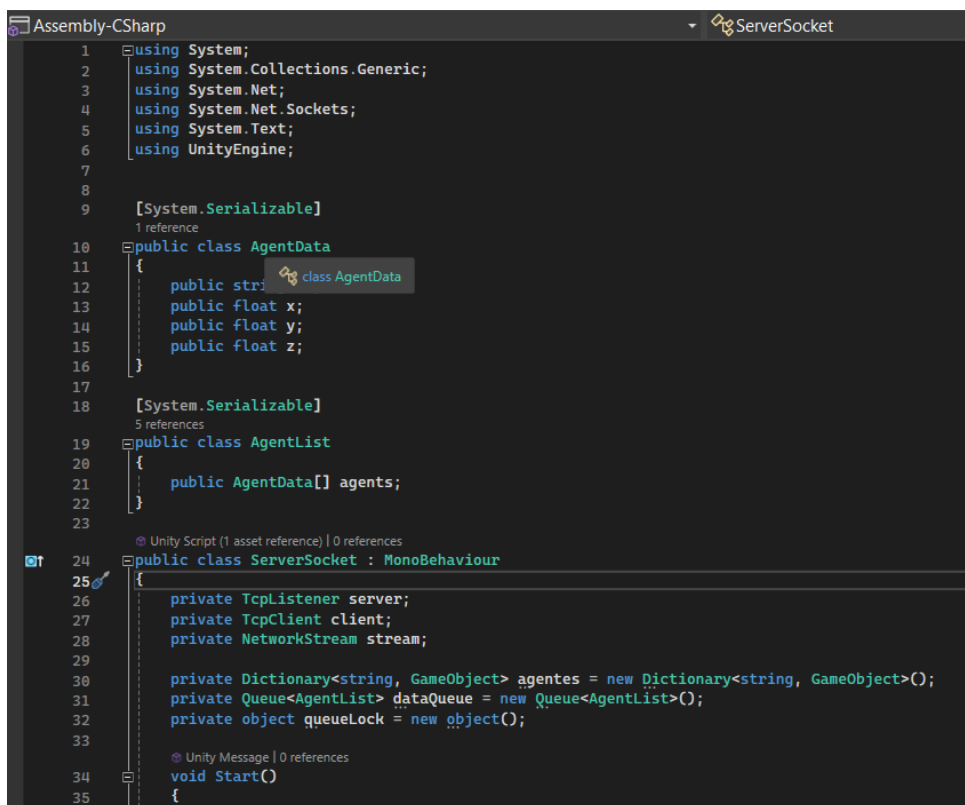


Servidor en un empty object



Assembly-CSharp

ServerSocket

Unity Message References

```
34 void Start()
35 {
36     foreach (GameObject obj in GameObject.FindObjectsOfType<GameObject>())
37     {
38         if (obj.name.StartsWith("agent_"))
39         {
40             agentes[obj.name] = obj;
41             Debug.Log("Agente registrado: " + obj.name);
42         }
43     }
44
45     server = new TcpListener(IPAddress.Any, 5005);
46     server.Start();
47     Debug.Log("Servidor iniciado en el puerto 5005");
48     server.BeginAcceptTcpClient(OnClientConnect, null);
49 }
50
51 1 reference
52 void OnClientConnect(IAsyncResult ar)
53 {
54     client = server.EndAcceptTcpClient(ar);
55     stream = client.GetStream();
56     Debug.Log("Cliente conectado");
57
58     byte[] buffer = new byte[1024];
59     stream.BeginRead(buffer, 0, buffer.Length, OnRead, buffer);
60 }
61
62 2 references
63 void OnRead(IAsyncResult ar)
64 {
65     int bytesRead = stream.EndRead(ar);
66     if (bytesRead <= 0) return;
67
68     byte[] buffer = (byte[])ar.AsyncState;
69     string json = Encoding.UTF8.GetString(buffer, 0, bytesRead);
70     Debug.Log("JSON recibido: " + json);
71 }
```

```

70     try
71     {
72         AgentList agentList = JsonUtility.FromJson<AgentList>(json);
73         lock (queueLock)
74         {
75             dataQueue.Enqueue(agentList);
76         }
77     }
78     catch (Exception e)
79     {
80         Debug.LogError("Error al parsear JSON: " + e.Message);
81     }
82
83     stream.BeginRead(buffer, 0, buffer.Length, OnRead, buffer);
84 }
85
86 // Unity Message | 0 references
87 void Update()
88 {
89     lock (queueLock)
90     {
91         while (dataQueue.Count > 0)
92         {
93             AgentList list = dataQueue.Dequeue();
94             foreach (var agent in list.agents)
95             {
96                 if (agentes.ContainsKey(agent.id))
97                 {
98                     agentes[agent.id].transform.position = new Vector3(agent.x, agent.y, agent.z);
99                 }
100             }
101         }
102     }
103 }

```

```

102     }
103
104 // Unity Message | 0 references
105 void OnApplicationQuit()
106 {
107     stream?.Close();
108     client?.Close();
109     server?.Stop();
110 }

```