Technical achievements

The most significant technical achievement was implementing real time sound communication. The infrastructure which captures sound from an in-built microphone, sends it over a TCP connection to a server which dispatches it to other clients participating in the conversation who play it back is non trivial.
Even though we leveraged the power of Unity for most of the 3D sound processing, combining this with a VoIP system is also a significant technical achievement.

Implementation details

Server

The server is written in JavaScript/Node.js and takes advantage of the asynchronous TCP/IP implementation provided by Node.js. It simply keeps track of all clients connected and forwards any incoming packets from one client to all the others using stream piping API provided by Node.js. The server is very simple (~50LOC) and hence doesn't need any design patterns.

Unity Application

This is written in C#. It allows users to login to the application.As soon as a user logs in they are assigned a user ID which is a number between 0-10.After the user logged in, if they were to make a call connection is made to tcp server. The tcp client is asynchronous and successfully receives and sends packets to the server. At the same time the microphone starts recording and these are put into byte array packets which is send to the server with the assigned identification number. The packets received are identified by their id number. Accordingly to the id number, these are passed onto the objects which appears on the screen as the people you are talking to. The user interface allows user to move these objects in 3D space. As they are being moved the sound source will be moved accordingly.

In the unity application we have implemented the .NET design patterns in our client. The .NET design patterns are separated into 3 different sets which are Creational, Structural and Behavioural.
We have implemented iterator which is part of the behavioural pattern.We sequentially access the packets that are stored in a byte array queue.