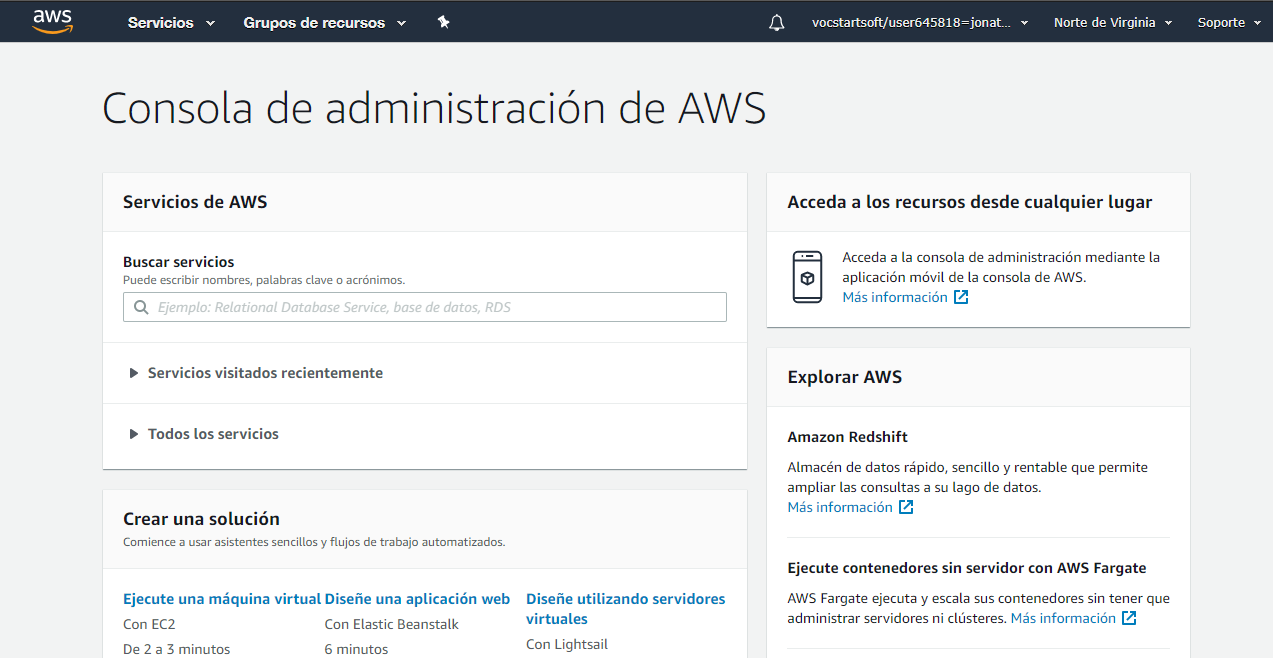
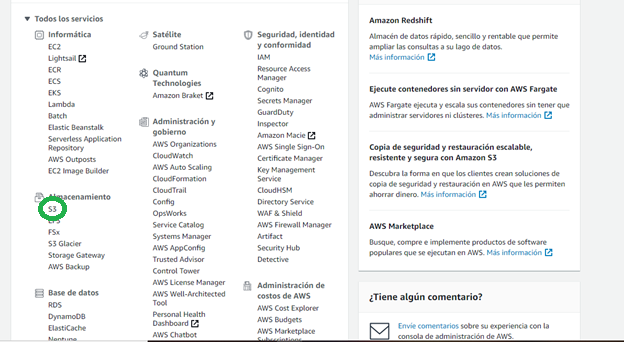
**Desplegar un sitio estático usando S3**

Para esto debemos tener cuenta en AWSEducate o AWS

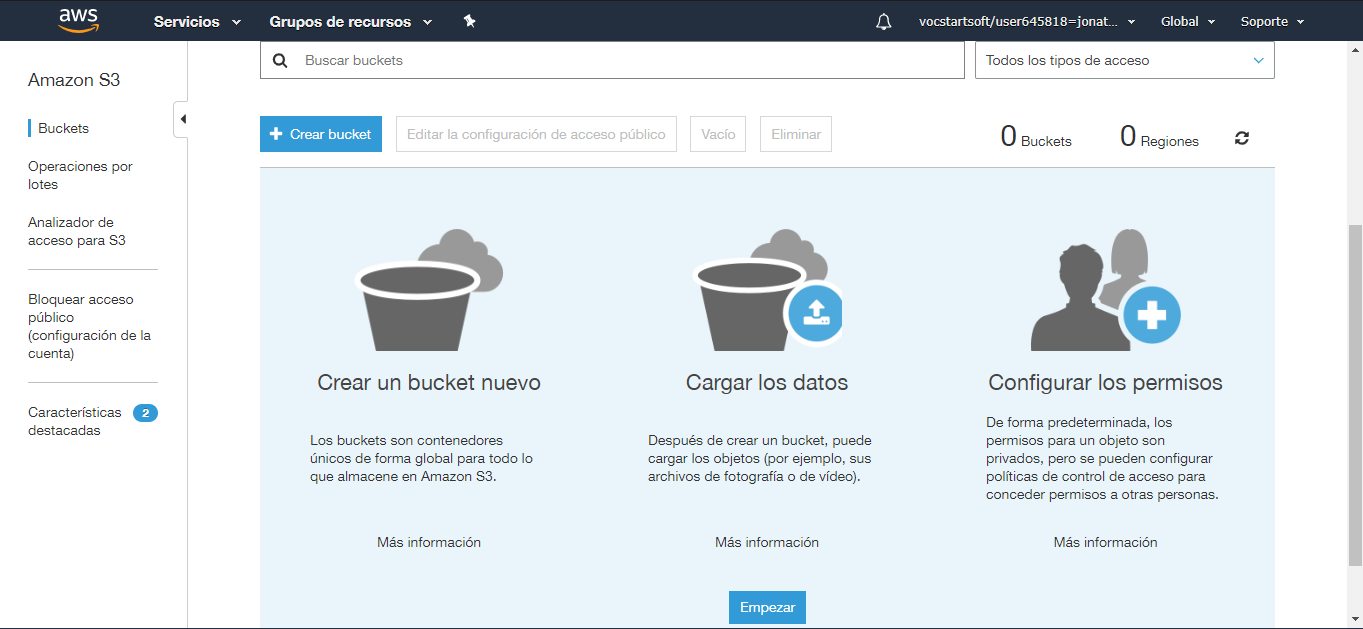
1. Lo primero que hacemos es abrir la consola de Administración de AWS.



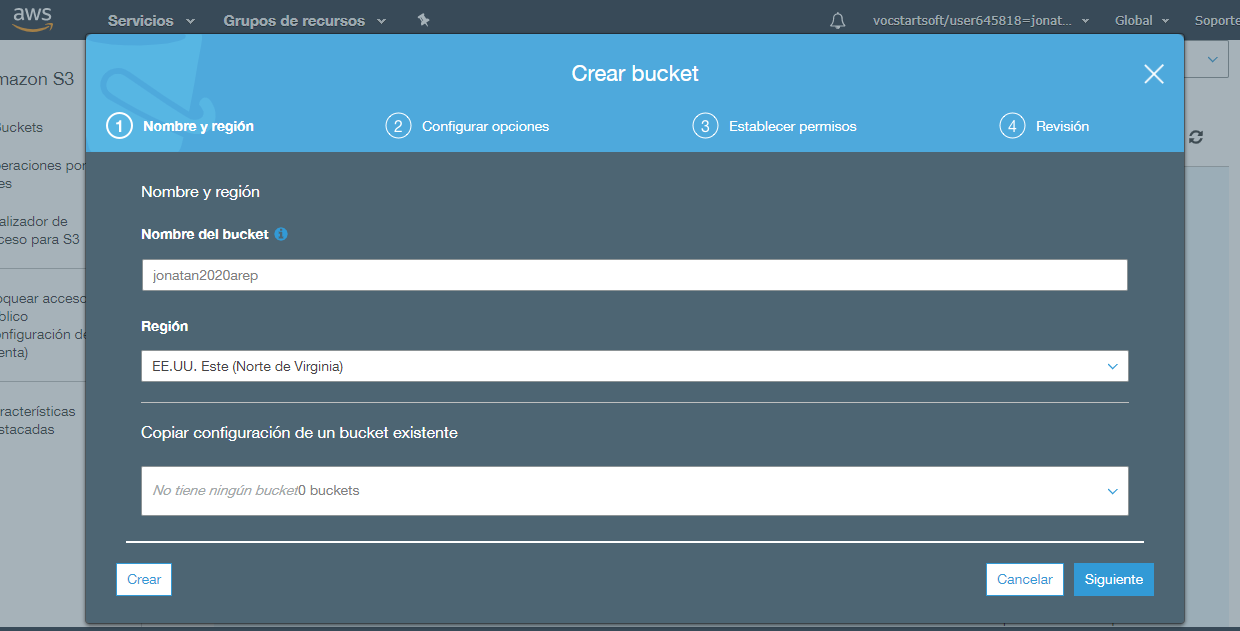
1. Nos dirigimos a la parte de Todos los servicios y seleccionamos S3 en la sección **‘Almacenamiento’**.

****

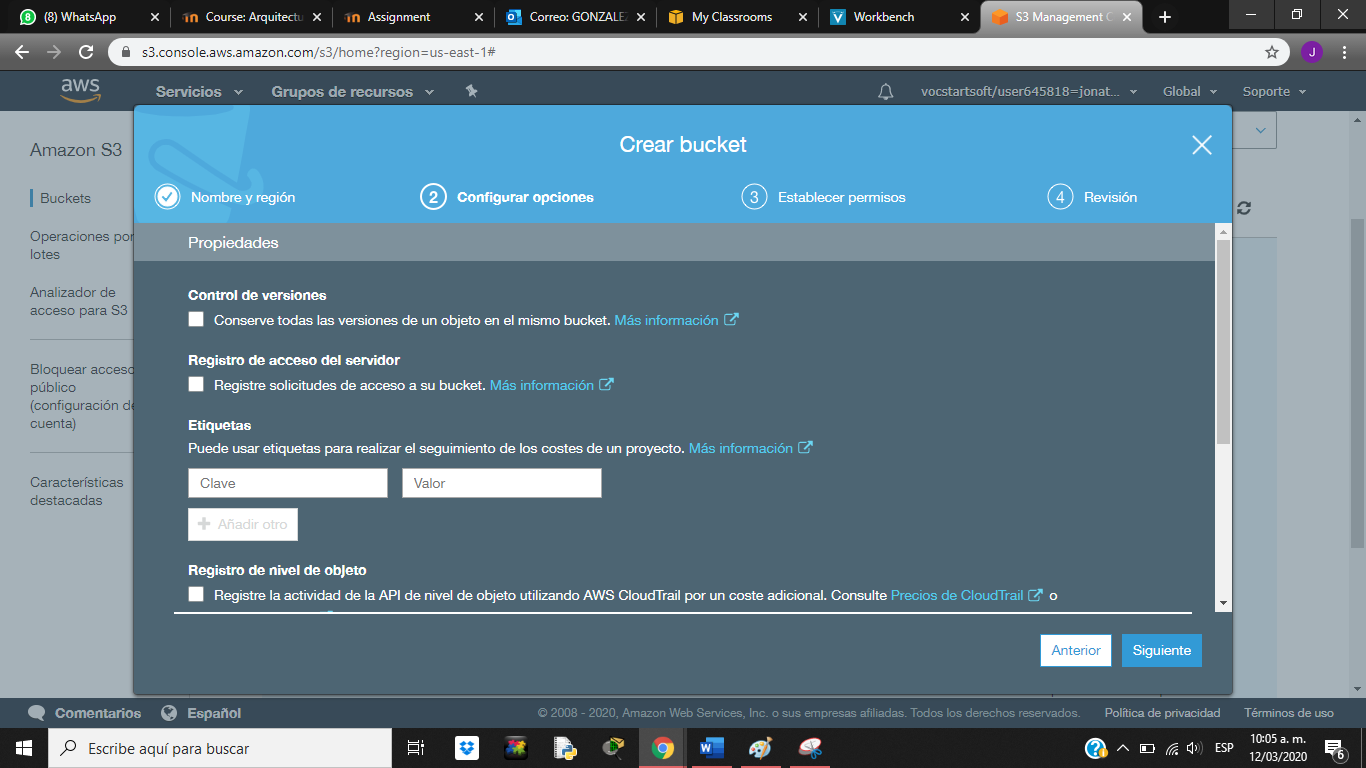
1. **Damos click en el botón ‘Creat Buket’**



1. Le asignamos un nombre al bucket, en mi caso le pondré ‘jonatan2020arep’, también le asignamos la región donde se desplegará el recurso estático, en mi caso la dejare en EE.UU. Este (Norte de Virginia) y luego damos clic en Siguiente.

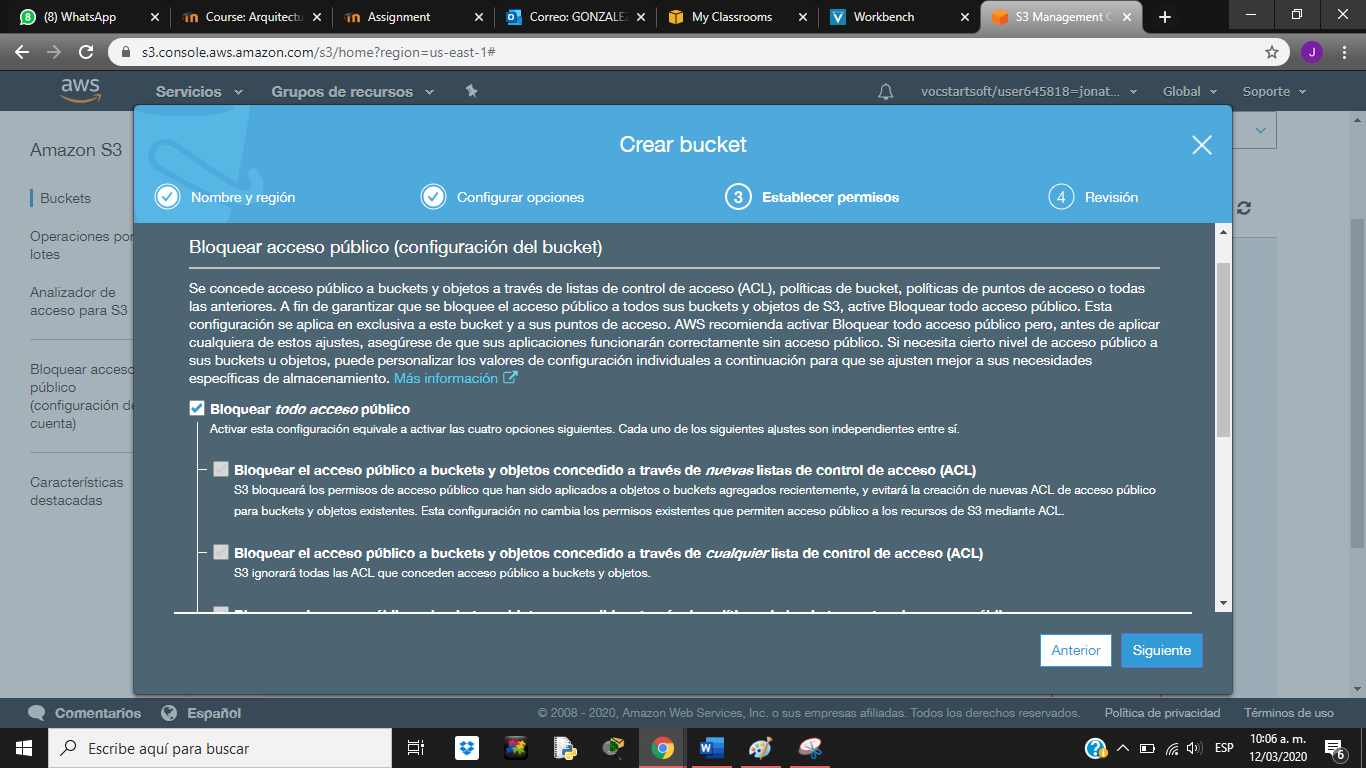


1. Para el caso de Configurar las opciones, no es necesario modificar los valores que vienen por defecto, simplemente damos click en Siguiente.

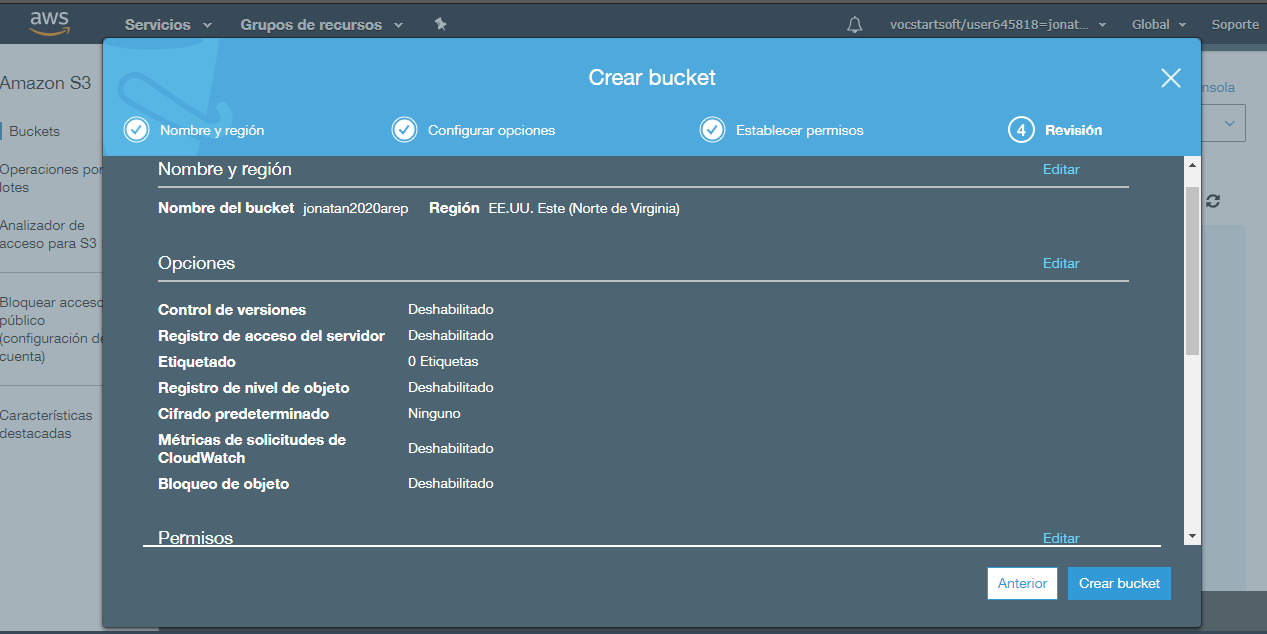


1. Desactivamos la opción de ‘Bloquear todo acceso público’ para poder dar acceso público y damos clic en Siguiente.

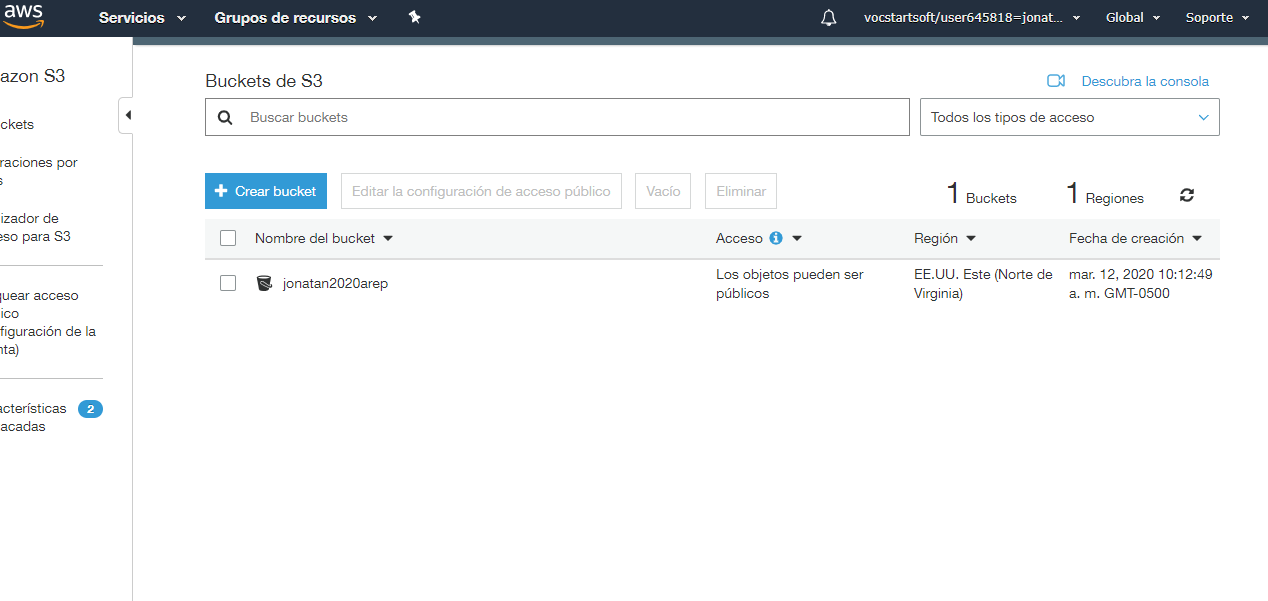
**Nota:** Al desactivar la opción de Bloquear todo acceso público se despliega una ventana y damos click en donde dice Reconozco que la configuración actual puede hacer que este bucket y los objetos que contiene se vuelvan públicos.



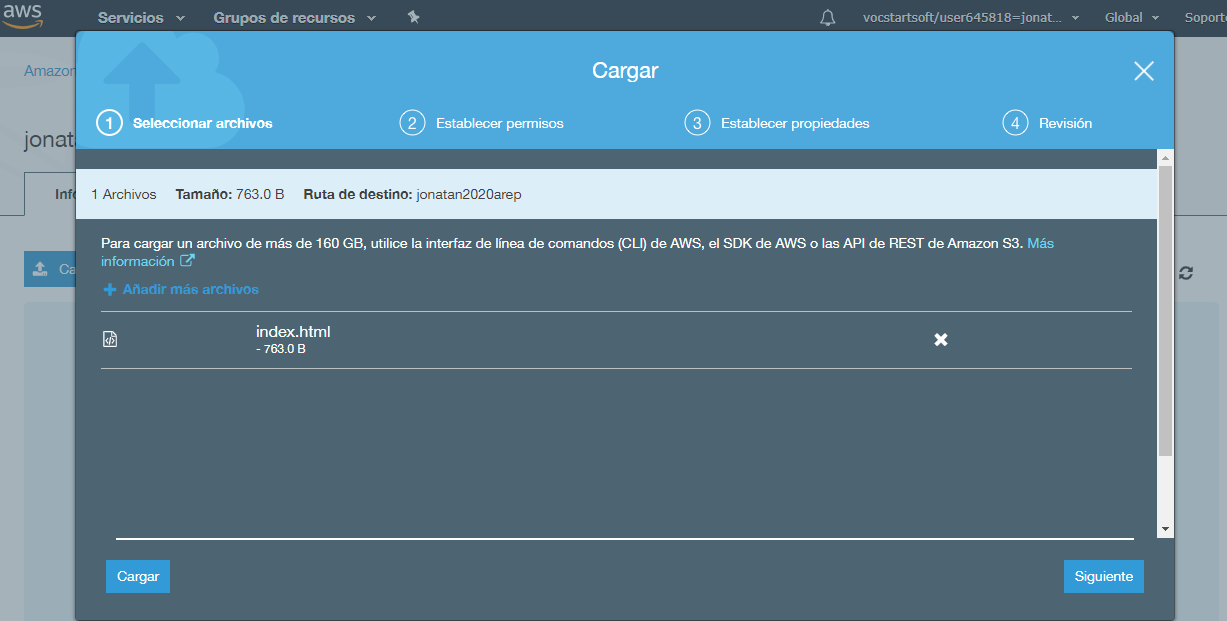
1. Revisamos que todo este configurado como nosotros queremos y damos click en **‘Crear bucket’**



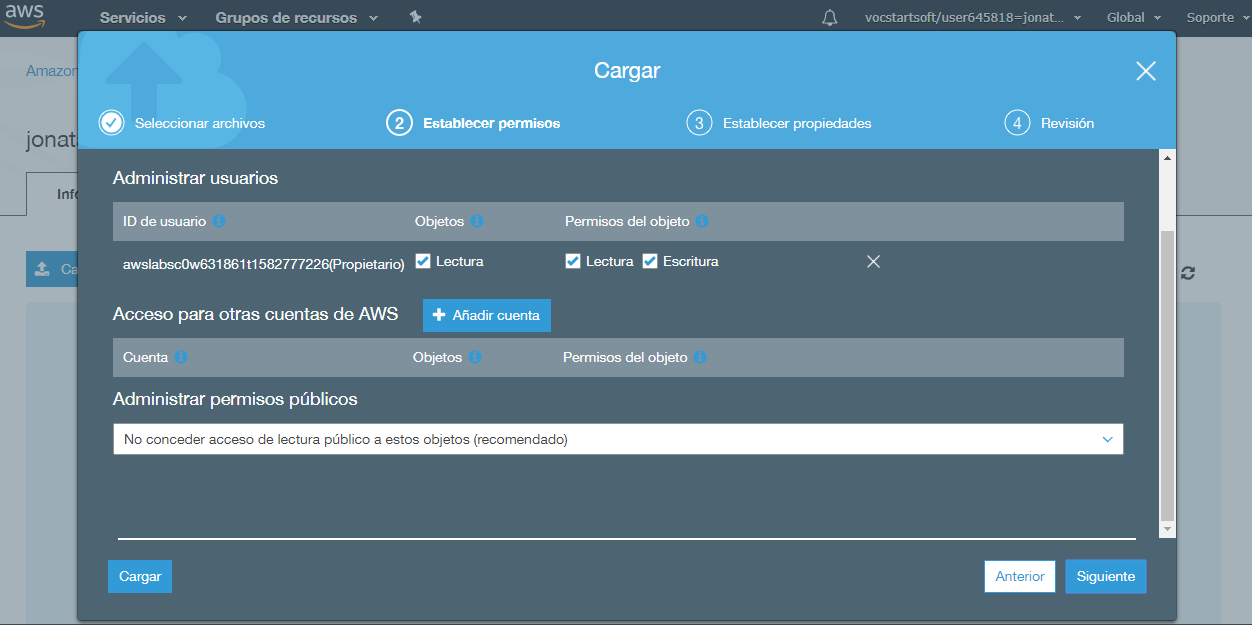
1. Una vez creado el bucket nos debe aparecer en la zona de buckets de S3



1. Entramos al bucket creado dando clic encima de él y damos clic en **‘Cargar’** para subir nuestro archivo estático, en este caso yo subiré un archivo index.html de un laboratorio pasado, después damos click en Siguiente cuando tengamos el archivo seleccionado.



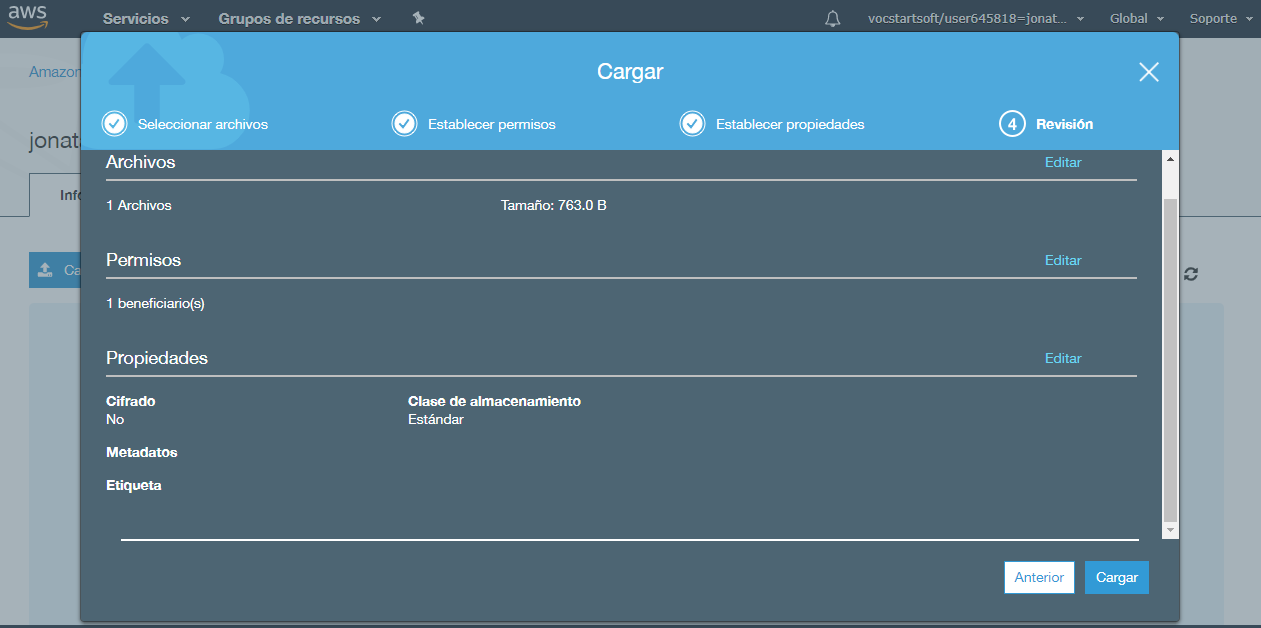
1. En la pestaña de cargar no debemos cambiar nada, solo damos clic en Siguiente.



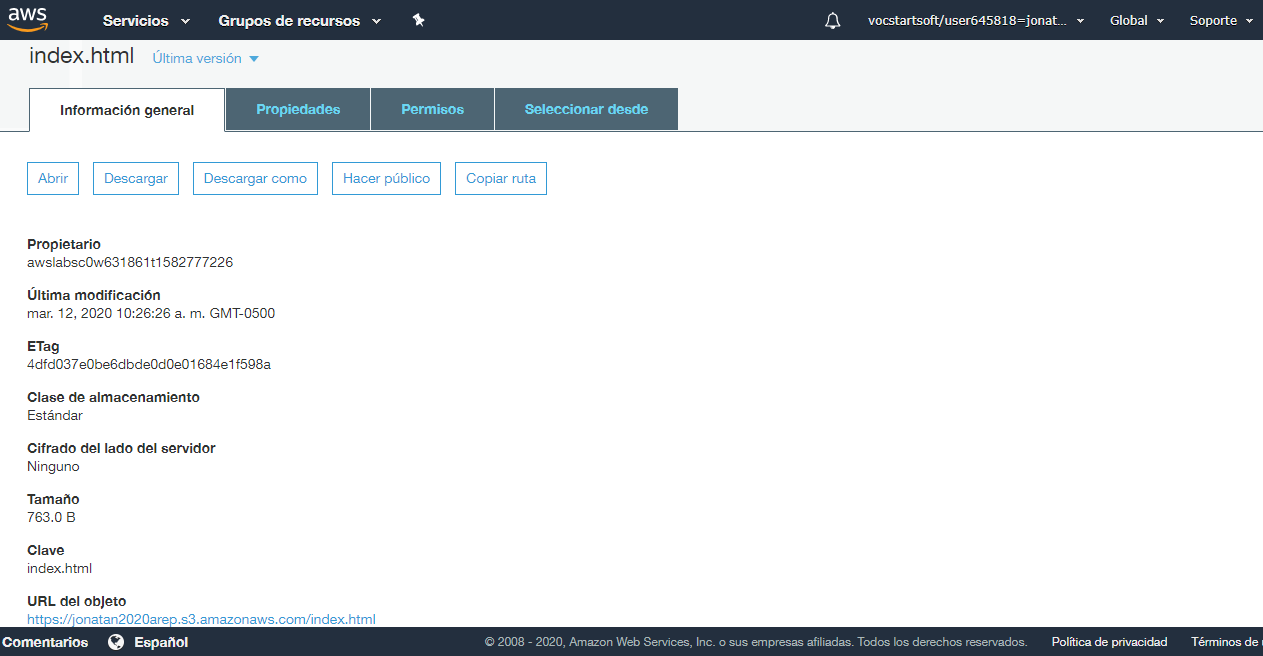
1. En la siguiente ventana de Cargar seleccionamos la opción **‘Estándar’** y damos click en siguiente.



1. Revisamos que las configuraciones estén como queremos y después damos click en Cargar.



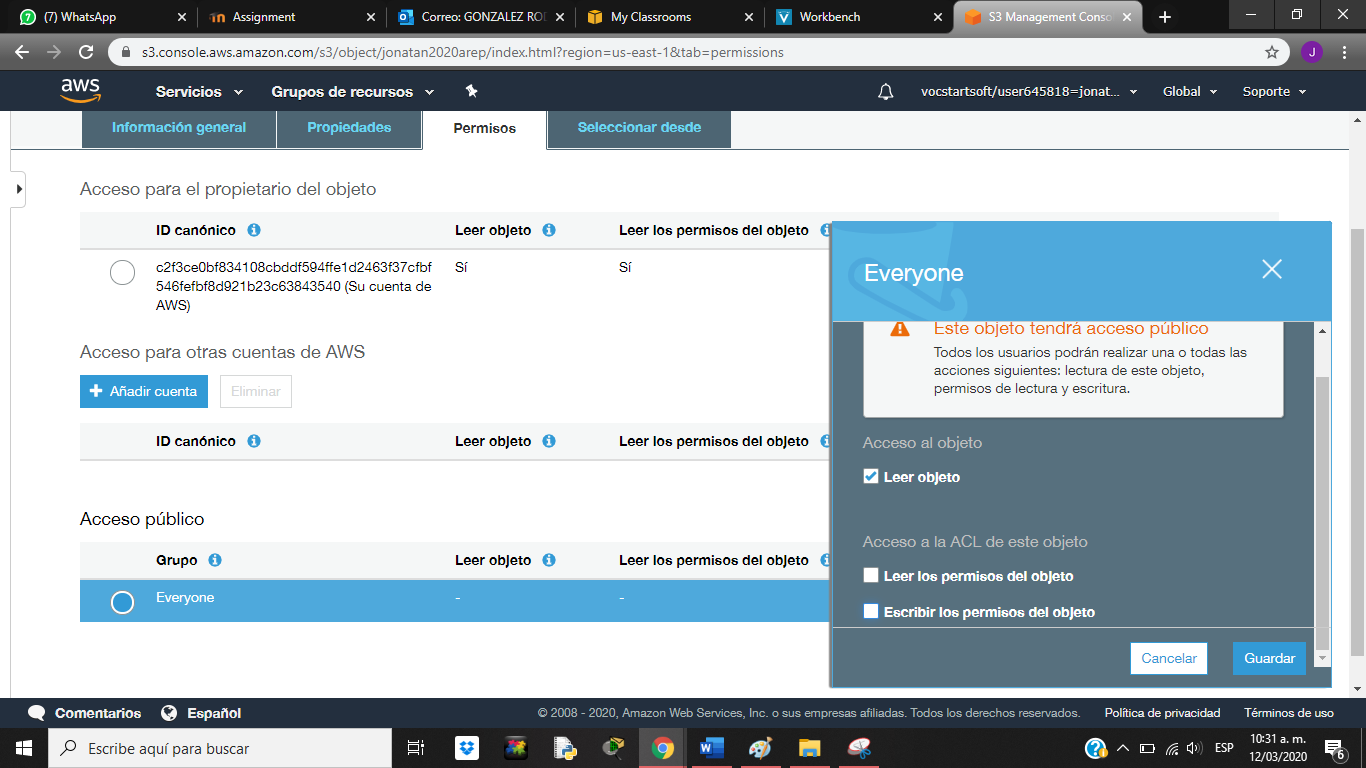
1. Una vez subido el archivo, debe mostrarlo en la información general de mi bucket creado anteriormente, damos clic sobre él para poder verlo.



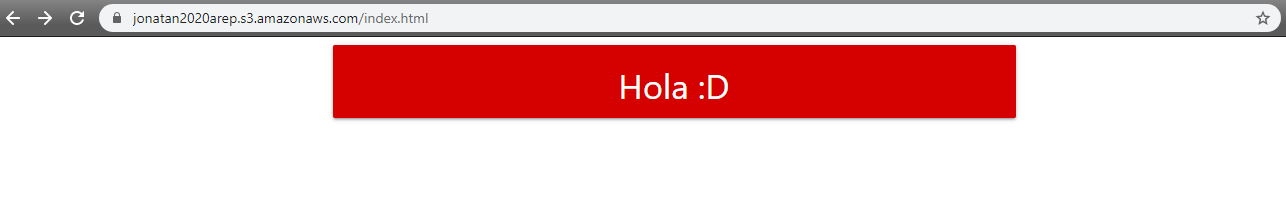
1. Al seleccionar la URL nos redirige a la siguiente página.



1. Para corregir lo anterior vamos a la pestaña de **‘Permisos’**, y en la sección de Acceso Público seleccionamos la opción de **‘Everyone’,** después se nos despliega un recuadro al lado derecho de la pantalla y damos click en Leer objeto y damos click en guardar.



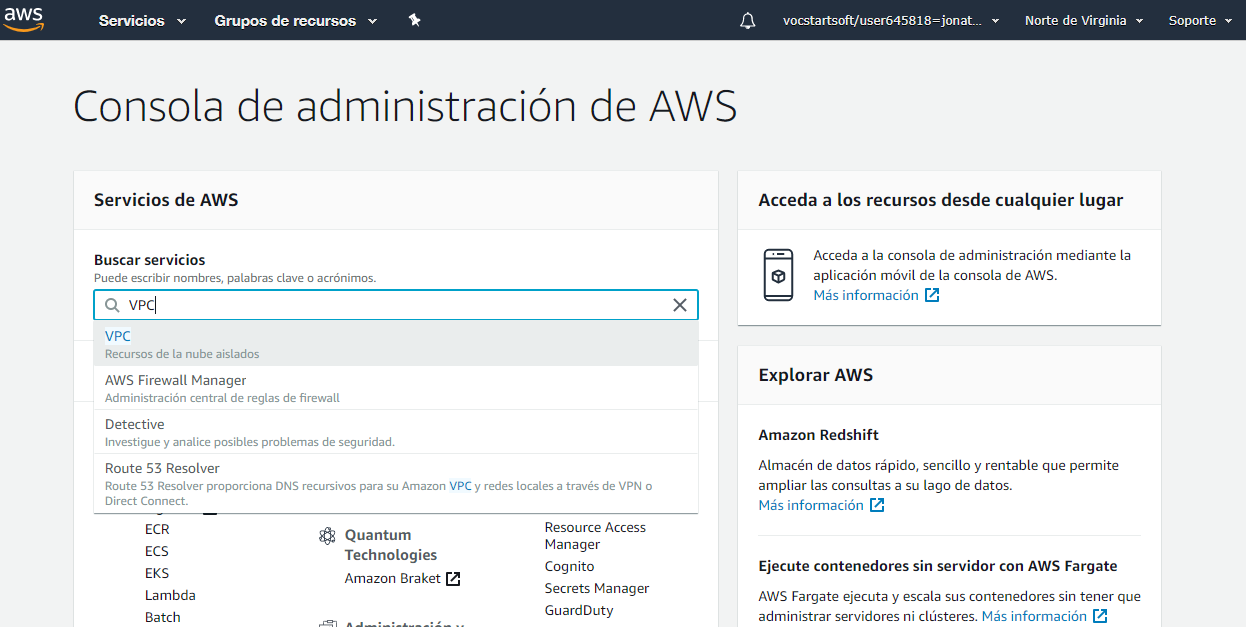
1. Volvemos y presionamos el link y nos sale la información correcta, en este caso sale mi index.htm con un botón que redirecciona a una imagen estática.

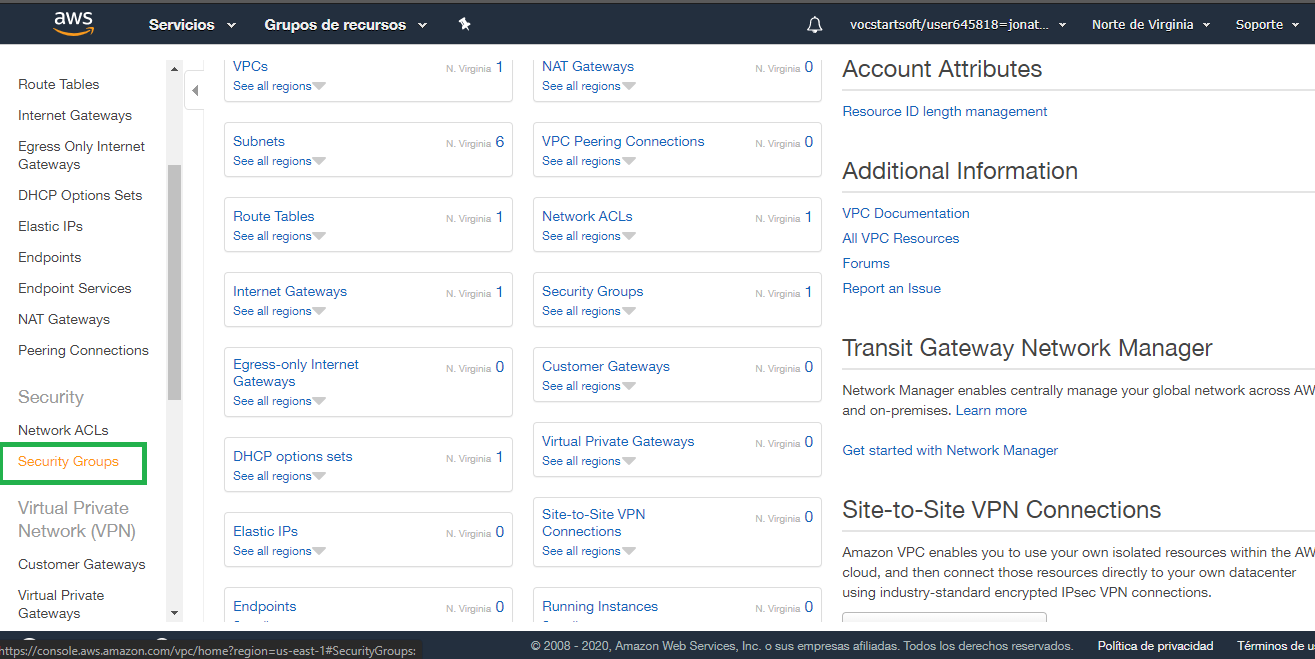


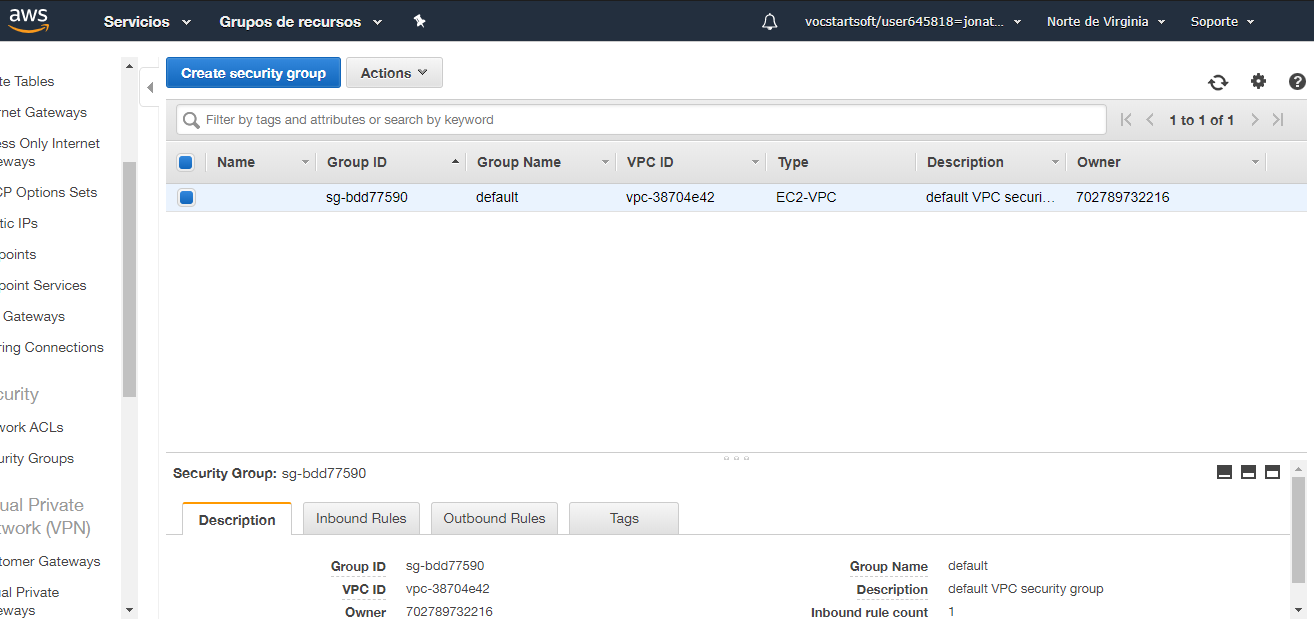
**Desplegar un formulario dinámico usando EC2 enlazado con una**

**base de datos y configurado con un VPC (Cloud Privado virtual)**

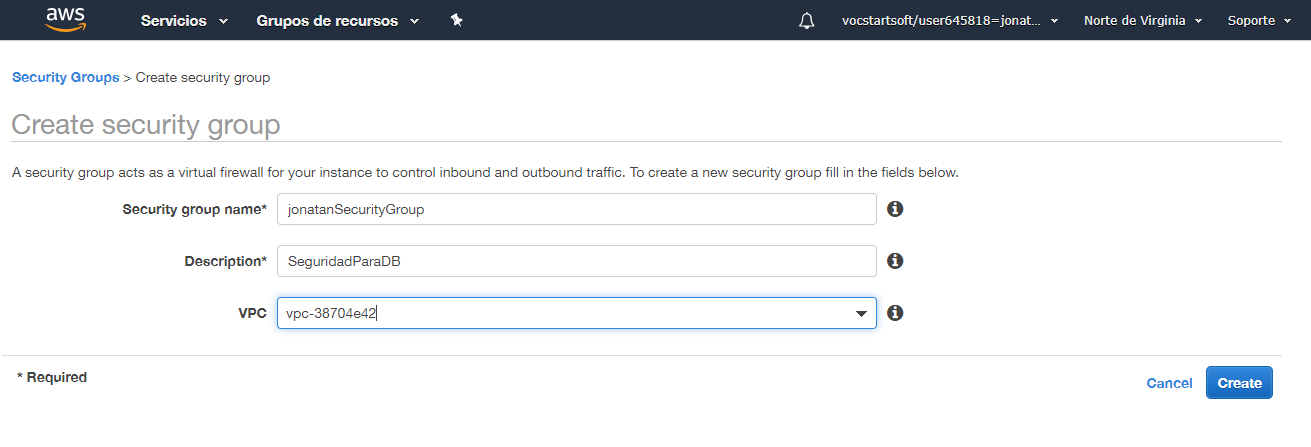
1. Una vez dentro de la consola, buscamos VPC, el cual nos ayudará a abrir los puertos para poder conectarnos a la base de datos.



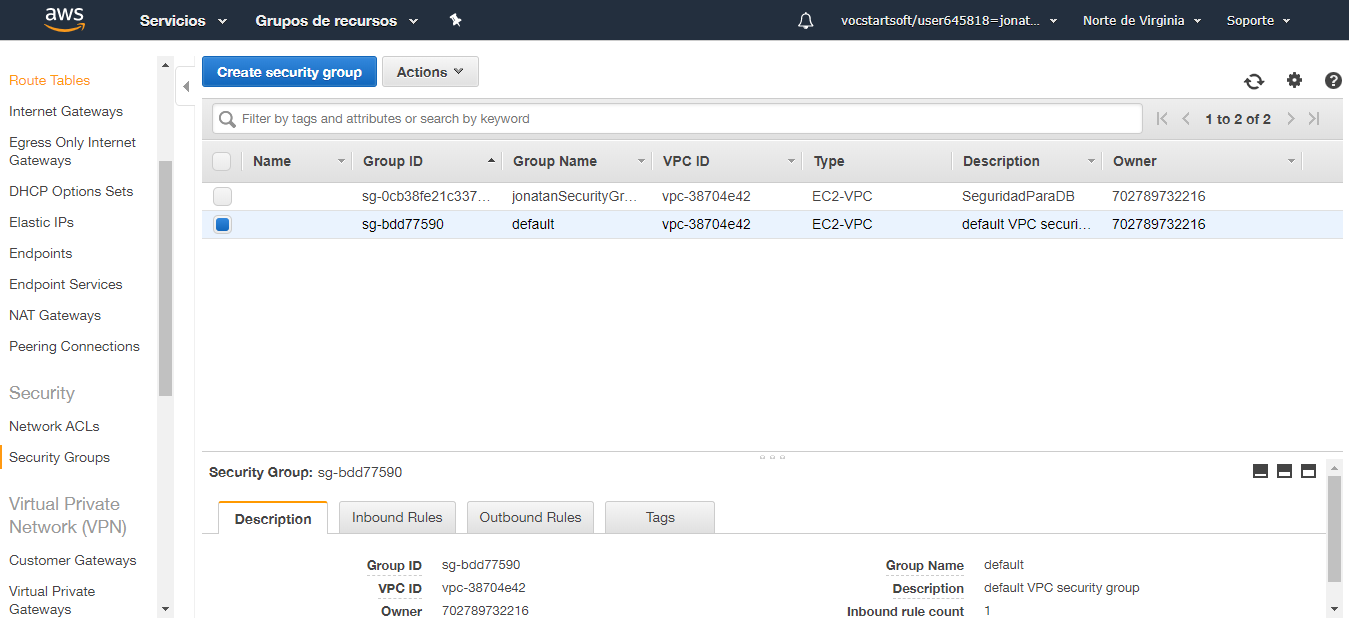
1. Nos dirigimos en la parte izquierda a la sección de **Security** y damos clic en **Security Groups**, como se muestra en la imagen.****
2. Creamos un grupo de seguridad, para esto damos clic en **Create security group** en la parte superior



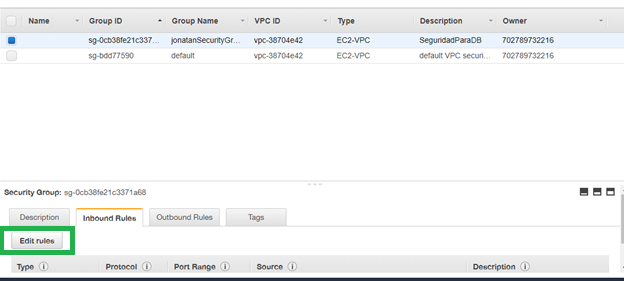
1. Configuramos los valores que piden, que es un nombre para el grupo, una descripción y la VPC, en este caso le asignamos la que aparece por defecto, luego presionamos el botón Create.



1. Una vez creado el grupo de seguridad, se puede observar en la sección de Grupos de Recursos.

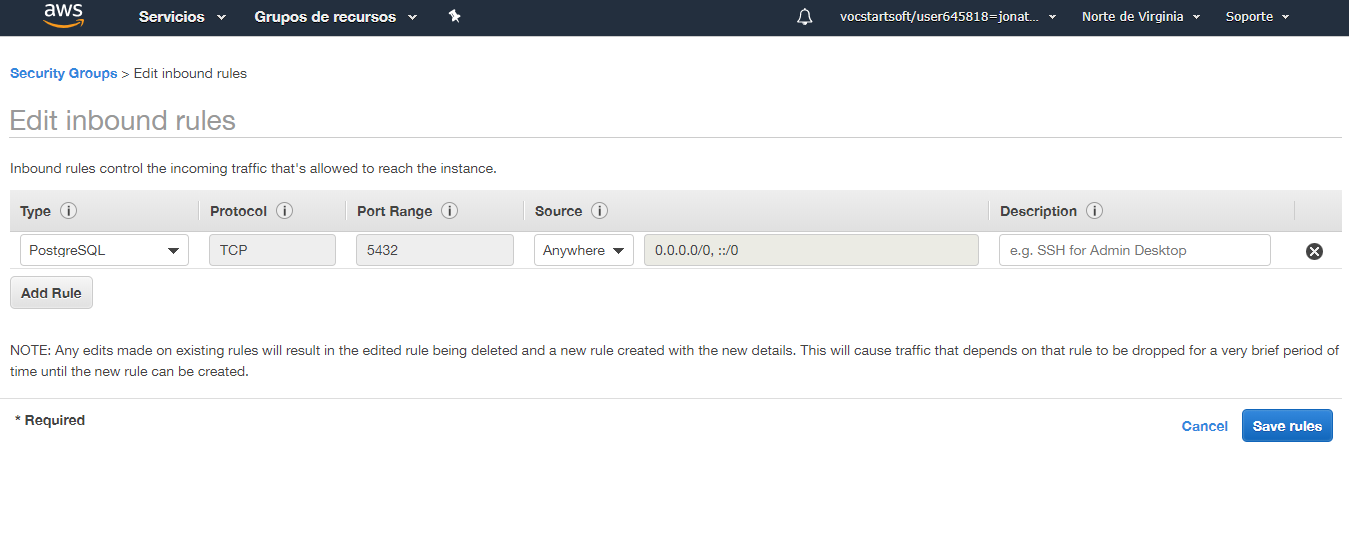


1. Después seleccionamos el grupo de Seguridad que creamos, y nos dirigimos en la parte inferior a la sección de **‘Inbound Rules’** y luego en la parte de **‘Edit Rules’.**

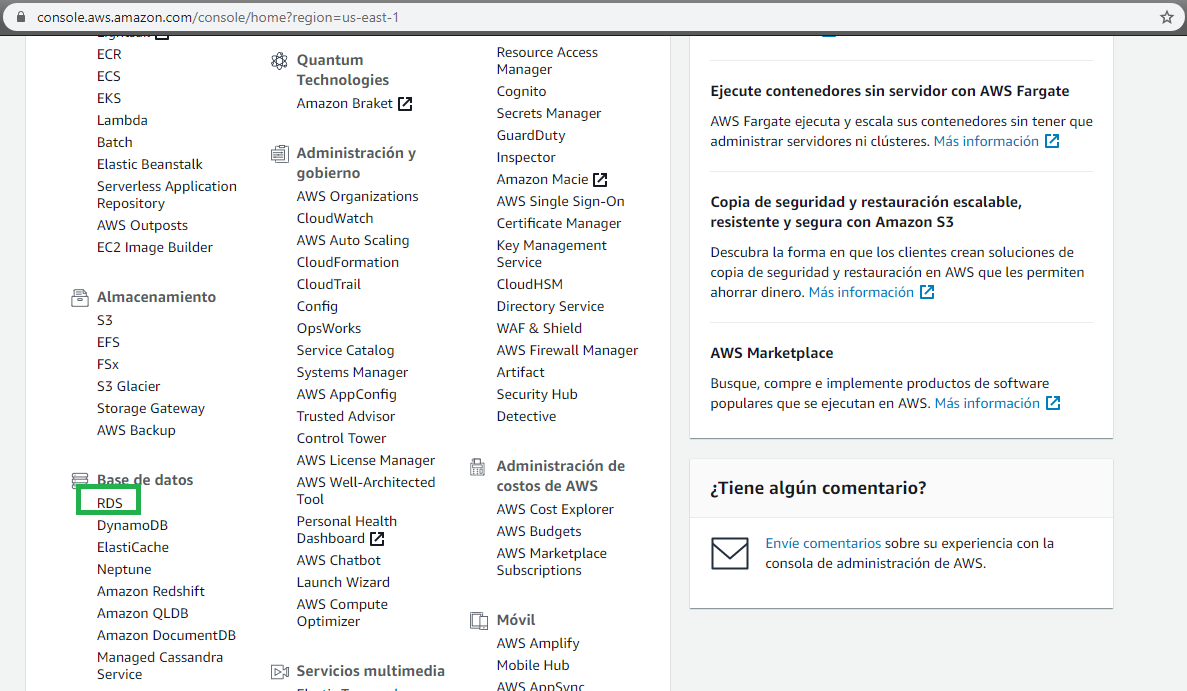


1. Después vamos a el botón de Add Rule y Creamos una regla con Type: PostgreSQL, Protocol: TCP, Port Range: 5432, **Source: Anywere** y una descripción si queremos, luego damos click en Save rules.

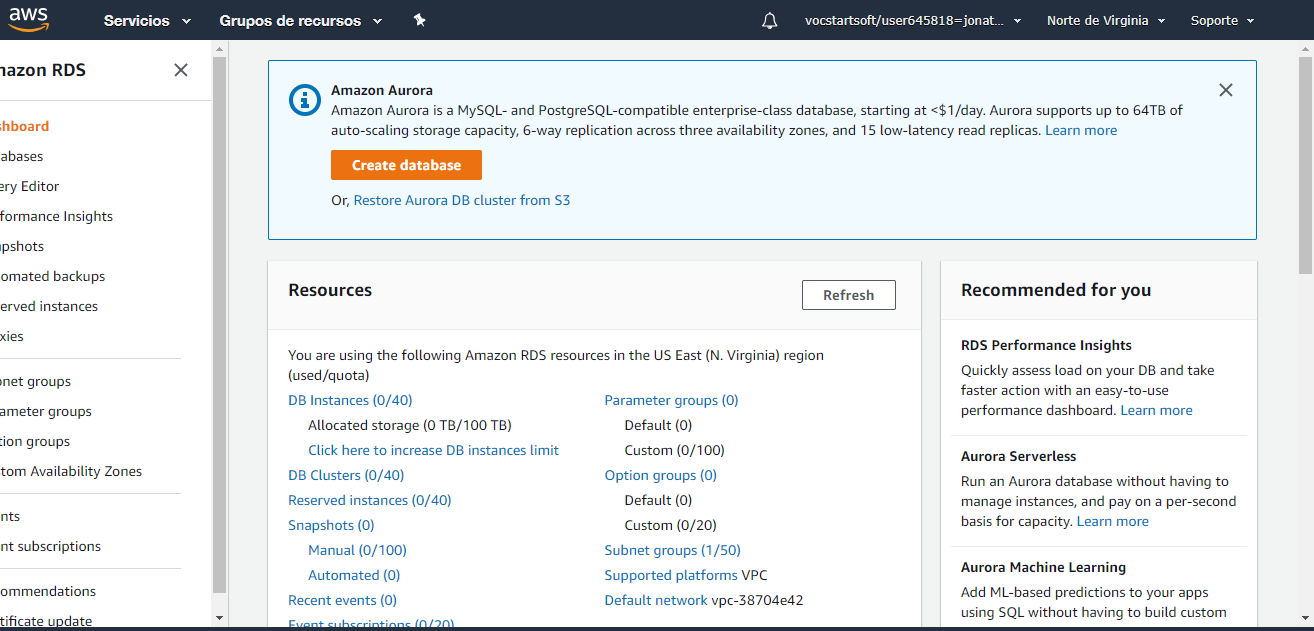
**Nota**: Debemos configurar la sección de Source como Anywere ya que esto es lo que nos deja que cualquiera ingrese a la página.



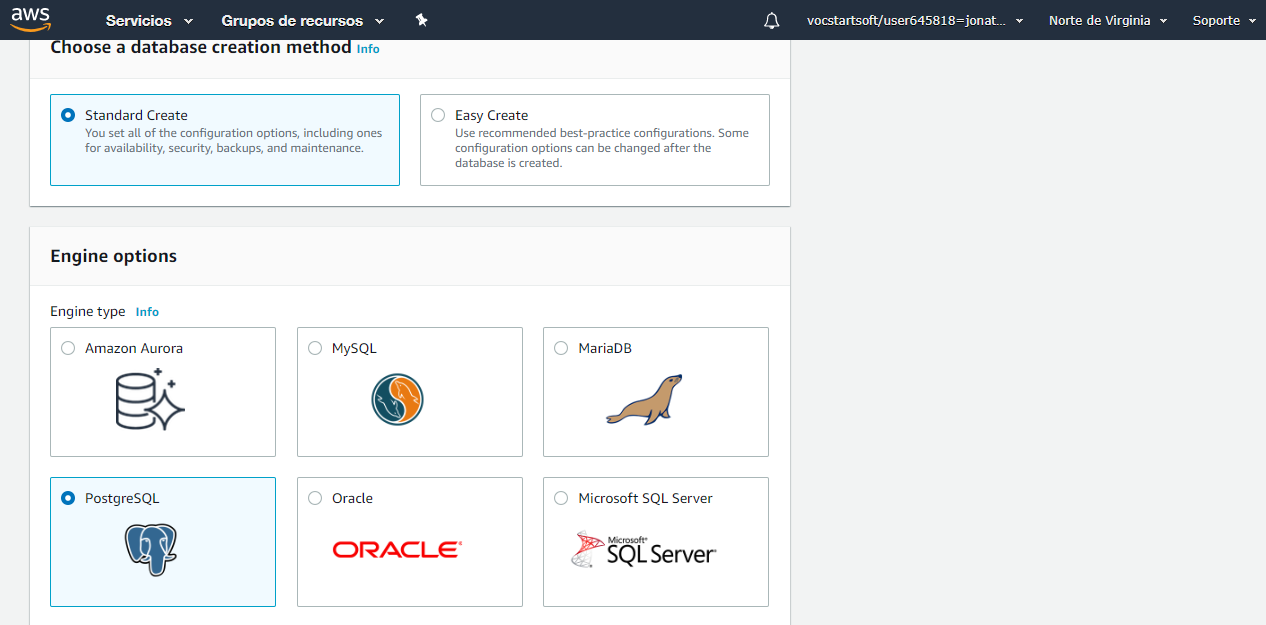
1. Volvemos a la consola de Administración de AWS, nos dirigimos a Todos los servicios y seleccionamos en Base de Datos **RDS**



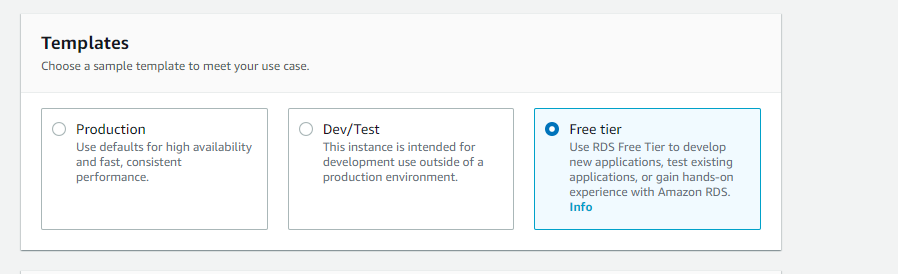
1. Damos click en Create Database



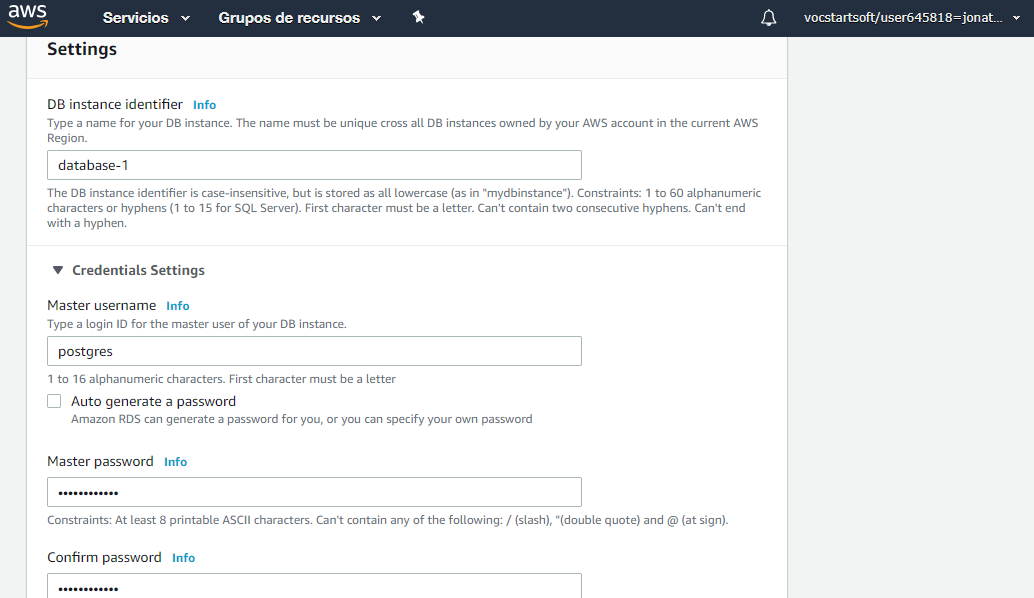
1. Aquí seleccionamos Standard create y PostgreSQL, que es con el motor que queramos trabajar, en mi caso será PostgreSQL.



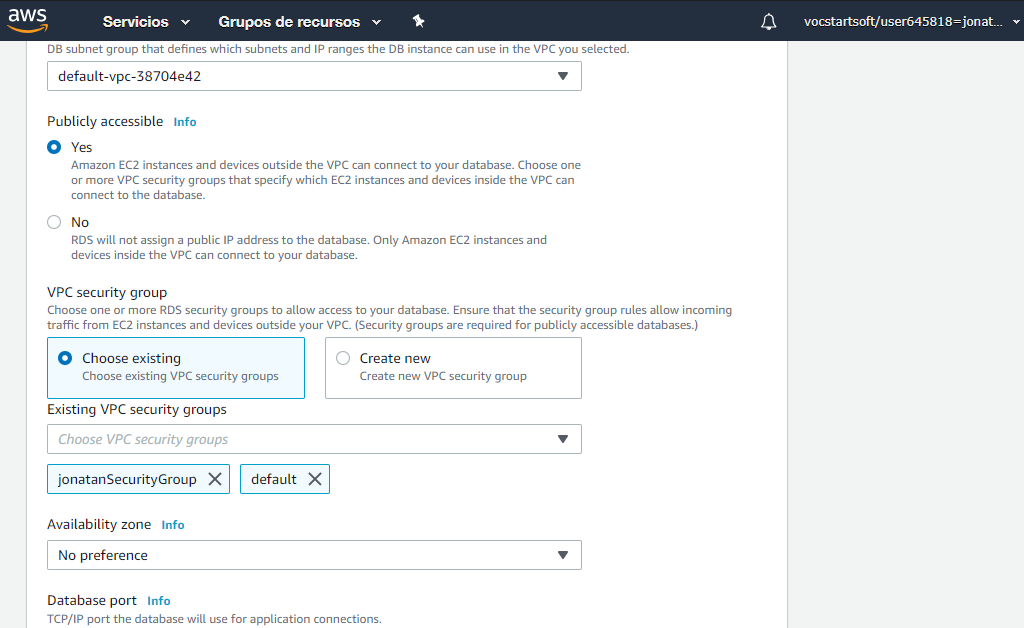
1. En la sección 'Templates' seleccionamos Free tier.

pass

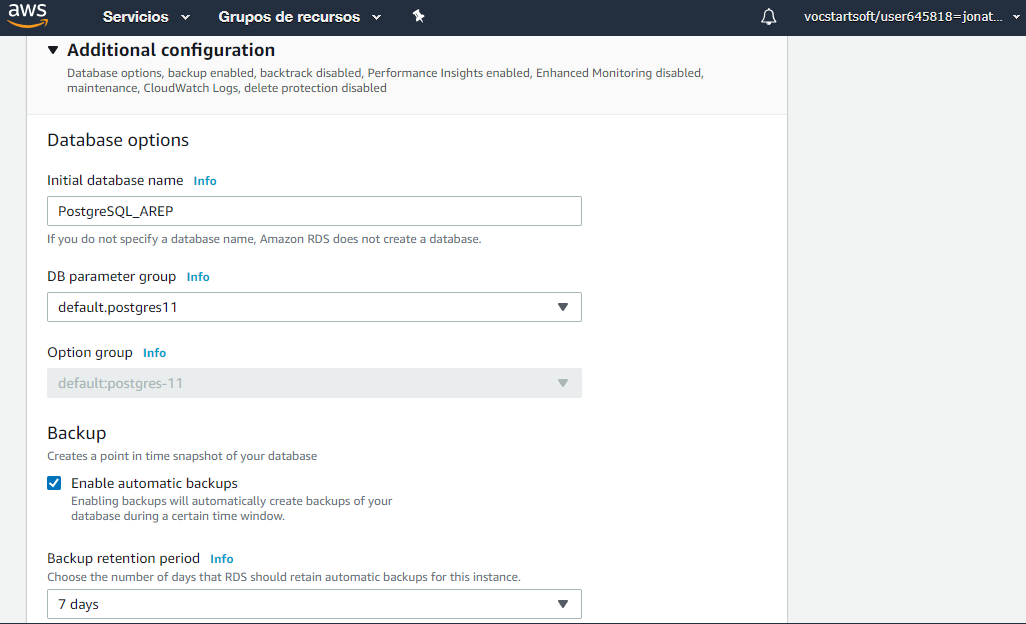
1. Debemos configurarle los nombres y password a la base de datos.



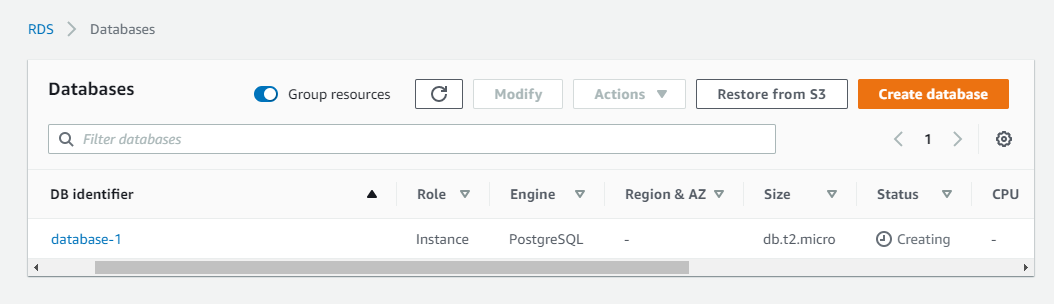
1. En la sección Conectivity, dejaremos que sea de acceso público, así que en 'Publicly accessible' cambiamos a **Yes**. En VPC Security group seleccionamos creada anteriormente.



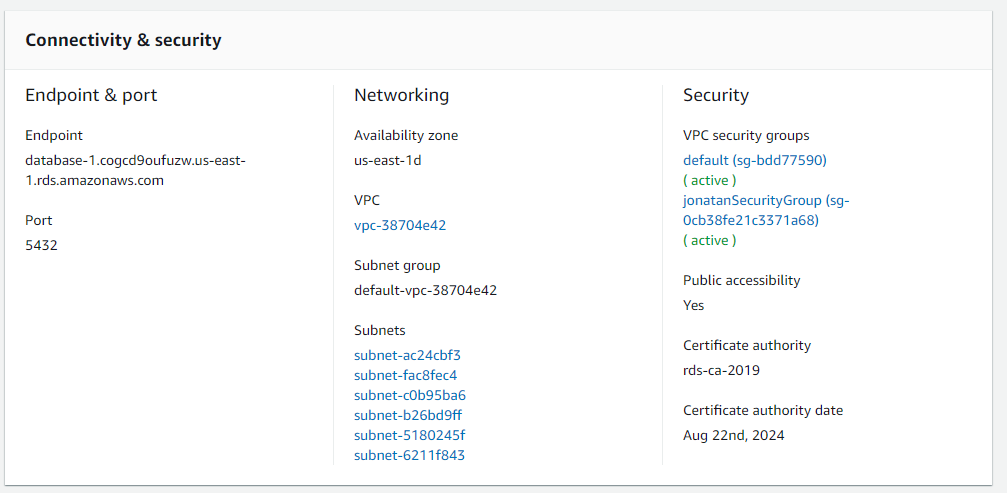
1. Finalmente asignamos un nombre a la base de datos, y damos en la parte final **‘Create database’**



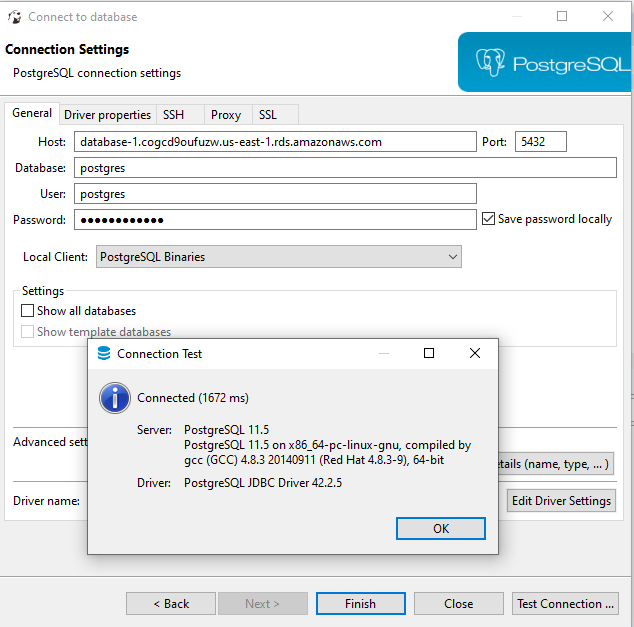
1. En la sección de Database podemos observar la nueva base de datos creada



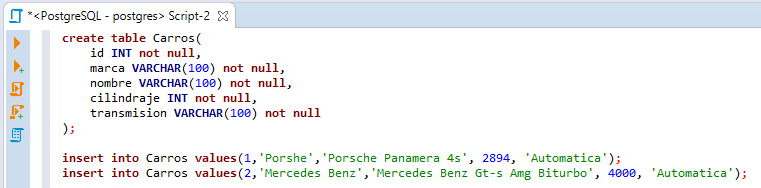
1. Cuando la sección de Status aparezca Available entramos a la base de datos dando click sobre el DB identifier, en esta parte nos aparecerán las credenciales de la base de datos.



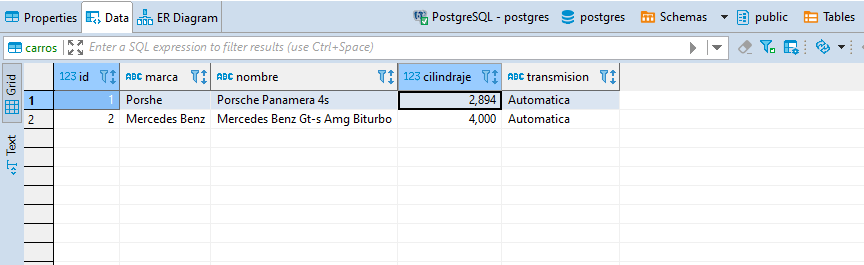
1. Ahora, abrimos DBeaver para conectarnos a la base de datos creada anteriormente, anotamos el host que es el que aparece en el Endpoint & port y el usuario y contraseña que dimos anteriormente, hacemos un test de conexión y observamos que esta conectada correctamente.



1. Una vez conectados a la base de datos, creamos un SQLEditor para crear tablas para utiizarlas en la API.



1. Podemos observar los datos en la base de datos



1. Ahora lo que hacemos es correr nuestra API localmente y debería aparecernos algo así, con los datos que le metimos a la base de datos.