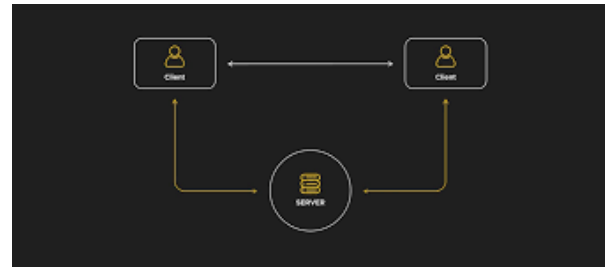


VIDEO CHAT APP - MAILER



What is our GOAL for this CLASS?

In this class, we have integrated the mailer API with the client side and tested the functionality to complete the video chat app.

What did we ACHIEVE in the class TODAY?

- Integrating the mailer API with the client
- End to end testing of the application

Which CONCEPTS/ CODING BLOCKS did we cover today?

- jQuery
- Ajax
- Heroku Deployment

How did we DO the activities?

In this class, we write some code in the client side (**index.ejs** and **script.js**) to integrate it with the mailer API that we have created in the last class!

Activity:

1. Our UI looks like this -



2. Add HTML code into **index.ejs** file to create one more icon next to the video and audio icons for inviting another user -

```
<div class="col-sm-12 col-md-12 col-lg-12 options">
  <!-- Icons -->
  <div id="stop_video" class="options_button">
    <i class="fa fa-video-camera"></i>
  </div>
  <div id="mute_button" class="options_button">
    <i class="fa fa-microphone"></i>
  </div>
  <div id="show_chat" class="options_button">
    <i class="fa fa-comment"></i>
  </div>
  <div id="invite_button" class="options_button">
    <i class="fas fa-user-plus"></i>
  </div>
</div>
```

3. Similarly, in **script.js**, let's create a click event handler for this button -

```
$("#stop_video").click(function () {  
    const enabled = myStream.getVideoTracks()[0].enabled;  
    if (enabled) {  
        myStream.getVideoTracks()[0].enabled = false;  
        html = `<i class="fas fa-video-slash"></i>`;   
        $("#stop_video").toggleClass("background_red");  
        $("#stop_video").html(html)  
    } else {  
        myStream.getVideoTracks()[0].enabled = true;  
        html = `<i class="fas fa-video"></i>`;   
        $("#stop_video").toggleClass("background_red");  
        $("#stop_video").html(html)  
    }  
})  
  
$("#invite_button").click(function() {  
  
})
```

4. Create an object in this click event that contains the URL and To data -

```
$("#invite_button").click(function() {  
    let data = {  
        url: "",  
        to: ""  
    }  
})
```

5. Fetch the URL of the current page into the **url** key of this data object -

```
$("#invite_button").click(function() {  
    let data = {  
        url: window.location.href,  
        to: ""  
    }  
})
```

6. Ask user's email ID with the **prompt()** function, and save it into the data object -

```
$("#invite_button").click(function() {  
    const to = prompt("Enter the email address")  
    let data = {  
        url: window.location.href,  
        to: to  
    }  
})
```

7. Create the AJAX request, making a post request to the server side on **/send-mail** URL

```
$("#invite_button").click(function() {  
    const to = prompt("Enter the email address")  
    let data = {  
        url: window.location.href,  
        to: to  
    }  
    $.ajax({  
        url: "/send-mail",  
        type: "post",  
        data: JSON.stringify(data),  
        dataType: 'json',  
        contentType: 'application/json',  
        success: function (result) {  
            alert("Invite sent!")  
        },  
        error: function (result) {  
            console.log(result.responseJSON)  
        }  
    })  
})
```

8. Deploy the code to Heroku by first pushing it to github and then deploying it from the dashboard

Manual deploy

Deploy the current state of a branch to this app.

Deploy a GitHub branch

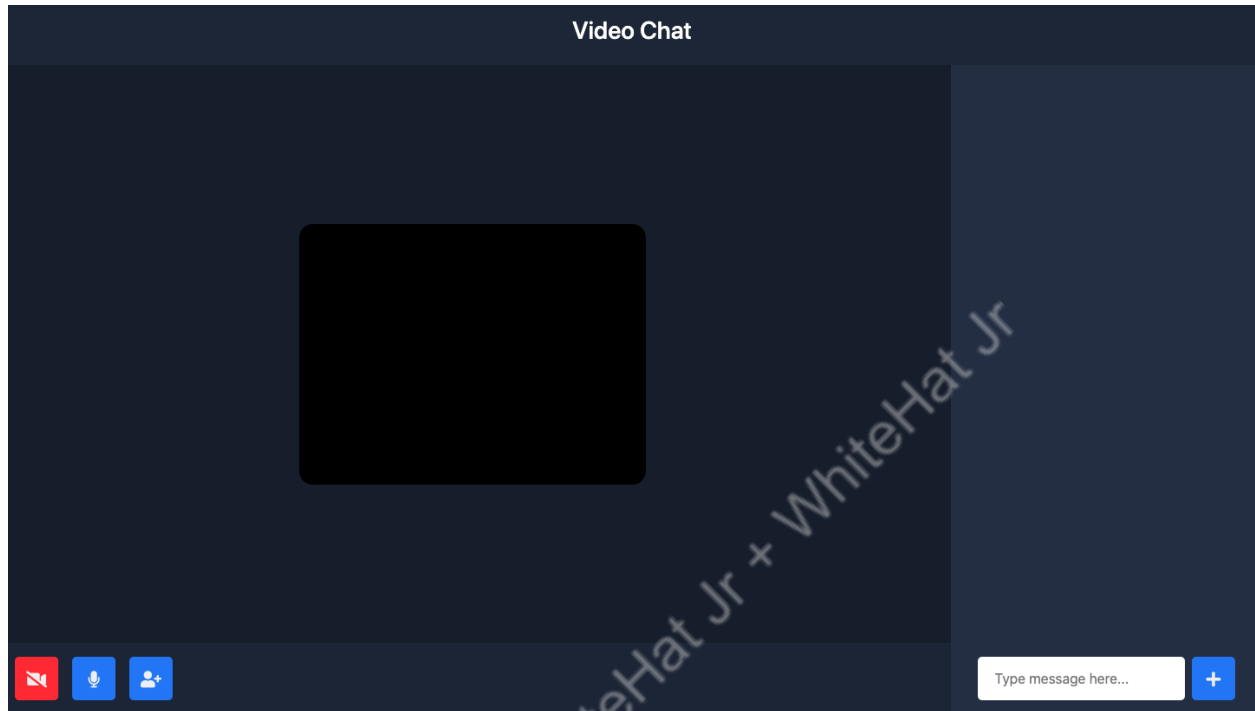
This will deploy the current state of the branch you specify below. [Learn more](#).

Choose a branch to deploy

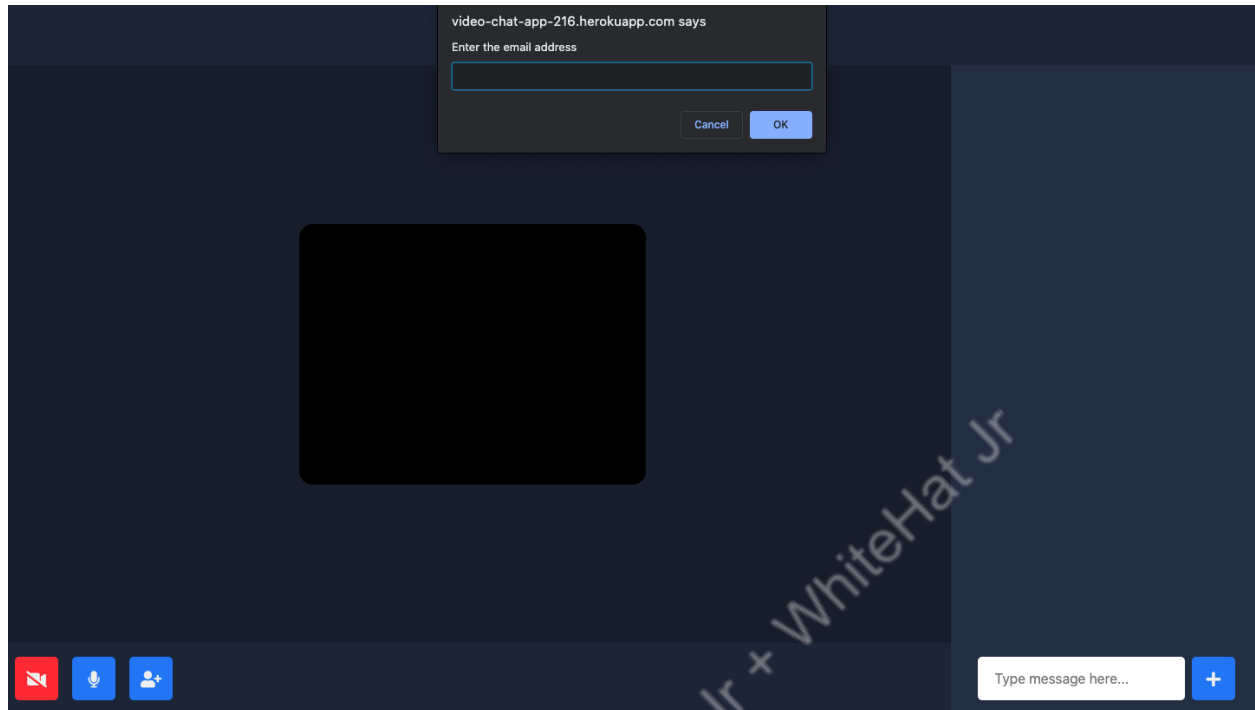
main

Deploy Branch

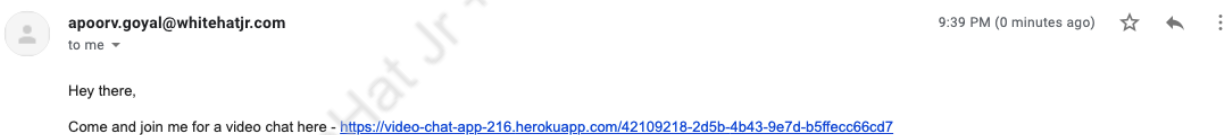
9. Open the app with the heroku's URL on the browser. You will now see the button that was just added to invite users -



10. Click on the invite button, and you will see the prompt



11. Enter the Email ID of the user that you want to invite for the video chat. Make sure you have access to this email ID too, so you can check if the email is sent or not -



12. With this, the video chat application is completed!

What's NEXT?

In the next class, we will learn about cyber security!

Expand Your Knowledge:

Explore more about AJAX requests [here](#).