**CAPSTONE PROJECT**



**PROJECT TITLE**

**MOVIE LISTING**

**TEAM - 3:**

* **IMMINNI SARITHA 20P31A1218**
* **SUBHA SARANYA BHIMALA 21P35A1201**
* **NEKKANTI SAI CHAITANYA 20P31A1239**
* **TEJASWINI THAMBABATHULA 20P31A1233**
* **KAKI SHIVA SATYA NARAYANA 21A95A0208**
* **CHIRUKOTI LIKITH 21P35A1202**
* **KOPPULA RAVINDRA REDDY 20MH1A0590**
* **NEETIPALLI SIVA GANESH 20MH1A0541**
* **PROJECT OVERVIEW**

You are provided with a “Movie listing” website which uses ReactJS as frontend, NodeJS as backend and MongoDB as database. Users can upload movie details where it uses the local storage to store the images. You are required to deploy this entire website into the cloud infrastructure (AWS) with proper scaling.

Use AWS S3 for storing images. You can use the multer-s3 library.

Replace local database with Atlas MongoDB cloud infrastructure to take the database into the cloud.

Deploy Backend in EC2 instance and attach Elastic IP to this instance. Deployment should be done using Docker.

Modify the Frontend code to be able to fetch data from Backend. Finally, deploy frontend using docker into EC2 instance.

Create Load balancer and attach load balancer to be able to properly scale the website traffic.

Use DNS to point to this IP.

Create a proper AWS deployment diagram and suggest methods to improve it.

Host docker images into AWS ECR/Docker hub.

* **TECHNOLOGIES USED**

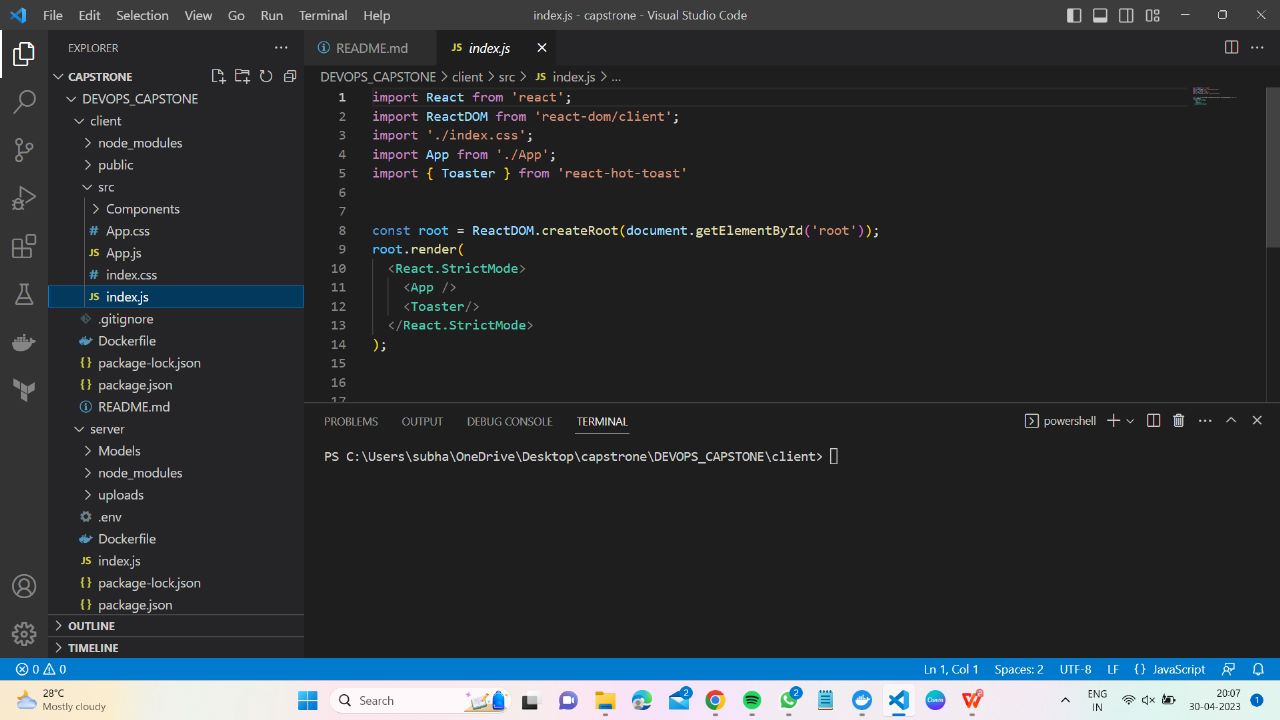
**ReactJS:** The React.js framework is an open-source JavaScript framework and library developed by Facebook. It’s used for building interactive user interfaces and web applications quickly and efficiently with significantly less code than you would with vanilla JavaScript.

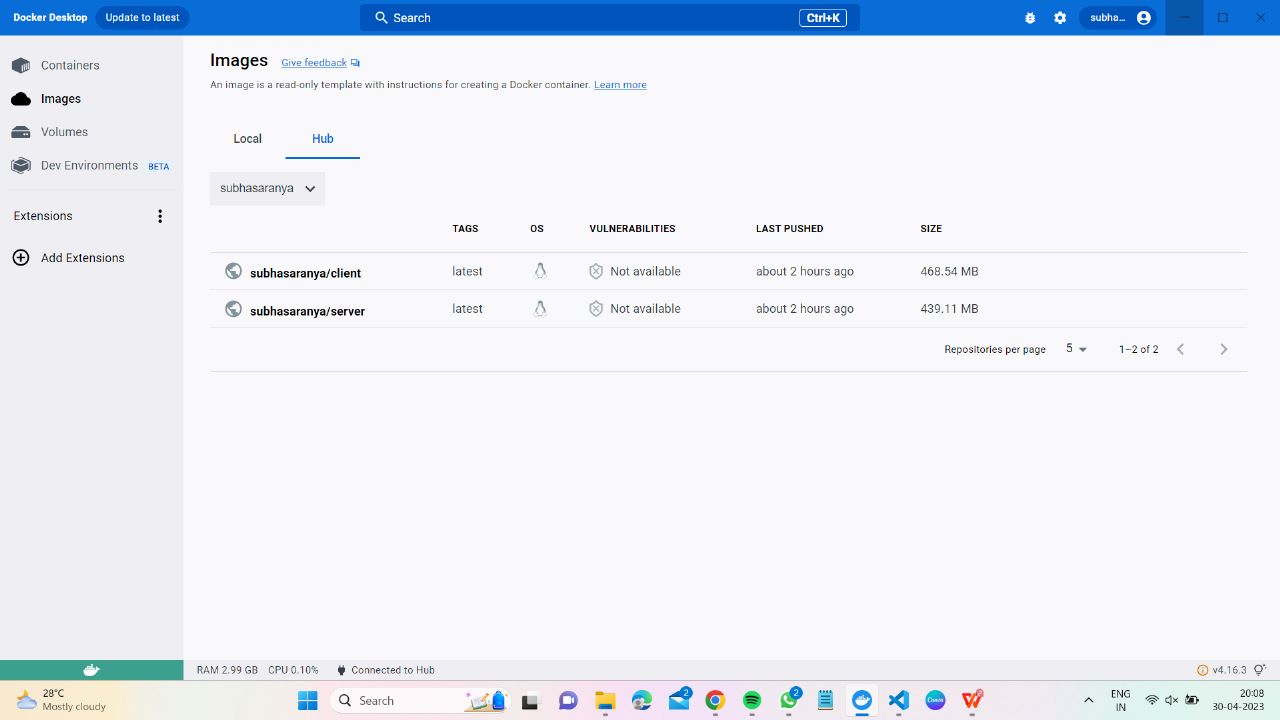
In our project.

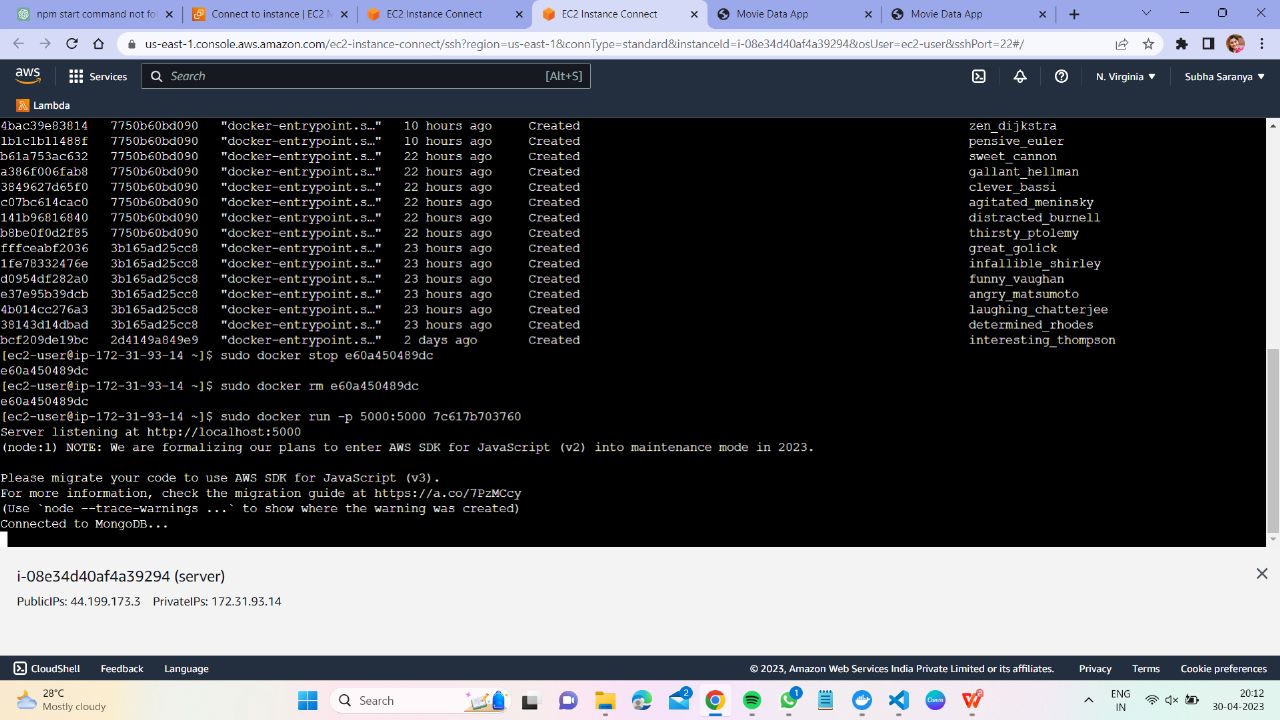
**NodeJS:** If we talk about any application then the part with which the user is interacting is basically the frontend of our website whereas there are many things which happen in the background or in the backend of our website. Basically, there are three parts of any application, one is Frontend with which the users are interacting, then comes to the backend server and backend database. For backend servers we can use NodeJS.

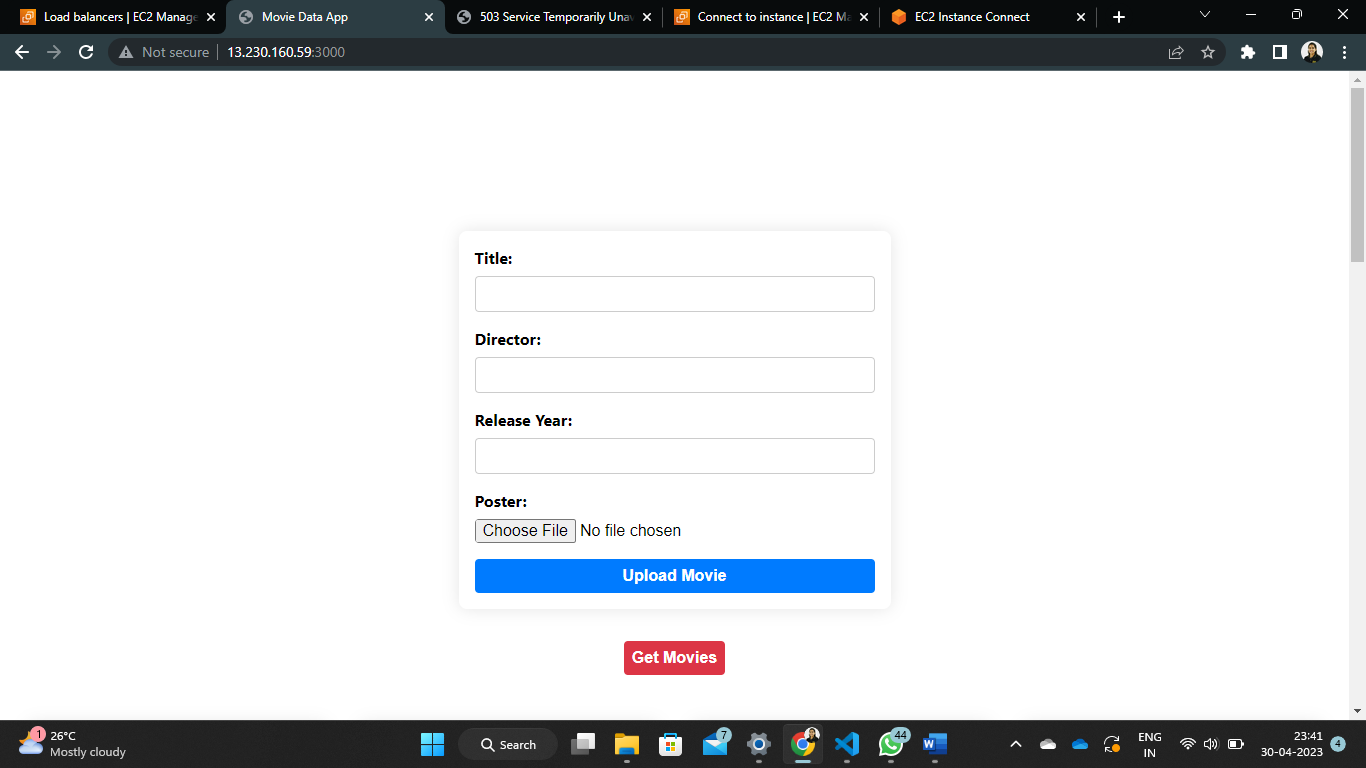
**MongoDB:** MongoDB is a document database used to build highly available and scalable internet applications. With its flexible schema approach, it’s popular with development teams using agile methodologies. Offering drivers for all major programming languages, MongoDB allows you to immediately start building your application without spending time configuring a database.

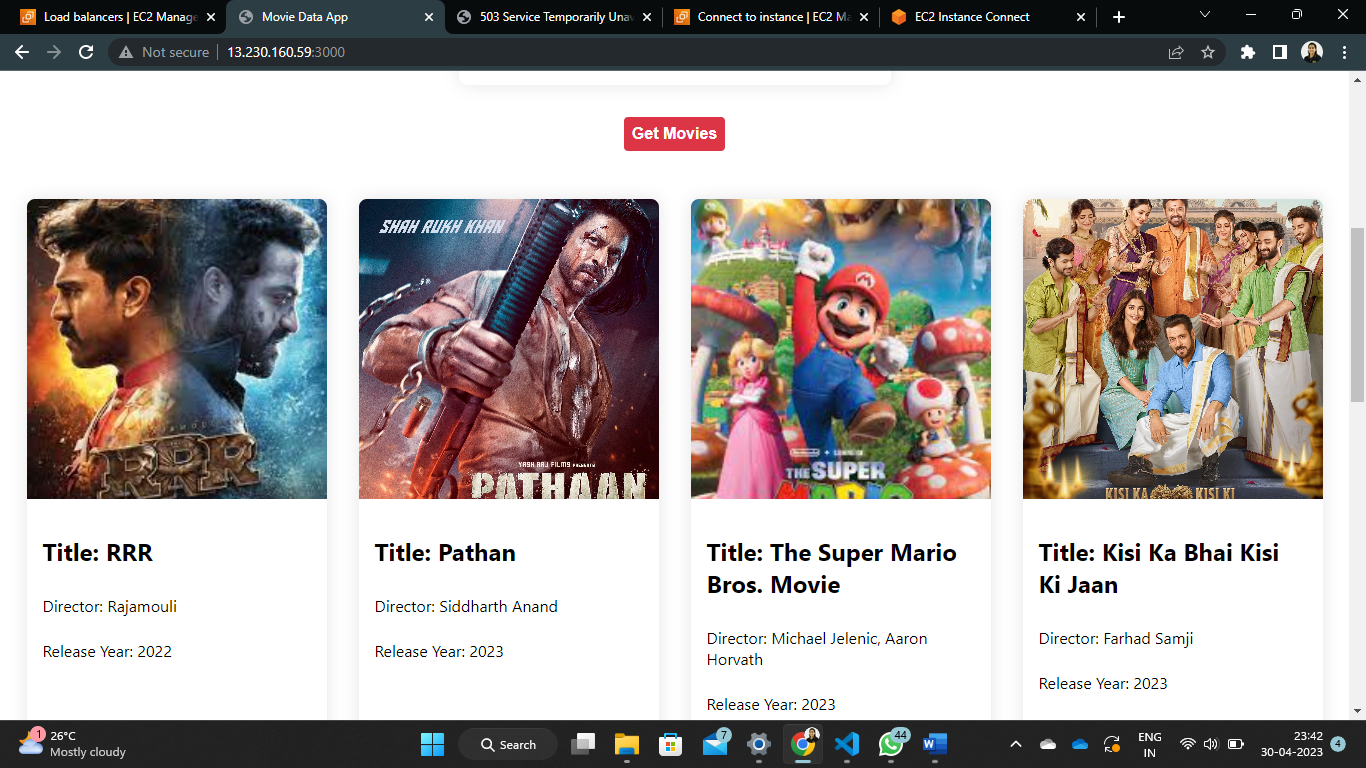
**PROOF OF CONCEPT**











* **LINK OF THE PROJECT**

**http://13.230.160.59:3000**

* **CONCLUSION**

In conclusion, the deployment of the frontend website for movie listing using AWS Cloud, Docker, and MongoDB has been a success. The use of these technologies has made the deployment process much more efficient, streamlined, and scalable.

AWS Cloud provided a flexible and scalable infrastructure that allowed us to easily deploy our website and scale it up or down based on demand. Docker enabled us to create a containerized environment for our application, making it easier to manage dependencies and deploy the application across different environments.

MongoDB provided a reliable and scalable database solution that allowed us to store and retrieve movie information in real-time. It also allowed us to easily scale our database as our application grew.

Overall, the deployment process was smooth, and we were able to launch our website successfully. The use of these technologies has made it possible to create a highly scalable and reliable movie listing website. We have also gained valuable experience in deploying applications on the cloud and containerized environments, which will be useful in future projects.