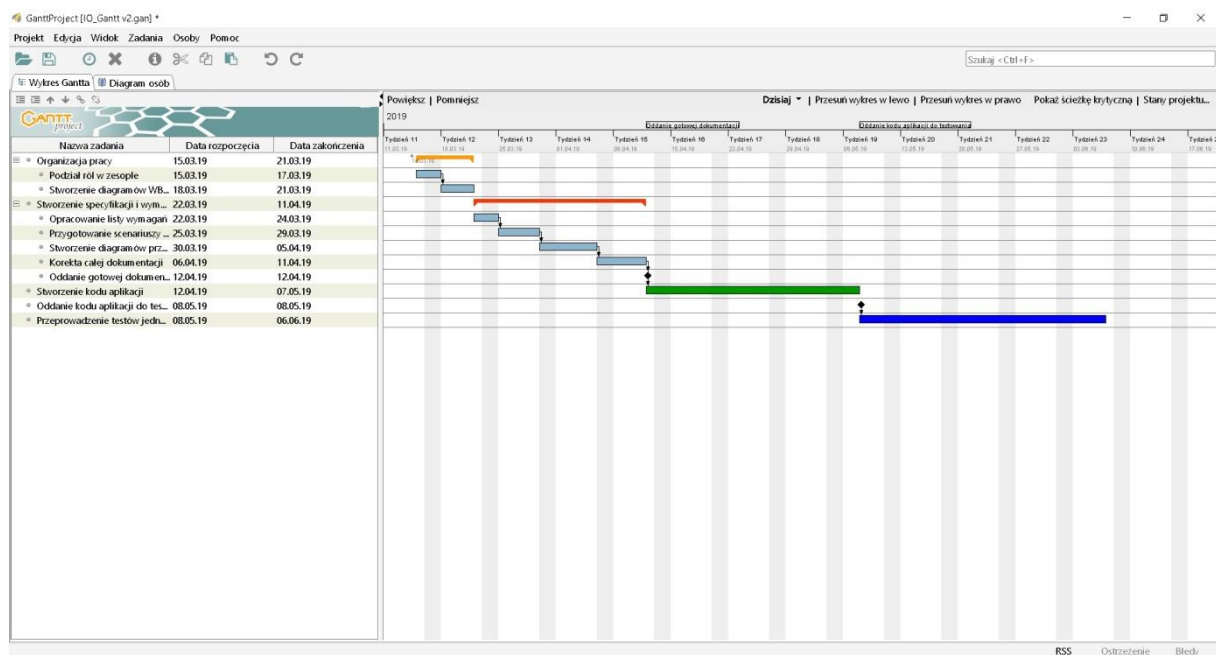
5.2. Plan pracy

Szczegółowy podział zadań oraz rozplanowanie ich w czasie przedstawione są na poniższych diagramach (WBS oraz Gantt).





Druga wersja wykresu Gantta:



// Zmiany w wykresie

Z racji na charakter projektu i rezygnację z testów integracyjnych czas przeznaczony na owe testy został rozdysponowany na tworzenie kodu aplikacji oraz testowanie systemowe i jednostkowe.