

KingBingo Web API v1.1 3/11/14

Supported Operations

- [Get Games](#)
- [Get Users](#)
- [Create User](#)
- [Join Game](#)
- [Get Number](#)
- [Quit Game](#)
- Get Card
- Submit Win
- Add Friend
- Remove Friend
- Create Game

Usage

- All json field names in the response will be lowercase only. All field names in request need to be lowercase as well.
- All responses will return a "command" parameter. This will allow you to standardize your json parsing depending on which request you are making.
- Get a list of all currently in progress games. Within each block of "game" data, there is a game_id.
- Create a user. This will give you a user_id that you can then join a game with. You can use "bingowizard" for login and/or "1" for user_id (see below) if you'd like to skip this step.
- Join a game using the user_id and game_id. This will return a "board" that you will use to construct your bingo card.
- Once you have your board, call use the get number operation to play the game!
- Be sure to check the response messages when getting errors to help troubleshoot your requests, and don't forget to validate your json.
- When initially working with any these services, use the "bingowizard" login or user_id of "1" if you do not want to create a user. This user has been joined to the game with game_id of "1".
- A monitoring service has been setup to send notifications in the event of outages, within 5 minutes. Error reports (specifically, what you were doing when it crashed) should be sent to joe@humboldttechgroup.com

Get Games

POST Request to <http://bingo.humboldttechgroup.com:1111/?cmd=allgames>

Returns all currently in progress games

- **games:** Array of game elements, each containing it's own game_id, win_limit, win_count, user_limit, user_count, and created_date.
- **game_id:** id of the game. used in joingame command to join user to the game.
- **win_limit:** number of wins allowed before game is closed
- **win_count:** number of wins so far in current game
- **user_limit:** limit of how many users are allowed to participate in current game
- **user_count:** number of users currently joined to the game
- **created_date:** timestamp of when game was created
- **status:** "ok" if successful, "error" if problem with request
- **message:** information about response

Sample request body:

Post Body not needed

Sample response data:

```
{
  "status": "ok",
  "command": "allgames",
  "message": "successfully retrieved list of active games",
  "games": [
    {
      "game_id": "1",
      "win_limit": "10",
      "win_count": "0",
      "user_limit": "10",
      "user_count": "9",
      "created_date": "2013-04-04 00:00:00"
    },
    {
      "game_id": "2",
      "win_limit": "5",
      "win_count": "2",
      "user_limit": "5",
      "user_count": "4",
      "created_date": "2013-04-03 00:00:00"
    }
  ]
}
```

Get Users

POST Request to `http://bingo.humboldttechgroup.com:1111/?cmd=allusers`

Returns all currently joined users in a given game

- **game_id**: Id of the game requested to retrieve list of users
- **timestamp**: UTC Time of request/response
- **status**: "ok" if successful, "error" if problem with request
- **users**: Array of user elements, each containing it's own user_id and login attributes.
- **user_id**: Id of user listed
- **login**: Username of user listed

Sample request body:

```
{
  "timestamp": "1234567890",
  "game_id": "1"
}
```

Sample response data:

```
{
  "status": "ok",
  "command": "allusers",
  "message": "list of users for game in progress",
  "game_id": "1",
  "timestamp": "1234567890",
  "users": [
    {
      "user_id": "1",
      "login": "bingowizard"
    },
    {
      "user_id": "2",
      "login": "bingobarbarian"
    }
  ]
}
```

Create User

POST Request to `http://bingo.humboldttechgroup.com:1111/?cmd=createuser`

Creates a user and returns a user_id, or existing user_id if given login name already exists.

Sample request body:

```
{
  "timestamp": "1234567890",
  "login": "bingowizard"
}
```

Sample response data:

```
{
  "status": "ok",
  "command": "createuser",
  "message": "user created successfully",
  "login": "bingowizard",
  "user_id": "1",
  "timestamp": "1234567890"
}
```

If user login already exists, the following response will be returned instead:

```
{
  "status": "ok",
  "command": "createuser",
  "message": "user already exists",
  "login": "bingowizard",
  "user_id": "1",
  "timestamp": "1234567890"
}
```

Join Game

POST Request to <http://bingo.humboldttechgroup.com:1111/?cmd=joingame>

Use this command to join a login to an existing game. A successful response will return the Game ID and a comma delimited list of numbers, in order, that make up the user's bingo card. The numbers should be placed in the card from top to bottom in a 5 x 5 grid. For example: using the below "board" data should produce a card with the following numbers in the "B" column, from top to bottom: 12, 1, 15, 2, and 9.

- **game_id**: Id of the game attempting to join.
- **user_id**: Id of user to join to game
- **timestamp**: UTC Time of request/response
- **board_id**: Returned id of generated board (bingo card)

- **board:** Comma-delimited list of numbers making up the generated board
- **status:** "ok" if successful, "error" if problem with request
- **message:** information about response

Sample request body:

```
{
  "timestamp": "1234567890",
  "game_id": "1",
  "user_id": "1"
}
```

Sample response data:

```
{
  "status": "ok",
  "command": "joingame",
  "message": "game sucessfully joined",
  "board_id": "1",
  "board":
"12,1,15,2,9,24,30,22,18,21,34,43,45,31,42,54,50,58,56,55,68,63,70,72,62",
  "game_id": "1",
  "user_id": "1",
  "timestamp": "1234567890"
}
```

If user has already joined game, the following will be returned instead, with the board data being the previously generated value.

```
{
  "status": "ok",
  "command": "joingame",
  "message": "user already joined game",
  "board_id": "1",
  "board":
"12,1,15,2,9,24,30,22,18,21,34,43,45,31,42,54,50,58,56,55,68,63,70,72,62",
  "game_id": "1",
  "user_id": "1",
  "timestamp": "1234567890"
}
```

Get Number

POST Request to <http://bingo.humboldttechgroup.com:1111/?cmd=getnumber>

Returns the next drawn number in a given game. The user_id must match that of a user who has been joined to the game of the requested game_id.

- **game_id**: Id of the game user is joined to
 - **user_id**: Id of user making request
 - **timestamp**: UTC Time of request/response
 - **status**: "ok" if successful, "error" if problem with request
 - **message**: information about response
 - **number**: the bingo number that has been drawn. B12, B6, I50, etc.
-

Sample request body:

```
{
  "timestamp": "1234567890",
  "game_id": "1",
  "user_id": "1"
}
```

Sample response data:

```
{
  "status": "ok",
  "command": "getnumber",
  "message": "newest number is",
  "game_id": "1",
  "number": "B12",
  "timestamp": "1234567890"
}
```

Quit Game

POST Request to <http://bingo.humboldttechgroup.com:1111/?cmd=quitgame>

Use this command to quit a login to a previously joined game.

- **game_id**: Id of the game attempting to quit.
 - **user_id**: Id of user to quit game
 - **timestamp**: UTC Time of request/response
 - **status**: "ok" if successful, "error" if problem with request
 - **message**: information about response
-

Sample request body:

```
{
  "timestamp": "1234567890",
  "game_id": "1",
  "user_id": "1"
}
```

```
}
```

Sample response data:

```
{
  "status": "ok",
  "command": "quitgame",
  "message": "game sucessfully quit",
  "game_id": "1",
  "user_id": "1",
  "timestamp": "1234567890"
}
```

If user has already quit game or not currently joined, the following will be returned instead.

```
{
  "status": "ok",
  "command": "quitgame",
  "message": "user is not joined to game",
  "game_id": "1",
  "user_id": "1",
  "timestamp": "1234567890"
}
```
