**REST API :**

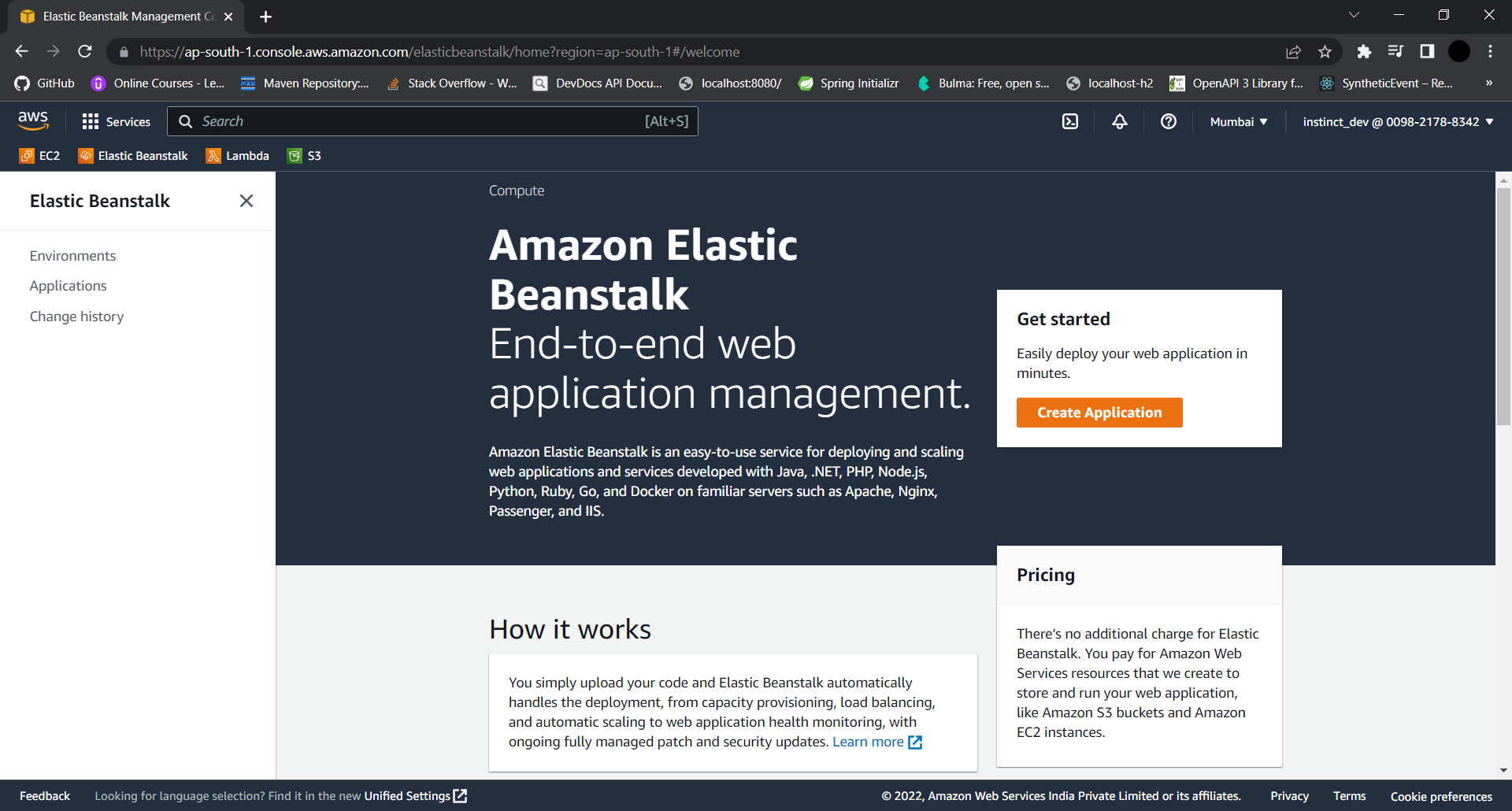
- execute maven goal: clean install // jar file is generated.

**FRONT END - REACT:**

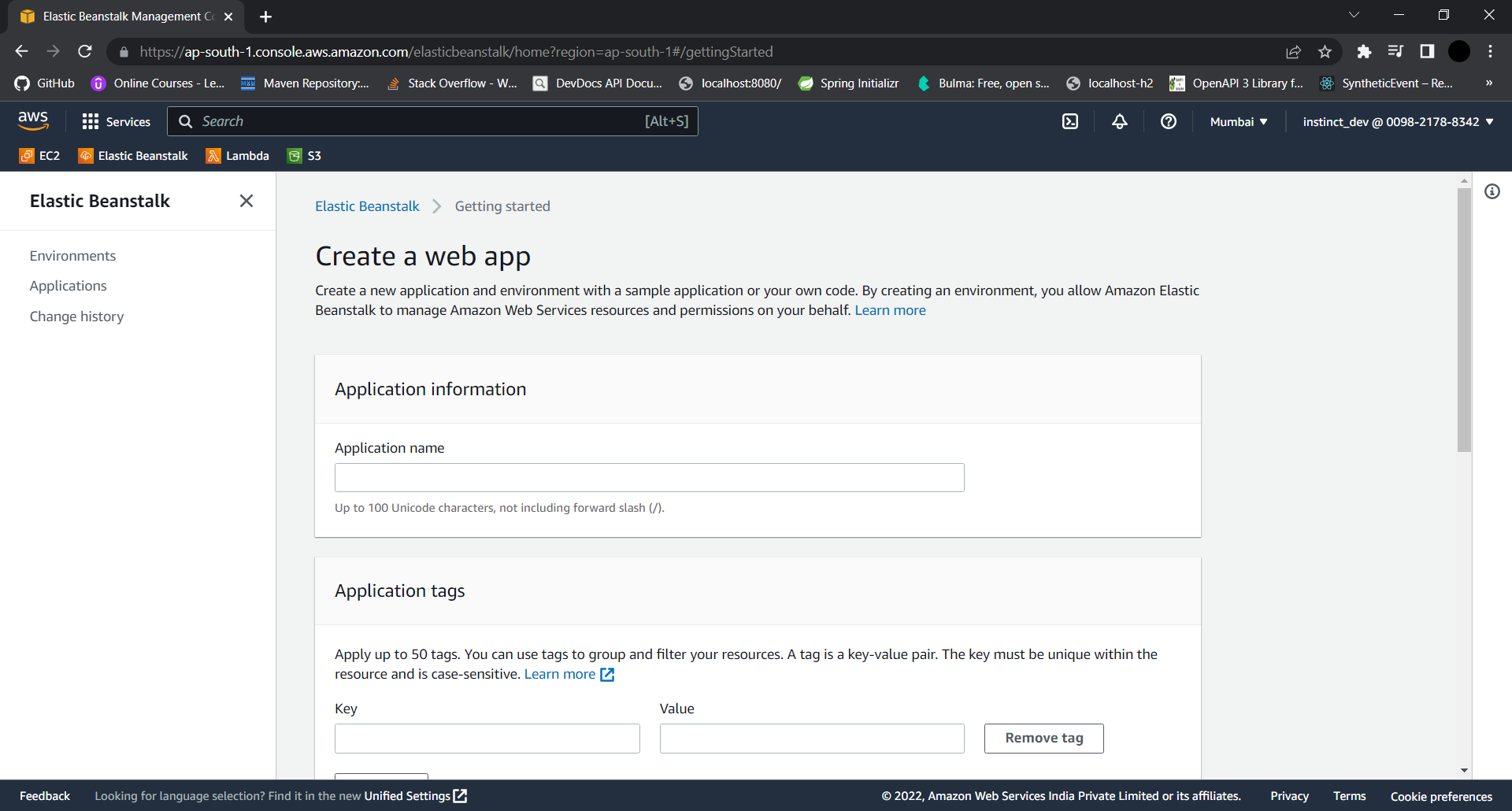
- npm run build // creates a build directory with a production build of your app.

**First push REST-API to AWS:**

Go to aws console and search for **Elastic Beanstalk,**

****

Click on **Create Application** button.



Set **Application Name**, Choose **Platform** as **Java,** Click on **Application Code** and Click on **upload your code.**

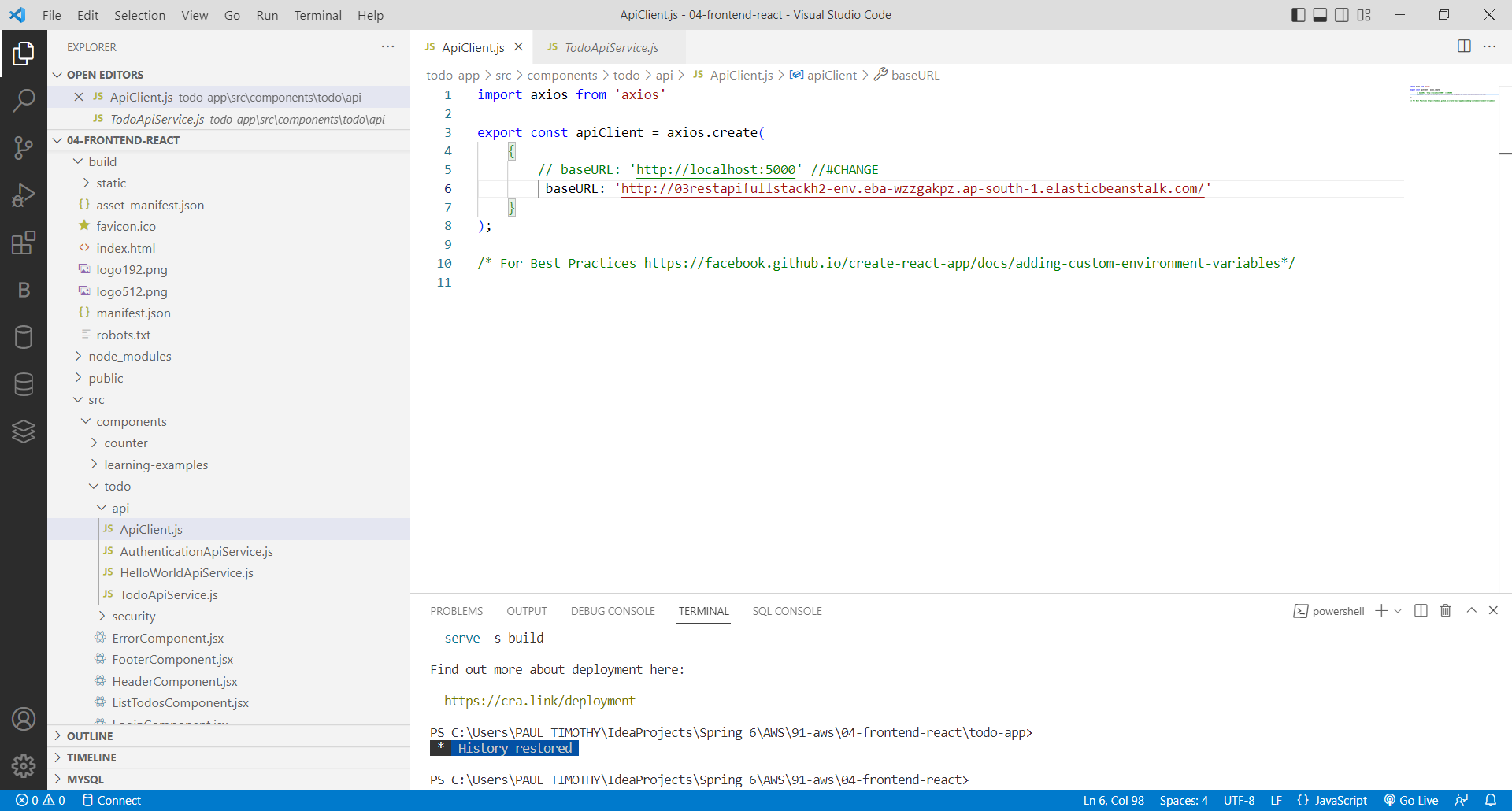
In **Source code origin,** give a **Version Label,** andupload the **.jar file.**

**Click on Create Application button.**

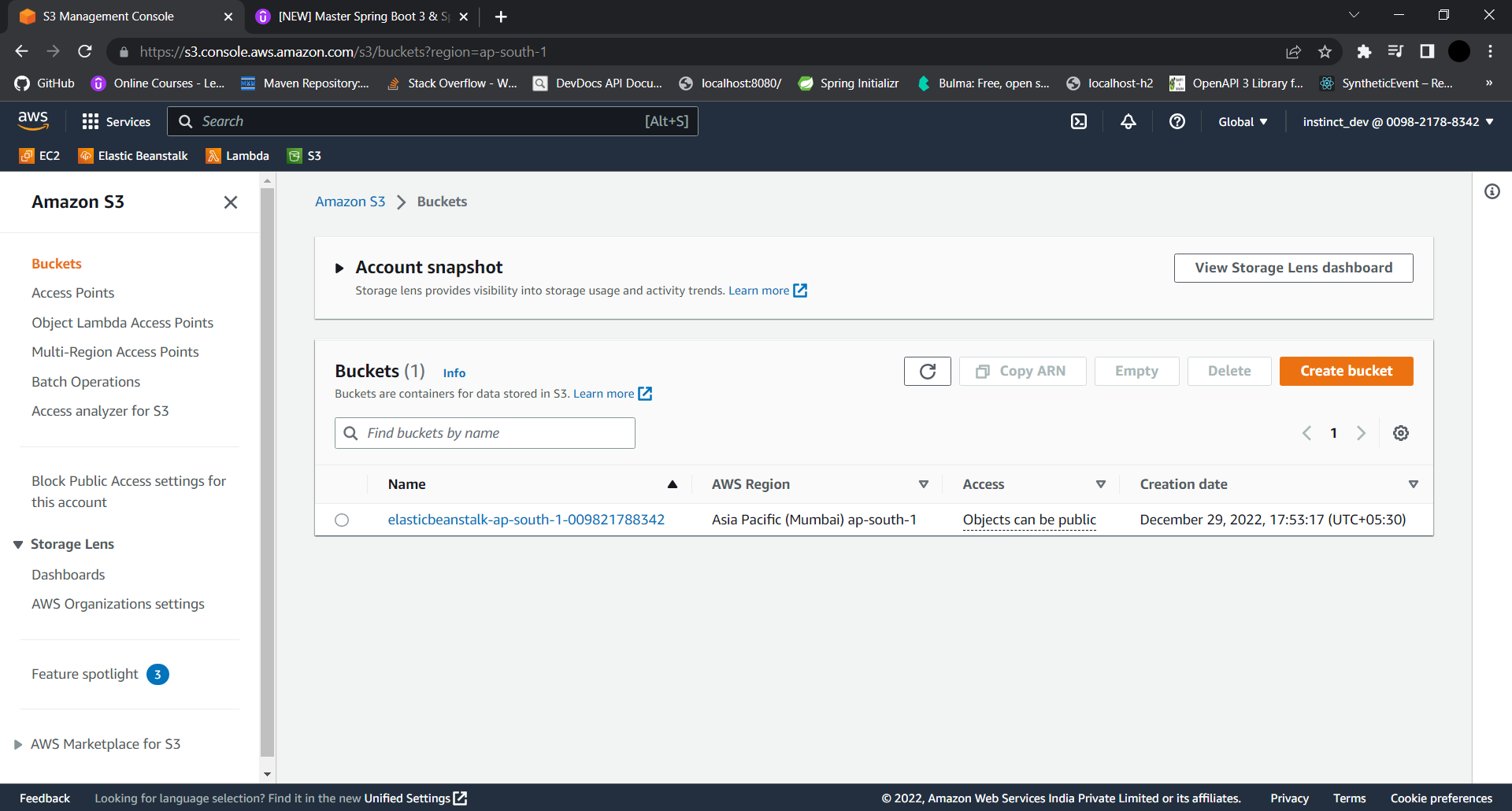
**We will get a link at the end to access our REST API, Like**

03restapifullstackh2-env.eba-wzzgakpz.ap-south-1.elasticbeanstalk.com

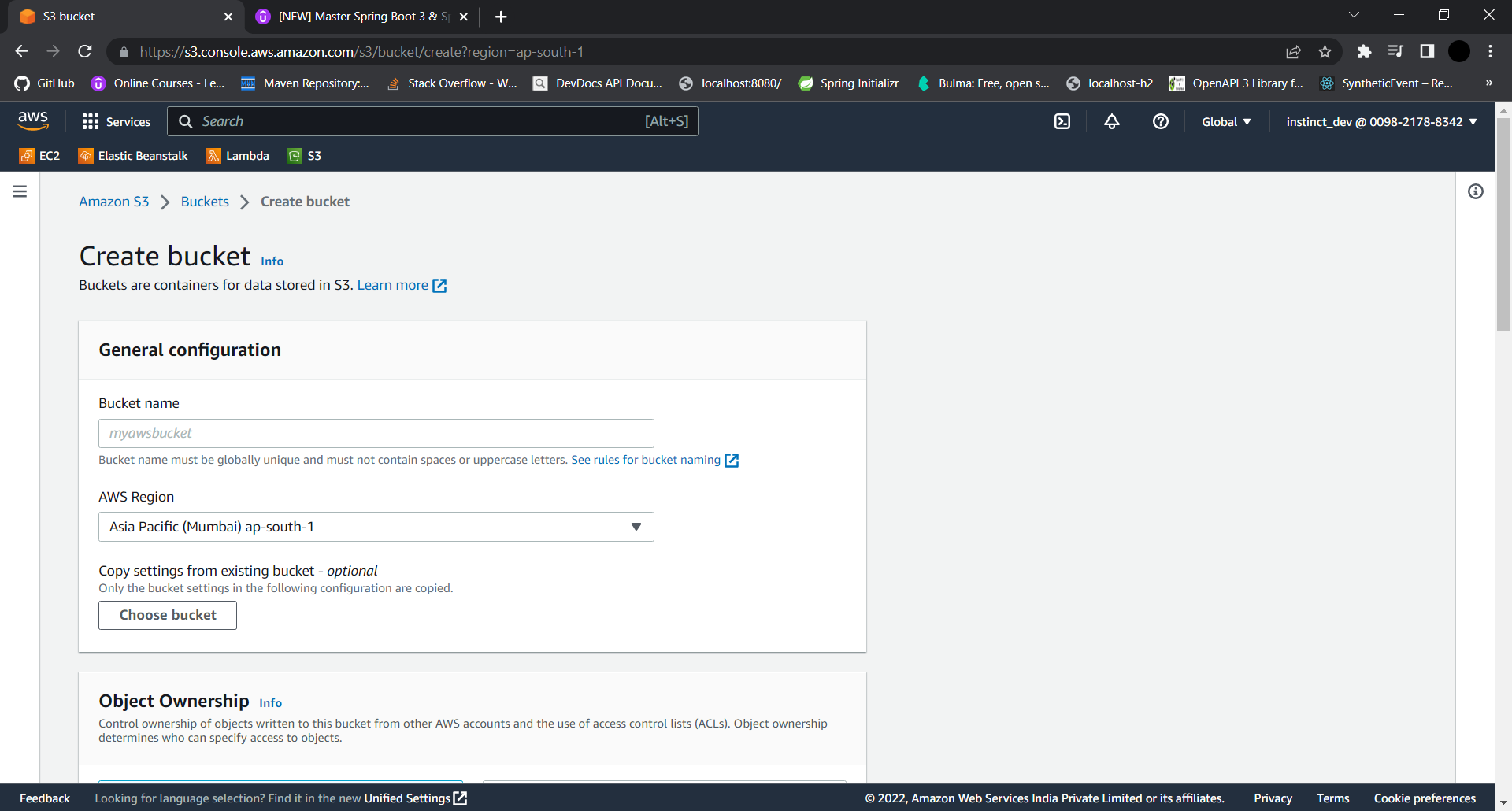
Now take the link and go to the **apiclient.js**, so that our front end communicates with cloud deployed REST-API. Now you can run your frontend and also work with rest-api and **run npm build**.

****

So, if we want to push **React Frontend** to **AWS**, Go to **S3**

****

Click on **create bucket** button on the right,

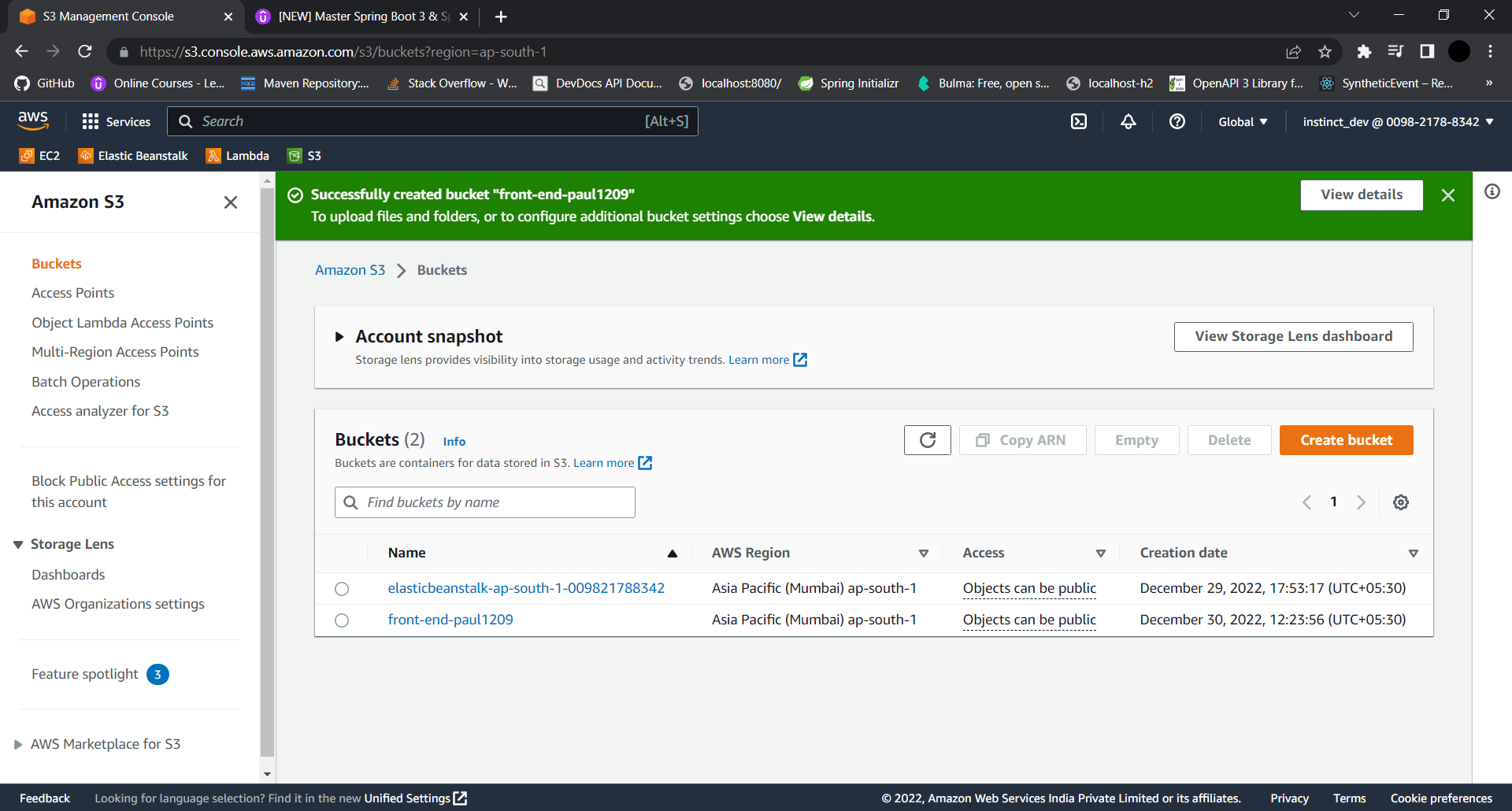


Set the **Bucket Name**(Gloablly Unique),

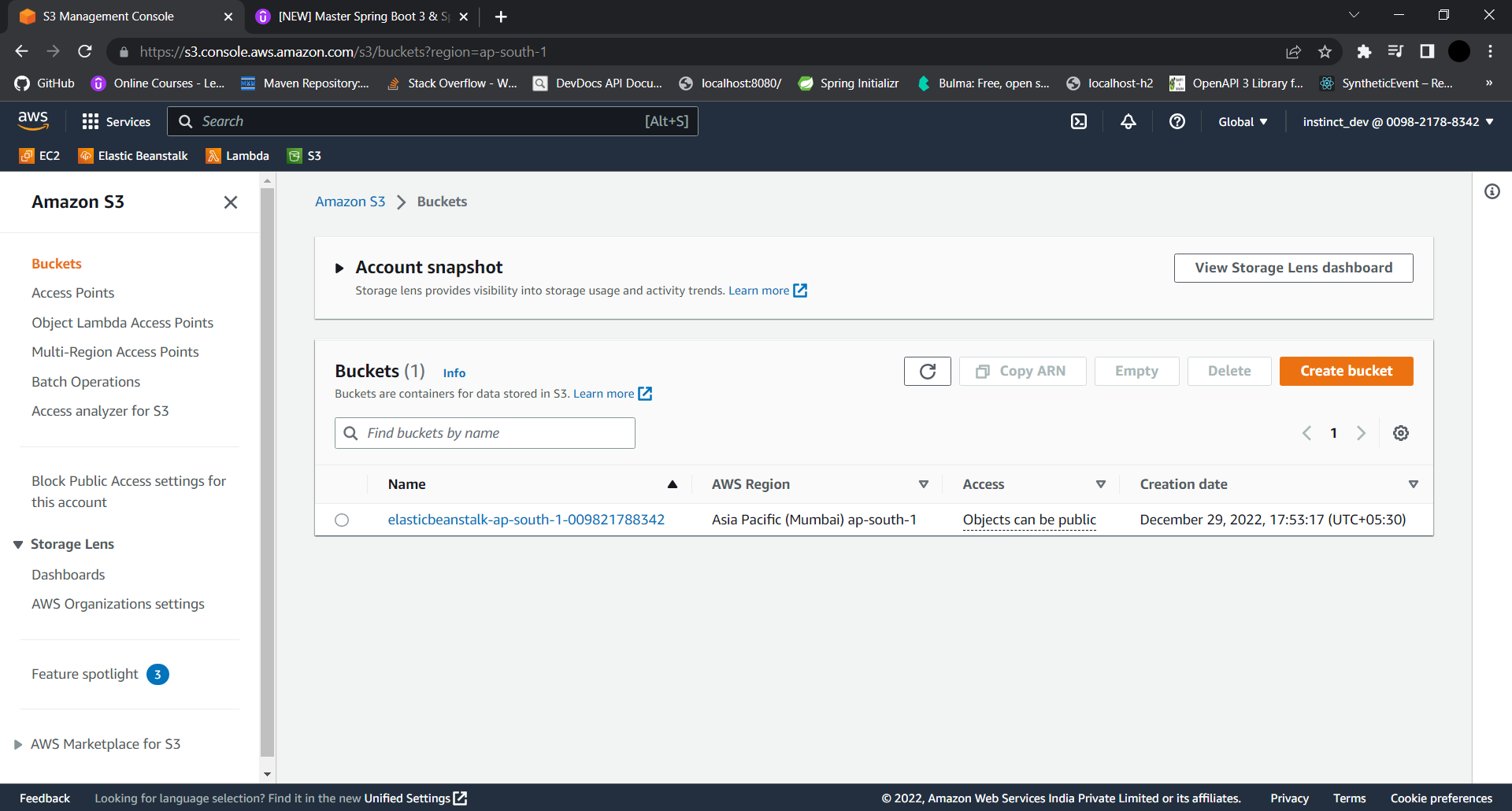
In **Block Public Access settings for this bucket**, **uncheck block all public access**, as we are making a rest-api, we should expose the data to outside world.

Click on **Create Bucket** button.

After creating our bucket,

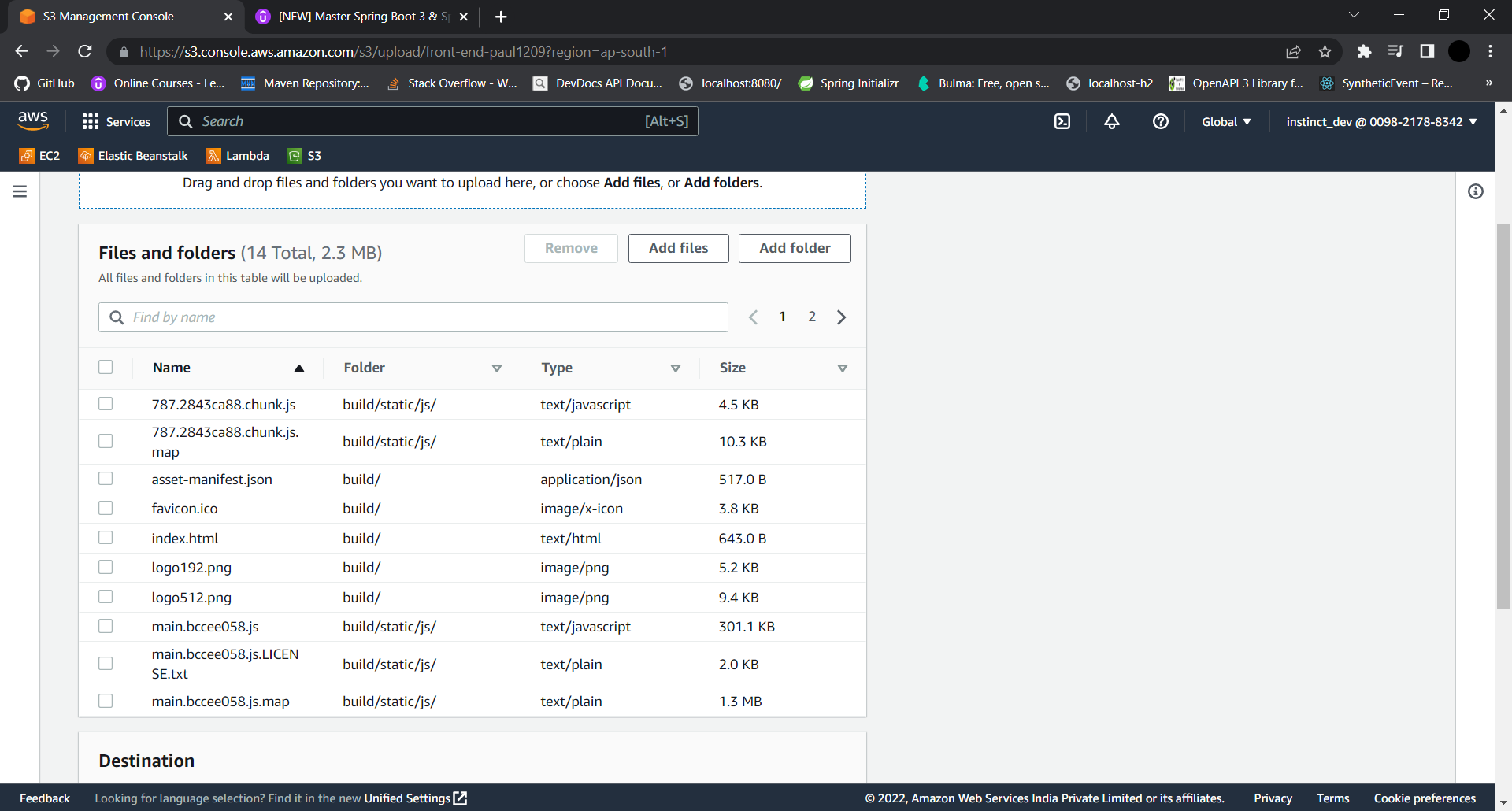


Click on the **name of the newly created** bucket,

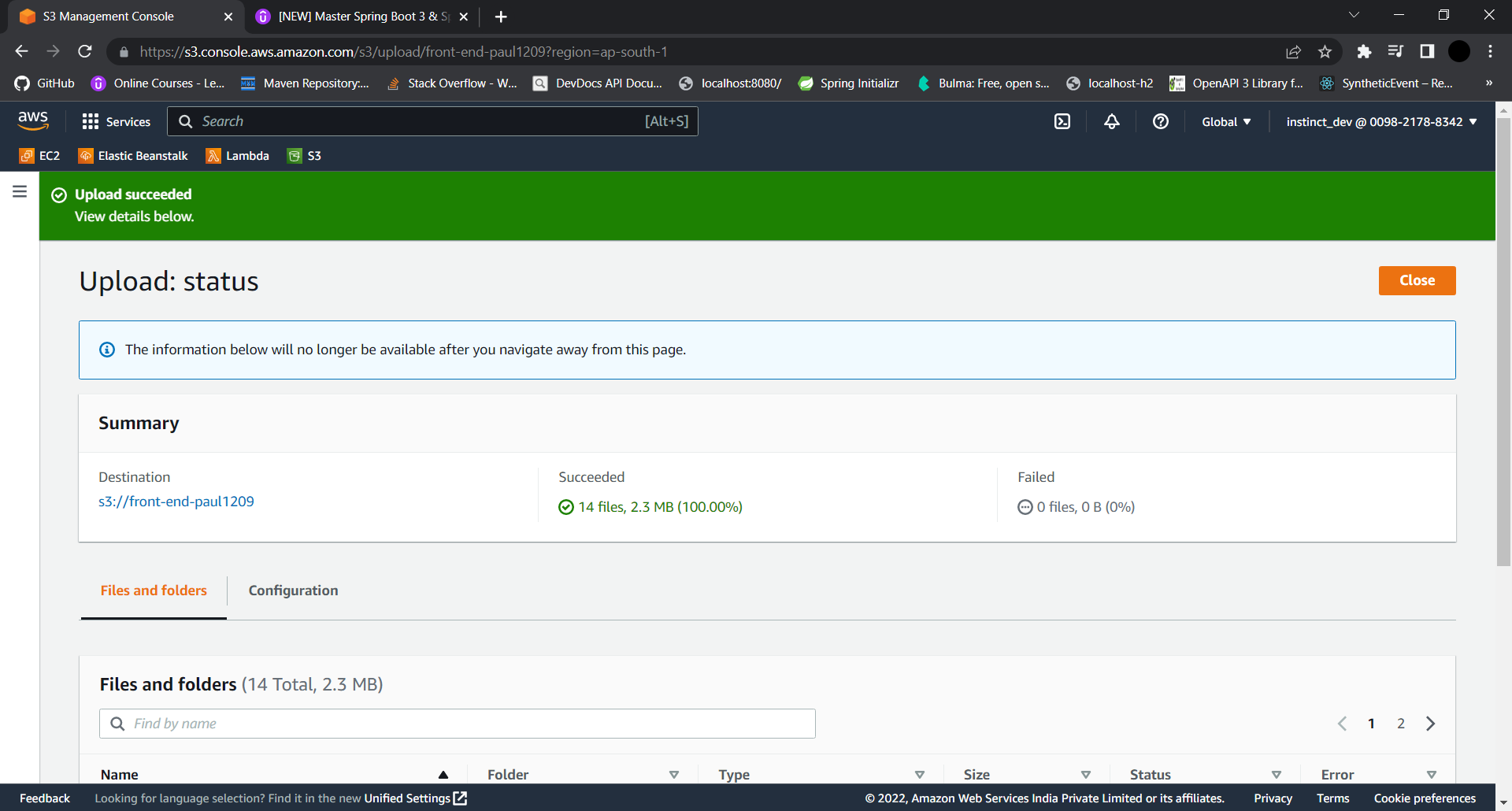


Click on the **upload button** and upload **all the files** on the **build** folder including **static folder** within the build folder.

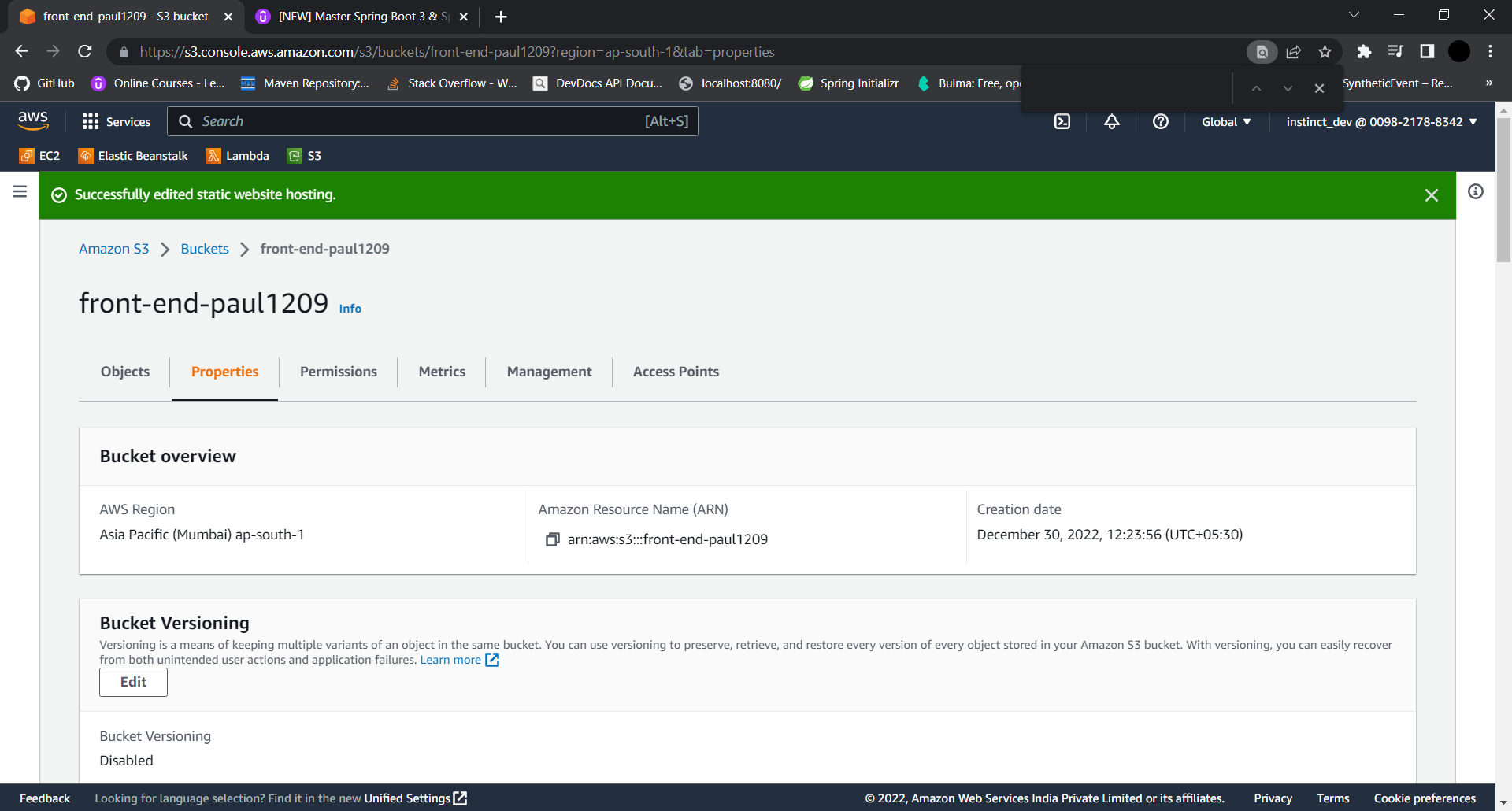
After uploading,



And click on **upload** button at the end.



Click on **close** on the right.



Now we need to **create a static website**,

Go to **properties**,

In **Static website hosting,** click on **edit** and **enable it.**

In **index document,** type **index.html** and **save the changes.**

It generates a link for our front-end application.

Go to **permissions**,

In **Bucket** **Policy**, Click on **edit**, refreshes to a new editor, click on **Add new statement** button.

Change value for following things:

“Principal”: “\*”

“Action”: “s3:GetObject”

Just above the editor there’s something called bucket ARN, copy that and add it

“Resource”: "arn:aws:s3:::front-end-paul1209/\*"

And click on **save changes**.

After saving, In permissions overview, the access will be set to **public** as a warning.

Now go back to **properties**, Scroll down to **static website hosting**, and our link is successful.