Next-Gen Classroom Middleware
User Guide

Team 11

Hoi Lung Lam, Sam Merren, Shahmir Ali Syed, Song Xen Hwa

Table Contents

End User Guide

Running the application

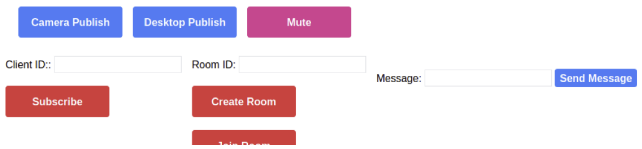
Theoretically after future implementations and improvements, the server would be hosted on a remote server and the client would just need to go to a website/link to access the server functionalities however since we are running it on local host, we need to do a yarn start after an “admin” has set up the server which is running currently (talked about later).

Once the server is up, we open a new terminal while keeping the current terminal in the background. In the new terminal, go into the client directory with “cd client”. Then proceed to enter “yarn start” to start the application.

Once the front end has loaded up we need the user to either create a room or join a room created by another user. More on rooms will be discussed below:

Client Directory

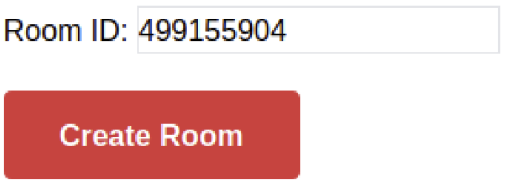
After running the commands in both the server and client directory as image above, the device will open a browser window which would look like the image below which represents the application.



Rooms

Creating Room

To create a room, just click the “Create Room” button which will generate a random integer of length 9



Joining Room

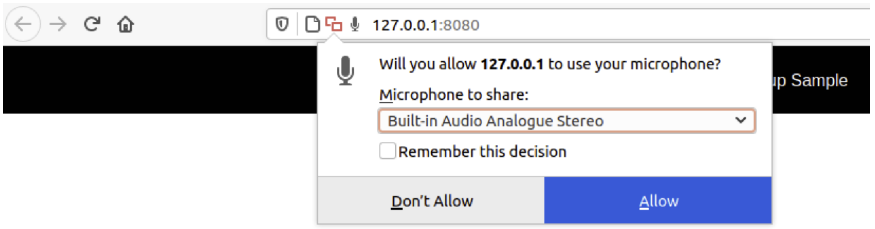
Ensure that the room id text box is not empty. To join the room, click the “Join Room” button.

Leave Room

To leave the room, just click the “Leave Room” button.

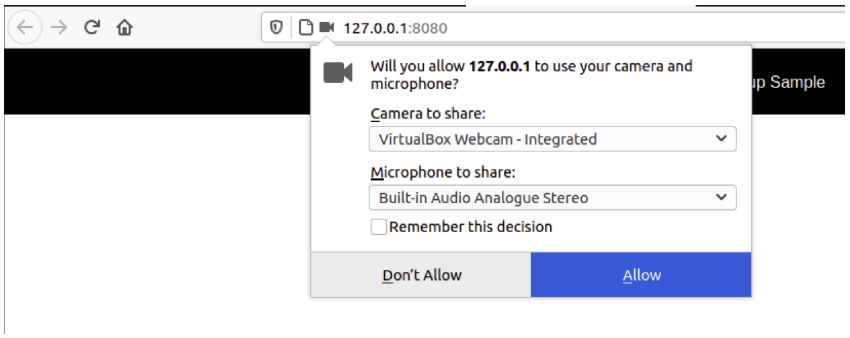
Enabling microphone

To enable the microphone, a pop up will appear which allows the user to select which microphone they preferred. Then proceed by clicking allow.



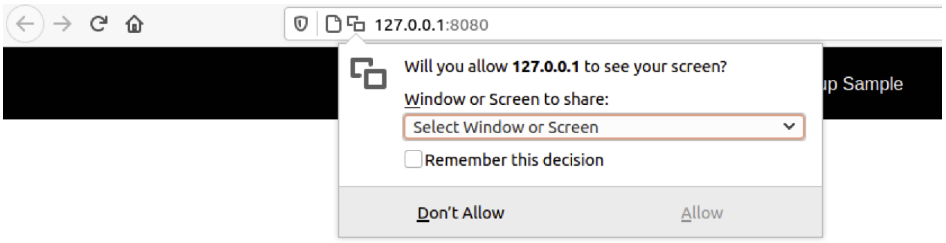
Enabling webcam

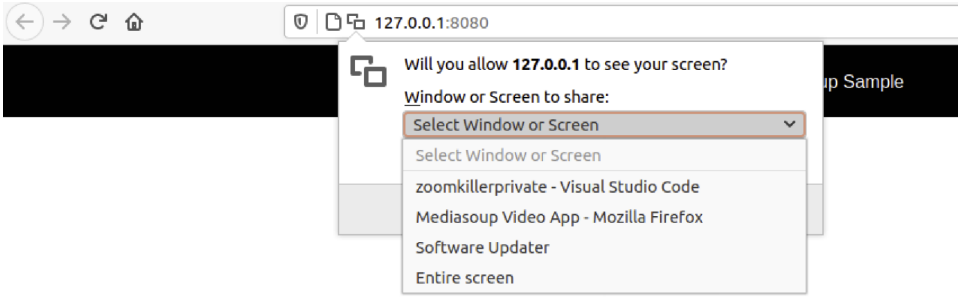
To enable the webcam, the user will need to join a room. After joining a room, click on the “Camera Publish” button. Then, a pop up will appear which allows the user to select which webcam they preferred. Proceed by clicking allow.



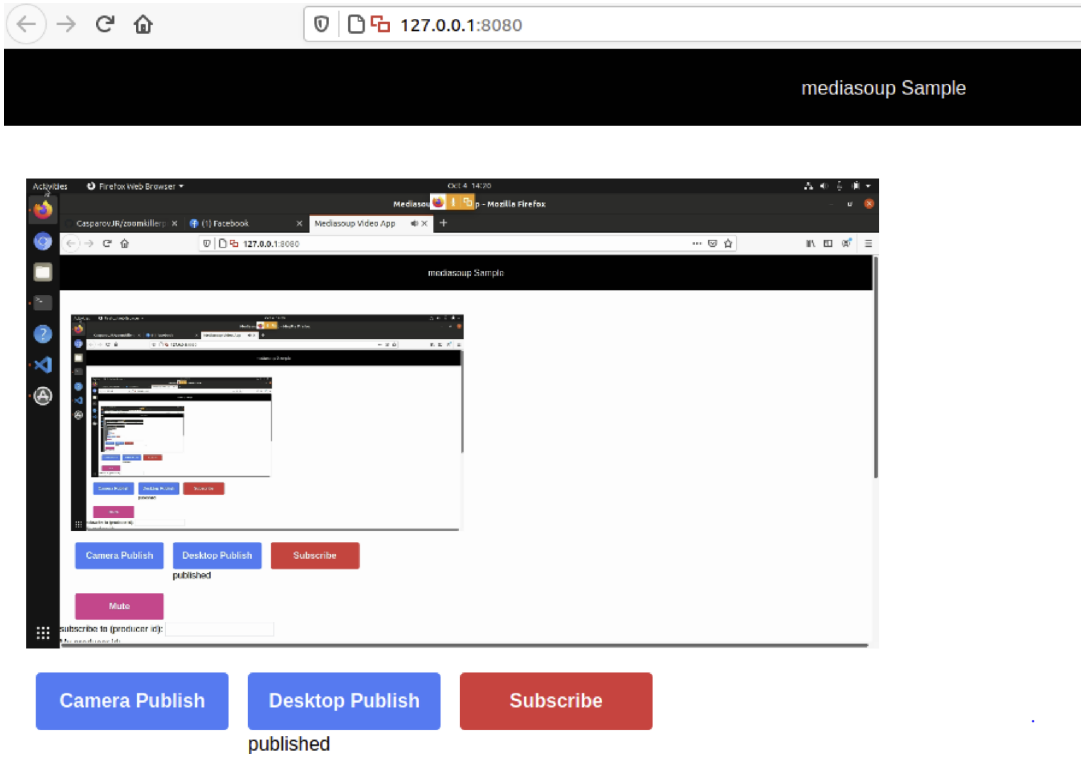
Enabling Screen Sharing

To enable screen sharing, the user will need to join a room. After joining a room, click on the “Desktop Publish”button. Then a pop up will appear to prompt the user to select which screen they would like to share. Select the desired screen to share and click allow.



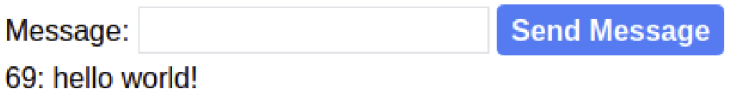


Once screen sharing is allowed, it will result as the image below.



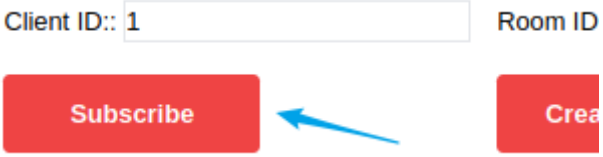
Sending Messages

First, ensure that the client ID is present. Then, input any message in the text box. To send the message, click on the “send Message” button. As a result, it will display the client id and the message inputted below the text box.



Subscribing

To watch others’ published video/audio stream within the room, click the ‘Subscribe’ button. Then others’ screens will be shown at the bottom section with their id.



Technical Guide

Cloning from GitHub

To clone the project from the group's repository in GitHub, You will first need to install Github packages in your Linux environment. After downloading the packages, clone the repository using the command "git clone ("http of the github")". If it manages to clone successfully, the output will be similar to the image shown below.

```
student@fit-vm:~$ git clone https://github.com/CasparovJR/zoomkillerprivate.git
Cloning into 'zoomkillerprivate'...
Username for 'https://github.com': SongXenHwa
Password for 'https://SongXenHwa@github.com':
remote: Enumerating objects: 19967, done.
remote: Counting objects: 100% (19967/19967), done.
remote: Compressing objects: 100% (14407/14407), done.
remote: Total 19967 (delta 4253), reused 19925 (delta 4219), pack-reused 0
Receiving objects: 100% (19967/19967), 52.69 MiB | 1.60 MiB/s, done.
Resolving deltas: 100% (4253/4253), done.
student@fit-vm:~$
```

Application Installation

To view the contents/ codes of the application. You will need to have Microsoft Visual Studios and install node.js packages in your Linux environment. To download Visual Studios, run the following below.

```
student@fit-vm:~$ sudo snap install code --classic
code 7f6ab548 from Visual Studio Code (vscode✓) installed
student@fit-vm:~$
```

To install node.js packages, run the following in the terminal

```
student@fit-vm:~$ sudo apt install npm
```

To be able to run the application, users will need to download yarn packages using the command below. Go into the cloned repository directory in the terminal and run the command.

```
student@fit-vm:~/zoomkillerprivate$ sudo npm install --global yarn

> yarn@1.22.15 preinstall /usr/local/lib/node_modules/yarn
> :; (node ./preinstall.js > /dev/null 2>&1 || true)

/usr/local/bin/yarn -> /usr/local/lib/node_modules/yarn/bin/yarn.js
/usr/local/bin/yarnpkg -> /usr/local/lib/node_modules/yarn/bin/yarn.js
+ yarn@1.22.15
updated 1 package in 5.906s
student@fit-vm:~/zoomkillerprivate$
```

```
student@fit-vm:~/zoomkillerprivate$ cd server
student@fit-vm:~/zoomkillerprivate/server$ yarn install
yarn install v1.22.15
[1/4] Resolving packages...
[2/4] Fetching packages...
[3/4] Linking dependencies...
warning " > ts-node@10.1.0" has unmet peer dependency "@types/node@*".
[4/4] Building fresh packages...
Done in 734.68s.
student@fit-vm:~/zoomkillerprivate/server$
```

To install yarn in the server directory, first go into the server directory of the clone repository by using “cd server”. Then input “yarn install” to install yarn dependencies.

Then similarly go into the client repository and do another yarn install, one problem may pop up which is an issue with the current node version being incompatible.

Issues Regarding Node Version

There might be issues related to the Node version which does not allow users to run and start the application. To solve this issue, go to the directory of the cloned project. Then input “sudo npm cache clean --force” in the terminal. Next, enter “sudo npm install -g n” which will reinstall node. After those two, enter “sudo n stable” to update the node version to the latest version.

```
student@fit-vm:~/zoomkillerprivate$ sudo npm cache clean -force
npm WARN using --force I sure hope you know what you are doing.
student@fit-vm:~/zoomkillerprivate$ sudo npm install -g n
/usr/local/bin/n -> /usr/local/lib/node_modules/n/bin/n
+ n@7.5.0
added 1 package from 2 contributors in 1.735s
student@fit-vm:~/zoomkillerprivate$ sudo n stable
  installing : node-v14.18.0
       mkdir : /usr/local/n/versions/node/14.18.0
       fetch : https://nodejs.org/dist/v14.18.0/node-v14.18.0-linux-x64.tar.xz
   installed : v14.18.0 (with npm 6.14.15)

Note: the node command changed location and the old location may be remembered in your current shel
         old : /usr/bin/node
         new : /usr/local/bin/node
To reset the command location hash either start a new shell, or execute PATH="$PATH"
student@fit-vm:~/zoomkillerprivate$
```

To check if the node version has been updated, run “node --version” in the terminal. As when the user guide is created (6 October 2021), the node version (v14.18.0) is able to run the application.

```
student@fit-vm:~/zoomkillerprivate$ node --version
v14.18.0
```

To get the server set up and running, first go into the server directory by entering “cd server”. After going into the server directory, start the server by doing “yarn dev” in the terminal. This starts the server which is ready for connections (this is the point the end user can then connect to).

Server Directory

```
student@fit-vm:~/zoomkillerprivate$ cd server
student@fit-vm:~/zoomkillerprivate/server$ yarn dev
yarn run v1.22.15
$ DEBUG=mediasoup:* ts-node src
Server started on port 8002
```