**SlotMachine OpenAPI**

default : [115.68.110.176:8090](http://115.68.110.176:8090/betting_time/1000)

[[목차]]

[**응답 규약**](#_e5gxfpi10jbe) **2**

[**엑세스 토큰**](#_jda1diq4tstq) **2**

[**전체 공지**](#_qucmidoxph3m) **2**

[**특정 유저(id)에 메시지 공지**](#_m04t5ygqgtd3) **3**

[**특정 유저(id)의 차단**](#_17vq3pubply) **3**

[**특정 유저(id)의 차단 해제**](#_lsl9z9setfzs) **4**

[**특정 유저(id)의 차단 여부 검사**](#_wtce49sdt6se) **4**

[**배팅시간 변경**](#_w9vtgjm9x8dy) **5**

[**배팅단위 금액 확인**](#_kfbj8it8r8jt) **5**

[**배팅단위 금액 설정**](#_25upuw6yuaqa) **5**

[**최대 배팅 가능 금액 조회**](#_v4fnyhi10uay) **6**

[**최대 배팅 가능 금액 설정**](#_g6g958y3n2cj) **6**

[**최대 지급 가능 금액 조회**](#_5uiti3dpy6ob) **7**

[**최대 지급 가능 금액 설정**](#_sqxljqpdwwks) **7**

[**Just one betting모드 조회**](#_tg20wwelehmy) **7**

[**Just one betting모드 설정**](#_o0oh1m45htz2) **8**

[**색상별 배당률 변경**](#_5vsw5zz43ix1) **8**

[**배당률 확인**](#_cgf9ii1tv1yz) **9**

[**환경설정 값 반환**](#_pq9wxu8z885q) **10**

[**종료 게임 목록**](#_7wmn05ndzpqa) **10**

[**게임 배팅 목록**](#_7wod717oe4t8) **12**

[**포인트 차감/지급 로그 파라미터**](#_l58cofqpgad0) **15**

[차감](#_i7jfhxtlym5s) 15

[지급](#_tx7jsikjsm2o) 15

[**환불목록 카운트**](#_vgd99e6cm3e2) **16**

[**환불목록**](#_n5xg02qfhymn) **16**

[**환불**](#_gf5eopaj7wiu) **18**

[**서비스 정지**](#_tse8o9b4bod) **18**

[**서비스 시작**](#_fsugj3cidwf) **19**

[**서비스 정지 확인**](#_e7vn27e918zx) **20**

[**욕설 광고 필터 등록**](#_yxviy8pm6smh) **20**

[**욕설 광고 필터 삭제**](#_um9ni79etr9t) **21**

# 응답 규약

성공의 경우 {“result”:true}

실패의 경우 {“result”:false} 또한 HTTP-CODE가 500번으로 전송됨.

모든 응답은 Content-Type이 application/json으로 응답함.

# 엑세스 토큰

모든 요청에는 access\_token이 반드시 필요함.

access\_token은 게임 서버의 config.json에서 설정함. “admin\_web\_server\_access\_token”에 설정해야함.

만약 엑세스 토큰이 없거나 올바르지 않다면 다음의 에러가 반환됨.

{"result":false,"error":"invalid access\_token"}

# 전체 공지

전체 공지 인스턴스 메시지를 전송함. 주로 정기점검 이전에 메시지를 전송함. message query가 존재하지 않는 경우 응답은 실패함.

|  |
| --- |
| [http://주소:포트/announce?message=helo](http://115.68.110.176:8090/announce?message=helo)&access\_token=엑세스토큰 |

요청 예)

[http://115.68.110.176:8090/announce?message=hello](http://115.68.110.176:8090/announce?message=helo)&access\_token=64756ea95f85f91ab64cc8f4a6275c5f

응답 예)

{"result":true,"message":"hello"}

# 특정 유저(id)에 메시지 공지

특정 유저에게만 인스턴스 메시지를 전송함. 주로 해당 유저에게 경고를 하거나 특정 메시지를 전당할 때 사용함. 해당 유저가 로그인 된 상태가 아니면 에러가 반환됨.

|  |
| --- |
| http://주소:포트/announce/아이디?message=메시지&access\_token=엑세스토큰 |

요청 예)

[http://115.68.110.176:8090/announce/jpwdnv?message=](http://115.68.110.176:8090/announce/jpwdnv?message=%EA%B4%91%EA%B3%A0%EC%84%B1)해킹하지 마세요&access\_token=64756ea95f85f91ab64cc8f4a6275c5f

응답 예)

{"result":true,"id":"jpwdnv","message":"해킹하지 마세요"}

{"result":false,"error":"not exist id or not logined"}

# 특정 유저(id)의 차단

특정 유저의 로그인을 차단하고자 할 경우에 다음 명령을 사용함. 현재 접속중이라면 게임 턴의 종료와 상관없이 차단함. 차단 유저는 DB에 기록하여 지속적으로 차단됨. 본 기능은 본 게임에만 적용되는 특성이므로 Portal시스템에서의 차단과는 별도로 운영되는 특성을 가짐. Portal에서 차단을 해제해도 본 기능에서 특정 유저의 차단이 적용되었다면 차단으로 간주됨.

차단만료날짜시간을 지정하고 싶다면 아래 예시를 참조. 단 차단만료날짜 시간의 포맷은 YYYY-mm-dd HH:ii:ss형태만 가능함. 가령 2019-1-3 오후 3시 30분인 경우 ‘2019-01-03 15:30:00’형태로 써야함. 해당 시간이 지나면 차단이 자동으로 해제됨.

|  |
| --- |
| http://주소:포트/block\_user/아이디?access\_token=엑세스토큰  http://주소:포트/block\_user/아이디/차단만료날짜시간?access\_token=엑세스토큰 |

요청 예)

http://115.68.110.176:8090/block\_user/xxx?access\_token=64756ea95f85f91ab64cc8f4a6275c5f

http://115.68.110.176:8090/block\_user/xxx/2018-11-22%2008:00:00?access\_token=64756ea95f85f91ab64cc8f4a6275c5f

응답 예)

{"result":true,"id":"xxx"}

{"result":true,"id":"xxx","expire\_time":"2018-11-22 08:00:00"}

만료시간을 지정한 경우 해당 만료시간을 경과했어도 해당 레코드는 drop\_users테이블에 기록되어 있음.

# 특정 유저(id)의 차단 해제

특정 유저의 차단된 상태를 해제함. Portal의 차단이 아닌 게임 시스템에서의 차단을 해제하는 명령임.

|  |
| --- |
| http://주소:포트/cancel\_block\_user/아이디?access\_token=엑세스토큰 |

요청 예)

http://115.68.110.176:8090/cancel\_block\_user/xxx?access\_token=64756ea95f85f91ab64cc8f4a6275c5f

응답 예)

{"result":true,"id":"xxx"}

# 특정 유저(id)의 차단 여부 검사

특정 유저의 차단된 상태를 검사함. Portal의 차단이 아닌 게임 시스템에서의 차단 여부를 검사하는 명령임.

|  |
| --- |
| [http://주소:포트/is\_block\_user/아이디?access\_token=](http://115.68.110.176:8090/is_block_user/xxx?access_token=64756ea95f85f91ab64cc8f4a6275c5f)엑세스토큰 |

요청 예)

<http://115.68.110.176:8090/is_block_user/xxx?access_token=64756ea95f85f91ab64cc8f4a6275c5f>

응답 예)

{"result":true,"id":"xxx","is\_blocked":true}

{"result":true,"id":"xxxx","is\_blocked":false}

# 배팅단위 금액 확인

슬롯머신은 Line을 추가할 때마다 배팅금액이 증가함. 1라인을 추가하는데 따르는 배팅 금액을 확인함. config.json보다 OpenAPI에 설정된 배팅 단위 금액이 우선하며 현재 시스템에 반영된 배팅 단위 금액을 반환함.

|  |
| --- |
| [http://주소:포트/](http://115.68.110.176:8090/betting_unit_amounts)unit\_bet\_amount?access\_token=엑세스토큰 |

요청 예)

<http://115.68.110.176:8090/unit_bet_amount?access_token=64756ea95f85f91ab64cc8f4a6275c5f>

응답 예)

{"result":true,"unit\_bet\_amount":100}

# 배팅단위 금액 설정

슬롯머신은 Line을 추가할 때마다 배팅금액이 증가함. 1라인을 추가하는데 따르는 배팅 금액을 변경함. config.json인 환경설정을 통해서도 변경할 수 있으나 OpenAPI를 통해서도 변경할 수 있으며 config.json에 명시된 배팅단위 금액보다 OpenAPI를 통해서 적용한 값을 우선적으로 적용함. 배팅 단위 금액을 변경하면 플레이어는 현재 제공된 hash에 대한 플레이가 완료된 다음 플레이부터 새로 설정된 배팅 단위 금액의 적용을 받게됨.

|  |
| --- |
| [http://주소:포트/](http://115.68.110.176:8090/betting_unit_amounts)unit\_bet\_amount/배팅단위금액?access\_token=엑세스토큰 |

요청 예)

[http://115.68.110.176:8090/](http://115.68.110.176:8090/betting_unit_amounts)unit\_bet\_amount/100?access\_token=64756ea95f85f91ab64cc8f4a6275c5f

응답 예)

{"result":true,"unit\_bet\_amount":100}

# 환경설정 값 반환

설정한 환경설정 정보를 반환함.

|  |
| --- |
| [http://주소:포트/get\_config](http://115.68.110.176:8090/get_config)?access\_token=엑세스토큰 |

요청 예)

<http://115.68.110.176:8090/get_config>?access\_token=64756ea95f85f91ab64cc8f4a6275c5f

응답 예)

{"result":true,"rows":[{"name":"unit\_bet\_amount","val":"100"}]}

|  |  |  |
| --- | --- | --- |
| **name** | **value 설명** | **비고** |
| unit\_bet\_amount | 배팅 단위 금액 | line을 하나 추가할 때마다 증가하는 배팅 금액 |

해당 값은 OpenAPI로 설정을 하지 않는 경우에는 존재하지 않을 수 있으며, 존재하지 않을 경우에는 비고란의 default값이 설정된 것으로 봐야함. 또한 이러한 환경설정은 config.json을 통해서도 제어가 가능하며, config.json보다 OpenAPI로 제어한 값이 더 우선순위가 높게 적용됨.

# 게임 배팅 목록

플레이어게 진행(spin)한 게임 배팅 목록을 한번에 40개씩 제공함. 진행(spin)하지 않은 게임은 제공하지 않음.

|  |
| --- |
| [http://주소:포트/](http://115.68.110.176:8090/get_config)game\_betting\_history?access\_token=엑세스토큰[&page=?] |

* page : (옵션) 없는 경우 기본값 1

요청 예시)

|  |
| --- |
| <http://115.68.110.176:8090/game_betting_history?access_token=64756ea95f85f91ab64cc8f4a6275c5f&page=1> |

응답 예시)

|  |
| --- |
| {  "result":true,  "game":"SlotMachine",  "channel\_name":"",  "channel\_code":"",  "page":1,  "rows":[  {  "game\_number":3,  "id":"test1",  "nick":"테스트1",  "is\_spinned":1,  "spin\_time":"2020-02-19 13:47:22",  "before\_user\_point":22491160,  "after\_user\_point":22491060,  "hash":"11498077bc124e739ff9f091c347d2ae789711c79420e169e9a98f8a7e814062",  "seed":"val(99,5,3,3,99|2,99,5,1,2|1,7,4,4,8)\_\_abaf5b444b5e60e63629806351ca6f4d",  "line":1,  "unit\_bet\_amount":100,  "profit":0,  "reward":0,  "amount":100,  "rtime":"2020-02-19 13:47:21",  "refund\_amount":0,  "refunded\_amount":0,  "refunded\_time":"0000-00-00 00:00:00"  },  {  "game\_number":2,  "id":"test1",  "nick":"테스트1",  "is\_spinned":1,  "spin\_time":"2020-02-19 13:47:18",  "before\_user\_point":22491260,  "after\_user\_point":22491160,  "hash":"259c6f20d9d18801185f53bace963180835acb9dc40ae954b7581b4903c77228",  "seed":"val(8,7,4,99,3|7,6,7,8,2|4,5,3,5,99)\_\_bc4060dbe5fb19ccfae018c1afbcf37d",  "line":1,  "unit\_bet\_amount":100,  "profit":0,  "reward":0,  "amount":100,  "rtime":"2020-02-19 13:47:17",  "refund\_amount":0,  "refunded\_amount":0,  "refunded\_time":"0000-00-00 00:00:00"  }  ]  } |

* $.result : true
* $.game : SlotMachine
* $.channel\_name : 채널 이름
* $.channel\_code : 채널 코드
* $.page : 현재 페이지
* $.rows[]
  + $.rows[].game\_number : 게임 번호
  + $.rows[].id : 플레이어 아이디
  + $.rows[].nick : 플레이어 닉네임
  + $.rows[].is\_spinned : SPIN여부(0:아니고, 1:예)
  + $.rows[].spin\_time : SPIN날짜 시간
  + $.rows[].before\_user\_point : 배팅 차감전 포인트
  + $.rows[].after\_user\_point : 배팅 차감후 포인트(reward는 반영 안됨)
  + $.rows[].hash : Hash데이터
  + $.rows[].seed : Hash의 기반이 되는 seed데이터
  + $.rows[].line : 배팅한 line의 수
  + $.rows[].unit\_bet\_amount : 단위 배팅 금액
  + $.rows[].profit : 플레이어 수익
  + $.rows[].reward : 당첨금
  + $.rows[].amount : 배팅 금액(일반적으로 line수 x 단위 배팅금액임)
  + $.rows[].rtime : 배팅 날짜시간
  + $.rows[].refund\_amount : 환불 금액
  + $.rows[].refunded\_amount : 환불여부. (SPIN을 했으나 환불 여부가 1인 경우 해당 플레이는 정상적으로 진행되지 않거나 정산되지 않았으므로 환불해야함.)
  + $.rows[].refunded\_time : 환불된 날짜 시간

# 포인트 차감/지급 로그 파라미터

게임에 배팅을 하거나 게임 결과 포인트(balance 또는 point)가 있는 경우 관련된 게임 차감/지급 로그와 함께 전송됩니다. 여기서는 게임의 포인트 차감/지급 항목에 대하여 설명합니다.

기본적으로 이러한 로그 정보는 ‘|’기호로 구분자를 구성하며 lvalue=rvalue형태의 항목을 가집니다.

가령 포인트 차감시 game=SlotMachine|channel\_code=?|game\_number=?|id=?|code=?|amount=?|line=?|unit\_bet\_amount=?...{생략}... 과 같은 형태로 로그가 전달됩니다.

## 차감

차감 요청은 유저가 배팅을 한 경우에 요청합니다.

|  |  |
| --- | --- |
| **변수** | **값** |
| game | SlotMachine |
| channel\_code | 채널 코드. 채널 코드가 없는 경우 blank(‘’)가 전달됩니다. |
| game\_number | 게임 번호(게임 식별용) |
| id | 유저 아이디 |
| code | 유저 코드 |
| amount | 차감 금액 (line x unit\_bet\_amount) |
| line | 배팅 line의 수 |
| unit\_bet\_amount | 배팅 단위 금액 |

## 지급

지급은 유저가 배팅에 당첨된 경우에 요청합니다.

|  |  |
| --- | --- |
| **변수** | **값** |
| game | SlotMachine |
| channel\_code | 채널 코드. 채널 코드가 없는 경우 blank(‘’)가 전달됩니다. |
| game\_number | 게임 번호(게임 식별용) |
| id | 유저 아이디 |
| code | 유저 코드 |
| line | 배팅 line수 |
| matched\_lines | 배팅된 line의 수 |
| profit | 수익(지급 금액 - 배팅금액) 단, 음수가 발생하면 0으로 처리 |
| reward | 지급 금액 (실제 지급되는 금액) |

# 환불목록 카운트

플레이를 진행(SPIN) 했으나 게임의 오동작으로 인하여 정상적으로 완료되지 않은 게임에 대한 환불 목록의 수를 제공함.

|  |
| --- |
| [http://주소:포트/refund\_](http://115.68.110.176:8090/refund_list)count?access\_token=엑세스토큰  [http://주소:포트/refund\_](http://115.68.110.176:8090/refund_list/%EC%95%84%EC%9D%B4%EB%94%94)count[/아이디](http://115.68.110.176:8090/refund_list/%EC%95%84%EC%9D%B4%EB%94%94)?access\_token=엑세스토큰 |

요청 예)

[http://115.68.110.176:8090/refund\_](http://115.68.110.176:8090/refund_list)count?access\_token=64756ea95f85f91ab64cc8f4a6275c5f

아이디가 없는 경우 응답 예)

{"result":true,"count":1}

아이디가 있는 경우 응답 예)

{"result":true,"count":0,"id":"aloha"}

# 환불목록

게임의 환불은 반드시 목록을 확인한 이후에 반드시 환불 OpenAPI를 호출하여 내부적으로 로그가 남을 수 있도록 해야함. refund\_amount는 게임이 종료(턴 완료)되면 자동으로 0으로 설정됨. 그러므로 배팅과 함께 refund\_amount의 금액이 설정됨에 주의해야함.

정말로 해당 금액을 환불해야 하는지에 대해서는 게임 서버가 죽은 경우 해야함.

또한 이미 환불한 금액이 있는지도 잘 살펴봐야함.

|  |
| --- |
| [http://주소:포트/refund\_list](http://115.68.110.176:8090/refund_list)?page=페이지&access\_token=엑세스토큰  [http://주소:포트/refund\_list/아이디](http://115.68.110.176:8090/refund_list/%EC%95%84%EC%9D%B4%EB%94%94)?page=페이지&access\_token=엑세스토큰 |

아이디가 없는 경우 요청 예)

<http://115.68.110.176:8090/refund_list>?access\_token=64756ea95f85f91ab64cc8f4a6275c5f

아이디가 없는 경우 응답 예)

|  |
| --- |
| {  "result":true,  "page":1,  "rows":[  {  "game\_number":3,  "id":"test1",  "nick":"테스트1",  "is\_spinned":1,  "spin\_time":"2020-02-19 13:47:22",  "before\_user\_point":22491160,  "after\_user\_point":22491060,  "hash":"11498077bc124e739ff9f091c347d2ae789711c79420e169e9a98f8a7e814062",  "seed":"val(99,5,3,3,99|2,99,5,1,2|1,7,4,4,8)\_\_abaf5b444b5e60e63629806351ca6f4d",  "line":1,  "unit\_bet\_amount":100,  "profit":0,  "reward":0,  "amount":100,  "rtime":"2020-02-19 13:47:21",  "refund\_amount":100,  "refunded\_amount":0,  "refunded\_time":"0000-00-00 00:00:00"  },  {  "game\_number":2,  "id":"test1",  "nick":"테스트1",  "is\_spinned":1,  "spin\_time":"2020-02-19 13:47:18",  "before\_user\_point":22491260,  "after\_user\_point":22491160,  "hash":"259c6f20d9d18801185f53bace963180835acb9dc40ae954b7581b4903c77228",  "seed":"val(8,7,4,99,3|7,6,7,8,2|4,5,3,5,99)\_\_bc4060dbe5fb19ccfae018c1afbcf37d",  "line":1,  "unit\_bet\_amount":100,  "profit":0,  "reward":0,  "amount":100,  "rtime":"2020-02-19 13:47:17",  "refund\_amount":100,  "refunded\_amount":0,  "refunded\_time":"0000-00-00 00:00:00"  }  ]  } |

아이디가 있는 경우 요청 예)

http://115.68.110.176:8090/refund\_list/xescgr?access\_token=64756ea95f85f91ab64cc8f4a6275c5f

아이다가 있는 경우 응답 예)

|  |
| --- |
| {  "result":true,  "page":1,  "rows":[  {  "game\_number":3,  "id":"xescgr",  "nick":"테스트1",  "is\_spinned":1,  "spin\_time":"2020-02-19 13:47:22",  "before\_user\_point":22491160,  "after\_user\_point":22491060,  "hash":"11498077bc124e739ff9f091c347d2ae789711c79420e169e9a98f8a7e814062",  "seed":"val(99,5,3,3,99|2,99,5,1,2|1,7,4,4,8)\_\_abaf5b444b5e60e63629806351ca6f4d",  "line":1,  "unit\_bet\_amount":100,  "profit":0,  "reward":0,  "amount":100,  "rtime":"2020-02-19 13:47:21",  "refund\_amount":100,  "refunded\_amount":0,  "refunded\_time":"0000-00-00 00:00:00"  },  {  "game\_number":2,  "id":"xescgr",  "nick":"테스트1",  "is\_spinned":1,  "spin\_time":"2020-02-19 13:47:18",  "before\_user\_point":22491260,  "after\_user\_point":22491160,  "hash":"259c6f20d9d18801185f53bace963180835acb9dc40ae954b7581b4903c77228",  "seed":"val(8,7,4,99,3|7,6,7,8,2|4,5,3,5,99)\_\_bc4060dbe5fb19ccfae018c1afbcf37d",  "line":1,  "unit\_bet\_amount":100,  "profit":0,  "reward":0,  "amount":100,  "rtime":"2020-02-19 13:47:17",  "refund\_amount":100,  "refunded\_amount":0,  "refunded\_time":"0000-00-00 00:00:00"  }  ]  } |

|  |  |  |
| --- | --- | --- |
| **column** | **설명** | **비고** |
| game\_number | 게임 번호(게임 식별자) |  |
| id | 유저 아이디 |  |
| nick | 닉네임 |  |
| is\_spinned | 플레이 여부 | 0:아니오, 1:예 |
| spin\_time | 플레이(스핀) 날짜시간 |  |
| before\_user\_point | 배팅 차감전 포인트 |  |
| after\_user\_point | 배팅 차감후 포인트 |  |
| hash | 해쉬코드 |  |
| seed | hash를 만들기 위한 플레이 시드 |  |
| line | 배팅 라인수 |  |
| unit\_bet\_amount | 단위 배팅 금액 |  |
| profit | 수익 | 플레이어 수익 |
| reward | 지불 금액(당첨금) | 지불 금액 |
| refund\_amount | 환불해야하는 금액 | 정상 게임 종료시 0 |
| refunded\_amount | 환불 완료된 금액 | 환불 OpenAPI를 사용해서 환불한 금액 |
| refunded\_time | 환불한 시간 | 환불 OpenAPI를 사용해서 환불한 시간 |

# 환불

본 OpenAPI를 사용하면 game\_betting\_history테이블에 해당 회차의 환불한 금액과 환불한 시간(호출 기준)을 설정함.

|  |
| --- |
| [http://주소:포트/refund/회차(웨이브)/아이디/](http://115.68.110.176:8090/refund/1542601875/xescgr/500)환불금액?access\_token=엑세스토큰 |

요청 예)

http://115.68.110.176:8090/refund/1234/xescgr/500?access\_token=64756ea95f85f91ab64cc8f4a6275c5f

응답 예)

{"result":true,"game\_number":"1234","id":"xescgr","refunded\_amount":"500"}

# 서비스 정지

서비스 정지를 수행하면 즉각적으로 신규 플레이어의 유입을 차단하고 플레이중인 플레이어도 연결을 끊음. 또한, 서비스가 재시작 되어도 DB의 service\_stop 테이블의 데이터를 가져와서 만약 데이터가 있는 경우 계속 서비스 정지 상태를 유지함. 그러므로 반드시 서비스 시작을 통해서 서비스를 가동해야함. 서비스가 정지가 되면 모든 유저는 웹 서버 단계에서 지속적으로 referesh하는 모드가 됨. 이후에 서비스가 시작되면 다시 게임모드로 변경됨. 메시지를 넣지 않는 경우 기본 메시지로 설정된 기본 메시지는 “Service is being inspected”이며 다국어 팩에서 번역되어 한국어의 경우 “서비스 점검중입니다…”가 제공됨.

|  |
| --- |
| [http://주소:포트/service\_stop](http://115.68.110.176:8090/service_stop)?message=메시지?access\_token=엑세스토큰 |

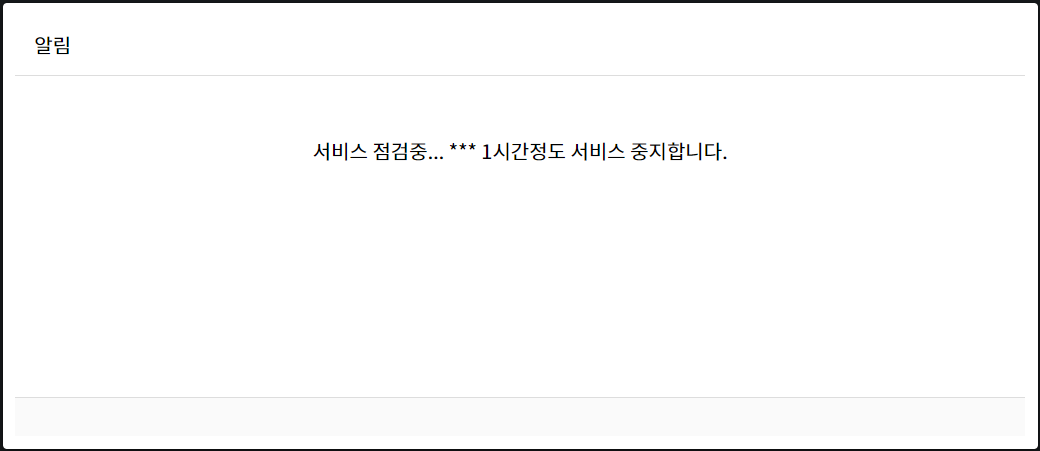
요청 예)

<http://115.68.110.176:8090/service_stop?message=1%EC%8B%9C%EA%B0%84%EC%A0%95%EB%8F%84%20%EC%84%9C%EB%B9%84%EC%8A%A4%20%EC%A4%91%EC%A7%80%ED%95%A9%EB%8B%88%EB%8B%A4>.&access\_token=64756ea95f85f91ab64cc8f4a6275c5f

응답 예)

{"result":true}

게임 클라이언트 화면 예)



# 서비스 시작

서비스를 시작함. DB의 service\_stop 테이블의 데이터를 모두 삭제하고 서비스를 가동함.

|  |
| --- |
| [http://주소:포트/service\_start](http://115.68.110.176:8090/service_start)?access\_token=엑세스토큰 |

요청 예)

<http://115.68.110.176:8090/service_start>?access\_token=64756ea95f85f91ab64cc8f4a6275c5f

응답 예)

{"result":true}

# 서비스 정지 확인

서비스 정지 명령 이후에 정말로 서비스가 정지되어 있는지를 확인.

마지막 게임의 턴이 종료되어야만 서비스를 점검을 시작할 수 있음. 서비스 정지 명령을 수행한 이후에 정말로 점검을 해도 되는지 여부는 서비스 정지 확인 명령을 통해서 확인하거나 또는 웹 게임을 직접 접근해 본 이후에 수행함. 단, 웹 게임을 직접 접근해도 테스팅 모드인 경우에는 접속이 가능할 수 있으므로 주의해야함.

|  |
| --- |
| [http://아이피:포트/is\_service\_stopped](http://115.68.110.176:8090/is_service_stopped)?access\_token=엑세스토큰 |

요청 예)

<http://115.68.110.176:8090/is_service_stopped>?access\_token=64756ea95f85f91ab64cc8f4a6275c5f

응답 예)

{"result":true,"service\_stopped":false}

{"result":true,"service\_stopped":false,"service\_stop\_message":"서비스 점검중입니다."}

{"result":true,"service\_stopped":true,"service\_stop\_message":"서비스 점검중입니다."}

message문자열이 empty가 아닌 경우 이번 게임을 마지막으로 게임을 더 이상 운영하지 않음. 만약 message문자열이 없다면, 게임은 계속해서 실행됨.

service\_stopped가 true인 경우 서비스 점검이 시작됨.