CLOUD AND SERVERLESS PROJECT DEMONSTRATION REPORT

PROJECT TITLE

BUILDING A VIDEO STREAMING SERVICE

ID:210030048 **NAME: Priyanka Bavineni**

ABSTRACT:

Building a serverless video streaming service using Amazon S3, React, and CloudFront. By ditching traditional server setups, we cut costs and complexity while boosting scalability. We'll walk through the architecture, highlighting user authentication, video transcoding, and CloudFront integration. With a focus on simplicity and efficiency, this solution delivers a seamless video streaming experience, perfect for today's demands.

INTRODUCTION:

Welcome, everyone. In today's digital world, video streaming has become a staple of our online experiences. But creating a video streaming service can be complex and costly, right? Not anymore. With Amazon Web Services (AWS), we have access to simple and affordable solutions for building video streaming platforms.

Today, we'll focus on how AWS enables us to create serverless video streaming services. By using services like Amazon S3 for storing videos, React for building interfaces, and CloudFront for speedy delivery, we can streamline the process and keep costs down. So, let's dive in and explore how AWS can revolutionize the way we deliver video content online.

SERVICES USED:

Amazon s3

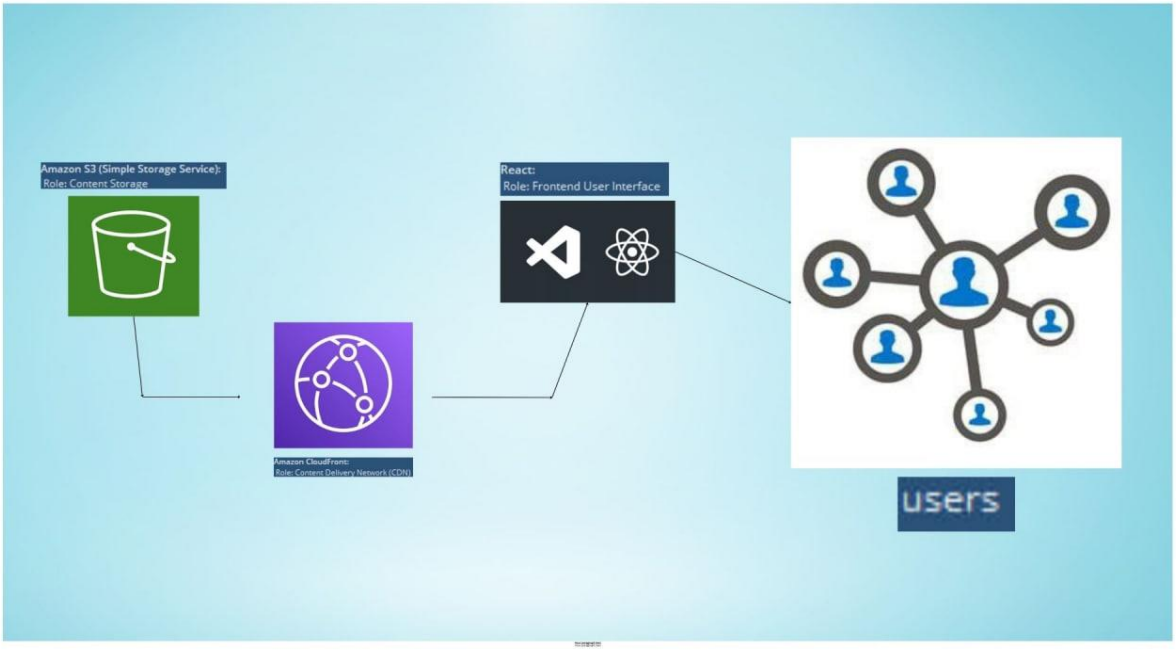
Amazon Cloudfront

OTHER TOOLS:

React js

Vs Code

FLOW DIAGRAM :



S3 Bucket:

Amazon S3 (Simple Storage Service):

- Role: Content Storage

- Amazon S3 will serve as the storage solution for your video files. You'll upload your video files to S3 buckets, which are highly scalable and durable.

- It provides a simple web services interface that allows you to store and retrieve any amount of data from anywhere on the web.

- S3 will handle the storage, reliability, and scalability of your video content.

Cloud Front:

Amazon CloudFront:

- Role: Content Delivery Network (CDN)

- CloudFront will act as the CDN to deliver your video content to viewers worldwide with low latency and high transfer speeds.

- It caches your video files in edge locations around the world, reducing the latency for users accessing your content from different geographical locations.

- CloudFront integrates seamlessly with S3, allowing you to deliver content stored in S3 buckets with improved performance and scalability**.**

React:

- Role: Frontend User Interface

- React will be used to build the user interface for your video streaming service.

- You'll create components and views using React to allow users to browse, search, and play video content.

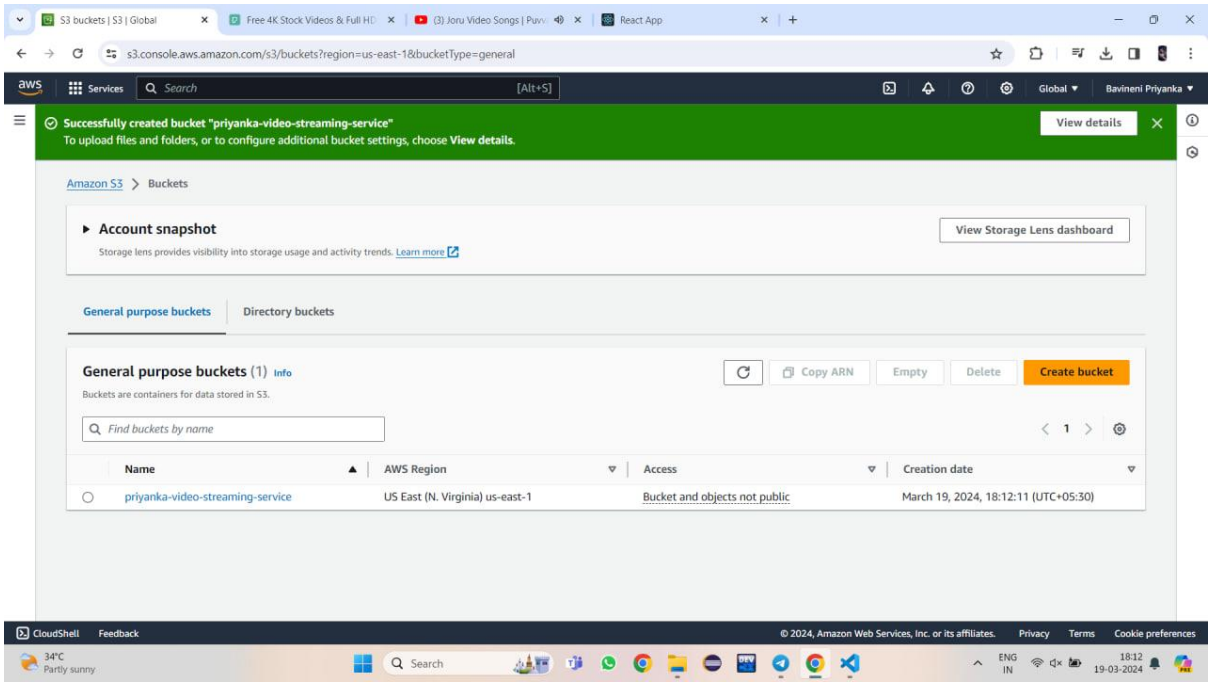
- React will handle the client-side rendering, state management, and user interactions for your application.

- It provides a responsive and interactive user experience, enabling users to seamlessly navigate through your video catalog and watch videos.

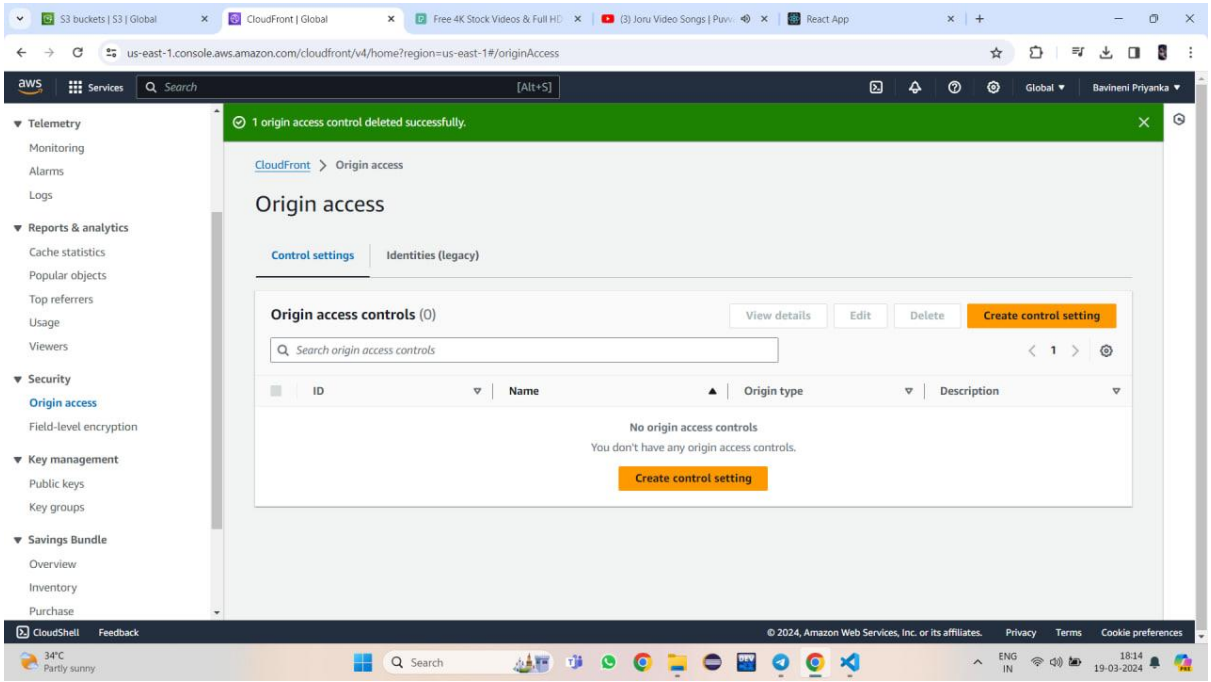
PROCEDURE:

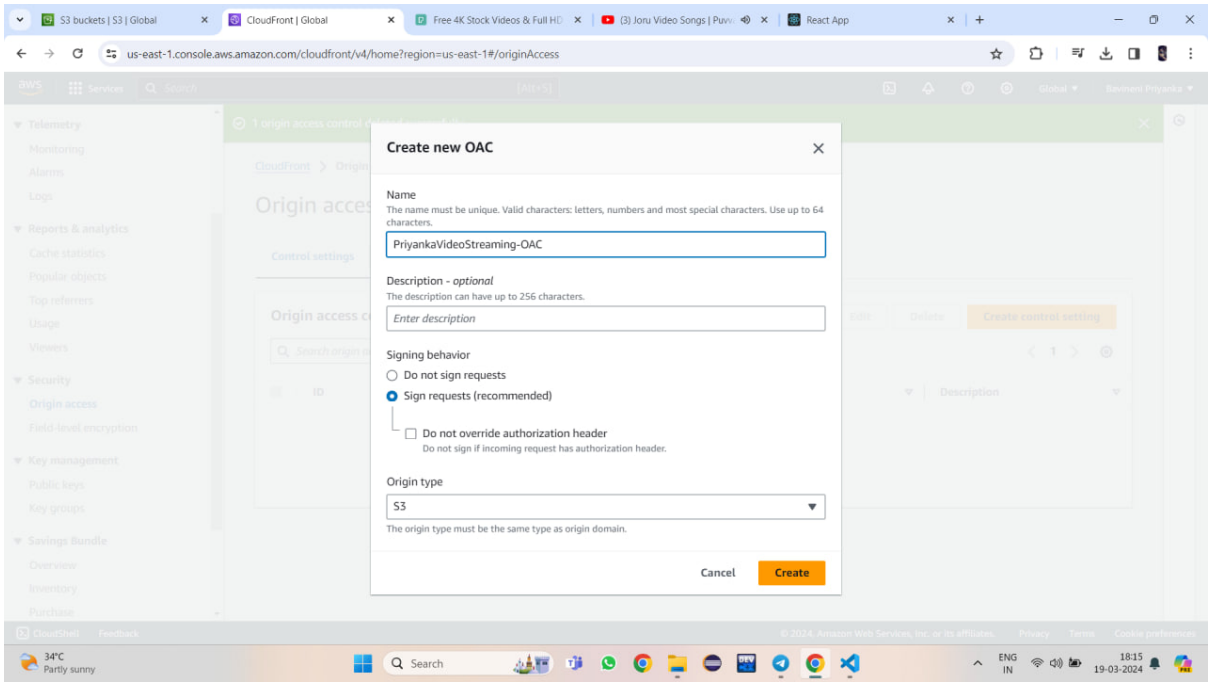
step1: Signin to Amazon console

Step2: Go to services and search for s3 bucket, and create a bucket

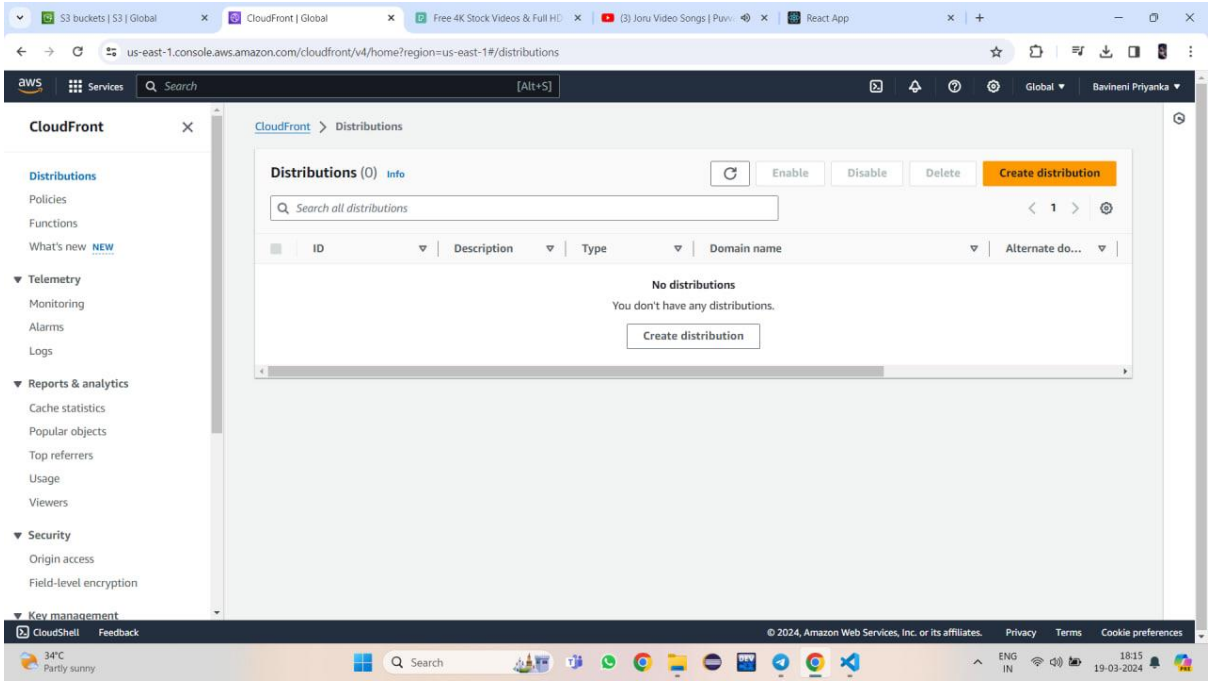


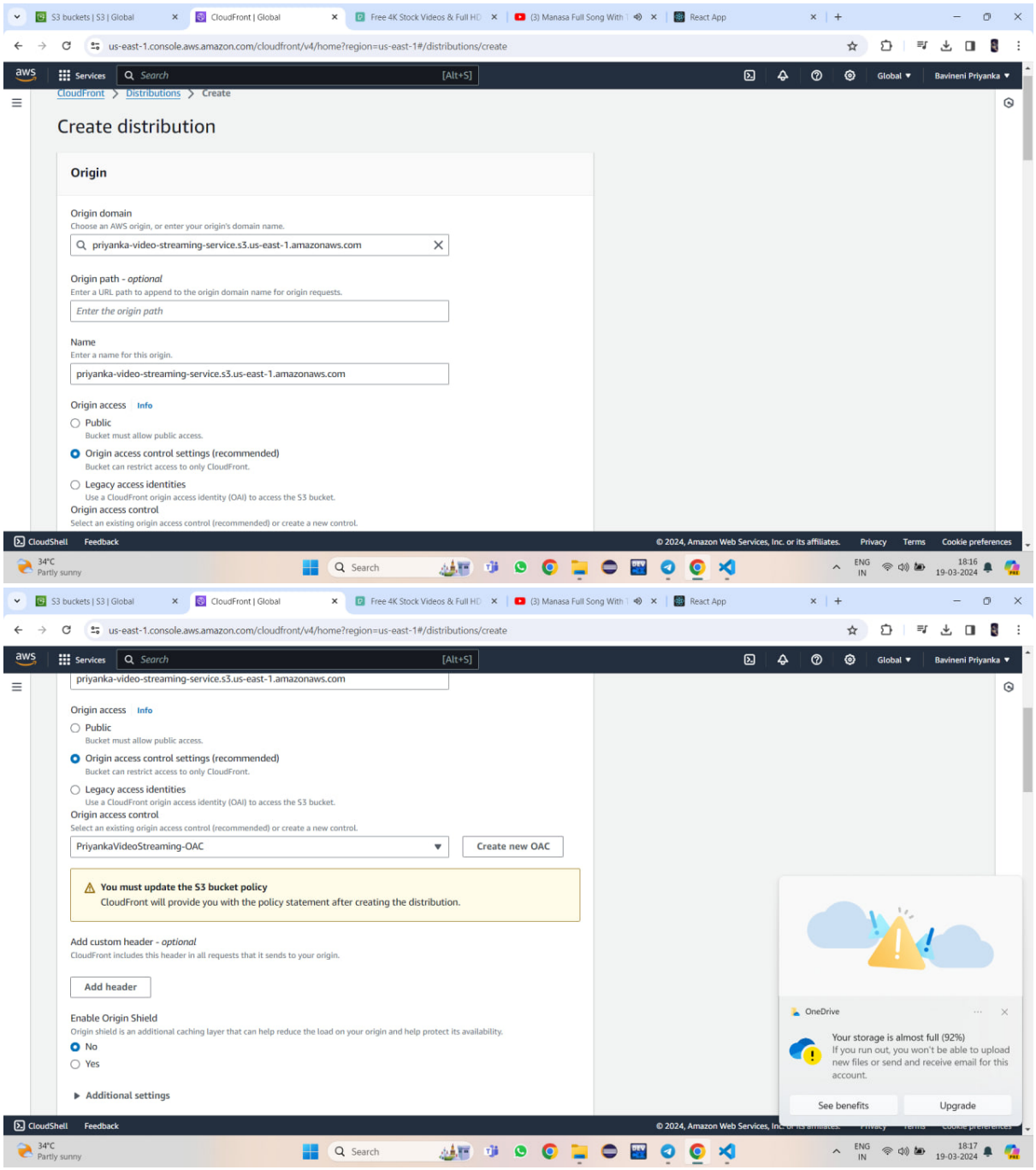
Step3: After that go to cloud front service in that go to origin access and create a control setting



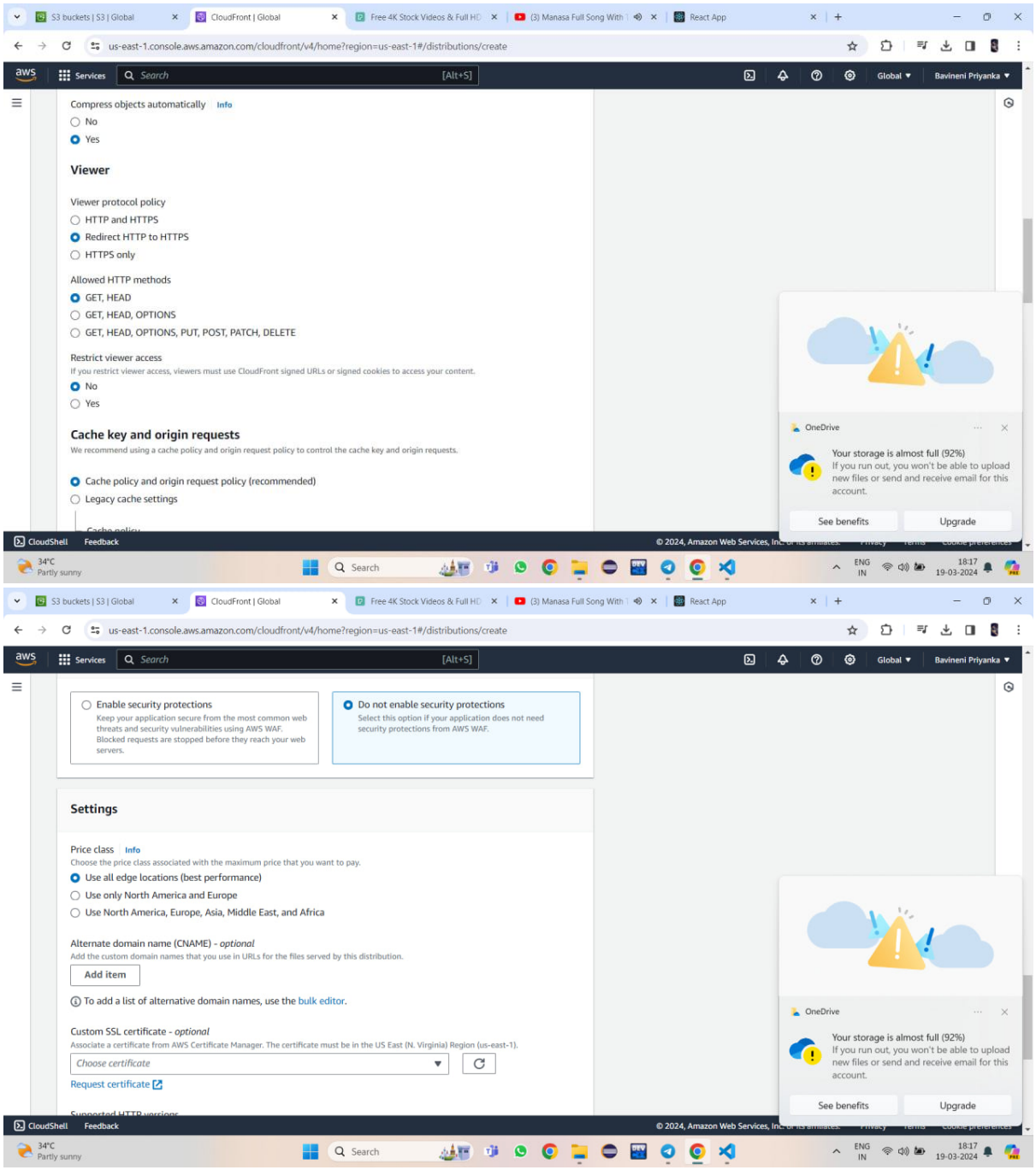


Step4: After creating control setting we have to create a distribution

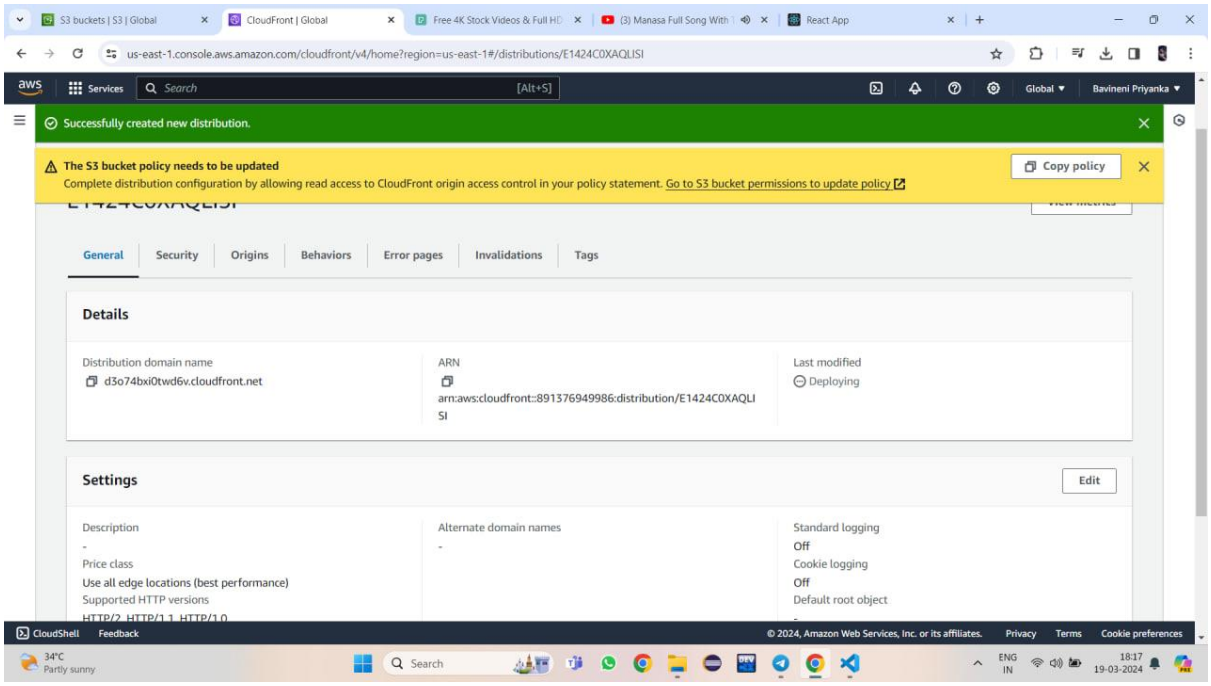



S3 buckets | S3 | Global
CloudFront | Global
Free 4K Stock Videos & Full HD
(3) Manasa Full Song With
React App
us-east-1.console.aws.amazon.com/cloudfront/v4/home?region=us-east-1#/distributions/create
aws
Services
Search
[Alt+S]
Global
Bavineni Priyanka
CloudFront > Distributions > Create
Create distribution
Origin
Origin domain
Choose an AWS origin, or enter your origin's domain name.
priyanka-video-streaming-service.s3.us-east-1.amazonaws.com
Origin path - optional
Enter a URL path to append to the origin domain name for origin requests.
Enter the origin path
Name
Enter a name for this origin.
priyanka-video-streaming-service.s3.us-east-1.amazonaws.com
Origin access Info
Public
Bucket must allow public access.
Origin access control settings (recommended)
Bucket can restrict access to only CloudFront.
Legacy access identities
Use a CloudFront origin access identity (OAI) to access the S3 bucket.
Origin access control
Select an existing origin access control (recommended) or create a new control.
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
34°C
Partly sunny
Search
ENG
IN
18:16
19-03-2024

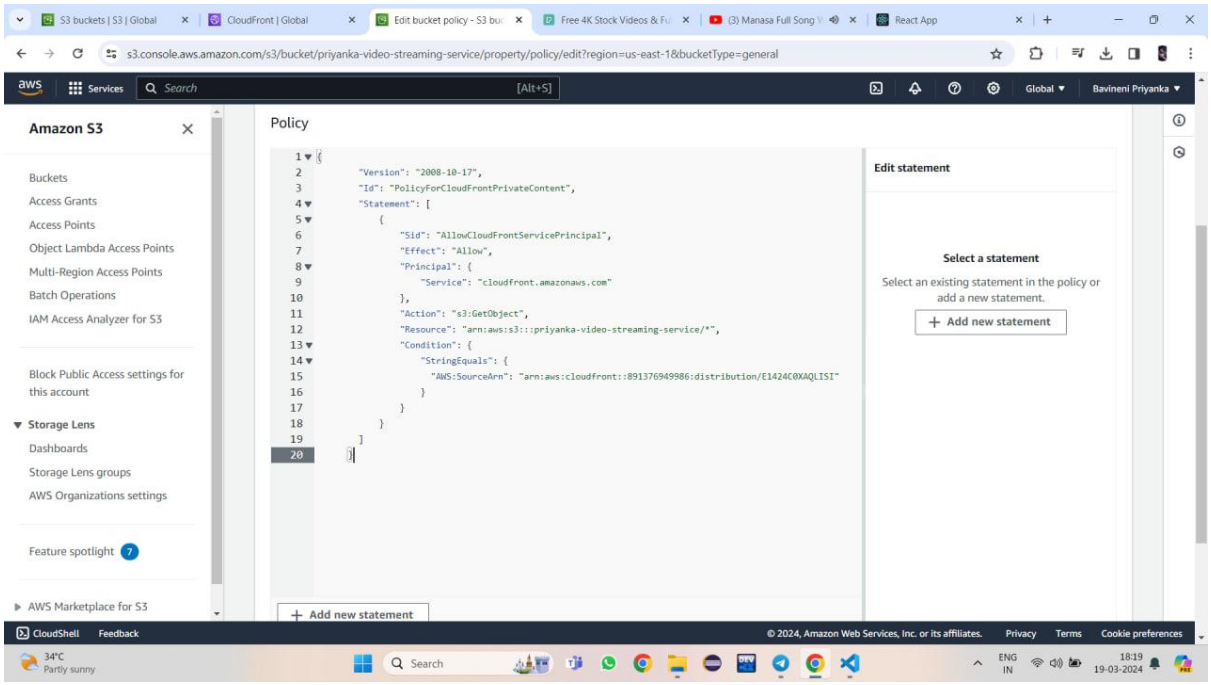
S3 buckets | S3 | Global
CloudFront | Global
Free 4K Stock Videos & Full HD
(3) Manasa Full Song With
React App
us-east-1.console.aws.amazon.com/cloudfront/v4/home?region=us-east-1#/distributions/create
aws
Services
Search
[Alt+S]
Global
Bavineni Priyanka
priyanka-video-streaming-service.s3.us-east-1.amazonaws.com
Origin access Info
Public
Bucket must allow public access.
Origin access control settings (recommended)
Bucket can restrict access to only CloudFront.
Legacy access identities
Use a CloudFront origin access identity (OAI) to access the S3 bucket.
Origin access control
Select an existing origin access control (recommended) or create a new control.
PriyankaVideoStreaming-OAC
Create new OAC
You must update the S3 bucket policy
CloudFront will provide you with the policy statement after creating the distribution.
Add custom header - optional
CloudFront includes this header in all requests that it sends to your origin.
Add header
Enable Origin Shield
Origin shield is an additional caching layer that can help reduce the load on your origin and help protect its availability.
No
Yes
Additional settings
OneDrive
Your storage is almost full (92%)
If you run out, you won't be able to upload new files or send and receive email for this account.
See benefits
Upgrade
CloudShell
Feedback
34°C
Partly sunny
Search
ENG
IN
18:17
19-03-2024


S3 buckets | S3 | Global
CloudFront | Global
Free 4K Stock Videos & Full HD
(3) Manasa Full Song With
React App
us-east-1.console.aws.amazon.com/cloudfront/v4/home?region=us-east-1#/distributions/create
aws
Services
Search
[Alt+S]
Global
Bavineni Priyanka
Compress objects automatically Info
No
Yes
Viewer
Viewer protocol policy
HTTP and HTTPS
Redirect HTTP to HTTPS
HTTPS only
Allowed HTTP methods
GET, HEAD
GET, HEAD, OPTIONS
GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE
Restrict viewer access
If you restrict viewer access, viewers must use CloudFront signed URLs or signed cookies to access your content.
No
Yes
Cache key and origin requests
We recommend using a cache policy and origin request policy to control the cache key and origin requests.
Cache policy and origin request policy (recommended)
Legacy cache settings
OneDrive
Your storage is almost full (92%)
If you run out, you won't be able to upload new files or send and receive email for this account.
See benefits
Upgrade
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
34°C
Partly sunny
Search
ENG
IN
18:17
19-03-2024

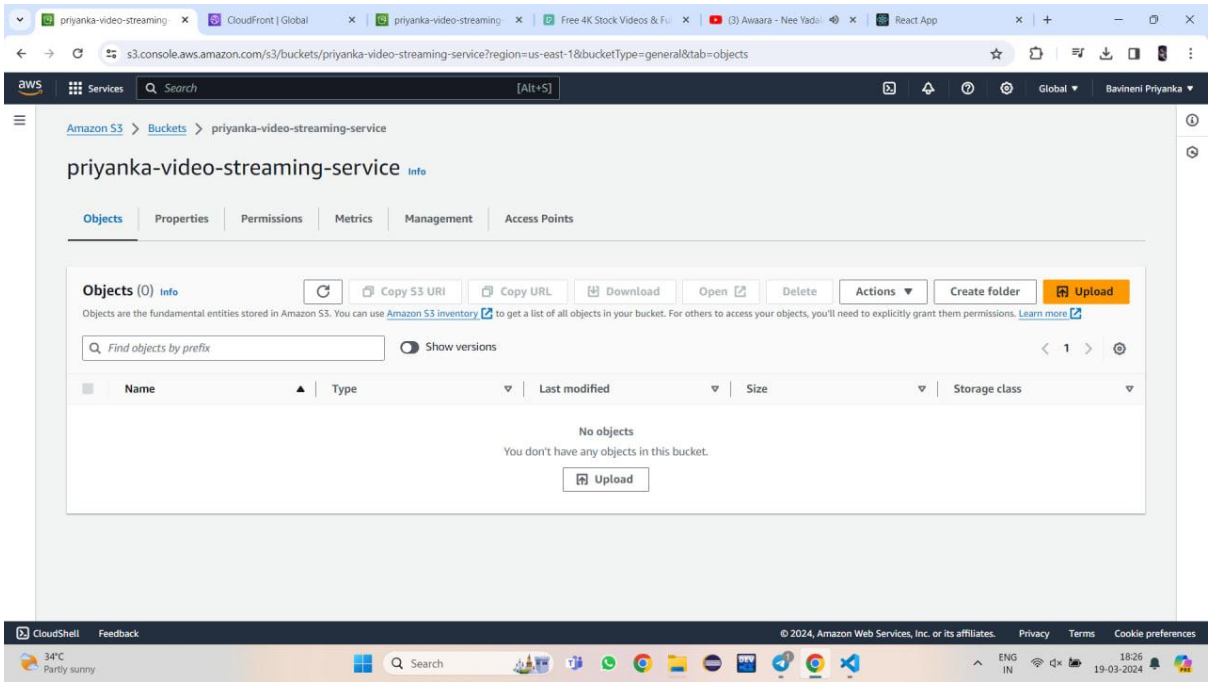
S3 buckets | S3 | Global
CloudFront | Global
Free 4K Stock Videos & Full HD
(3) Manasa Full Song With
React App
us-east-1.console.aws.amazon.com/cloudfront/v4/home?region=us-east-1#/distributions/create
aws
Services
Search
[Alt+S]
Global
Bavineni Priyanka
Enable security protections
Keep your application secure from the most common web threats and security vulnerabilities using AWS WAF. Blocked requests are stopped before they reach your web servers.
Do not enable security protections
Select this option if your application does not need security protections from AWS WAF.
Settings
Price class Info
Choose the price class associated with the maximum price that you want to pay.
Use all edge locations (best performance)
Use only North America and Europe
Use North America, Europe, Asia, Middle East, and Africa
Alternate domain name (CNAME) - optional
Add the custom domain names that you use in URLs for the files served by this distribution.
Add item
To add a list of alternative domain names, use the bulk editor.
Custom SSL certificate - optional
Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1).
Choose certificate
Request certificate
OneDrive
Your storage is almost full (92%)
If you run out, you won't be able to upload new files or send and receive email for this account.
See benefits
Upgrade
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
34°C
Partly sunny
Search
ENG
IN
18:17
19-03-2024

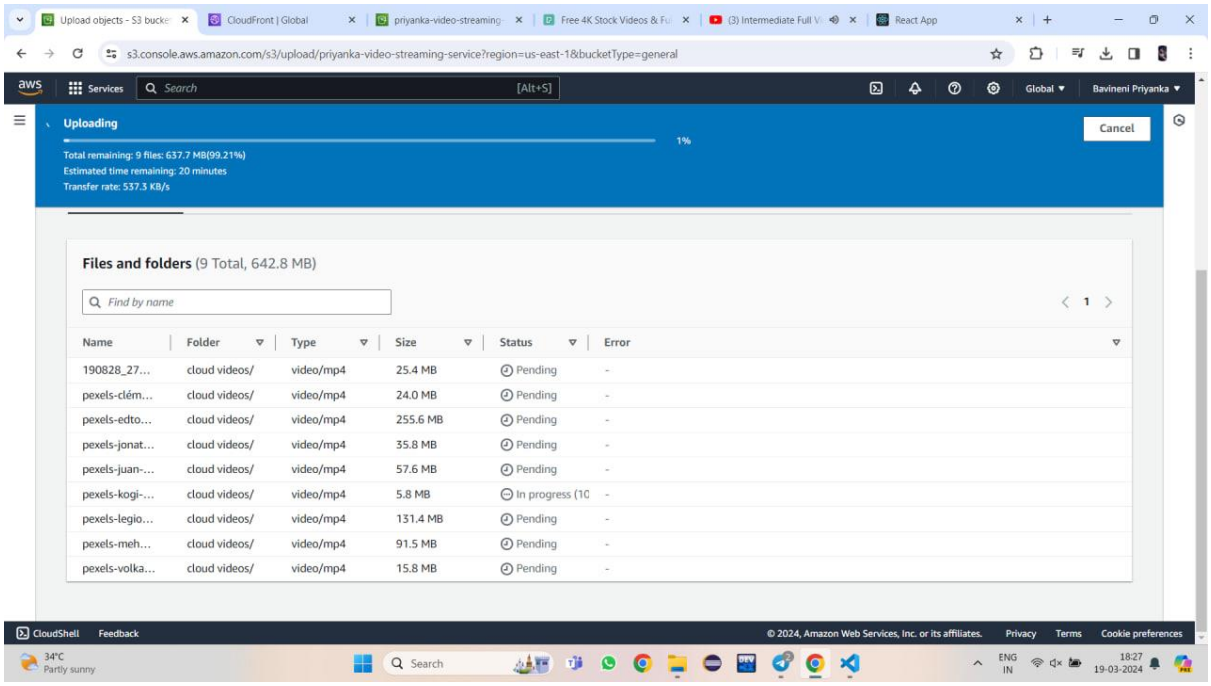


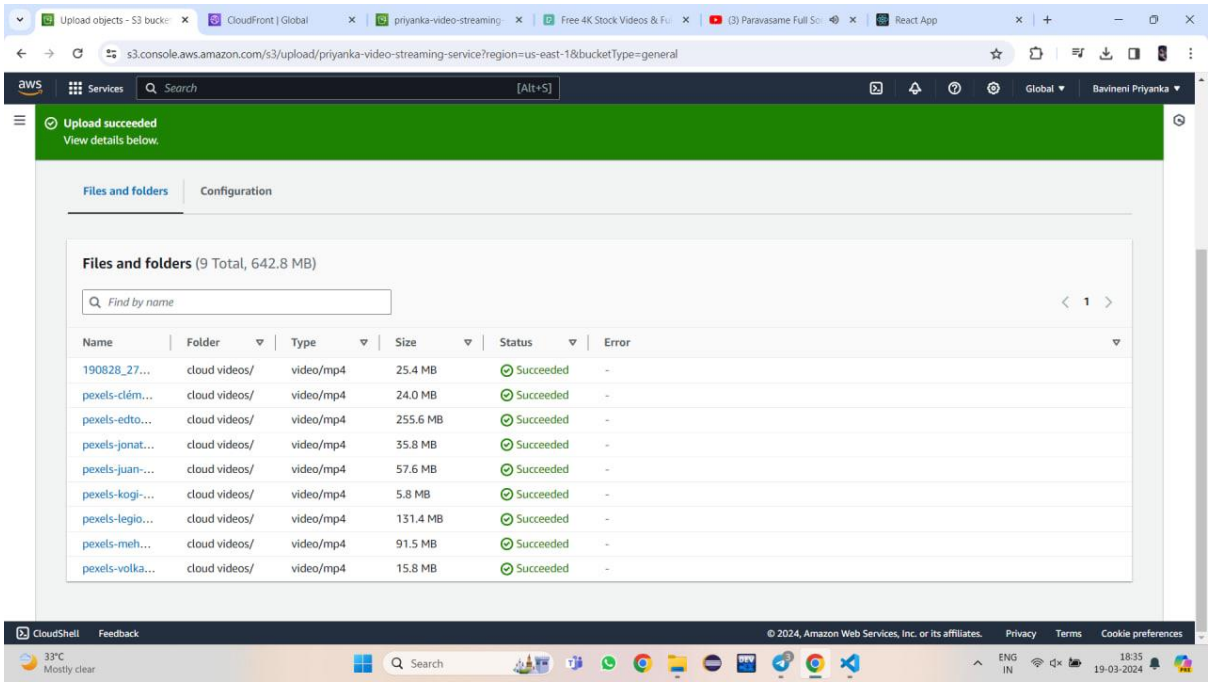
Step-5: After creating distribution we have to modify the policy of s3, for that copy the policy and paste it in s3 policy



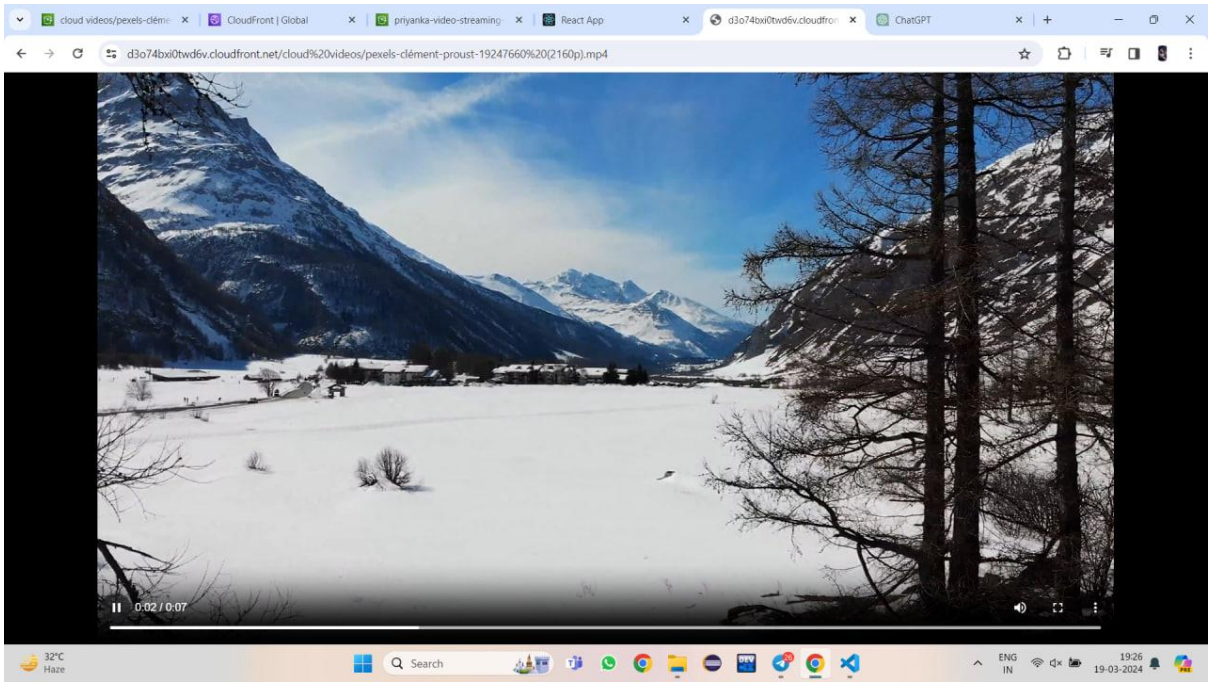
Step6: Go to s3 bucket and click on it and select upload option to select the video file







Step7: Go to Distribution and copy the domain name and paste it in a web browser, and copy the key name and paste it in the web browser then we get like this.



Step8: After getting this we have to go to VS code and create a react app for that we use these commands:

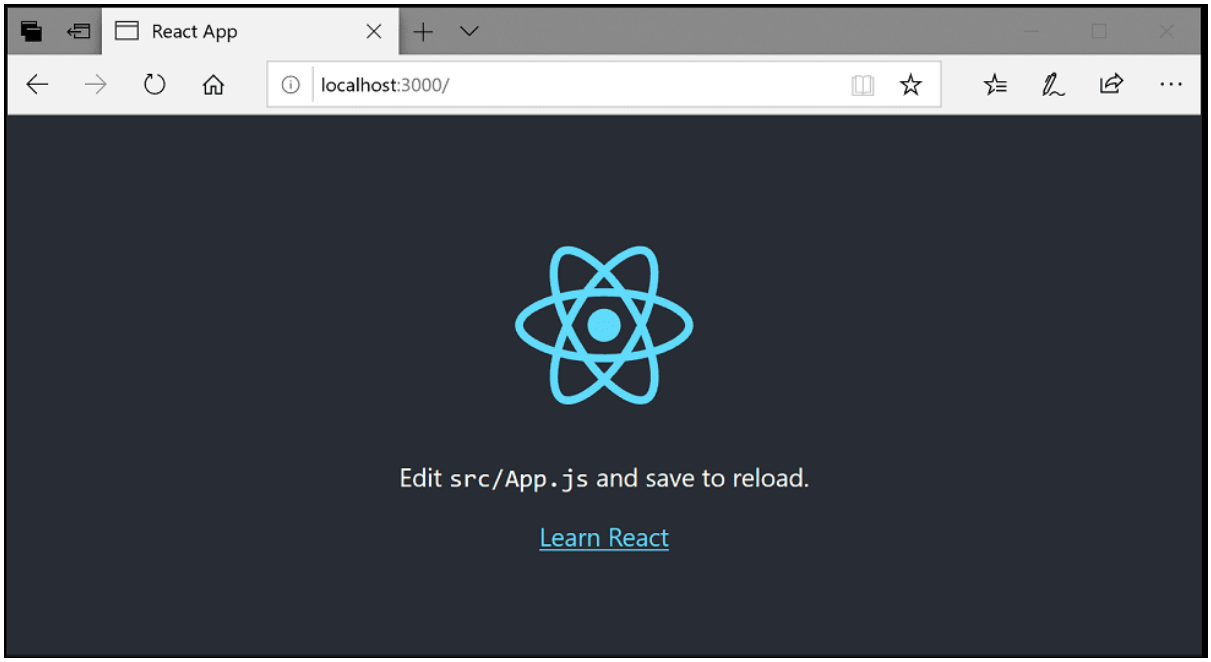
for creating

npx create-react-app my-video-app

for start the app we have to use these commands:

cd my-video-app
npm start

then we get a page like this



Delete the unnecessary files and start to modify the app.js and app.css

code for App.js:

```
import './App.css';
function App() {
 return (
  <div className="App">
   <h2>PRIYANKA NATURE BLOG!!!</h2>
   <video width="700px" height="400px" controls>
```

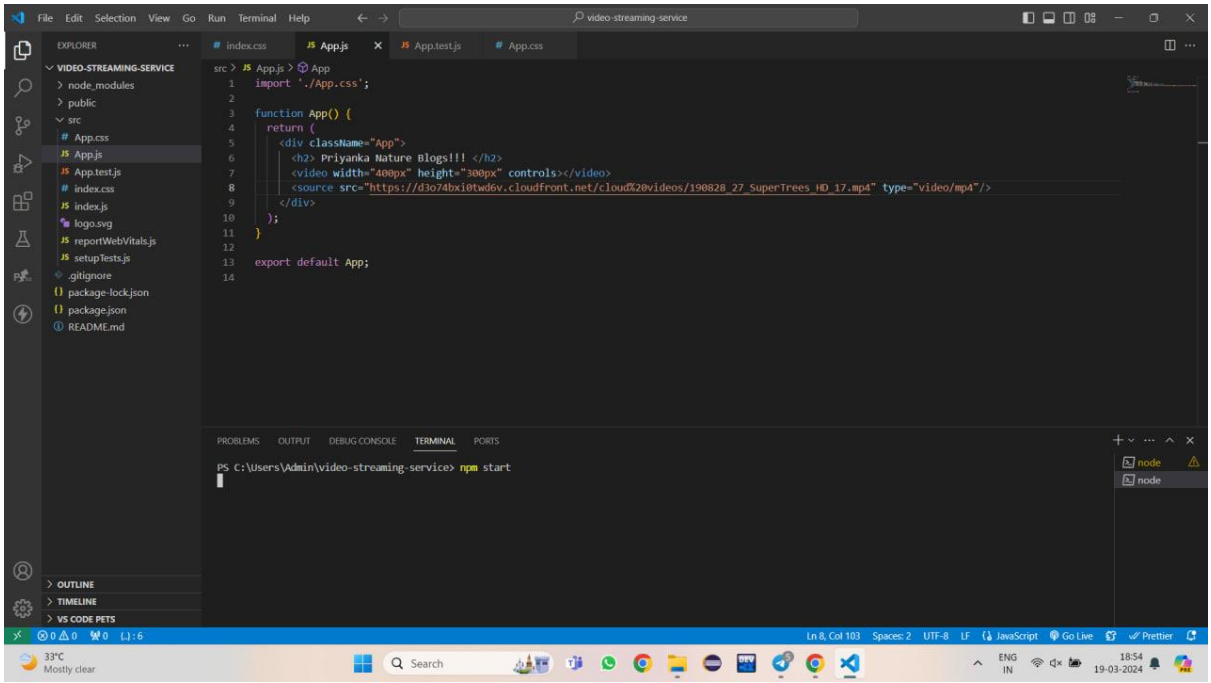
```
      <source src="https://d3o74bxi0twd6v.cloudfront.net/cloud%20videos/pexels-cl%C3%A9ment-proust-19247660%20(2160p).mp4" type="video/mp4"/>
    </video>
   </div>
  );
}

export default App;
```

code for App.css:

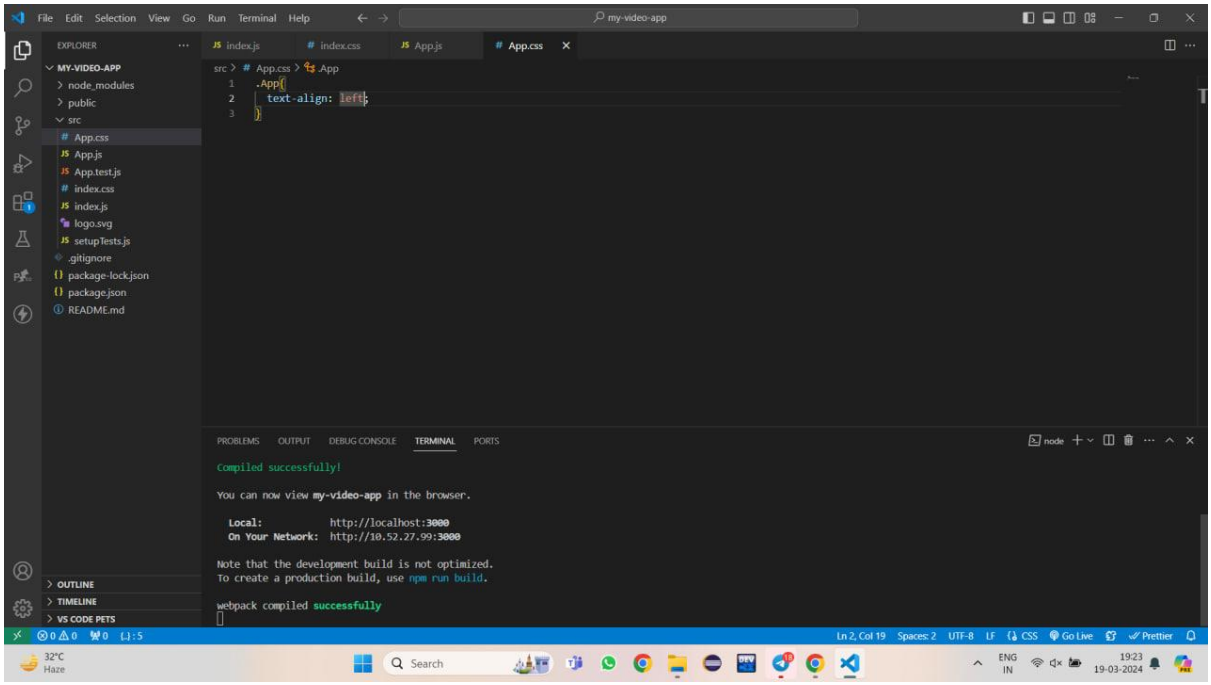
```
.App{
  text-align: left;
}
```

Step9:After modifying the code we have to save and refresh the web page then we get output

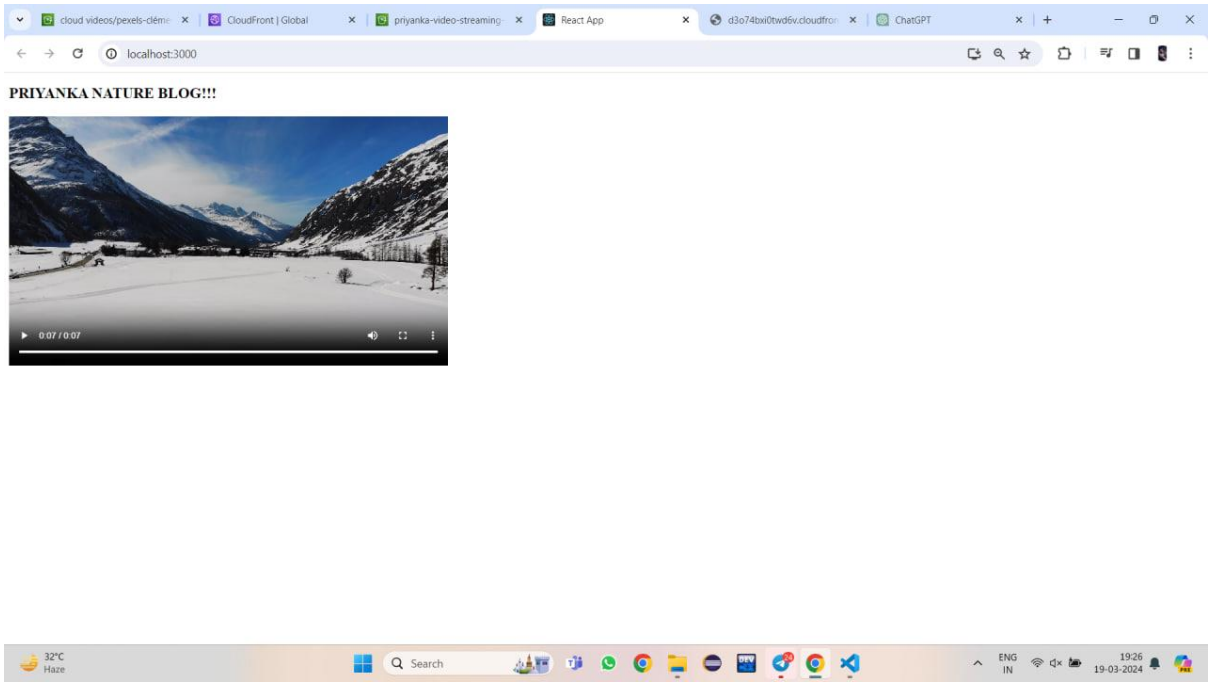

File Edit Selection View Go Run Terminal Help
video-streaming-service
EXPLORER
VIDEO-STREAMING-SERVICE
node_modules
public
src
App.css
App.js
App.test.js
index.css
index.js
logo.svg
reportWebVitals.js
setupTests.js
.gitignore
package-lock.json
package.json
README.md
index.css
App.js
App.test.js
App.css
src > App.js > App
import './App.css';
function App() {
return (

<h2> Priyanka Nature Blogs!!! </h2>
<video width="400px" height="300px" controls></video>
<source src="https://d3o74bxi0twd6v.cloudfront.net/cloud%20videos/190828_27_SuperTrees_HD_17.mp4" type="video/mp4"/>

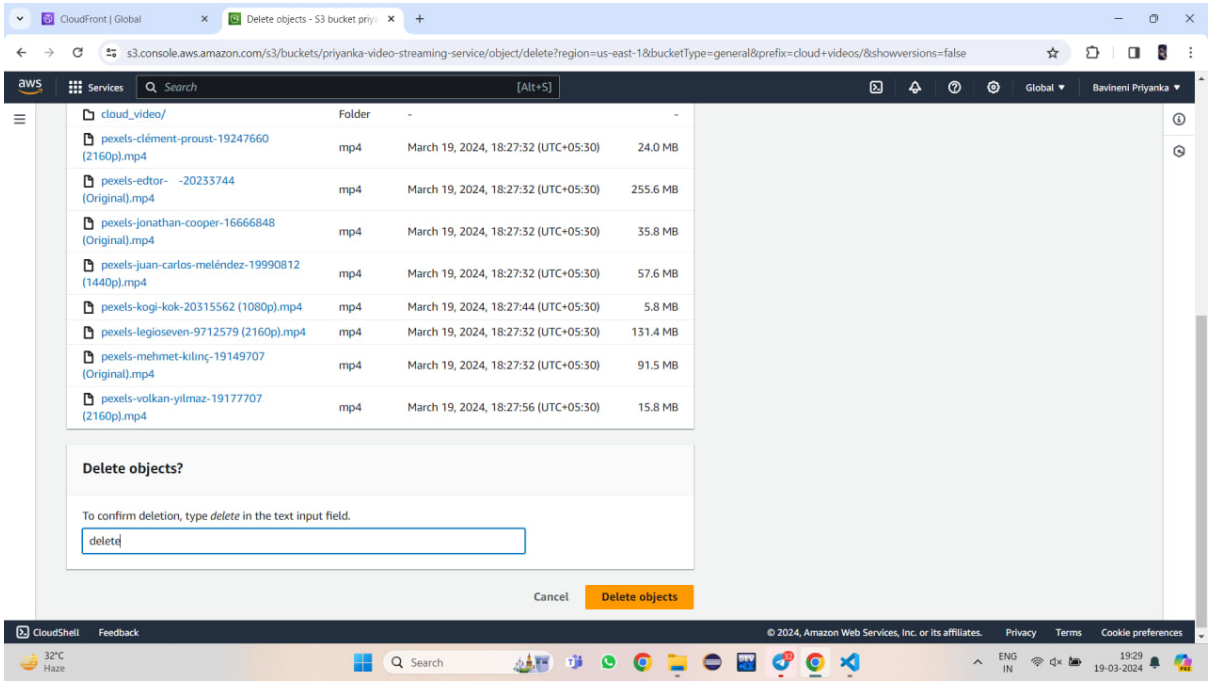
);
}
export default App;
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\Admin\video-streaming-service> npm start
node
node
OUTLINE
TIMELINE
VS CODE PETS
Ln 8, Col 103 Spaces: 2 UTF-8 LF JavaScript Go Live Prettier
33°C
Mostly clear
Search
ENG
IN
18:54
19-03-2024


File Edit Selection View Go Run Terminal Help
my-video-app
EXPLORER
MY-VIDEO-APP
node_modules
public
src
App.css
App.js
App.test.js
index.css
index.js
logo.svg
setupTests.js
.gitignore
package-lock.json
package.json
README.md
index.js
index.css
App.js
App.css
src > App.css > .App
.App{
text-align: left;
}
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
node
Compiled successfully!
You can now view my-video-app in the browser.
Local: http://localhost:3000
On Your Network: http://10.52.27.99:3000
Note that the development build is not optimized.
To create a production build, use npm run build.
webpack compiled successfully
OUTLINE
TIMELINE
VS CODE PETS
Ln 2, Col 19 Spaces: 2 UTF-8 LF CSS Go Live Prettier
32°C
Haze
Search
ENG
IN
19:23
19-03-2024

OUTPUT:



DELETING THE SERVICES:



LINKDELN ARTICLE LINK: https://www.linkedin.com/pulse/video-streaming-service-priyanka-bavineni-6tkbc

YOUTUBE VIDEO LINK: https://youtu.be/bNO1TD7v4xE

Conclusion: In conclusion, leveraging Amazon Web Services (AWS) to create a serverless video streaming service offers a game-changing solution for businesses and developers alike. By harnessing the power of AWS services such as Amazon S3, React, and CloudFront, we can significantly reduce complexity and costs while enhancing scalability and efficiency. This architecture empowers us to deliver seamless video streaming experiences that meet the demands of today's digital landscape. With AWS, building and managing a robust video streaming platform has never been more accessible, enabling us to focus on delivering high-quality content to our users without the burden of traditional server setups.