OLD new connection flow:

1. connect to socket server
2. if client is remote host, client send 'host' message
3. server received, if there is no host, save client id as hostID and authenticated client
4. if client is player, client send 'new player' message
5. sever received, broadcast 'new player' message to all clients
6. if client received 'new player' message, do nothing
7. else if remote host received, spawn player, save to array and send 'move player' message

NEW new connection flow:

1. connect to socket player
2. if server in remote mode, if hostID is 'none', send host = true message to the first client, hostID = client ID
3. else if hostID is not none, send host = false to client
4. client received host message, stored host value then run gameLoop()

Username/UserID getting and storing flow:

1. client login successfully, server send login message contain username to client
2. client received, client emit 'new player' message contain username and userID to server
3. server emit 'new player' message contain socketID, userID and username to host
4. host spawn player, giving it X/Y and direction, add to array then emit 'move player' contain id, username, x, y, direction
5. server relay the message, all client received then find if player existed, add or modified with the received value

Remote host flow:

Player move interpolation

1. onKeyDown send to server
2. server relay to host
3. host player.setMoving(true) and emit to all client
4. client player.setMoving(true)
5. host and client auto move the player
6. onKeyUp or collision send to server
7. server player.setMoving(true) and emit coordinate to all client
8. client set all the parameter server sent
9. both client and server stop auto move player

Game logic running at a fixed rate

1. get time from last loop to now (delta)
2. divide delta to (1000/60) to get how many 'tick' (60tick/second) need to be run (range from 0/1/2/5...)

Old move player

1. client emit input to server, at the same time move the player on screen
2. server relay to host
3. host move the player
4. client emit stop to server
5. server to host, host stop the player and send the coordinate to all client
6. client will 'teleport' the player to the coordinate server sent

NEW move player

1. input emit from client
2. server relay to host
3. host calculate time and place the player will be, include in the collision

Render bullet destruction flame:

1. when mapCollision is true
2. use the same technich in move player to draw each stage of the flame in 0.3 second
3. if theres 2 collision at the same time?
4. RenderingBulletDestroyed run all the time check 4 collision

Separate into client/server/common