

Rock-Paper-Scissors Game API Documentation

Overview

This is an API documentation for the Rock-Paper-Scissors game. The game can be played by two bots. Each game consists of several rounds, where the bots make their moves simultaneously. After each round, the outcome is decided based on the moves of the bots.

Rules

- You are not allowed to use any random number generators, or try to make a series of random numbers yourself.
- Each game will be 2,000 rounds
- You will have 0.5 seconds after receiving the previous game's results to send your next input. Failure to meet this will disqualify your bot from the tournament.

Base URL

All endpoints are prefixed with:

```
http://<ip_address>.<port_number>
```

Where **<ip_address>** and **<port_number>** will be given at the start of the match.

Endpoints

1. Start a New Game

you don't have to worry about starting the game, this will be done for you

- **Endpoint:**

```
POST /start_game
```

- **Payload:**

- **num_rounds** (Optional): An integer representing the number of rounds the game should have. Defaults to 5 if not provided.

- **Response:**

- **game_id**: A unique identifier for the game.

- **bot1_id**: A unique identifier for the first bot.
- **bot2_id**: A unique identifier for the second bot.
- **message**: A message indicating the game has started.

you will receive **bot_id** and **game_id** to send your posts with once the game has started

- **Example Request:**

```
{
  "num_rounds": 2000
}
```

- **Example Response:**

```
{
  "game_id": "1",
  "bot1_id": "12345",
  "bot2_id": "67890",
  "message": "Game started!"
}
```

2. Play a Round

Endpoint:

```
POST /play_round/<game_id>/<bot_id>
```

Example Responses:

If the bot is disqualified due to a timeout:

```
{
  "error": "Bot 1 was disqualified due to timeout."
}
```

If both bots have played and the game isn't over yet:

```
{
  "your_name": "Bot 1",
  "outcome": "You win",
  "your_move": "Rock",
  "opponent_move": "Scissors",
  "score": [1, 0]
}
```

If the game has finished after the round:

```
{
  "message": "Game over! Winner: Bot 1",
  "score": [1000, 900]
}
```

If the opponent has yet to move:

```
{
  "status": "Waiting for opponent"
}
```

Errors

- If the provided game ID does not exist, a response with a status code of **404** and a message `{"error": "Game not found!"}` is returned.
- If an invalid move is provided, a response with a status code of **400** and a message `{"error": "Invalid move!"}` is returned.
- If an invalid bot ID is provided, a response with a status code of **404** and a message `{"error": "Invalid Bot ID!"}` is returned.

Notes

- The game automatically ends after the specified number of rounds.
- The game uses the traditional rules of Rock-Paper-Scissors to decide the winner of each round.
- The game ensures that each bot has a unique ID.