Role play as an expert in C++ and web development, for building tools and libraires. We want to develop a simple C++ DLL/Shared Object that is capable of connecting to a native c++ application to an external service, and this external service can also connect to a web browser allowing it to visualize any information coming from the native c++ application, and/or send events back to the native application from the browser to control the native c++ app execution and/or events.

What would be a strategy to use for the external application, that will allow for easy integration/use between both a native c++ app and a browser.

What would be a good simple quick interface for a c++ class to wrap around a WebSocket server, that would allow for the basic functionality:  
  
Start a server if it is not online

Shutdown the server

Connect to server

Send data to server via JSON objects

Receive data back from server that could be sent from a browser.

Please provide a IWebsocket.cpp and IWebsocket.h that matches your proposed interface

Could you also provide a schema that we could use for the JSON messaging, so that we could keep all communiation between the c++ app and the browser consistent.

Can you provide a design document that could be used to help build Webscoket interface code as well as give detailed instructions on how to create and parse messages to and form the server

Using the design document attached, can you provide a tier 1 basic implmanetation of a node server that will enable basic functionalities so we can demonstrate that the basic communication pipeline between browser, native c++ app are working.

mkdir websocket-demo

cd websocket-demo

npm init -y

npm install ws uuid