

Protocollo DGDBP

Venturini - D'annibale

18/08/2024

Contents

1	Descrizione	2
2	Richieste	2
2.1	1xx - Game operations	2
2.1.1	102 - find opponent	2
2.1.2	103 - update game	3
2.1.3	104 - search game	3
2.1.4	105 - quit search opponent	3
2.2	2xx - User operations	4
2.2.1	201 - new user	4
2.2.2	202 - update user	4
2.2.3	203 - update user preferences	4
3	Risposte	4
3.1	2xx - Success	4
3.1.1	200 - OK	4
3.1.2	201 - Created	4
3.1.3	202 - Game created	5
3.1.4	203 - Game found	5
3.1.5	204 - User created	5
3.2	4xx - Client errors	5
3.2.1	400 - Bad request	5
3.2.2	404 - Not found	5
3.3	5xx - Server errors	5
3.3.1	500 - Internal server error	5
4	Tipi enumerativi	5
4.1	chessboardStyle	5
4.2	piecesStyle	5
4.3	esito	5
4.4	motivo	6

1 Descrizione

Il Dan Gigi DataBase Protocol è un semplice protocollo per la comunicazione tra client e server del progetto Ingegneria della scacchiera.

I messaggi sono di tipo testuale in formato json ed hanno la seguente struttura:

- header : { tipologia : x, codice : c};
- body : {}

La struttura del body .

Nella parte seguente verrà trattata la specifica di ogni body, la quale può variare in base al tipo di messaggio e al codice dell'operazione, sulla falsa riga delle API web.

Nella sezione finale di questo documento, sono riportati tutti i tipi di dato utilizzati non standard, che sono stati definiti come enumerazioni.

2 Richieste

2.1 1xx - Game operations

2.1.1 102 - find opponent

body : { user : u_id, time_duration: x, time_increment: y } Dove u_id è una stringa e x ed y sono interi.

Le risposte possibili a questa richiesta sono:

- 200 - OK
- 400 - Bad request
- 404 - Not found
- 500 - Internal server error

2.1.2 103 - update game

body : { game_id : g_id, moves : , esito : e, motivo : m }

Dove g_id è un intero, il valore di moves è un array di mosse, e è di tipo [esito](#) , m è di tipo [motivo](#) . Esiste una dipendenza tra i valori di esito e motivo: con la notazione $e \rightarrow m$ valgono le seguenti regole;

1. "W" \vee "B" \rightarrow "checkmate" \vee "wonOnTime" \vee "quitmate"
2. "D" \rightarrow "stalemate" \vee "insufficientMaterial" \vee "threefoldRepetition" \vee "50moveRule"
3. "NF" \rightarrow "NF"

Le risposte possibili a questa richiesta sono:

- [200 - OK](#)
- [404 - Not found](#)
- [500 - Internal server error](#)

2.1.3 104 - search game

body : { game_id : g_id }

dove g_id è un intero.

Le risposte possibili a questa richiesta sono:

- [203 - Game found](#)
- [404 - Not found](#)
- [500 - Internal server error](#)

2.1.4 105 - quit search opponent

body : { user : u_id, time_duration: x, time_increment: y } Dove u_id è una stringa e x ed y sono interi.

Le risposte possibili a questa richiesta sono:

- [200 - OK](#)
- [400 - Bad request](#)
- [404 - Not found](#)
- [500 - Internal server error](#)

2.2 2xx - User operations

2.2.1 201 - new user

body : { username : u_id, }

Le risposte possibili a questa richiesta sono:

- [204 - User created](#)
- [500 - Internal server error](#)

2.2.2 202 - update user

body : { nome : n, cognome : c, username : u_id, new_username: u_id, elo : points }

Dove n,c,u_id sono stringhe e points è un intero.

Le risposte possibili a questa richiesta sono:

- [200 - OK](#)
- [400 - Bad request](#)
- [404 - Not found](#)
- [500 - Internal server error](#)

2.2.3 203 - update user preferences

body : { username : u_id, chessboard_style : c_st, pieces_style : p_st }

Dove u_id è una stringa, c_st è di tipo [chessboard_style](#) mentre p_st è un valore di tipo [pieces_style](#)

Le risposte possibili a questa richiesta sono:

- [200 - OK](#)
- [404 - Not found](#)
- [500 - Internal server error](#)

3 Risposte

3.1 2xx - Success

3.1.1 200 - OK

body : {}.

3.1.2 201 - Created

body : {message : x}

Dove x è una stringa.

3.1.3 202 - Game created

body : {game_id : x}
dove x è un intero.

3.1.4 203 - Game found

3.1.5 204 - User created

body : {user_id : x}
dove x è un intero.

3.2 4xx - Client errors

3.2.1 400 - Bad request

body : {message : x}
Dove x è una stringa.

3.2.2 404 - Not found

body : {message : x}
Dove x è una stringa.

3.3 5xx - Server errors

3.3.1 500 - Internal server error

body : {message : x}
Dove x è una stringa.

4 Tipi enumerativi

4.1 chessboardStyle

Un valore dell'insieme "blue", "brown", "black".

4.2 piecesStyle

Un valore dell'insieme "neo", "pixel"

4.3 esito

Un valore nell'insieme {"W","D","B","NF"} (che ndicano white win, draw, black win, not finished)

4.4 motivo

Un valore nell'insieme {"checkmate", "wonOnTime", "quitmate", "stalemate", "insufficientMaterial", "threefoldRepetition", "50moveRule", "NF"}