

## بسمه تعالی

فاز دوم مینی پروژه

نحوه ی اجرا کردن :

ابتدا envirement مجازی را با دستور زیر فعل میکنیم تا همه ی وابستگی ها فعال شود و نیاز به نصب جداگانه نداشته باشد .

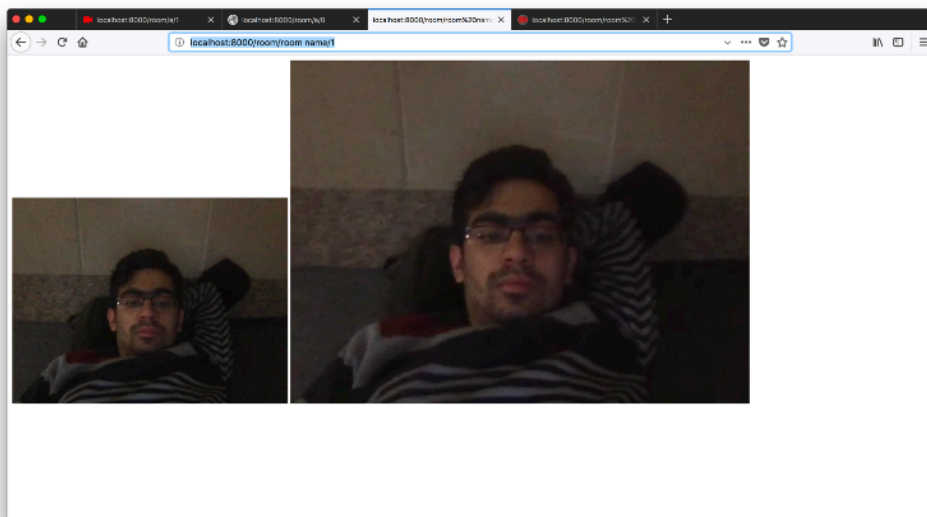
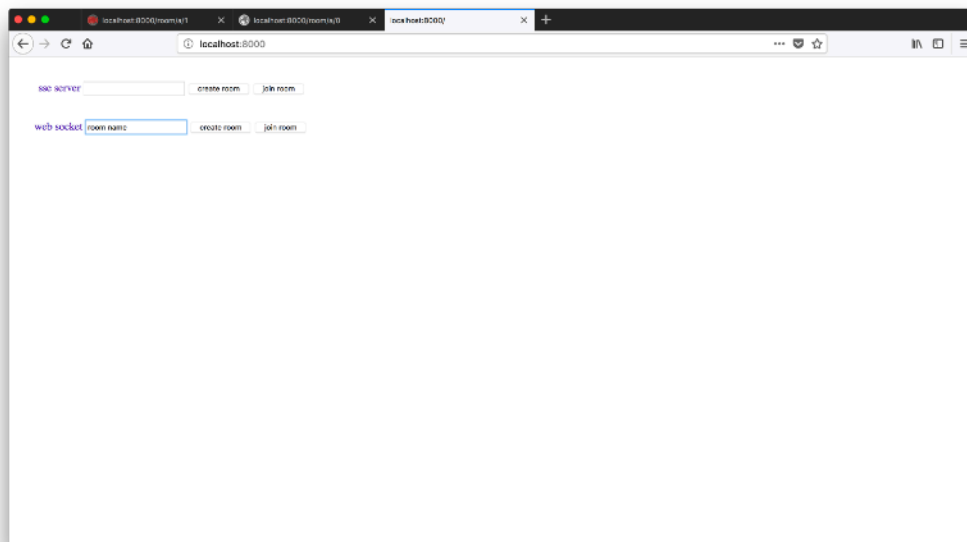
`. venv/bin/activate`

بعد سرور پایتون را با دستور زیر اجرا میکنیم .

`gunicorn main:app --worker-class gevent -w 1 --bind 0.0.0.0:8000`

سپس با مراجعه به ادرس `localhost:8000` صفحه ی زیر برای شما باز میشود.

نام روم جدید را وارد کرده و میتوانی روم جدید بسازید یا جویین شوید.



کد و پیاده سازی :

زمانی که صفحه لود میشود یک کانکشن وب سوکت با سرور برقرار میشود.  
و تابع سند یک جیسون را از طریق سوکت برای سرور ارسال میکند.

```
var socket = io.connect('http://' + document.domain + ':' + location.port + "/webSigChannel");
```

```
function sendbySocket(json, type){  
    console.log("send by socket called");  
    socket.emit(type, json);  
}
```

توابع ارسال و دریافت سیگنال ice را به مانند شکل زیر تغییر میدهیم.

```
function gotIceCandidate(event) {  
    if(event.candidate != null) {  
        sendbySocket(JSON.stringify({'ice': event.candidate, 'uuid': uuid, "room":"{{name}}"}), "ice");  
    }  
}
```

```
socket.on("ice", function(data){  
    console.log("in ice data is ", data);  
    var json = JSON.parse(data);  
    console.log("ice is :", json.ice );  
    if(json.uuid == uuid || json.room != "{{name}}")  
return;  
    peerConnection.addIceCandidate(new  
RTCIceCandidate(json.ice)).catch(errorHandler);  
})
```

توابع ارسال و دریافت سیگنال sdp را به مانند شکل زیر تغییر میدهیم.

```
function createdDescription(description) {
    console.log("createdDescription")
    peerConnection.setLocalDescription(description).then(function()
    {
        sendbySocket(JSON.stringify({'sdp':
        peerConnection.localDescription, 'uuid': uuid,
        "room":"{{name}}"}), 'sdp');
    }).catch(errorHandler);
    }

socket.on("sdp", function(data){
    var json = JSON.parse(data);
    // console.log("parsed is :", json.uuid );
    if(json.uuid == uuid || json.room != "{{name}}") return;
    console.log("i got other side sdp message", data);
    peerConnection.setRemoteDescription(new
    RTCSessionDescription(json.sdp)).then(function() {
    peerConnection.createAnswer().then(createdDescription).catch(err
    orHandler);
    }).catch(errorHandler);
    })
})
```

از uuid برای شناسایی فرستنده استفاده میشود و به صورت زیر ساخته میشود.

```
function uuid() { //az git
    function s4() {
        return Math.floor((1 + Math.random()) *
    0x10000).toString(16).substring(1);
    }
    return s4() + s4() + '-' + s4() + '-' + s4() + '-' + s4() +
    '-' + s4() + s4() + s4();
}

var uuid = uuid();
```