

# Fitxa de Requisits: Control de Moviment de Jugador via WebSocket

---

Jonathan Rueda Neiro

Data: 2025-05-14

## 1. Objectiu Principal

Desenvolupar una aplicació client-servidor amb Node.js i WebSockets que permeti controlar el moviment d'un jugador mitjançant les tecles de fletxa. El client envia els moviments al servidor, que els emmagatzema en una base de dades MongoDB. Si el jugador deixa de moure's durant 10 segons, es considera que la partida ha acabat.

El servidor calcula la distància en línia recta entre el punt inicial i final de la partida i envia aquesta informació al client. Tots els esdeveniments rellevants es registren amb winston tant per consola com en un fitxer de log.

## 2. Requisits Funcionals (Què ha de fer)

- RF-01: El servidor ha de permetre connexions WebSocket entrants en un port configurable.
- RF-02: El client s'ha de poder connectar al servidor via WebSocket.
- RF-03: El client ha de capturar els esdeveniments de les tecles de fletxa i enviar la nova posició del jugador al servidor.
- RF-04: Cada moviment enviat pel client ha de contenir: playerId, timestamp, i posició (x, y).
- RF-05: El servidor ha de registrar cada moviment rebut com un document en MongoDB, amb un camp gameId que permeti agrupar els moviments d'una mateixa partida.
- RF-06: El servidor ha de detectar quan passen 10 segons sense rebre moviments del mateix jugador i considerar la partida finalitzada.
- RF-07: Un cop finalitzada la partida, el servidor ha de calcular la distància en línia recta entre el primer i l'últim punt i enviar aquesta distància al client.
- RF-08: El servidor ha d'enviar una notificació al client quan la partida acabi (amb gameId i distànciaFinal).
- RF-09: El servidor ha de mantenir l'estat de la partida activa per cada jugador de manera independent.

### 3. Requisits No Funcionals (Com ho ha de fer)

- RNF-01 (Logging): El servidor ha d'usar la llibreria winston per registrar: connexions/desconnexions de clients, moviments rebuts, errors, inici i final de partida, distància calculada. Els logs han d'anar a consola i fitxer game\_server.log.
- RNF-02 (Persistència): La base de dades utilitzada ha de ser MongoDB. Cada moviment s'ha de guardar com un document independent amb: playerId, timestamp, posició, gameId.
- RNF-03 (Configurabilitat): Paràmetres com el port del servidor i la connexió a MongoDB han de ser fàcilment modificables mitjançant variables d'entorn o fitxer .env.
- RNF-04 (Robustesa): El servidor ha de continuar funcionant encara que es produeixin errors puntuals de connexió o base de dades (registrant l'error).

### 4. Format del Missatge JSON (Client → Servidor)

```
{
  "playerId": "jugador123",
  "position": { "x": 100, "y": 200 },
  "timestamp": "2025-05-14T10:00:00.000Z"
}
```

### 5. Format de la Resposta Final de Partida (Servidor → Client)

```
{
  "type": "game_end",
  "gameId": "abc123",
  "finalDistance": 374.16
}
```

### 6. Fitxers Sol·licitats

- websocket\_server.js: implementa el servidor WebSocket, la gestió de partides, el càlcul de distància, logging amb winston i la persistència amb MongoDB.
- websocket\_client.js: connecta al servidor, captura esdeveniments de teclat (fletxes), envia les posicions i rep notificacions de final de partida.