

# Play

## POST /play/settings

---

### Req

```
{  
  base coordinate: x-double, y-double, z-double  
  userid: string (경기 주최자. 즉, 현재 게임 실행중인 유저)  
  userLoginToken: string  
  balance: double  
  time: int64  
}
```

### Res

#### Success{

```
code: '200'  
message: '경기를 시작합니다.'
```

경기 코드 발행

```
}
```

#### Fail{

```
code: '402'  
message: '잔액 부족' → 결제 페이지로 이동 또는 경기 종료  
→ 최소 2시간
```

- **GET** users/checkCredit에서 잔액 조회에 성공했으나 잔액이 부족한 경우

```
}
```

## POST /play/judgement @auth

---

### Req

```
{
```

```
    userid: string (경기 주최자. 즉, 현재 게임 실행중인 유저)
    userLoginToken: string
    image: int
    gamecode: string
}
```

## Res

```
Success{
    code: '200'
    message: null

    판정 결과 음성알림
}
```

```
Fail{
    code: '412'
    message: '공 인식 실패', '스트라이크 존 리셋 필요'
    error sound 발생
}
```

## PUT /play/status @auth

---

### Req

```
{
    userid: string (경기 주최자. 즉, 현재 게임 실행중인 유저)
    userLoginToken: string
    balance: int64
    status: enum (현 게임 상태 - 0: 종료, 1: 재개, 2: 멈춤)
}
```

### Res

```
Success{
    code: '200'
```

message:

1: '경기를 재개합니다.'

2: '경기를 중단합니다' → checkCredit api

0: '경기가 종료되었습니다.'

각 상태 간 변경

}

**Fail**{

code: '402'

message: '잔액 부족' → 결제 페이지로 이동 또는 경기 종료

code: '401'

message: '권한 없음', 만약 경기를 주최한 계정이 아닌 경우 경기 현황 제어에 대한 권한을 갖지 못함

}

## GET /play/onair

---

**Res**

**Success**{

code: '200'

message: null

data:[

  'result' : 'S', 'B'

  'speed': '120'

  'gamecount' : '[[1,4],[1,2,1],[1,0,0]]'

    // [[score],[ballcount,b,s,o],[advancestate,1b,2b,3b]]

  'strike zone' : [[x1,y1],[x2,y2],[x3,y3],[x4,y4]]

]

}

```
Fail{  
  code: '408'  
  message: '서버로 부터 데이터를 받아오지 못함'  
}
```