

C++ Code

```
1 //  
2 // server.cpp  
3 // TP C++  
4 // Eric Lecolinet - Telecom ParisTech - 2016.  
5 //  
6  
7 #include <memory>  
8 #include <string>  
9 #include <iostream>  
10 #include <sstream>  
11 #include <algorithm>  
12 #include "tcpserver.h"  
13 #include "MediaManager.h"  
14  
15 const int PORT = 3331;  
16  
17  
18 int main(int argc, char* argv[])  
19 {  
20  
21     auto myManager = std::make_shared<MediaManager>();  
22     myManager->createPhoto("Photo1", "montsouris.jpg", 48.8, 2.3);  
23     myManager->createVideo("Video1", "video.mp4", 120);  
24  
25     auto* server = new TCPServer([&](std::string const& request, std::string& response) {  
26  
27         std::cout << "Requête reçue: " << request << std::endl;  
28  
29         std::stringstream ss(request);  
30         std::string command, name;  
31         ss >> command >> name; // Découpe "SEARCH nom" ou "PLAY nom"  
32  
33         std::stringstream resStream;  
34  
35         if (command == "SEARCH") {  
36             // On utilise une stringstream pour capturer l'affichage  
37             myManager->displayObject(name, resStream);  
38             response = resStream.str();  
39         }  
40         else if (command == "PLAY") {  
41             myManager->playObject(name, resStream);  
42             response = resStream.str();  
43         }  
44         else {  
45             response = "Unknown command: " + command;  
46         }  
47  
48         // IMPORTANT : Nettoyer les '\n' et '\r' car ils cassent le protocole  
49     });  
50 }
```

```
50     std::replace(response.begin(), response.end(), '\n', ' ');
51     std::replace(response.begin(), response.end(), '\r', ' ');
52
53     return true; // Garder la connexion ouverte
54 );
55
56 std::cout << "Starting Server on port " << PORT << std::endl;
57 int status = server->run(PORT);
58 if (status < 0) {
59     std::cerr << "Could not start Server on port " << PORT << std::endl;
60     return 1;
61 }
62
63 return 0;
64 }
65
```