1. Language chosen and why?

I used JavaScript for the given assessment Node Js for backend, React Js for front end and express to create APIs. My main purpose was to enhance and revisit my skills with JavaScript with building apps from scratch and wanted to explore the assessment as well. My overall Idea was to be efficient in coding with ability to reuse code and focus on getting the work done with less efforts. The reason of choosing react js as frontend framework was to have the ability to fetch api continuously without restarting the app and focusing on the look of front end. Node js was a good choice for backend to have the ability use less amount of code and make the process of building backend faster using inbuild modules and libraries, I also used express as a middleware to make api connection between to react js from node js since express since helps making the server side application faster and easier and it is a good node js framework.

To read the file I used XMLtoJS to convert xml to javascript object. Here I have hardcoded the xml response to save the time and thought process. It can be done better via using iterator like getElementbyid, etc on the dom parser. Once the res collected it is stored to global varable and being exported though modules using module.exports. Further given modules are access by express js. Once called, variable are sent as a response to url( localhost: 8080) using get requests.

Later using fetch API from react app running on port 3000 urls are fetched to gathered the required data from backing to display it on html front page just the way they are for the proof of concept.

1. High level design. (A visio diagram or hand drawn image)

While exploring the assessment I came up with idea to build an web app video and calling application

More of like the image shown below:

A collage of a person

Description automatically generated with low confidence

Here is a work flow diagram of the project showing how the xml data is converted to and processed to the frontend react:

Graphical user interface, application

Description automatically generated

Starting from the designing, the application will have:

## NavBar

NavBar with a dropDown on the left side, a searchBar in the middle and user image on right hand side. Code For the NavBar is located in devCode/Client/my-app/src/Component/Navbar

Graphical user interface

Description automatically generated

## DropDown

Dropdown has three main components Call, Video and Settings. Code for DropDown is in devCode/Client/my-app/src/Component/DropDown

Graphical user interface, text

Description automatically generated

Video:

Video menu collects the following

Audio: Input, Output, Microphone: mute , Volume: 70

Conference: Call.BookingId,

Peripherals. ConnectedDevice.Status & Name' to turn on and off or switch path manually

Video: ActiveSpeaker, Presentation, Selfview

Cameras: Camera, SpeakerTrack

The Path for Video is: devCode/Client/my-app/src/Component/Videos

Graphical user interface

Description automatically generated

## Calls:

Calls menu collectd the following

Audio: Input, Output, Microphone: mute , Volume: 70

Conference: Call.BookingId,

Peripherals. ConectedDevice.Status & Name' to turn on and off or switch path manually

The path for Calls is : devCode/Client/my-app/src/Component/Calls

Graphical user interface

Description automatically generated

Both call.js and video.js fetches the xml using express

1. How to deploy the solution.
   1. GO to ../devCode/server> node app.js, you will see the console.logs for data retrieved
   2. Go to ../devCode/Client/my-app/src> npm start this is the react front end
2. Screenshots of formatted data presentation, in case we are not able to run the solution.

1 AudioInput: { '$': { item: '2', maxOccurrence: 'n' }, EcReferenceDelay: [ '0' ] }

2 AudioOutputOn {

'$': { item: '5', maxOccurrence: 'n' },

Channels: [ '2' ],

Connector: [

{ \_: 'Line.1', '$': [Object] },

{ \_: 'Line.2', '$': [Object] },

{ \_: 'Line.3', '$': [Object] },

{ \_: 'Line.4', '$': [Object] },

{ \_: 'HDMI.1', '$': [Object] }

],

Input: [

{ '$': [Object], Gain: [Array] },

{ '$': [Object], Gain: [Array] },

{ '$': [Object], Gain: [Array] },

{ '$': [Object], Gain: [Array] }

],

Loudspeaker: [ 'On' ],

Name: [ 'Loudspeaker' ],

VolumeControlled: [ 'On' ]

}

3 Volume 70

4 BookingId 15909

5 DeviceStatus [ [ 'Connected' ], [ 'Connected' ], [ 'Connected' ], [ 'Connected' ] ]

6 DeviceName [

[ 'Cisco TelePresence Touch' ],

[ 'Precision 60 Camera' ],

[ 'Precision 60 Camera' ],

[ 'SpeakerTrack 60' ]

]

7 ActiveSpeaker [ 'UpperCenter' ]

8 CameraInfo [

{

'$': { item: '1', maxOccurrence: 'n' },

Capabilities: [ [Object] ],

Connected: [ 'True' ],

Flip: [ 'Off' ],

MacAddress: [ '18:8B:9D:D4:73:8C' ],

Manufacturer: [ 'Cisco' ],

Model: [ 'Precision 60 Camera' ],

SerialNumber: [ 'FTT192804FZ' ],

SoftwareID: [ 'HC8.0.1.1e47efe, 2016-01-18' ]

},

{

'$': { item: '2', maxOccurrence: 'n' },

Capabilities: [ [Object] ],

Connected: [ 'True' ],

Flip: [ 'Off' ],

MacAddress: [ '18:8B:9D:D4:97:08' ],

Manufacturer: [ 'Cisco' ],

Model: [ 'Precision 60 Camera' ],

SerialNumber: [ 'FTT192804IH' ],

SoftwareID: [ 'HC8.0.1.1e47efe, 2016-01-18' ]

},

{

'$': { item: '3', maxOccurrence: 'n' },

Capabilities: [ [Object] ],

Connected: [ 'False' ],

MacAddress: [ '' ],

Manufacturer: [ '' ],

Model: [ '' ],

SerialNumber: [ '' ],

SoftwareID: [ '' ]

},

{

'$': { item: '4', maxOccurrence: 'n' },

Capabilities: [ [Object] ],

Connected: [ 'False' ],

MacAddress: [ '' ],

Manufacturer: [ '' ],

Model: [ '' ],

SerialNumber: [ '' ],

SoftwareID: [ '' ]

},

{

'$': { item: '5', maxOccurrence: 'n' },

Capabilities: [ [Object] ],

Connected: [ 'False' ],

MacAddress: [ '' ],

Manufacturer: [ '' ],

Model: [ '' ],

SerialNumber: [ '' ],

SoftwareID: [ '' ]

},

{

'$': { item: '6', maxOccurrence: 'n' },

Capabilities: [ [Object] ],

Connected: [ 'False' ],

MacAddress: [ '' ],

Manufacturer: [ '' ],

Model: [ '' ],

SerialNumber: [ '' ],

SoftwareID: [ '' ]

},

{

'$': { item: '7', maxOccurrence: 'n' },

Capabilities: [ [Object] ],

Connected: [ 'False' ],

MacAddress: [ '' ],

Manufacturer: [ '' ],

Model: [ '' ],

SerialNumber: [ '' ],

SoftwareID: [ '' ]

}

]