**Steps to be done to run the project :**

- Run the docker desktop

**\* Thing to be done in build-server/scriptjs :**

- In script js give the region(of bucket created) ,access key, secret key

- You have to give the project\_id in script js where you have to write the project it in the env  
- Create the bucket in the aws and paste the name into the **Bucket   
-** After creating the bucket to make is publically accesible you have to give the acces by adding the following policy:   
- You can create the access key and credential by creating the iam user and creating the access key.   
  
{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Principal": "\*",

"Action": "s3:GetObject",

"Resource" : "arn:aws:s3:::vercel-clone-jd/\*"

},

{

"Sid" : "AllowWebAccess",

"Effect": "Allow",

"Principal": "\*",

"Action": "s3:GetObject",

"Resource" : "arn:aws:s3:::vercel-clone-jd/\*"

}

]

}

- Replace the resource with the new bucket generated

**\* Thing to be done in aws:**

- create the ecr and push the image created inside the builder server folder.

- while pushing you might get the error so do the aws configure and paste the new access key and secret key   
- Create the ecs cluster

- Now copy the uri image and create the new task in the task defination and paste the uri and give the name.

**\* Thing to be done in s3-proxy/index.js:   
  
-** in the base path give the s3 bucket url upto the output and not the subdomain

- copy the esclient from the script.js.

- in config : copy the cluster arn and also the arn of new task created in the task defination .

- create the new task and in the network copy the subnet and security group (You don’t have to actually create the new task just for copying the info we are doing it)

- get the image name from the task defination json.   
  
- run Aiven redis

Bucket

Iam

Ecs

Ecr

Task

Cluster

Vpc

rediss://default:AVNS\_cKnXkMN0FdIhQxAuOGw@redis-33-jemmy33-9bc1.j.aivencloud.com:10261