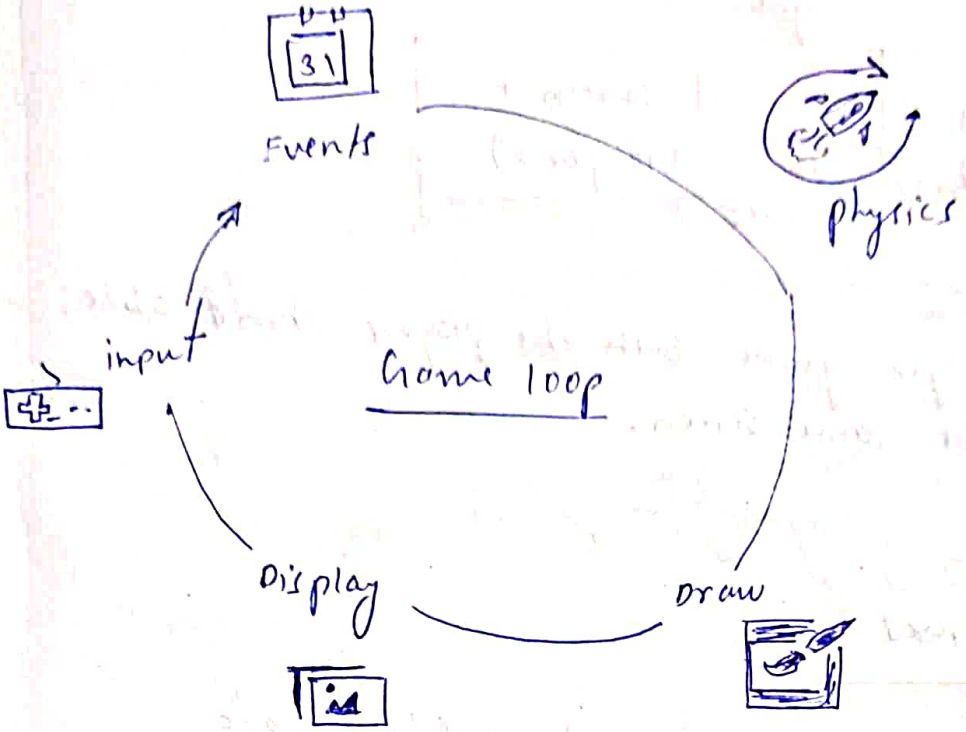
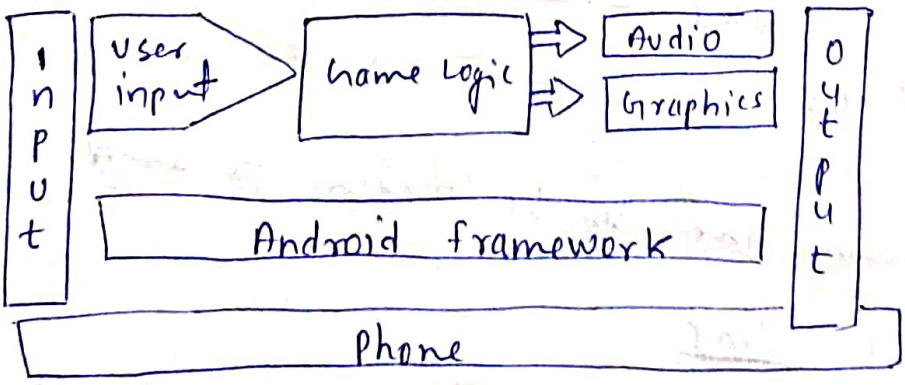
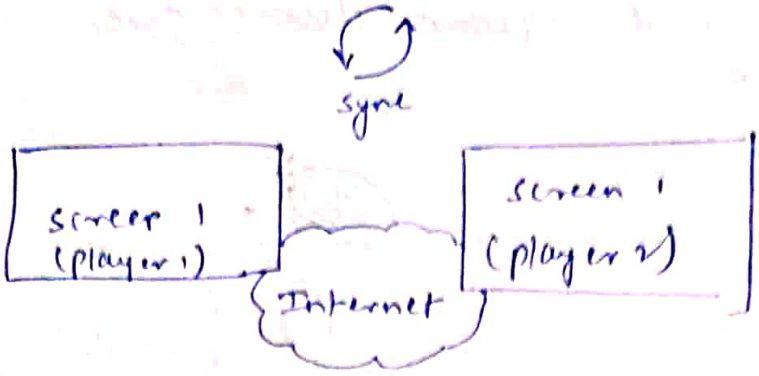
Online games System - Architecture.




```
while(true) {
    check_input()
    update_game_state()
    render_screen()
}
```

Note:- Tools :- Unity / UnrealEngine



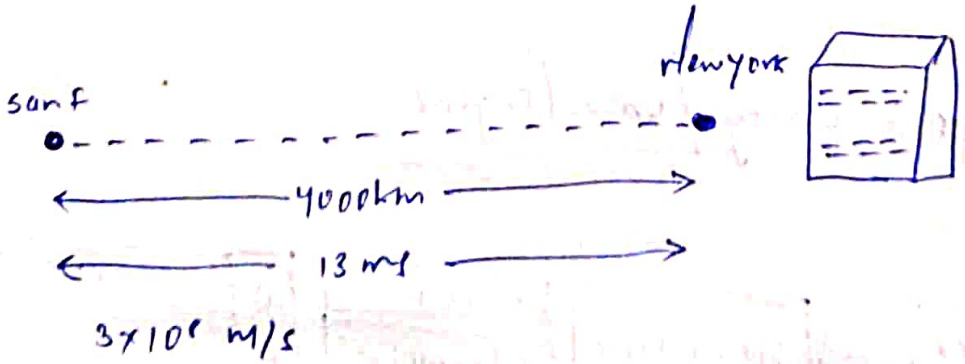


① In multiplayer game both the players should able to see the same screen.

Types of Games

① strategic games :- clash of Clans, etc
② slow turn games :- chess, carrom ... etc.
③ first person games :- pubg, coo ... etc.

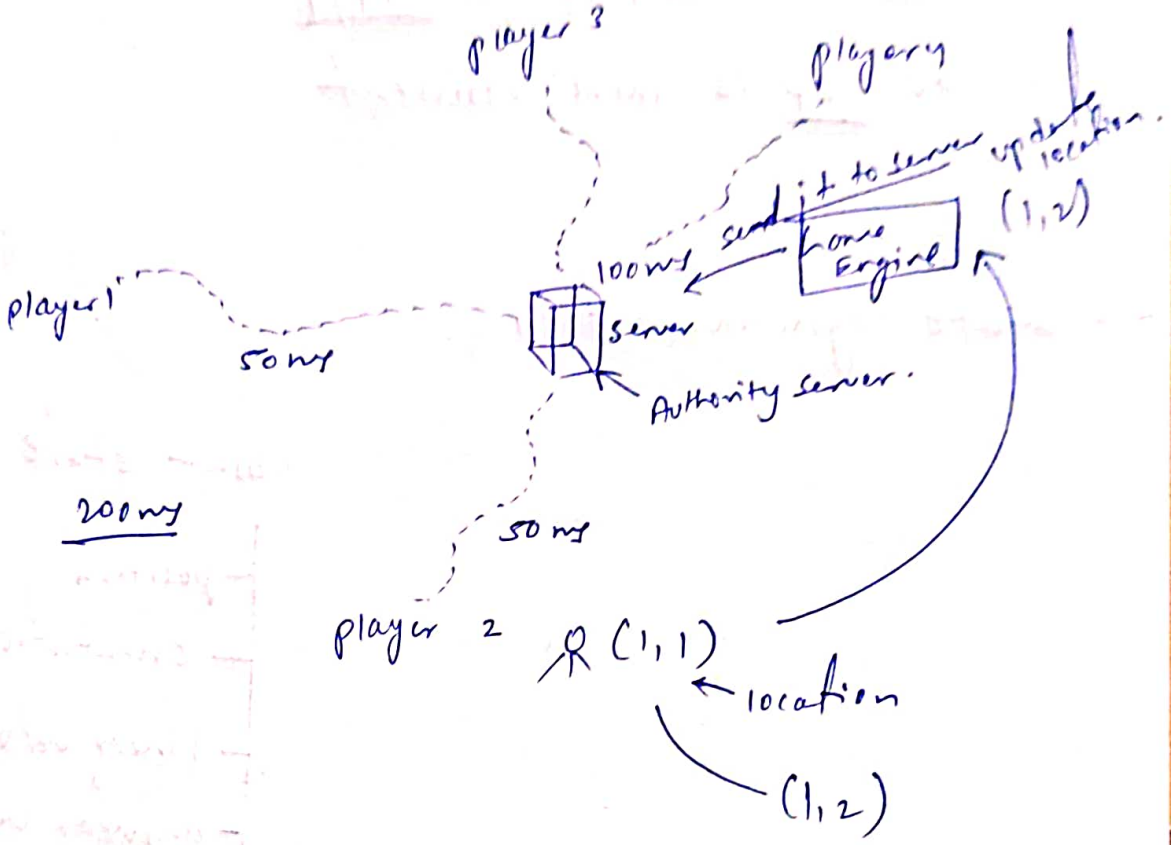
* suppose user1 wants to send the data user 2



Here, minimum time can be taken to send message is 13ms

less than 13ms is not possible because light speed is 3x10⁸ m/s.

- To send a message also, it should go through the OSI layers which is an protocols.



Authority Server

1. Connects all player
2. Responsible for all operations
3.

Game Engine

1. Suppose one player change its location then G.E update it send it to the server and server update to all the players.

→ Deterministic Lock step.

Note:- If the bandwidth is not strong then we can lose packets in TCP/IP

So UDP is more efficient

→ state synchronization.

Object state

- position
- orientation
- Linear velocity
- angular velocity

sample code that how, objects are working.

```
function handleInput (Input_input) {
    if(input == press_B)
     {
        if( !isJumping_ && !isDucking_)
          {
             // jump ....
          }
     }
    else if (input == press_Down)
     {
        if( !isJumping_)
          {
             isDucking = true ;
             setGraphics (Image_Duck );
          }
        else
          {
             isJumping = false;
             setGraphics (Image_Dive);
          }
     }
    else if (input == Release_Down )
      {
         if (isDucking_)
           {
              // stand ---
           }
      }
}.
```