

Michał Graniczka

21.06.2018

Dokumentacja opisująca wszystkie komendy i komunikacje klient-serwer gry TicTacToe Online

SPIS ZAWARTOŚCI

- I. Uwagi ogólne
- II. Komendy podstawowe
 - a. CMD_LOGIN
 - b. CMD_CREATE_ROOM
 - c. CMD_JOIN_GAME
 - d. CMD_LEAVE_GAME
 - e. CMD_SEND_MSG
 - f. CMD_MAKE_MOVE
 - g. CMD_LOGOUT
 - h. CMD_DOWNLOAD_ROOMS
- III. Game Loop
 - a. GL_OPPONENT_CONNECTED
 - b. GL_OPPONENT_MOVED
 - c. GL_OPPONENT_SENT_MSG
 - d. GL_OPPONENT_LEFT
 - e. GL_GAME_OVER

I. Uwagi ogólne

Przed wysłaniem jakiegokolwiek komendy do serwera należy ją przekształcić na format heksadecymalny. Serwer również odpowiada w formie ciągów heksadecymalnych.

II. Komendy podstawowe

Spis komend wysyłanych do serwera przez klienta

Ogólny schemat komend:

id_komendy[|parametr1|parametr2...|parametrn]

#CMD LOGIN

Loguje użytkownika do serwera gry.

Odbiera dwie odpowiedzi - potwierdzenie poprawnego przebiegu polecenia oraz id nadane użytkownikowi, który się połączył.

W przypadku napotkania błędów zwraca jedną odpowiedź.

ID komendy = 0

Dozwolona konstrukcja:

0|user_ip

Pozytywna odpowiedź serwera:

01

user_id

Negatywna odpowiedź serwera:

00

Parametry:

1. user_ip - ip użytkownika wysyłającego komendę i próbującego zalogować się do serwera gry. [string]

#CMD CREATE ROOM

Tworzy pokój do gry, w którym wywołujący użytkownik staje się hostem.

Obiera dwie odpowiedzi - potwierdzenie poprawnego przebiegu polecenia oraz id nadane tworzonemu przez użytkownika pokojowi.

W przypadku napotkania błędów zwraca jedną odpowiedź.

ID komendy = 1

Dozwolona konstrukcja:

1|user_ip|room_name|host_name|game_type|password_flag[|password]

Pozytywna odpowiedź serwera:

11

room_id

Negatywna odpowiedź serwera:

10

Parametry:

1. user_ip - ip użytkownika wysyłającego komendę. [string]
2. room_name - nazwa tworzonego pokoju. [string]
3. host_name - nazwa użytkownika tworzącego pokój. [string]
4. game_type - liczba dziesiętna z zakresy 0-2 informująca o rodzaju planszy gry. (0 - 3x3, 1 - 4x4, 2 - 6x6) [int]
5. password_flag - flaga informująca o tym, czy pokój ma być chroniony hasłem czy nie. (0 - nie chroniony, 1 - chroniony) [int]
- [6.] password [opcjonalne] - w przypadku password_flag==1, należy podać hasło jako ostatni parametr. Nie może być pustym ciągiem lub nullem. [string]

#CMD JOIN GAME

Pozwala na połączenie się z istniejącym pokojem jako gość/drugi gracz.

W przypadku, gdy pokój nie jest chroniony hasłem odbiera wiadomość potwierdzenia.

W przypadku, gdy pokój jest chroniony hasłem odbiera wiadomość potwierdzenia, wysyła hasło, odbiera wiadomość weryfikacji hasła.

Gdy użytkownik łączy się z pokojem do hosta wysyłana jest informacja o tym.

ID komendy = 2

Dozwolona konstrukcja:

2|user_ip|room_id|user_name|user_id

Pozytywna odpowiedź serwera jeżeli pokój jest chroniony hasłem:

22

>Należy wówczas podać hasło do przesłania serwerowi.

Odpowiedź jeżeli hasło jest "puste":

PASSWORD EMPTY

23

Odpowiedź jeżeli hasło błędne:

PASSWORD WRONG

20

Odpowiedź jeżeli hasło poprawne:

PASSWORD OK

> do hosta pokoju wysłany jest komunikat

GL_OPPONENT_CONNECTED|joined_user_id opisany poniżej

21

Pozytywna odpowiedź serwera jeżeli pokój nie jest chroniony hasłem:

> do hosta pokoju wysłany jest komunikat GL_OPPONENT_CONNECTED

21

Negatywna odpowiedź serwera:

20

Parametry:

1. user_ip - ip użytkownika wysyłającego komendę. [string]
2. room_id - id pokoju, do którego próbuje połączyć się użytkownik. [int]
3. user_name - nazwa użytkownika łączącego się z pokojem. [string]
4. user_id - id użytkownika łączącego się z pokojem. [int]

#CMD LEAVE GAME

Komenda służąca do opuszczenia i usunięcia pokoju z serwera gry.

Po otrzymaniu komunikatu potwierdzenia wysyła komunikat o zamknięciu pokoju do znajdujących się w nim graczy.

Po jej wykonaniu pokój zostaje usunięty z serwera i nie można już w nim grać.

ID komendy = 3

Dozwolona konstrukcja:

3|user_ip|room_id|user_id

Odpowiedź serwera:

Gdy zlecającym jest host pokoju i żaden gość nie dołączył do pokoju:

9

> do hosta wysłany jest komunikat o zamknięciu pokoju

31

Gdy zlecającym jest host i w pokoju jest inny gracz:

9

> do hosta wysłany jest komunikat o zamknięciu pokoju

> do gościa wysłany jest komunikat GL_OPPONENT_LEFT

31

Gdy zlecającym jest gość:

9

> do gościa wysłany jest komunikat o zamknięciu pokoju

> do hosta wysłany jest komunikat GL_OPPONENT_LEFT

31

Negatywna odpowiedź serwera:

30

Parametry:

1. user_ip - ip użytkownika wysyłającego komendę. [string]
2. room_id - id zamykanego pokoju. [int]
3. user_id - id użytkownika zlecającego zamknięcie pokoju. [int]

#CMD_SEND_MSG

Komenda służąca do komunikacji gracz-gracz, znajdujących się w tym samym pokoju gry użytkowników.

Wiadomość wysyłana jest do podanego pokoju, do użytkownika o danym id.

ID komendy = 5

Dozwolona konstrukcja:

5|user_ip|room_id|user_id|message

Pozytywna odpowiedź serwera:

> do drugiego użytkownika pokoju przesyłana jest komunikat GL_OPPONENT_SENT_MSG

51

Negatywna odpowiedź serwera:

50

Parametry:

1. user_ip - ip użytkownika wysyłającego komendę. [string]
2. room_id - id pokoju, do którego wysłana będzie wiadomość. [int]
3. user_id - id użytkownika który wysyła wiadomość. [int]
4. message - wiadomość, która będzie wysłana do drugiego użytkownika pokoju (nie zlecającego). [string]

#CMD MAKE MOVE

Komenda służąca do zakomunikowaniu o podjętym ruchu podczas gry.

Wysyła informacje zarówno o tym, kto wykonał ruch, gdzie i jaki znaczek wstawił na pole.

Wywołanie tej komendy informuje graczy czy przeciwnik wykonał ruch oraz czy gra dobiegła końca.

ID komendy = 6

Dozwolona konstrukcja:

6|user_ip|room_id|user_id|board_index_x|board_index_y|user_character

Pozytywna odpowiedzi serwera:

> do drugiego gracza wysłana jest komunikat GL_OPPONENT_MOVED

61

Jeżeli gra zakończyła się:

> do obu graczy wysyłany jest komunikat GL_GAME_OVER

Negatywna odpowiedź serwera:

60

Parametry:

1. user_ip - ip użytkownika wysyłającego komendę. [string]
2. room_id - id pokoju, w którym wykonany jest ruch. [int]
3. user_id - id użytkownika, który wykonał ruch. [int]
4. board_index_x - współrzędne x pola planszy, w którym gracz wykonał ruch. [int]
5. board_index_y - współrzędne y pola planszy, w którym gracz wykonał ruch. [int]
6. user_character - znaczek, którym posługuje się gracz. ("X" lub "O")

#CMD LOGOUT

Komenda wylogowuje użytkownika z serwera gry i zamyka nawiązane połączenie.

Odbiera jedynie kod potwierdzenia.

ID komendy = 8

Dozwolona konstrukcja:

8|user_ip|user_id

Pozytywna odpowiedź serwera:

81

Negatywna odpowiedź serwera:

80

Parametry:

1. user_ip - ip użytkownika wysyłającego komendę. [string]
2. user_id - id użytkownika wylogowującego się. [int]

#CMD DOWNLOAD ROOMS

Komenda służy do pobrania od serwera informacji o wszystkich utworzonych pokojach.

Odbierany jest kod potwierdzenia, liczba pokoiów na serwerze,

ID komendy = 10;

Dozwolona konstrukcja:

10|user_ip|user_id

Pozytywna odpowiedź serwera:

101

rooms_count

> odpowiedź odebrana rooms_count razy:

room_id|room_name|room_host_name|room_game_type|password_info|number_of_players

101

Negatywna odpowiedź serwera:

100

Parametry:

1. user_ip - ip użytkownika wysyłającego komendę. [string]
2. user_id - id użytkownika ubiegającego się o pobranie listy pokoi. [int]

Parametry odebranych danych pokoju:

1. room_id - id pobieranego pokoju. [int]
2. room_name - nazwa pobieranego pokoju. [string]
3. room_host_name - nazwa hosta pobieranego pokoju. [string]
4. room_game_type - typ gry pobieranego pokoju. ("3x3" itd.) [string]
5. password_info - informacja czy pokój chroniony hasłem czy nie. ("Yes"/"No")
6. number_of_players - ilość graczy grających w pobieranym pokoju. (max 2) [int]

III. Game Loop

Spis komunikatów od serwera do klienta, wykorzystywanych jedynie w kontekście pokoju gry podczas rozgrywki

Podczas grania klient użytkownika nie tylko wysyła komunikaty do serwera i odbiera odpowiedzi ale również przechodzi w stan nasłuchiwanie komunikatów GL (Game Loop). Służą one synchronizacji danych wyświetlanych w oknach graczy.

##GL OPPONENT CONNECTED

Komunikat informuje klienta hosta pokoju, że inny gracz o danej nazwie dołączył i gra może się zacząć.

ID komunikatu = 201

Dozwolona konstrukcja:

201|user_name

Parametry:

1. user_name - nazwa gracza, który wykonał ruch. [string]

##GL OPPONENT MOVED

Komunikat informuje klienta o ruchu przeciwnika użytkownika - kto wstawił gdzie i co.

ID komunikatu = 202

Dozwolona konstrukcja:

202|user_name|board_index_x|board_index_y|user_character

Parametry:

1. user_name - nazwa gracza, który wykonał ruch. [string]
2. board_index_x - współrzędne x komórki planszy, w której gracz wstawił swój znak. [int]
3. board_index_y - współrzędne y komórki planszy, w której gracz wstawił swój znak. [int]
4. user_character - znak gracza, który został wstawiony. ("X" lub "O")

##GL OPPONENT SENT MSG

Komunikat informuje klienta o tym, że przeciwnik użytkownika wysłał wiadomość oraz co ona zawierała.

ID komunikatu = 203

Dozwolona konstrukcja:

203|user_name|message

Parametry:

1. user_name - nazwa gracza wysyłającego wiadomość. [string]
2. message - wiadomość wysłana przed graczem. [string]

##GL OPPONENT LEFT

Komunikat informuje klienta, że jeden z graczy opuścił pokój i że jest on zamykany.

ID komunikatu = 204

Dozwolona konstrukcja:

204

##GL GAME OVER

Komunikat informuje klienta, że gra dobiegła końca oraz z jakim skutkiem.

ID komunikatu = 205

Dozwolona konstrukcja:

205|winner_id|winner_name|winner_score

Parametry:

1. winner_id - id zwycięzcy gry. [int]
2. winner_name - nazwa zwycięzcy gry. [string]
3. winner_score - ilość punktów zwycięzcy gry. [int]

W przypadku remisu:

```
> winner_id = -1;
```

```
> winner_name = null;
```

```
> winner_score = -1;
```