Livesitter FullStack Assignment

- Documentation for A RTSP Livestream App with Overlay

Setup

- Backend

- Open backend terminal folder in terminal and run

```
npm install
```

- Change the environment variables in `.env` file in backend folder.
- change MongoDB URI in `.env` file in backend folder.
- Change the port in `.env` file in backend folder.
- To start the server run

```
  npm run dev
```

- Frontend

- Open frontend terminal folder in terminal and run

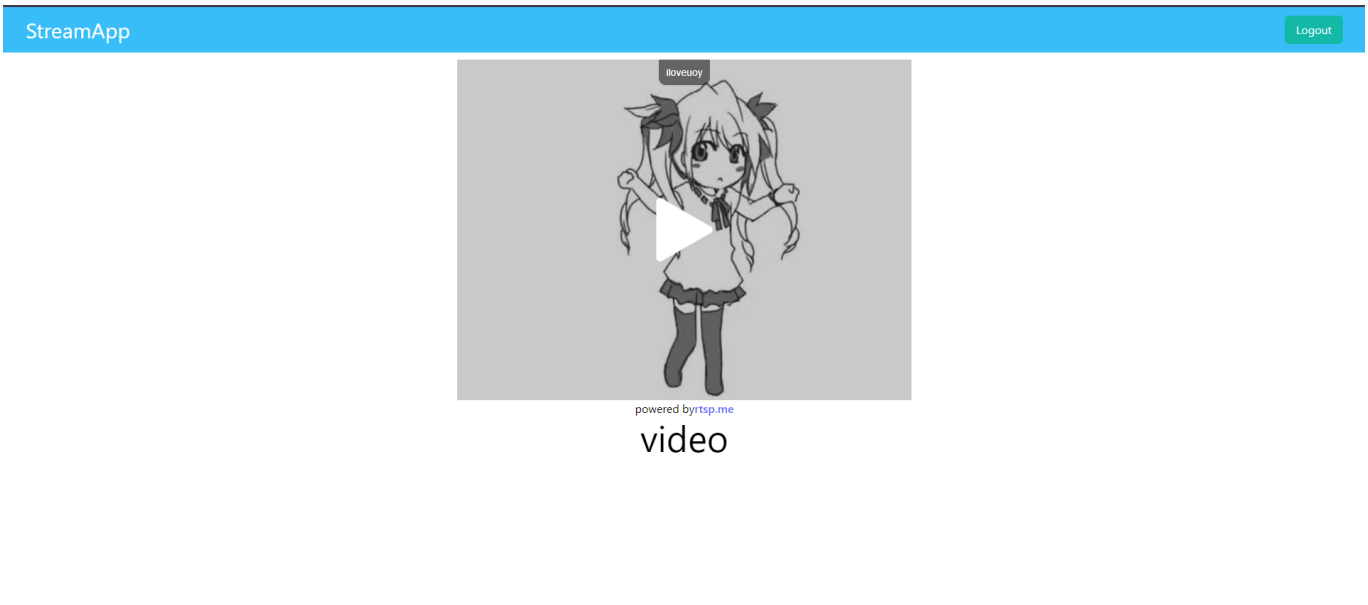
```
npm install
```

- To start react app run

```
  npm run dev
```

Using the App

- To view the app go to `http://localhost:{port assigned by react}` usually `http://localhost:5173`
- Then go to `/signip` route and register.
- You will be directed to the login screen and login with your credentials.
- Now you will be in the app at `/overlay` route.

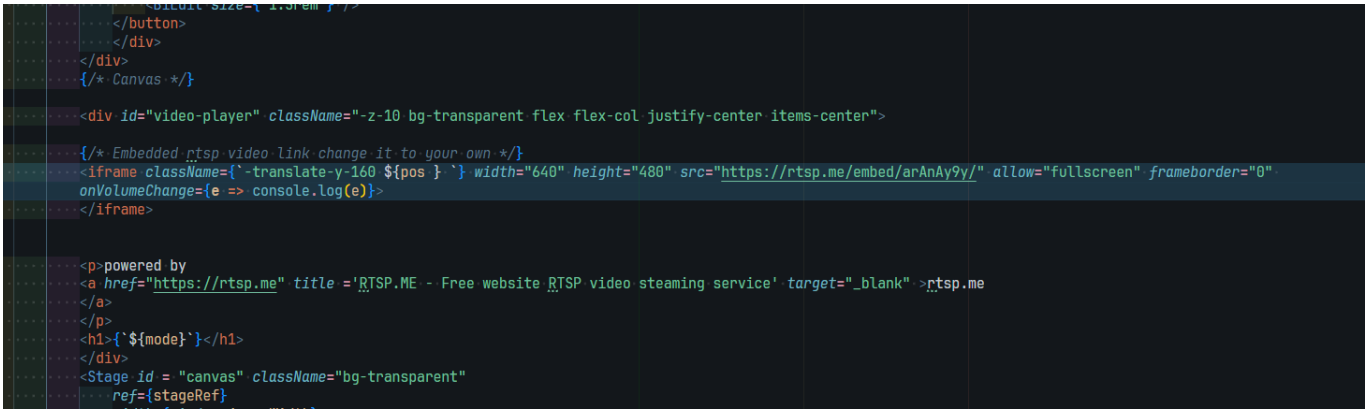


App Main Page

- Now you can start the livestream.
- The PLay, Pause, Volume controls are Fullscreen controls are available.

RTSP URL

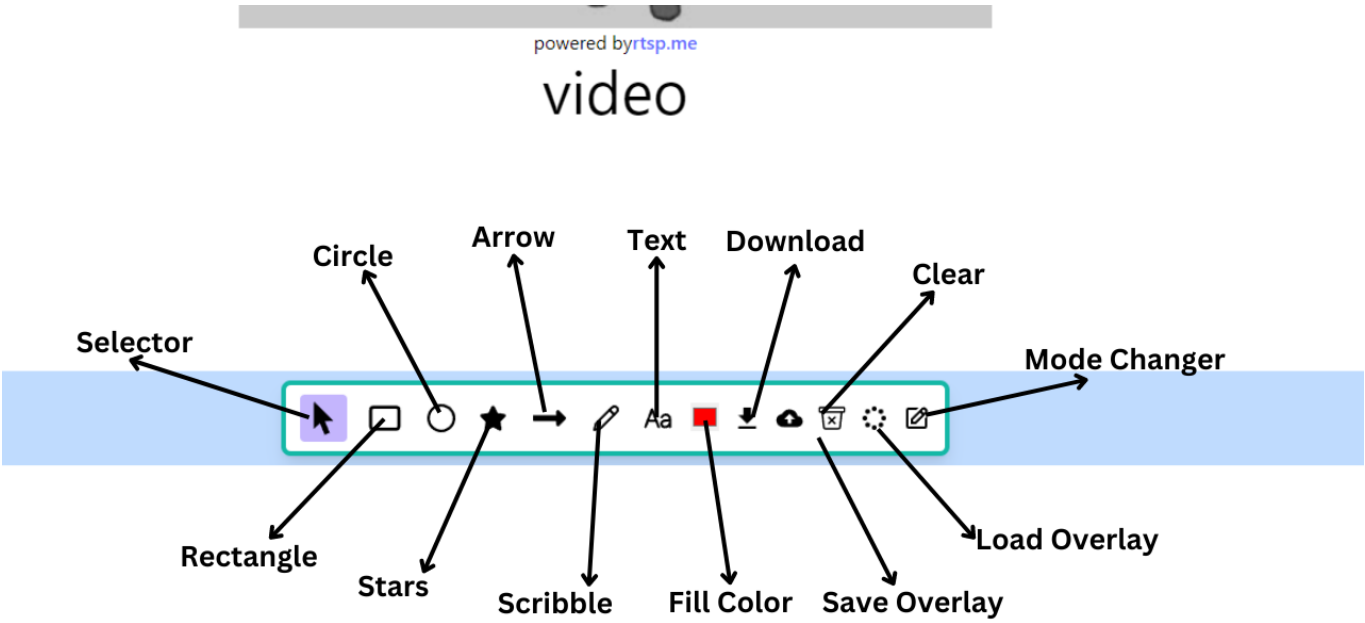
- I am using `RTSP.me` for generating rtsp to http livestream.
- So I have emmbedded the url in the `overlay.jsx` in the app as `iFrame` component.
- To change the url, just change the url in `overlay.jsx` with the new url from `RTSP.me` and it will change the url in the app.
- It is located after overlay button components and before stage component.



```
        </button>
        </div>
    </div>
    {/* Canvas */}

    <div id="video-player" className="-z-10 bg-transparent flex flex-col justify-center items-center">

    {/* Embedded rtsp video link change it to your own */}
    <iframe className={`-translate-y-160 ${pos} `} width="640" height="480" src="https://rtsp.me/embed/arAnAy9y/" allow="fullscreen" frameborder="0"
    onVolumeChange={e => console.log(e)}>
    </iframe>


    <p>powered by
    <a href="https://rtsp.me" title ='RTSP.ME - Free website RTSP video steaming service' target="_blank" >rtsp.me
    </a>
    </p>
    <h1>{`${mode}`}</h1>
    </div>
    <Stage id = "canvas" className="bg-transparent"
        ref={stageRef}
```

Overlay

- Before we start putting overlays lets look at the `Overlay board` below.



overlay board

1. `Selector` - Use selector to activate selector mode, move around and resize the shapes. Make sure you are normally in the selector mode.
2. `Rectangle` - For drawing Rectangle shapes.
3. `Circle` - For drawing Circle shapes.
4. `Stars` - This will spawn draggable stars on the screen.
5. `Arrow` - For drawing Arrows.
6. `Scribble` - For drawing Scribbles.
7. `Text` - For adding text on the screen.
8. `Fill Color` - For choosing fill color of shapes
9. `Download` - To download the overlay as image.
10. `Save` - To save the overlay in database.
11. `Clear` - To clear or delete the overlay from database.
12. `Load` - To load the overlay from database.
13. `Mode Selector` - To switch between overlay and video mode.

- `Important Note`

- So to start putting the overlay on the screen, click on the `Mode Selector` button to activate overlay mode and it will say `"Overlay"` above the video.
- To Access the video controls click on the `Mode Selector` button again to go back to video mode and it will say `"Video"` below the video.

- `Deleting Overlay`

- To delete the overlay from database click on the `Clear` button and then click on the `Save` button.

Some Issues

- Its good to be transparent so i should talk about features that require more work to be done.

1. The `Text` feature is not working as expected.
    - You can only have one proper text element.
    - Adding multiple text does not work properly.
    - As they share some states like the text value, color etc.
    - So multiple unique texts are not possible yet.
2. I have used MERN stack foe the project that means I used NodeJS and ExpressJS as Backend Framework.
    - The recommended tech stack for backend framework was Python Flask.
    - But I was more Comfortable with NodeJS and ExpressJS also we don't have much time for the assignment project and i had to focus on making the project which was a challenging one.
    - So I decided to use MERN stack for the project.