

# Documentación Mensajes del server.

## MENSAJES MANDADOS POR EL SERVER:

### 1. CONECCTIONSTATUS:

- a. Se manda cuando el server ha recibido la petición de conexión del servidor.
- b. Campos:
  - i. connectionStatus -> true

### 2. DRAWMAP:

- a. Se manda en la creación de la sala para pintar el mapa
- b. Campos.
  - i. event -> "DRAWMAP"
  - ii. posX -> Array con las posiciones X de todos los objetos
  - iii. posY -> Array con las posiciones Y de todos los objetos
  - iv. height -> Array con todas las alturas de los objetos
  - v. width -> Array con todas las anchuras de los objetos
  - vi. myType -> Array con el tipo de objeto pasado a String entre:
    - 1. GROUND,WALL,OBSTACLE,POWERUP,OBSTACLEPOINT,SLOPE,GENERICPOWERUP,DOOR,TRAPDOOR,TRAMPOLINE,FINISH

### 3. ENTERLOBBY:

- a. Sirve para comunicar al cliente el caracol que tiene seleccionado.
- b. Campos:
  - i. event -> "ENTERLOBBY"
  - ii. snail -> String que indica el tipo de caracol, actualmente {NORMAL,TANK}

### 4. FINISH:

- a. Se manda cuando se ha acabado la carrera con el resultado
- b. Campos:
  - i. event -> "FINISH"
  - ii. winner -> boolean si has ganado o no
  - iii. time -> tiempo que has hecho
  - iv. maxTime -> Tiempo máximo para ganar el circuito en caso singlePlayer

### 5. GROUNDCOLLISION:

- a. Se manda cuando se entra en contacto por primera vez con un suelo para poner la animación.
- b. Campos:
  - i. event -> "GROUNDCOLLISION"

### 6. LOSESHIELD:

- a. Se manda cuando teníamos escudo y lo perdemos
- b. Campos:
  - i. event -> "LOSESHIELD"

## 7. MYRECORDS:

- a. Se manda ante una petición de ver tus mejores tiempos.
- b. Campos:
  - i. event -> "MYRECORDS"
  - ii. nameMap -> Array con el nombre de los distintos mapas. Son Strings
  - iii. myTime -> Array con los tiempos de cada mapa. Son Integers

## 8. OBJECTUSED:

- a. Se manda cuando se ha usado un objeto para quitar el indicador de que todavía se tiene dicho objeto y pintar su activación en la pantalla
- b. Campos:
  - i. event -> "OBJECTUSED"
  - ii. type -> String con los siguientes tipos:
    - 1. SHIELD, STAMINA, WEIGHT, LETUCCE, SPEED, INK, NULL

## 9. OBSTACLEUPDATE:

- a. Se manda cuando ha habido una actualización en el estado de los fuegos.
- b. Campos:
  - i. event -> "OBSTACLEUPDATE"
  - ii. id -> int con el índice del obstáculo
  - iii. estate -> String con los siguientes estados:
    - 1. ACTIVE, NOTACTIVE, Preactivate

## 10. OBSTACLECOLLISION:

- a. Se manda cuando el caracol choca con un obstáculo para la animación de cansado
- b. Campos:
  - i. event -> "OBSTACLECOLLISION"

## 11. RECORDS:

- a. Se manda ante la petición de ver los records de un mapa.
- b. Campos:
  - i. event -> "RECORDS"
  - ii. playerName -> Array de nombres de los jugadores. Strings
  - iii. time -> Array de tiempo de los jugadores. Integers
  - iv. mapName -> String nombre del mapa
  - v. myTime -> Integer, tu tiempo en ese mapa. 0 si no has jugado

## 12. SLOPECOLLISION:

- a. Se manda cuando se entra en contacto por primera vez con una cuesta para poner la animación.
- b. Campos:
  - i. event -> "SLOPECOLLISION"
  - ii. degrees -> int con los grados de la cuesta

## 13. SNAILUPDATE:

- a. Se manda información sobre si hay que cambiar las animaciones si te quedas sin stamina
- b. Campos:

- i. event-> "SNAILUPDATE"
- ii. runOutStamia -> true o false si te quedas sin stamina
- iii. recoverStamina -> true o false si recuperas la stamina

#### 14. TAKEPOWERUP:

- a. Se manda cuando el caracol colisiona con un power up. Te manda el power up que se ha creado, se debe borrar la caja sorpresa del escenario.
- b. Campos:
  - i. event -> "TAKEPOWERUP"
  - ii. type -> una String con el tipo de power up creado entre:
    - 1. {SHIELD,STAMINA,WEIGTH,LETUCCE,SPEED,INK,NULL}

#### 15. TICK:

- a. Se manda cada refresco de la sala con las posX e Y de nuestro caracol y su stamina
- b. Campos:
  - i. event -> "TICK"
  - ii. posX -> float con tu posX
  - iii. posY -> float con tu posY
  - iv. stamina -> float con tu stamina

#### 16. UPDATEDOOR:

- a. Se manda un evento para cambiar la animación del phaser cuando la puerta se abre o cierra.
- b. Campos:
  - i. event -> "UPDATEDOOR"
  - ii. id -> un int con el id

#### 17. UPDATETRAMPOLINE:

- a. Se manda un evento para cambiar la animación del phaser cuando el trampoline se activa.
- b. Campos:
  - i. event -> "UPDATETRAMPOLINE"
  - ii. id -> un int con el id
  - iii. estate -> String con los siguientes estados:
    - 1. ACTIVE,NOTACTIVE,PREAMIVATE

#### 18. UPDATETRAPHDOOR:

- a. Se manda un evento para cambiar la animación del phaser cuando la trampilla se abre o cierra.
- b. Campos:
  - i. event -> "UPDATETRAPHDOOR"
  - ii. id -> un int con el id

#### 19. WALLCOLLISION:

- a. Se manda cuando se entra en contacto por primera vez con una pared para poner la animación.
- b. Campos:
  - i. event -> "WALLCOLLISION"

#### 20. WAITINGROOMSTART:

- a. Se manda mientras se espera a que la sala se llene.

- b. Campos:
  - i. event -> "WAITINGROOMSTART"
  - ii. roomName -> String con el nombre

## MENSAJES MANDADOS POR EL CLIENTE:

### 1. SINGLEPLAYER:

- a. Se recibe el nombre del jugador y la sala, se envia al darle a que quieres jugar solo.
- b. Campos:
  - i. playerName
  - ii. roomName

### 2. UPDATEINPUT:

- a. Se recibe cada vez que el jugador interactua con el cliente
- b. Campos:
  - i. isSprinting
  - ii. useObject

## MENSAJES PARA EL MULTIJUGADOR:

- TICKMULTI:
  - Se anda en el tick del multi
  - Campos:
    - event -> "TICKMULTI"
    - posX -> Array con las posX de todos los jugadores
    - posY -> Array con las posY de todos los jugadores
    - stamina -> Array con la stamina
    - name -> Array con los nombres de todos los jugadores
- SNAILUPDATESMULTI:
  - Se manda para informar a los clientes de si un caracol se ha quedado sin stamina.
  - Campos:
    - event -> "SNAILUPDATESMULTI"
    - runOutStamina -> boolean para indicar que no tienes stamina
    - recoverStamina -> boolean para indicar que has recuperado la stamina
    - id -> int identificador del personaje al que le ha ocurrido
- FINISHMULTI:
  - Se le manda a alguien cuando acabe la partida.
  - Campos:
    - event -> "FINISHMULTI"
    - time -> int con tu tiempo
    - record -> integer con tu record
    - points -> int con puntos conseguidos en la carrera
    - positionNames -> Array de strings con los nombres ordenados por posición

- positionTime -> Array de Integers con el tiempo de cada uno
- WAITINGROOMSTART:
  - Se manda cuando has entrado a la sala
  - Campos:
    - event -> "WAITINGROOMSTART"
    - roomName -> String con el nombre de la sala
- SLOPECOLLISIONMULTI:
  - Se manda a todos cuando alguien choca con un acuesta
  - Campos:
    - event -> "SLOPECOLLISIONMULTI"
    - id -> Int con el id del jugador que ha chocado
    - degrees -> Integer con los grados de la cuesta.
- WALLCOLLISIONMULTI:
  - Se manda a todos cuando alguien choca con un acuesta
  - Campos:
    - event -> "WALLCOLLISIONMULTI"
    - id -> Int con el id del jugador que ha chocado
- GROUNDCOLLISIONMULTI:
  - Se manda a todos cuando alguien choca con un acuesta
  - Campos:
    - event -> "GROUNDCOLLISIONMULTI"
    - id -> Int con el id del jugador que ha chocado
- PLAYERENTER:
  - Se manda cuando un jugador ha entrado a la sala.
  - Campos:
    - event -> "PLAYERENTER"
    - name -> String con el nombre del player
- PLAYERLEFT:
  - Se manda cuando un jugador ha dejado la sala.
  - Campos
    - event -> "PLAYERLEFT"
    - name -> String con el nombre del player
- MULTIROOMSFULL:
  - Se manda si has buscado una sala pero no encuentra ninguna
  - Campos:
    - event -> "MULTIROOMSFULL"