Programowanie Sieciowe

# Serwer do obsługi usługi IRC

Z51. Demonstracja działania

Maciej Bogusławski (331362)
Hubert Kaczyński (331386)
Bartosz Żelazko (331457)

# Spis treści

# Wstęp

W tym dokumencie znajduje się opis działania funkcjonalności związanych z działaniem serwera IRC w ramach projektu z przedmiotu programowanie sieciowe. Ze względu na fakt, iż prace naszego zespołu udało się przyspieszyć ukazane zostało działanie wszystkich funkcji opisanych w dokumentacji wstępnej projektu.

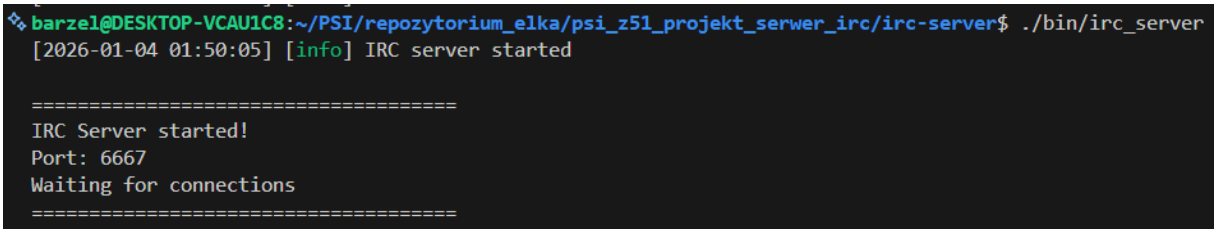
W celu wizualizacji działania użyto narzędzia weechat które jest zewnętrzną implementacją klienta IRC. Dodatkowo w tym dokumencie znajdują się zrzuty ekranu terminala, na których widoczne są logi generowane przez bibliotekę spdlog.

# Prezentacja działania

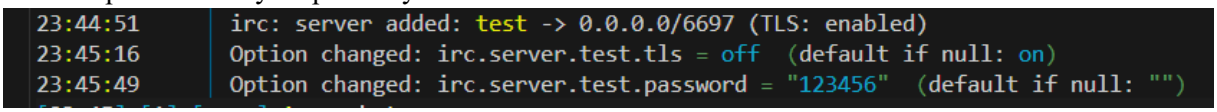
Aby zaprezentować działanie programu postanowiliśmy posłużyć się przypadkami użycia przedstawionymi w punkcie pod tytułem „Przypadki użycia" w dokumentacji wstępnej. W celu uzyskania większej czytelności w logach z serwera uwzględniono jedynie logi z poziomem info i wyższym.

# Otwarcie połączenia klient- serwer

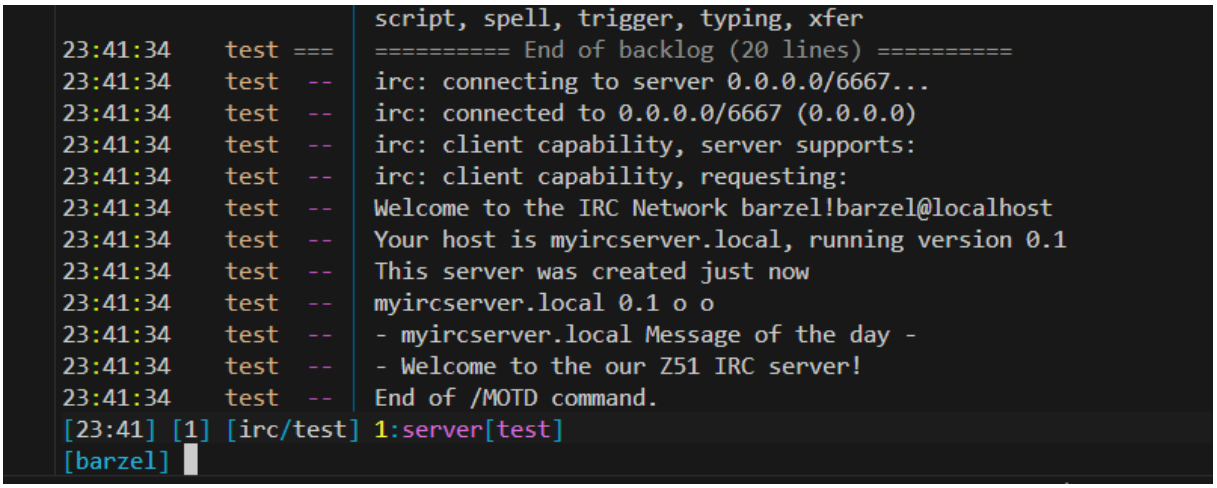
Na samym początku pracy z naszym programem uruchamiany jest serwer IRC, który po uruchomieniu loguje fakt, iż rozpoczął on pracę oraz wypisuje na terminal parametry z jakimi pracuje. Poniżej zamieszczony został zrzut ekranu przedstawiający elementy wypisane na terminal chwilę po uruchomieniu serwera IRC.



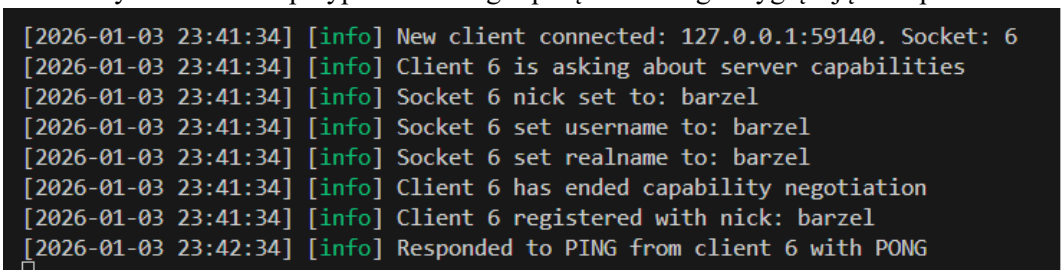

Następnie w ramach demonstracji uruchomiony został program weechat, aby móc rozpocząć pracę z serwerem. Na samym początku pracy należy skonfigurować parametry takie jak adres IP i port na jakim działa serwer, wyłączenie TLS oraz ustawienie hasła dla serwera. Przykładowy proces konfiguracji został zaprezentowany na poniższym zrzucie ekranu.




Po dokonaniu wstępnej konfiguracji w narzędziu weechat możemy zainicjalizować połączenie z serwerem IRC poprzez wykonanie polecenia */connect <nazwa serwera>* w weechat. Komenda ta powoduje nawiązanie połączenia poprzez wysłanie widomości CAP PASS USER oraz NICK. Działanie tej komendy z poziomu weechat pokazuje zdjęcie na następnej stronie.




Od strony serwera w przypadku takiego połączenia logi wyglądają w sposób ukazany poniżej.

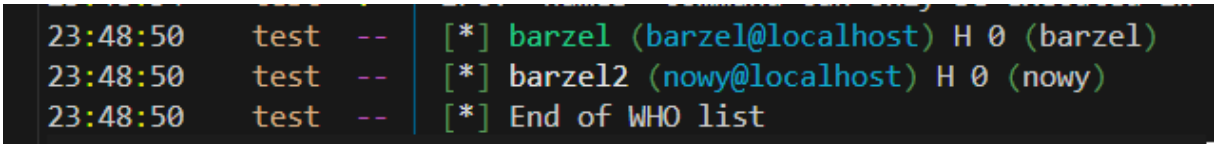



W przypadku w którym występuje kolizja nazw użytkowników i/lub nicków serwer podczas próby zalogowania odpowie odpowiednim błędem. Błąd ten jest poprawnie interpretowany przez klienta weechat, a on wyświetla użytkownikowi następujące wiadomości.

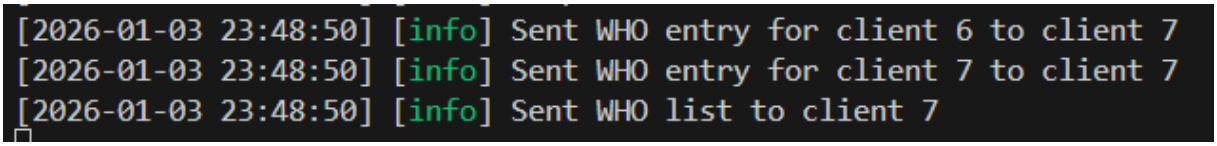

Warto zwrócić uwagę na fakt, iż weechat, jako dość zaawansowany klient, po otrzymaniu błędu mówiącego o nieoryginalnym nickname sam zaproponuje nowy nickname. Jak widać na powyższym zrzucie ekranu, zaproponowany nowy nickname już jest oryginalny i udało się połączyć z serwerem. Od strony serwera w takiej sytuacji logowane są przedstawione powyżej zdarzenia.

## Wysłanie wiadomości pomiędzy dwoma klientami

Po poprawnym połączeniu się z serwerem możemy wysyłać wiadomości prywatne pomiędzy użytkownikami. Na samym początku, aby dowiedzieć się, jacy użytkownicy znajdują się już w sieci, możemy skorzystać z komendy */who*. Komenda ta służy do wyszukiwania użytkowników. Przykładowy wynik użycia tej komendy z poziomu klienta weechat znajduje się poniżej.

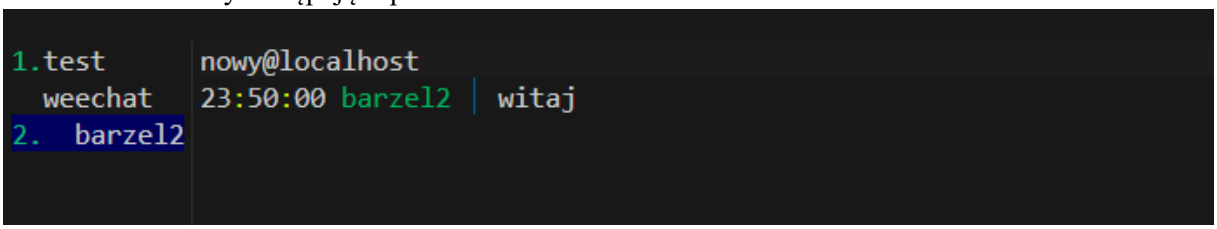



W trakcie obsługi tego zapytania serwer generuje następujące logi.

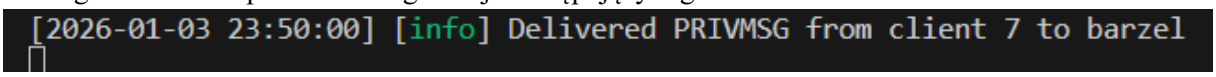



Wiedząc już jacy klienci znajdują się już w naszej sieci, korzystając z komendy /msg <użytkownik> <treść wiadomości> możemy wysłać komunikat PRIVMSG, który poskutkuje wysłaniem wiadomości prywatnej do wyspecyfikowanego użytkownika. Klient wysyłający wiadomości zobaczy następujący komunikat w weechat.


A odbiorca zobaczy następujące powiadomienie:

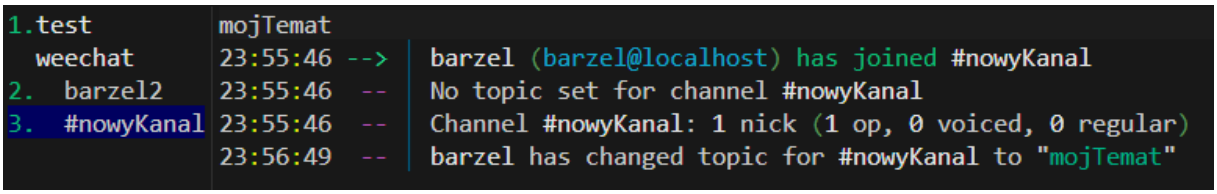



Obsługa PRIVMSG przez serwer generuje następujący log:

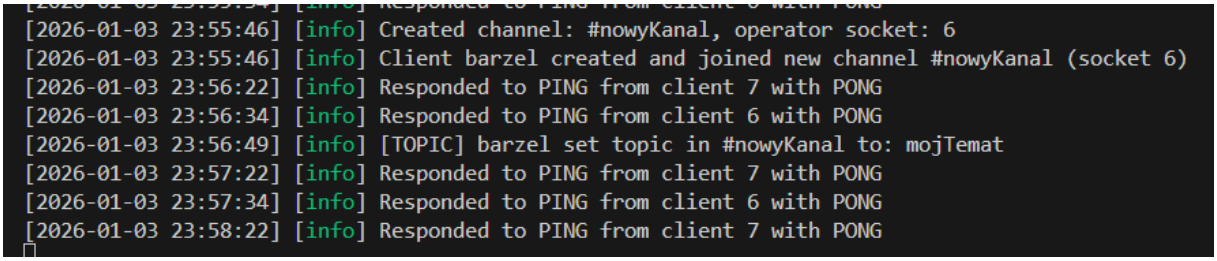



## Stworzenie kanału

Aby stworzyć kanał należy wykonać operację */join* gdzie jako argument podana jest nazwa nieistniejącego kanału. Wysłanie takiego komunikatu do serwera spowoduje stworzenie nowego kanału a osoba tworząca z automatu zostanie administratorem tego kanału. W weechat ten fakt jest symbolizowany przez znak @ przed nickiem użytkownika. Dodatkowo po stworzeniu kanału możemy skorzystać z komendy */topic* aby nadać temat dla danego kanału. Wykonanie obu tych komend widziane z poziomu klienta zostało zaprezentowane na następnej stronie.

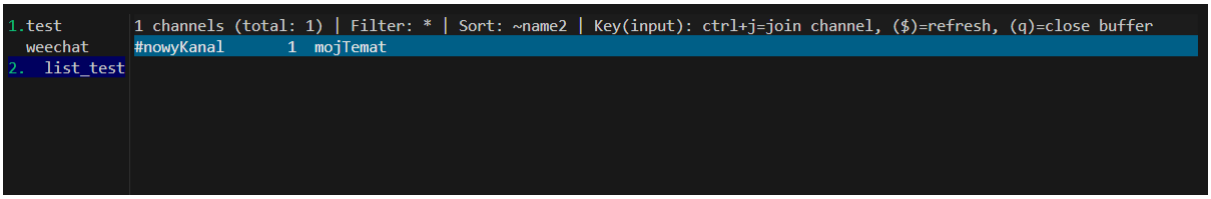



Operacje te po stronie serwera pozostawiają następujące logi.

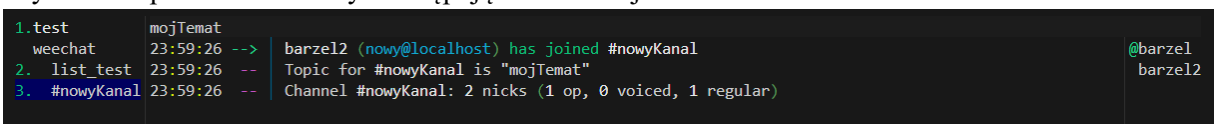



## Dołączenie do kanału

Aby otrzymać listę kanałów dostępnych na serwerze należy wysłać wiadomość LIST. Na tą wiadomość serwer powinien odpowiedzieć wysyłając listę wszystkich kanałów wraz z ich tematami. Przykładowa wizualizacja takiej listy wygenerowana z poziomu weechat znajduje się poniżej.




Następnie możemy dołączyć do kanału poprzez wysłanie wiadomości JOIN. Po dołączeniu do kanału użytkownik powinien zobaczyć następujące informacje.



## Wysłanie wiadomości do kanału

Wysłanie wiadomości na kanał odbywa się poprzez wysłanie PRIVMSG gdzie zamiast nazwy użytkownika podawana jest nazwa kanału. Przykład wysłania wiadomości użytkowników na kanał został zamieszczony poniżej.

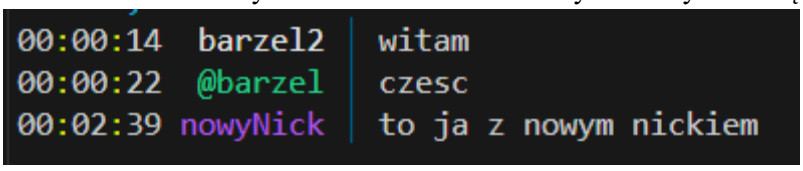

Serwer natomiast w takiej sytuacji powinien wygenerować następujące logi:

```
[2026-01-04 00:00:14] [info] Delivered PRIVMSG from client 7 to channel #nowyKanal (1 members)
[2026-01-04 00:00:22] [info] Delivered PRIVMSG from client 6 to channel #nowyKanal (1 members)
```

Na tym etapie postanowiliśmy także zaprezentować jak wygląda zmiana nazwy użytkownika widocznej dla innych. Aby ją zmienić należy wykonać wysłać wiadomość NICK. Wysłanie takiej wiadomości zostanie zalogowane przez serwer w następujący sposób.

```
[2026-01-04 00:01:49] [info] Socket 6 nick set to: nowyNick
```

Po zmianie nicku i wysłaniu wiadomości inni użytkownicy zobaczą zmianę nicków.



# Opuszczenie kanału

Celem opuszczenia kanału należy wysłać do serwera wiadomość PART jako argument podając jako argument nazwę kanału z którego chcemy wyjść. W takiej sytuacji klient weechat powinien zaprezentować następujący komunikat.

```
00:04:08      <--  barzel2 (nowy@localhost) has left #nowyKanal (WeeChat 4.1.1)
```

Serwer powinien zalogować następującą informację:

```
[2026-01-04 00:04:08] [info] [PART] Client barzel2 is leaving channel #nowyKanal - reason: WeeChat 4.1.1
[2026-01-04 00:04:08] [info] Socket 7 left channel #nowyKanal
```

Po opuszczeniu kanału nie możemy już wysyłać w nim wiadomości. Próba wysłania wiadomości do serwera, który został przez nas już opuszczony skończy się zwróceniem przez serwer odpowiedniego błędu . Błąd ten weechat wyświetla w sposób następujący:

```
00:04:08      <--  barzel2 (nowy@localhost) has left #nowyKanal (WeeChat 4.1.1)
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
00:06:09  barzel2  abc
00:06:09       --  #nowyKanal: Cannot send to channel
```

W takim przypadku serwer stworzy następujący log:

```
[2026-01-04 00:06:09] [warning] PRIVMSG command from client 7: not in channel #nowyKanal
```

# Usunięcie użytkownika z kanału

Aby usunąć użytkownika z kanału administrator może wykorzystać komendę */kick.* Wysyła ona do serwera wiadomość o tej samej nazwie która w przypadku podania odpowiednich parametrów poskutkuje usunięciem użytkownika. Udane wykonanie tej komendy zostało zaprezentowane poniżej.

```
00:21:30      <--  nowyNick has kicked barzel2 (nowyNick)
```

W tym przypadku w nawiasie przedstawiony jest nick administratora, który zadecydował o wyrzuceniu użytkownika. W takiej sytuacji serwer generuje poniższe logi.

```
[2026-01-04 00:21:30] [info] [KICK] nowyNick is kicking barzel2 from #nowyKanal - reason: nowyNick
[2026-01-04 00:21:30] [info] Socket 6 kicked socket 7 from channel #nowyKanal
```

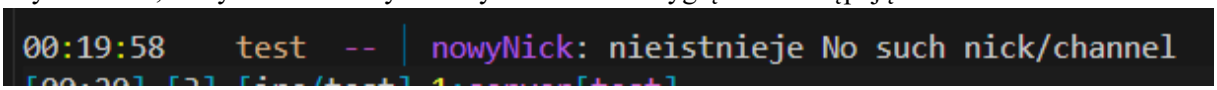
W przypadku gdy użytkownik bez praw administratora postanowi wykonać komendę *kick*, serwer nie pozwoli mu na to i odeśle kod błędu który jest poprawnie interpretowany przez weechat.

```
00:11:44       --  #nowyKanal: You're not channel operator
```

Serwer natomiast w takiej sytuacji stworzy następujący log:

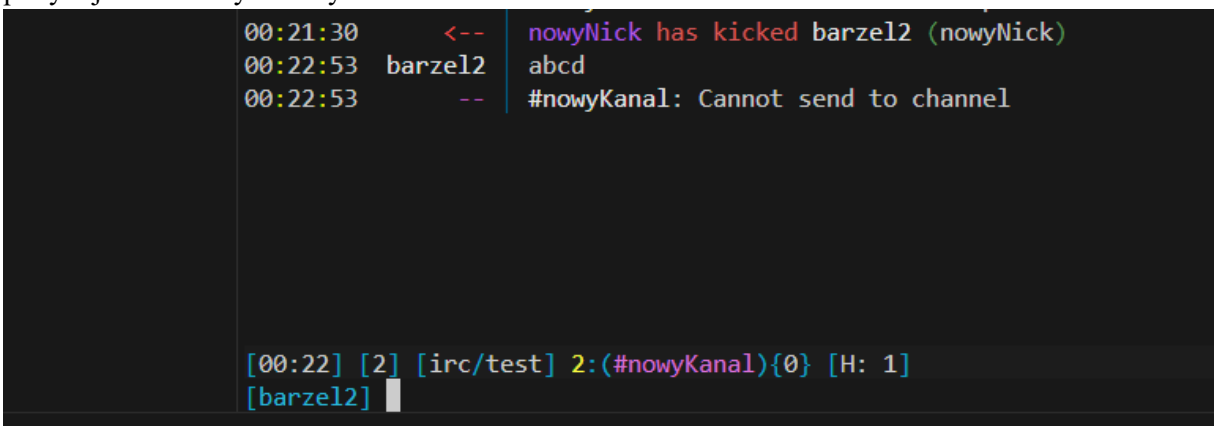
```
[2026-01-04 00:11:44] [info] [KICK] Client barzel2 is not channel operator in #nowyKanal
```

Podobnie sprawa będzie wyglądać gdy administrator poda niepoprawną lub nieistniejącą nazwę użytkownika, który ma zostać wyrzucony. W weechat wygląda to następująco:




Server:

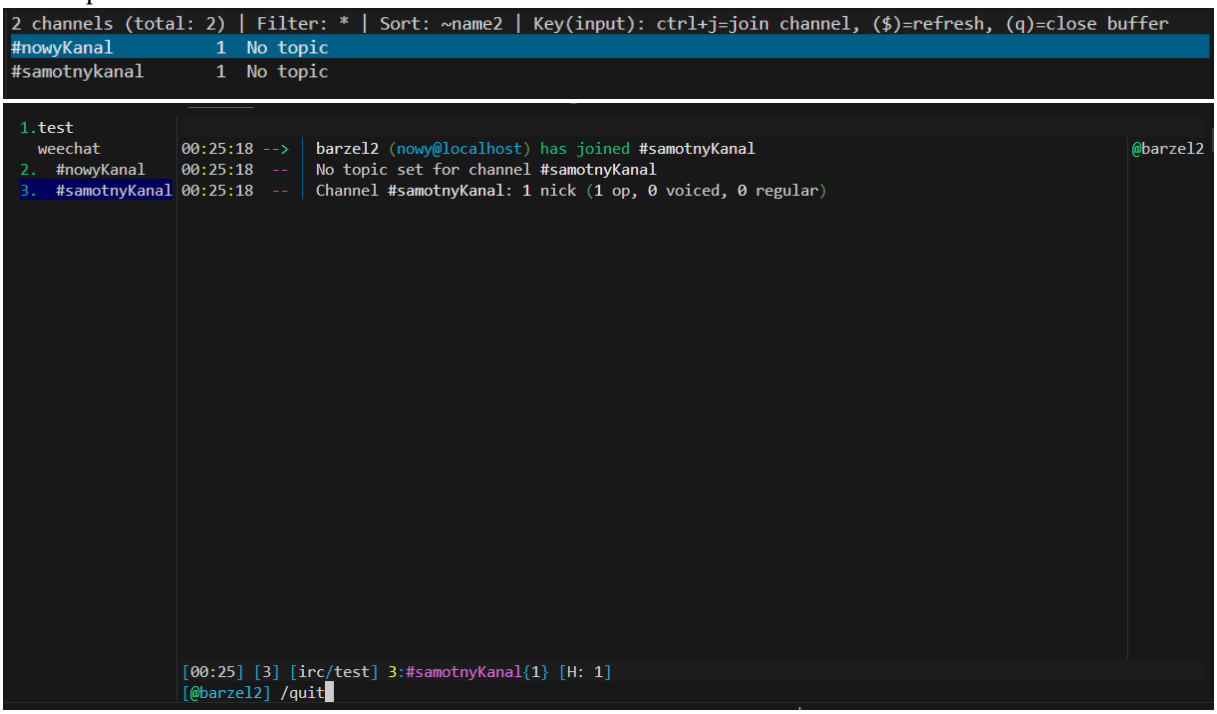

Podobnie jak w przypadku wyjścia po wyrzuceniu użytkownik nie może wysyłać wiadomości do kanału po tym jak został wyrzucony.



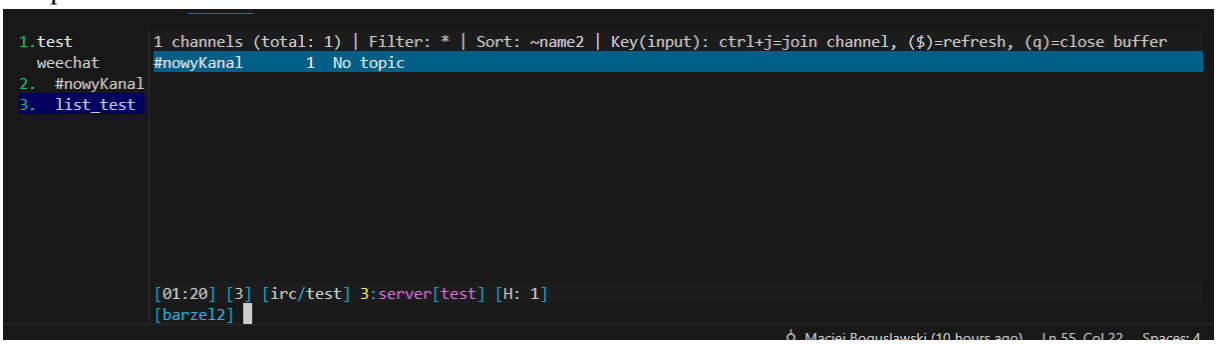

## Zakończenie połączenia

Aby zakończyć połączenie z serwerem należy wysłać wiadomość QUIT. Wysłanie wiadomości tej skutkuje także usunięciem użytkownika ze wszystkich kanałów, których był członkiem lub administratorem. Dodatkowo gdy liczba użytkowników kanału spadnie do zera serwer automatycznie usuwa dany kanał. Zakończenie połączenia i następujące po nim wyjście i usunięcie pustego kanału zostało zaprezentowane na poniższych zrzutach ekranu.

Przed quit:



Po quit:




Logi serwera: