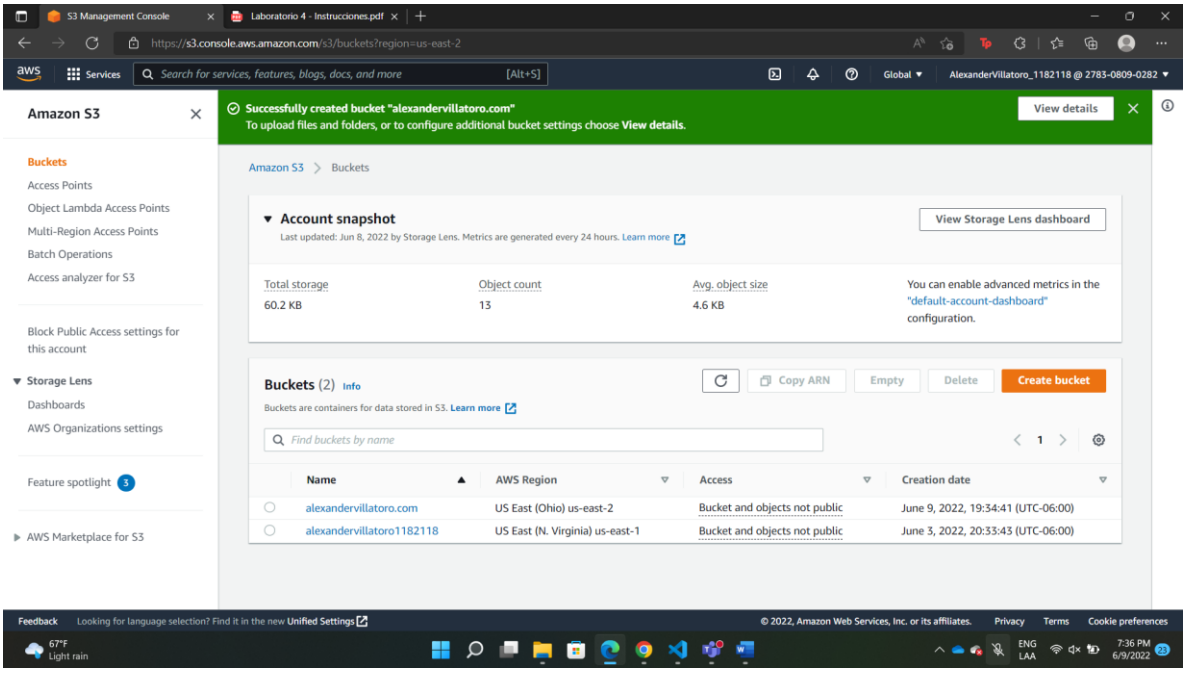
Universidad Rafael Landívar
Tradición Jesuita en Guatemala

Facultad de Ingeniería
Ingeniería en informática y sistemas
Virtualización, sec 01
Jornada vespertina

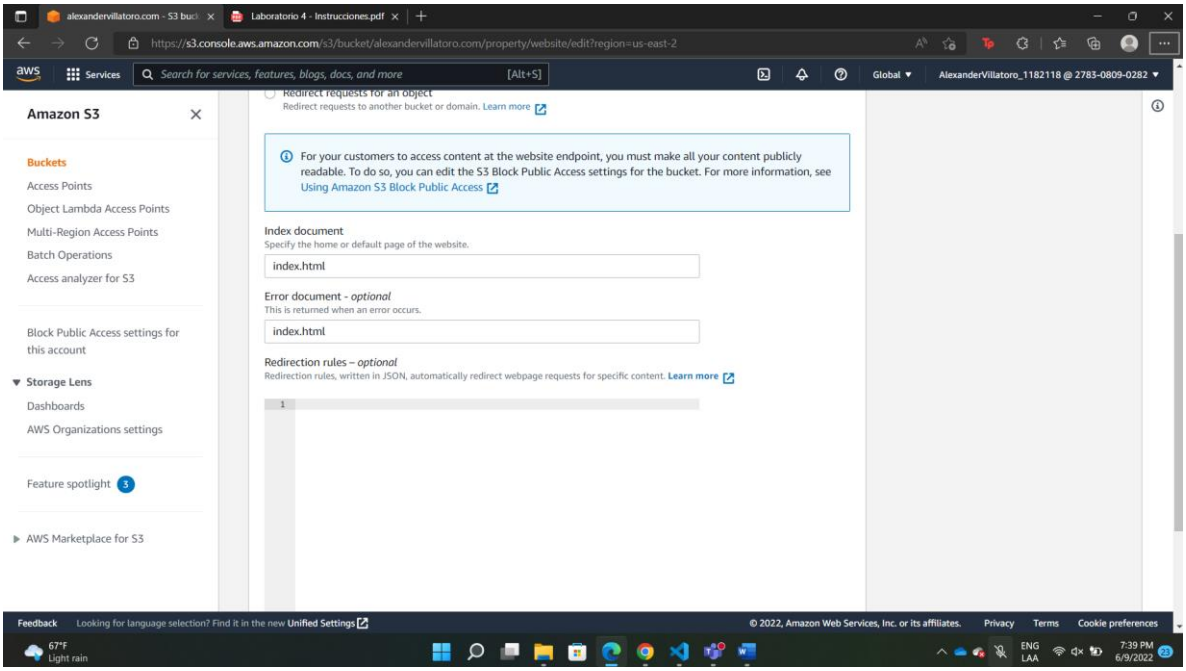
Nombre: Alexander Villatoro
Carné: 1182118

Laboratorio # 4

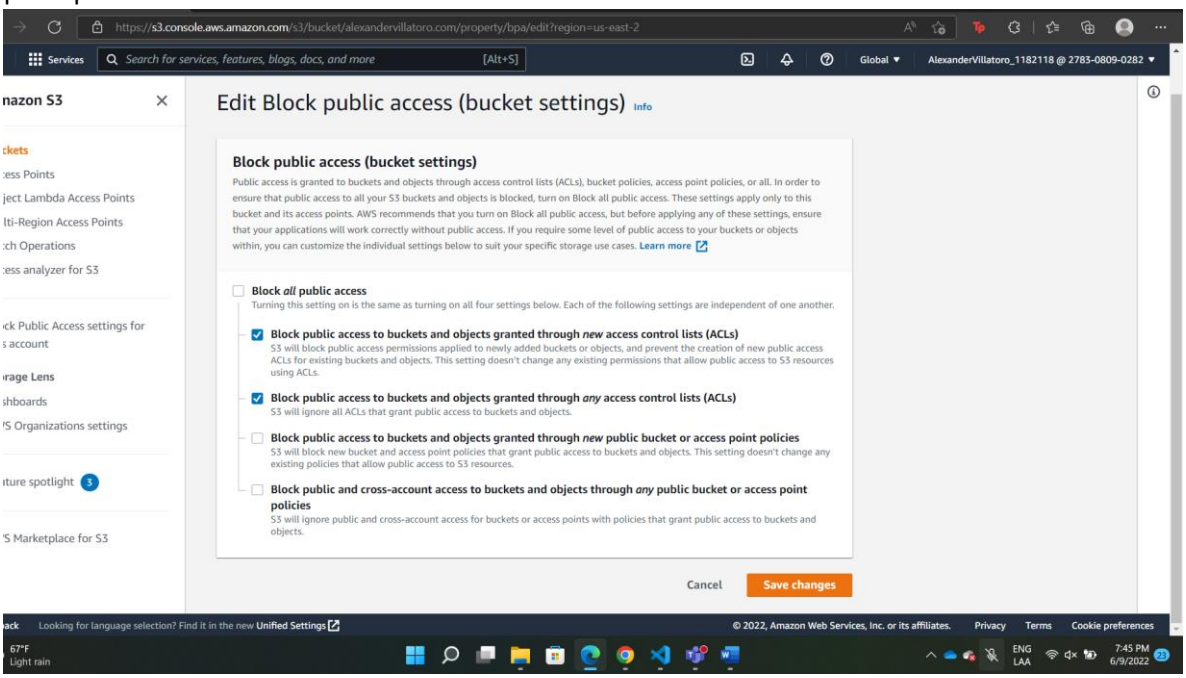
Crear un nuevo bucket con su nombre completo y dominio .com. (cokaescobar.com).
Denegar versionamiento, logs, encriptación, métricas CloudWatch y acceso público.



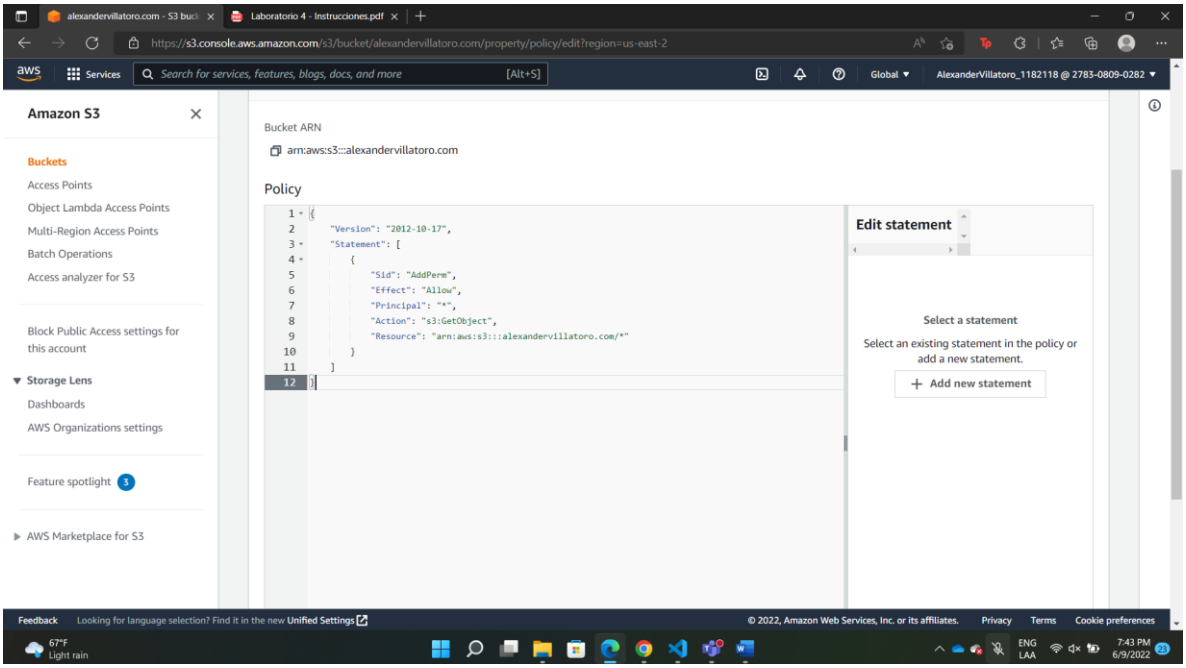
Subir al bucket un archivo con código html de nombre index.html.
En Propiedades, habilitar el Static website hosting, colocando el archivo index.htm como Index Document.

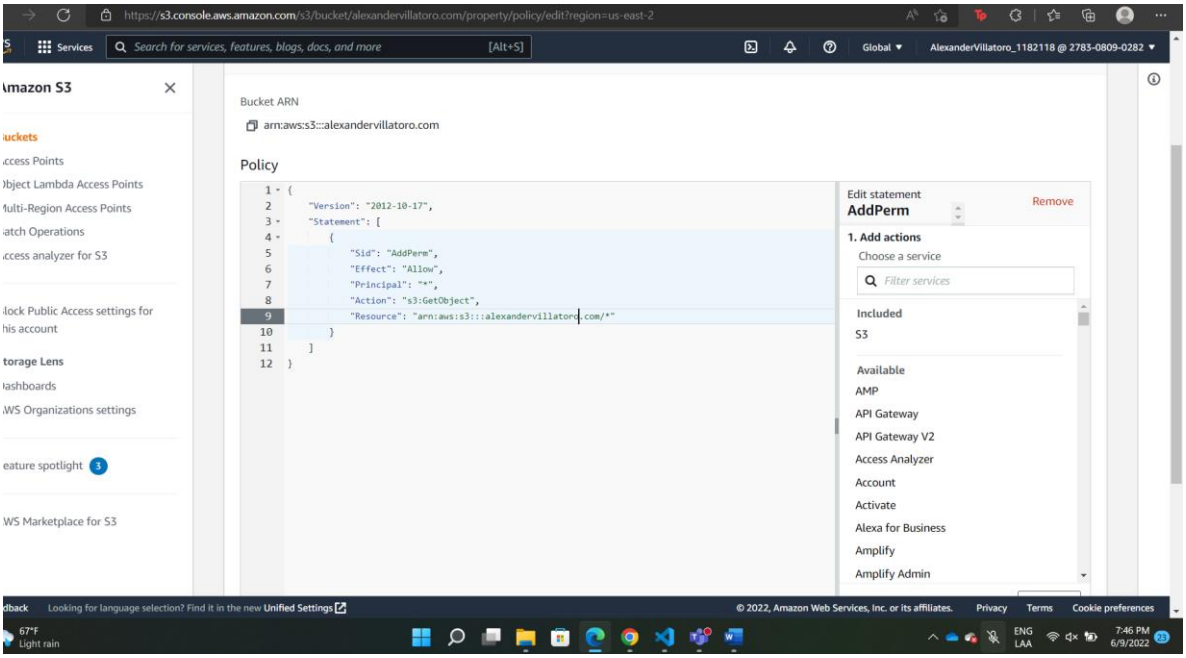


En Permisos, editar los permisos de Public Access desactivando las siguientes opciones. • Block public access to buckets and objects granted through new public bucket or access point policies • Block public and cross-account access to buckets and objects through any public bucket or access point policies

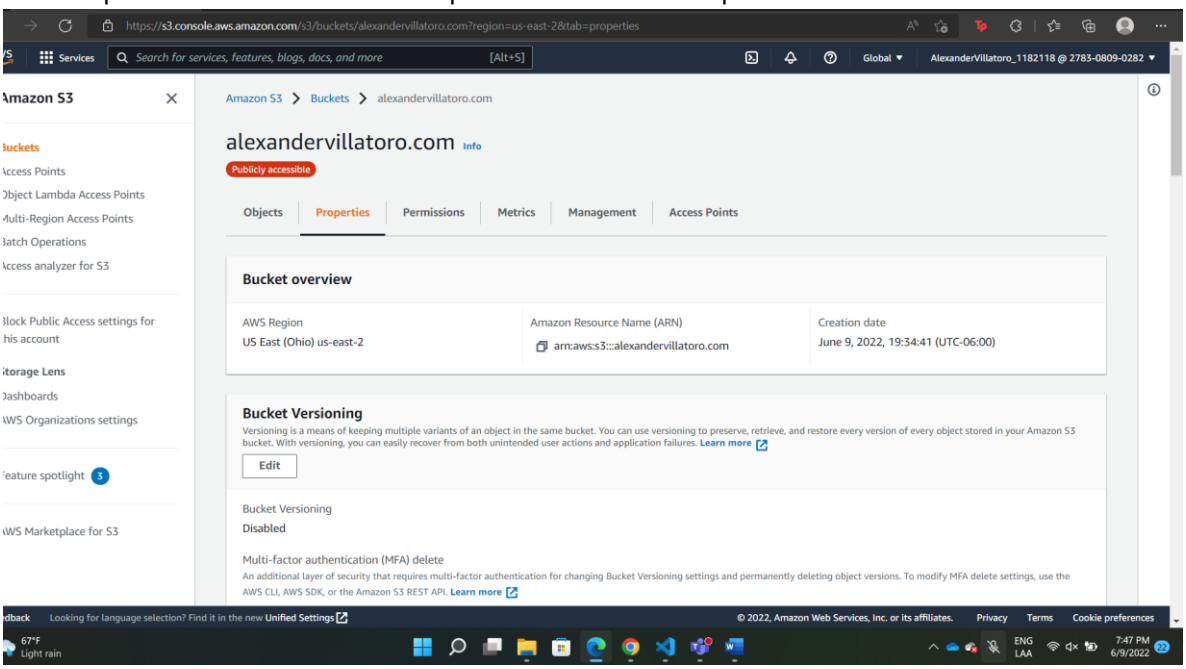


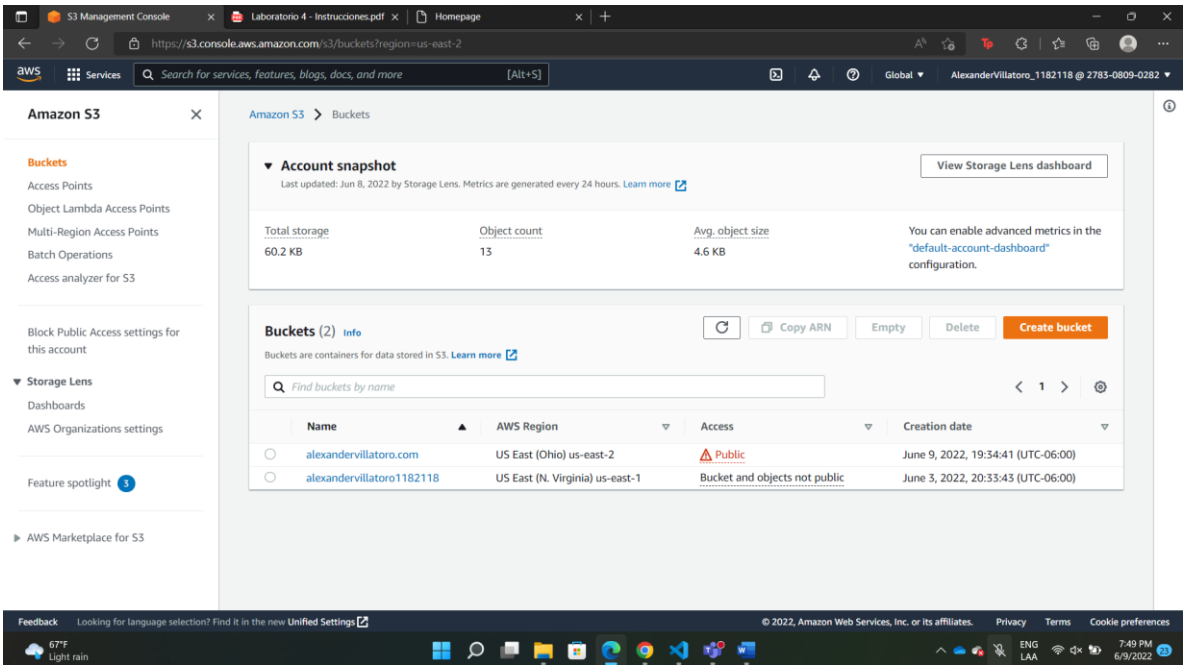
En Bucket Policy, copier y pegar el JSON adjunto en su portal. Sustituir el nombre del bucket en la línea 9.

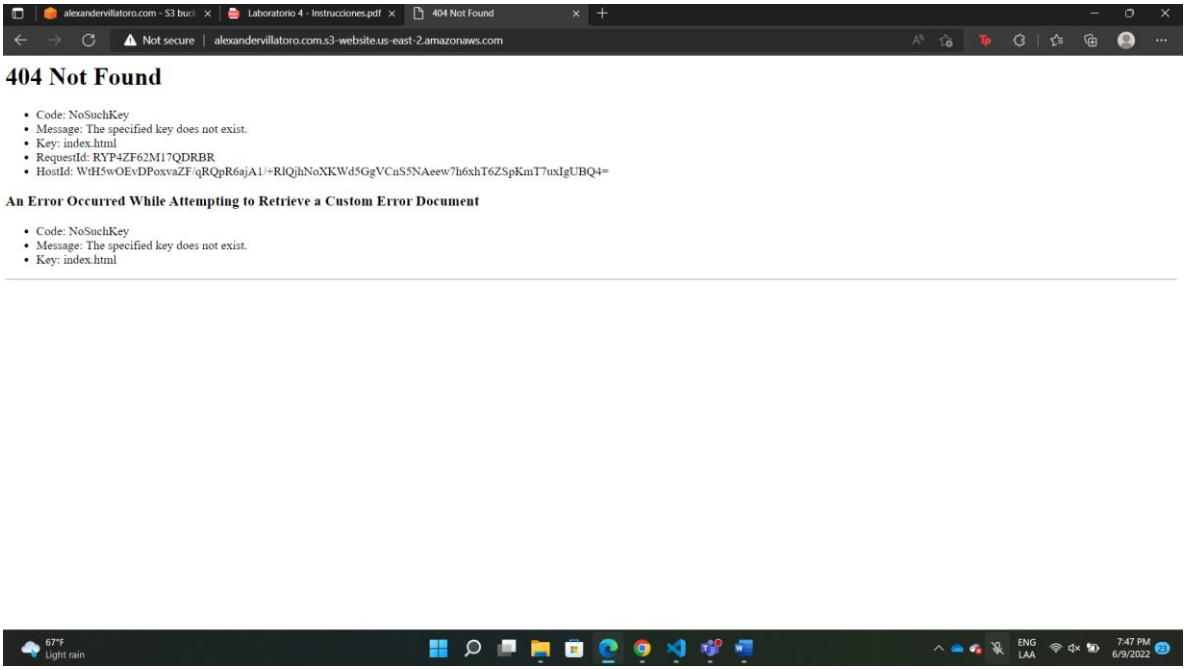




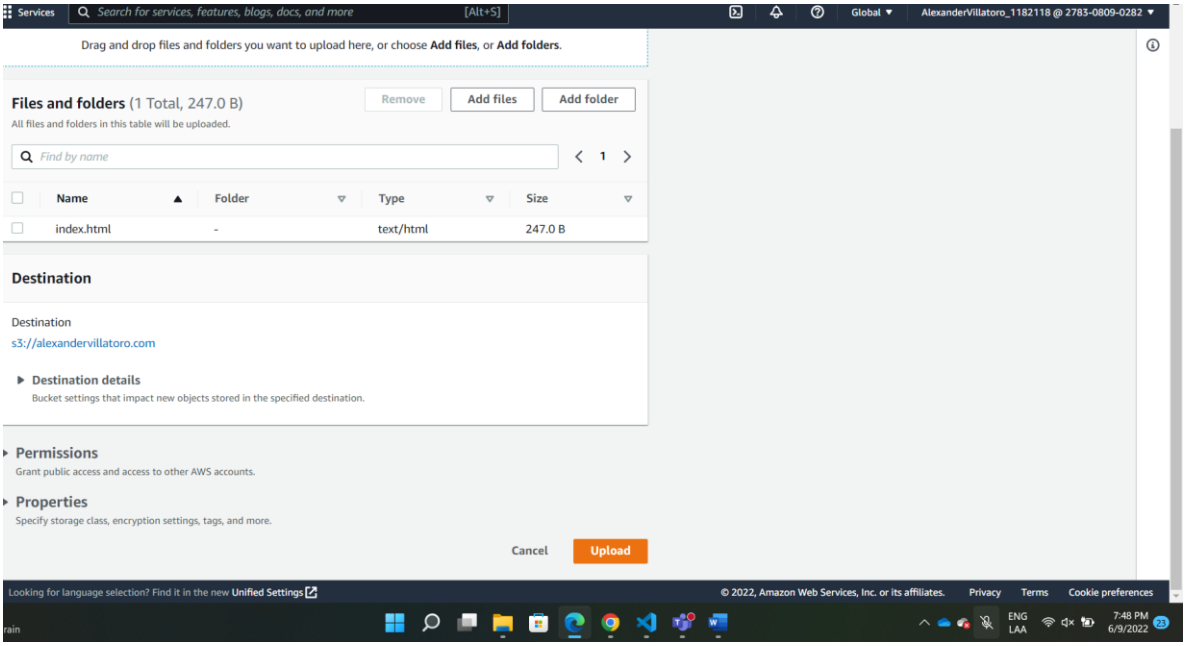
Validar que el bucket sea reconocido por AWS como bucket público

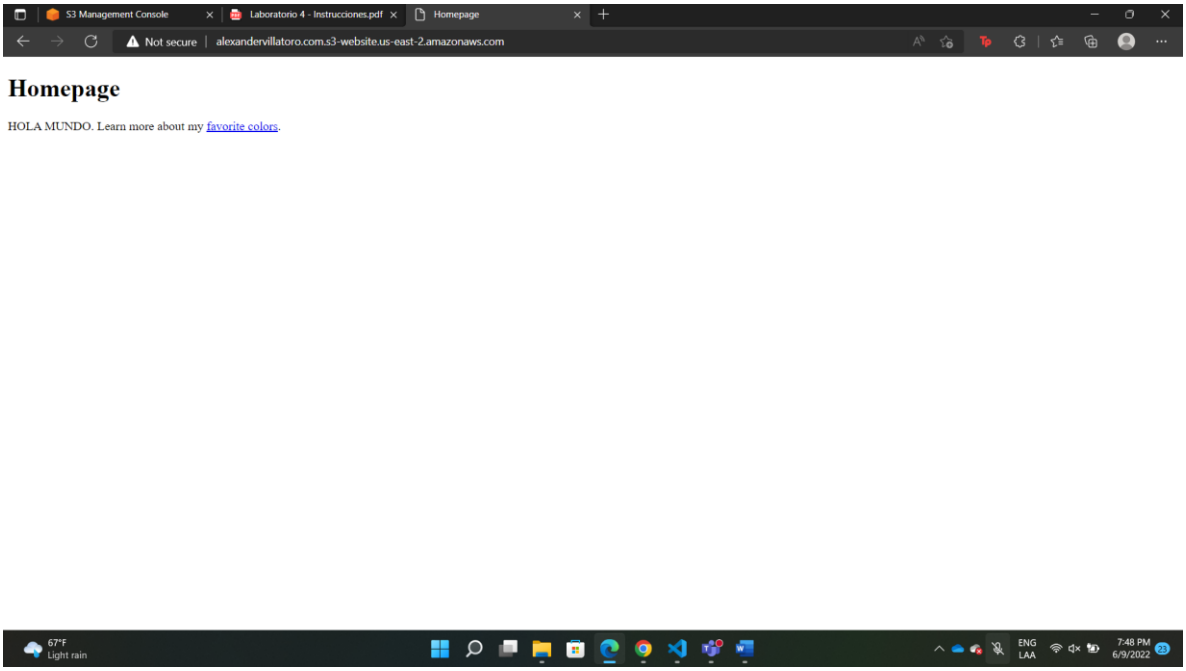






Obtener el endpoint del bucket y probar su funcionalidad.

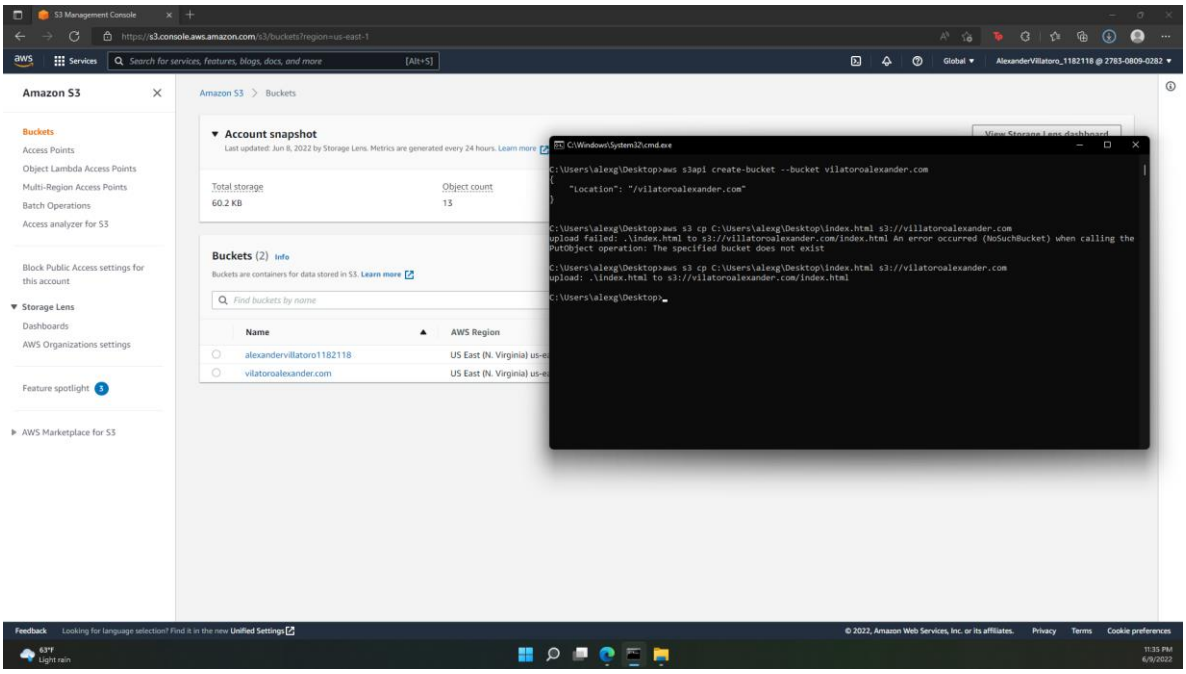




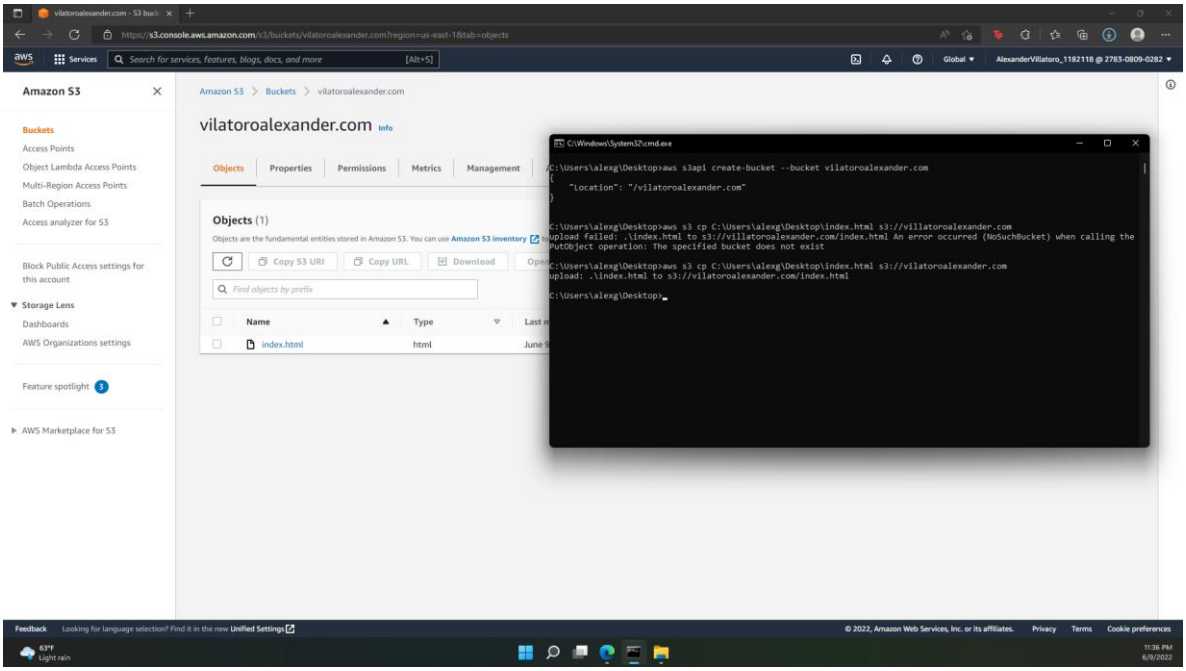
Parte 2

Repetir toda la parte 1 del laboratorio utilizando la AWS CLI.

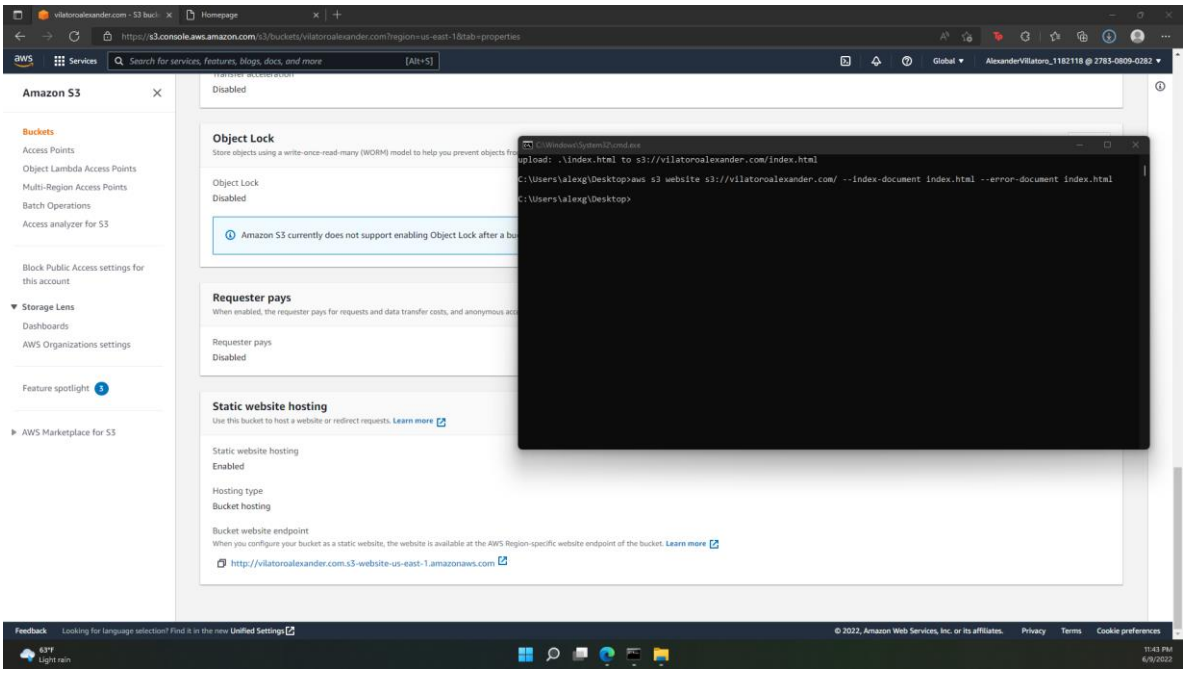
a. Los pasos correspondientes a permisos de acceso público y bucket policy pueden realizarse de forma manual.
b. b. Dejar por escrito los comandos utilizados y los screenshot correspondientes para validar las configuraciones.



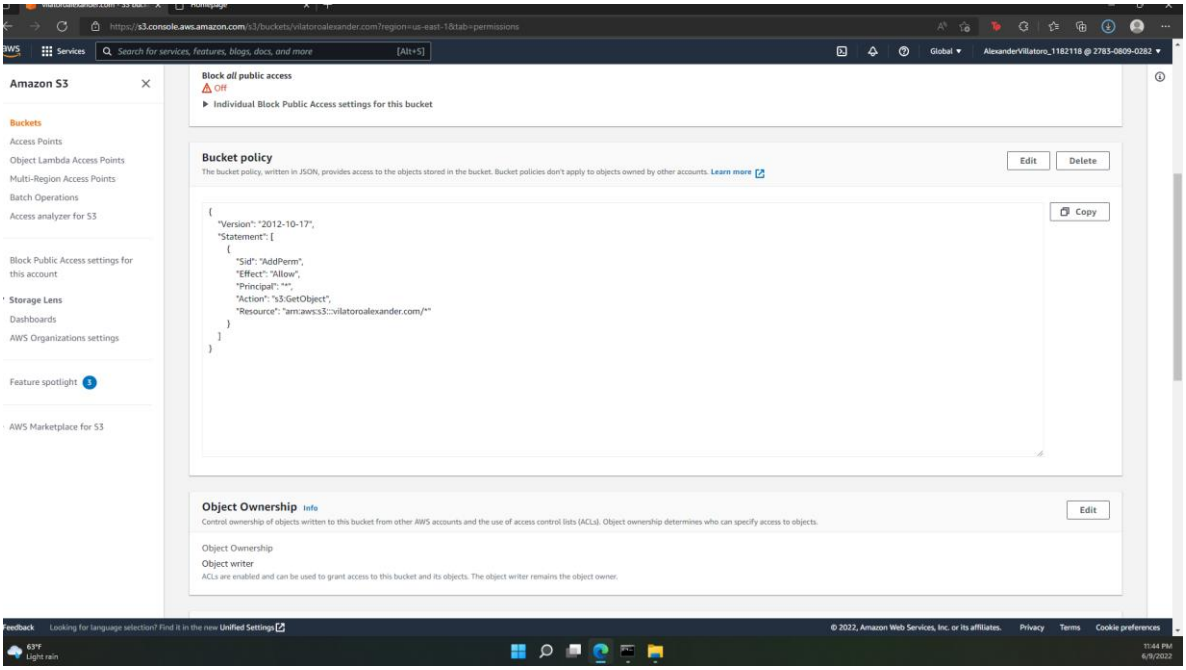
El comando create-bucket crea un bucket con un nombre dado bajo el parámetro –bucket/



El comando aws s3 cp C:\Users\alexg\Desktop\index.html s3://vilatoroalexander.com nos indica que se va a subir un archivo llamado index.html en el bucket (s3://vilatoroalexander.com) en donde se va a estar subiendo el archivo html.



Se utilizo el siguiente comando aws s3 website s3://vilatoroalexander.com/ --index-document index.html --error-document index.html en donde se habilita el static wesite hosting.



Parte 3

La empresa guatemalteca Molinos del Sur desea migrar una de sus aplicaciones legacy a nueva tecnología, la cual debe ser alocada en un servidor cloud, con sistema operativo Windows y bajo cualquiera de los proveedores siguientes: Microsoft Azure, Amazon Web Services, Google Cloud Platform.

Los requerimientos del sistema son los siguientes:

• Altamente disponible

• Seguridad en acceso.

• Active Directory se encuentra en servidor local de la empresa.

• Cantidad máxima de 100 usuarios concurrentes.

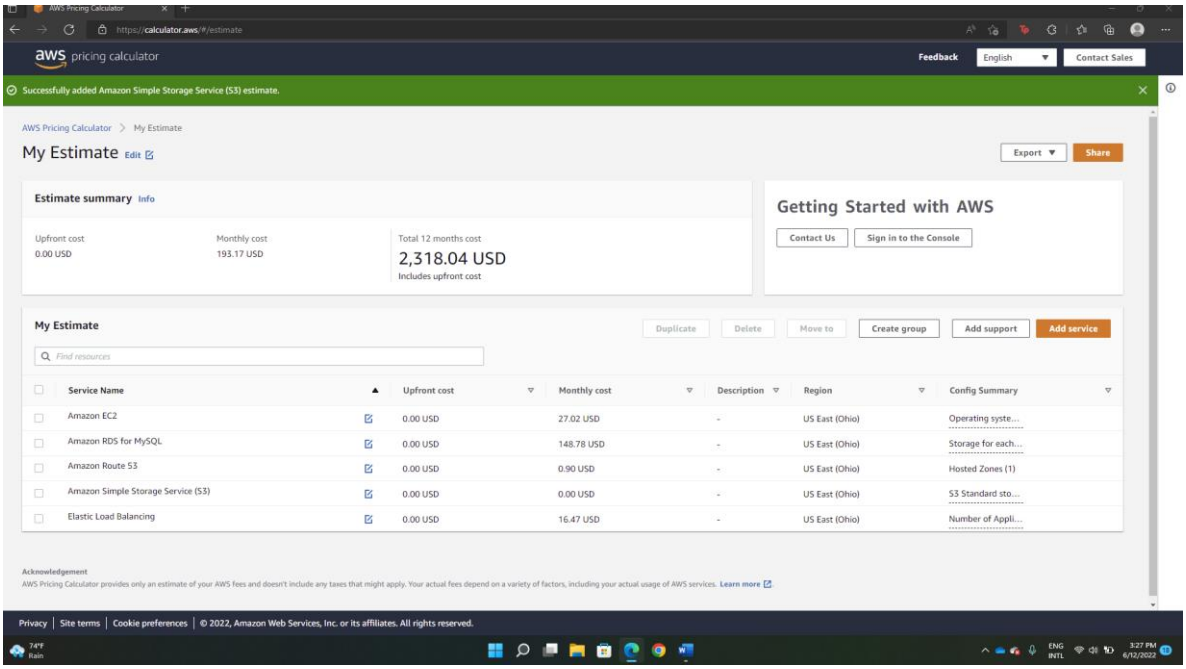
La aplicación debe alocar en el mismo servidor la aplicación y la base de datos, la base de datos actualmente es manejada en MySQL y el cliente está abierto a nuevas propuestas en caso exista una opción más eficiente en rendimiento y costo para los requerimientos ya descritos.

Entregables

• Utilizando lo aprendido en la sección teórica del curso, justificar el mejor servidor, gestor de base de datos y proveedor cloud para dar solución al problema presentado por Molinos del Sur.

• Utilizar la calculadora de costos de cada uno de los proveedores para justificar la parte económica. Validar y justificar el mejor costo mensual.

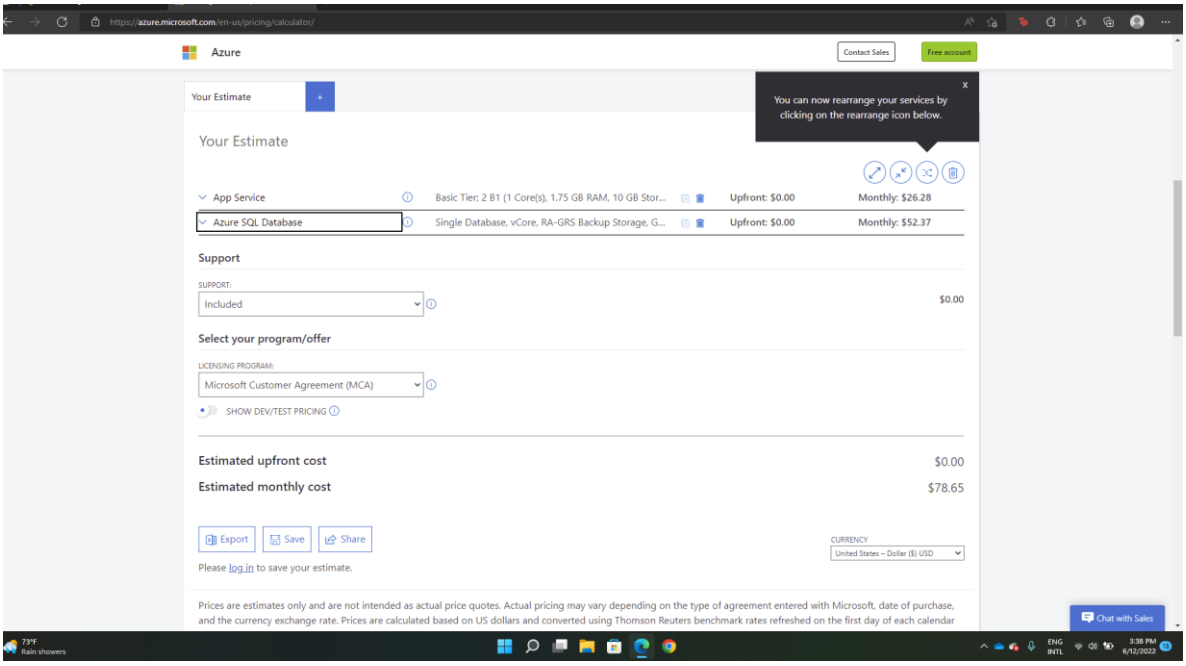
AWS



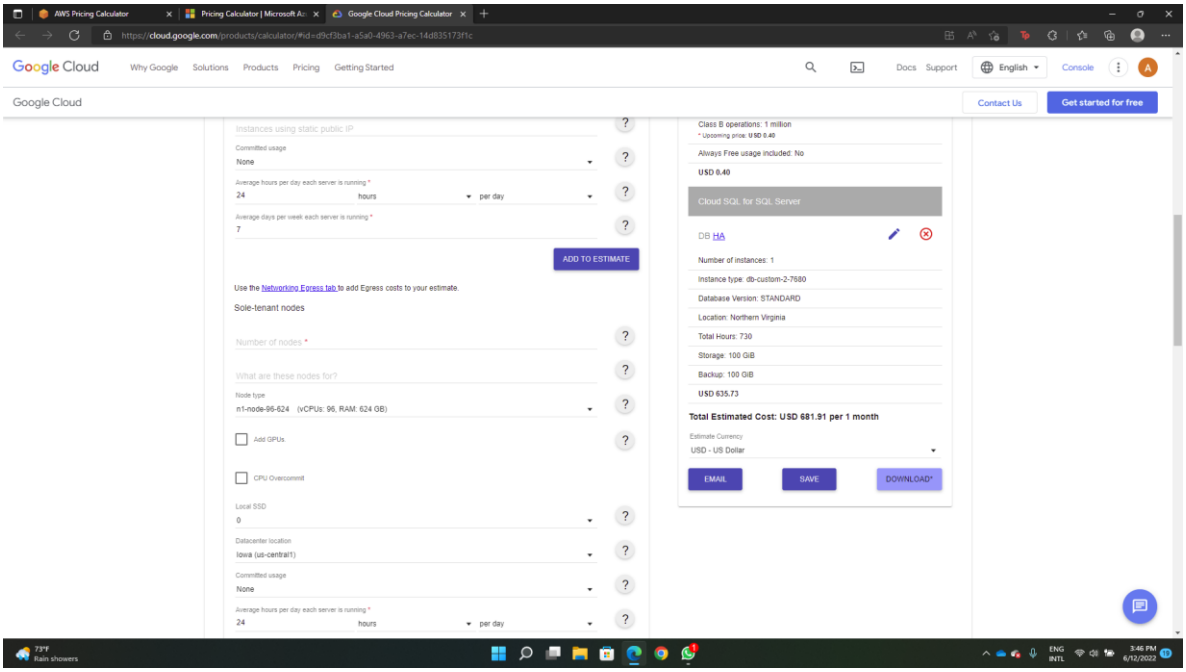
[AWS Pricing Calculator]

Se necesita un balanceador de cargar por las 2 instancias, un RDS de 100GB, frontend en s3 y zona alojada en route 53 para el dominio al que se tiene que acceder.

AZURE



GPC



AppEngine nos da la opción de poder hostear una pagina web o móvil, cloud storage es para almacenar los objetos que se necesitan, y el servidor SQL para el almacenamiento de 100 GB de datos.

Conclusión: podemos darnos cuenta que en costos es mejor Azure al tener los servicios necesarios al menor costo mensual y poder hostear paginas o aplicaciones escalables al menor costo.