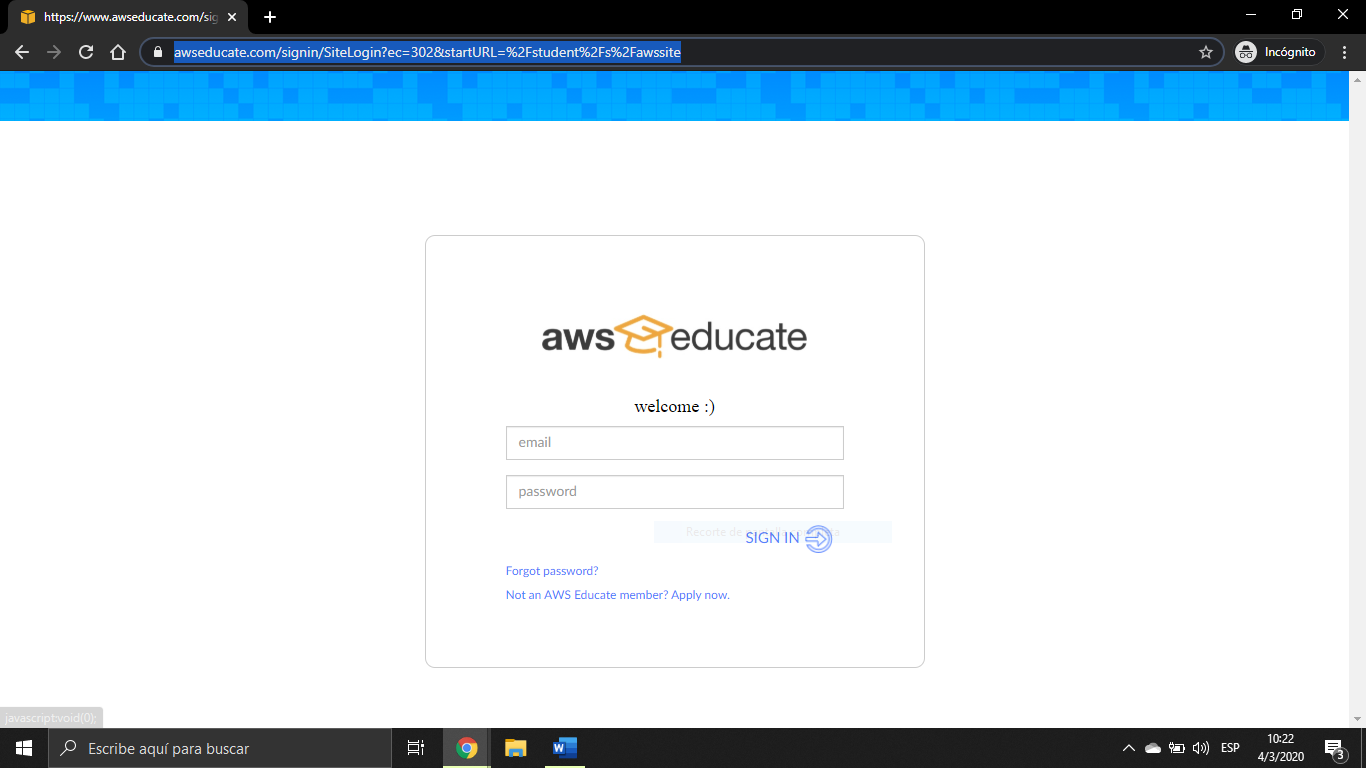
Iniciar sesión en AWS eduacte

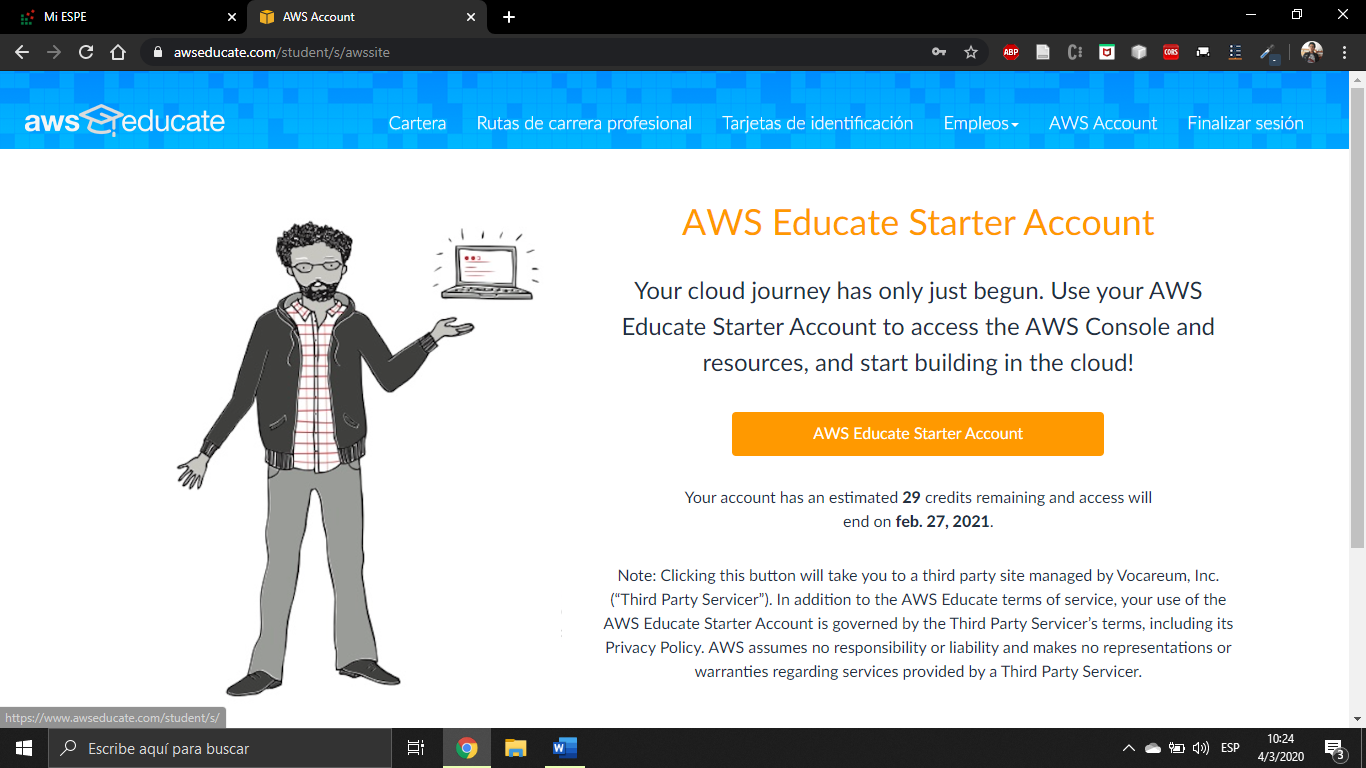
Link: <https://www.awseducate.com/signin/SiteLogin?ec=302&startURL=%2Fstudent%2Fs%2Fawssite>



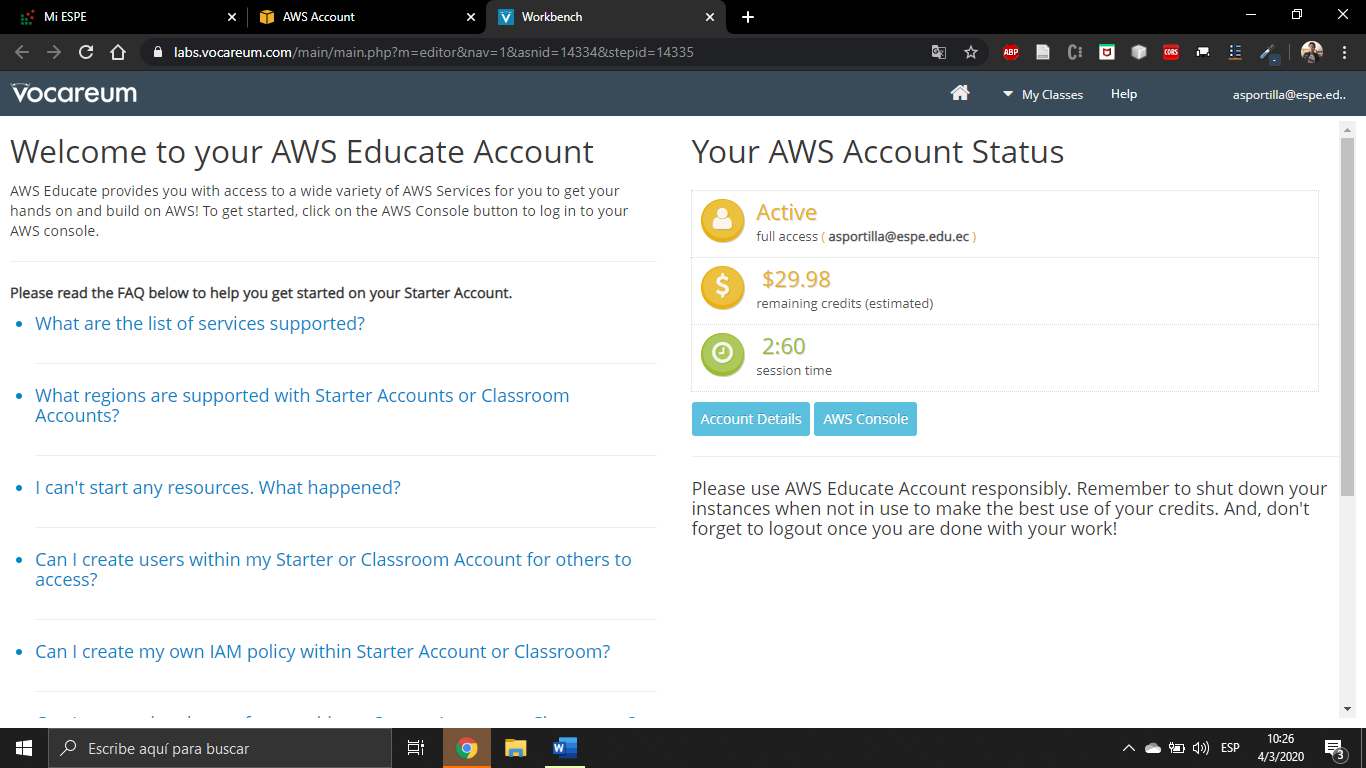
Usuario: asportilla@espe.edu.ec

Contraseña: @portilla0979350864

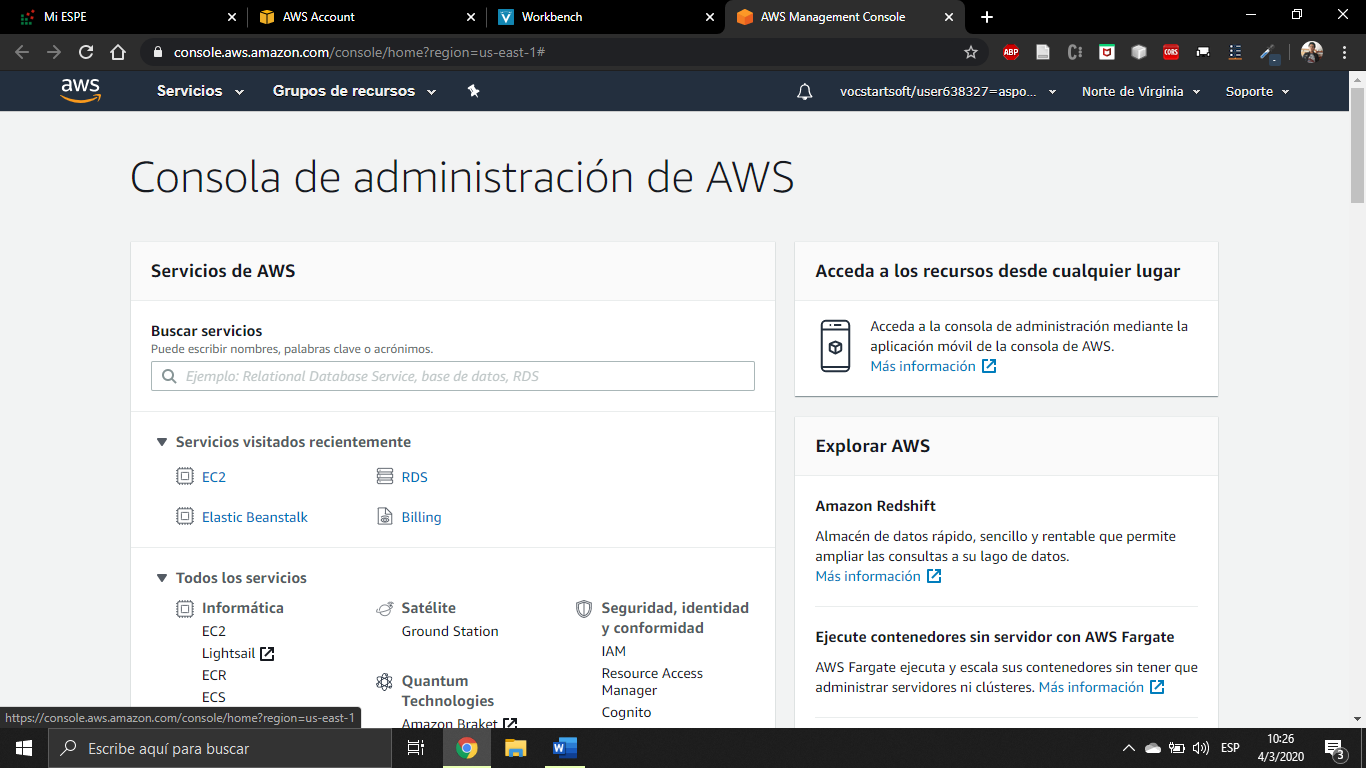
Seleccionar la opción ***AWS Account***



Seleccionar la opción ***AWS Educate Starter Account***

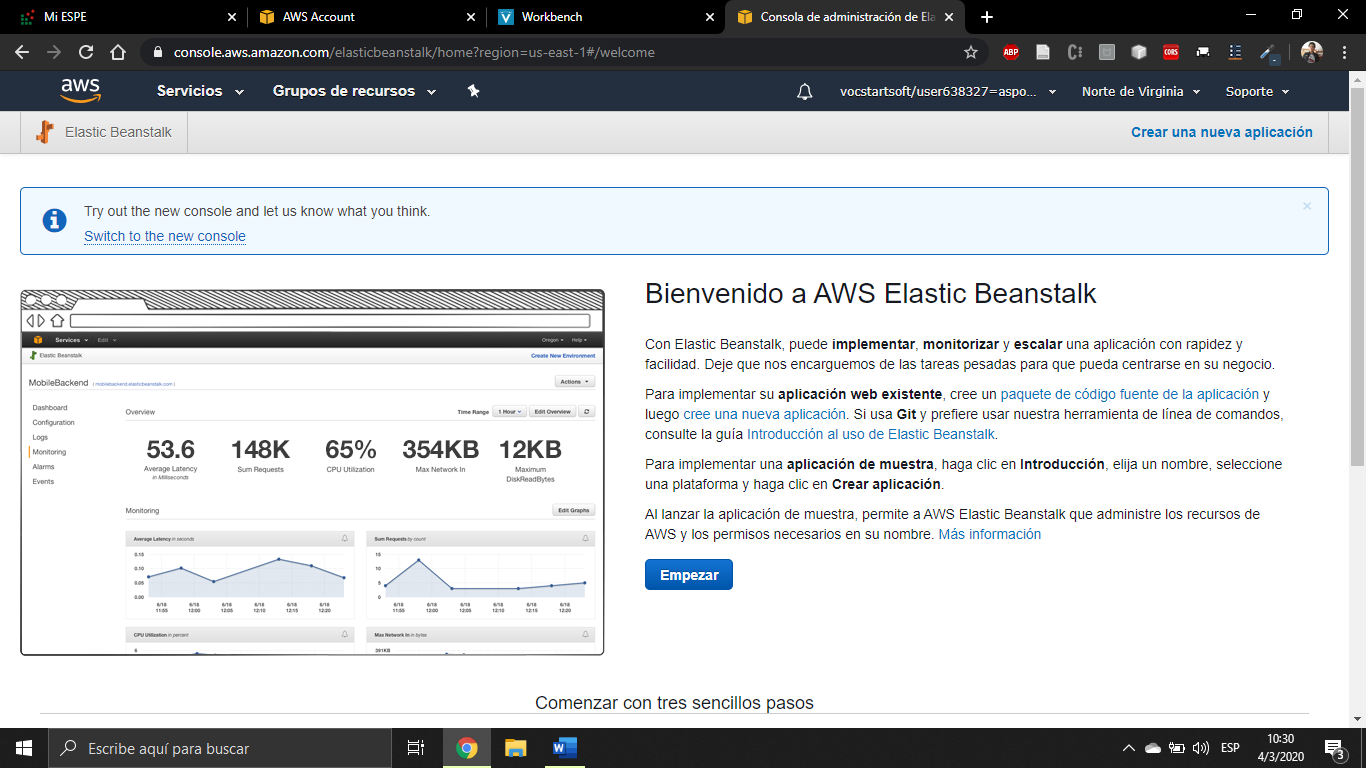


Seleccionar la opción ***AWS Console***

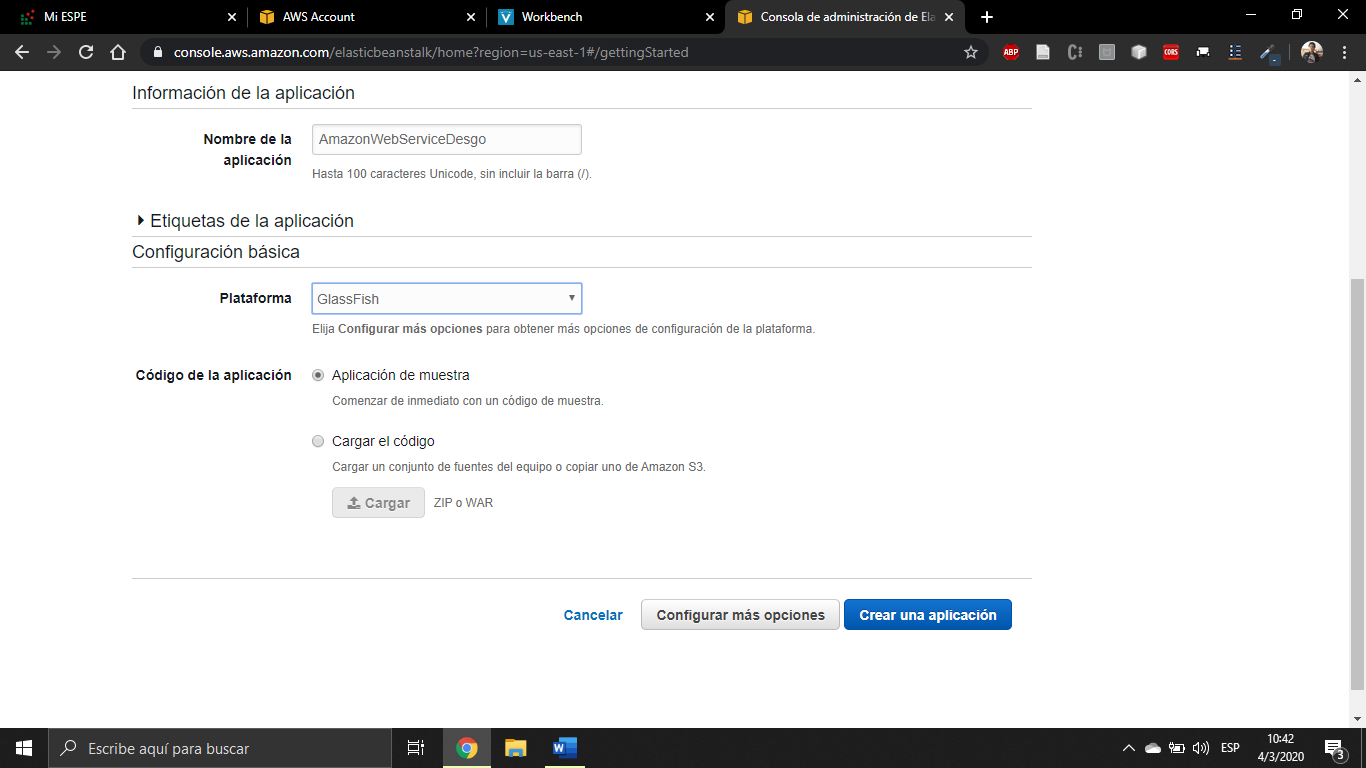


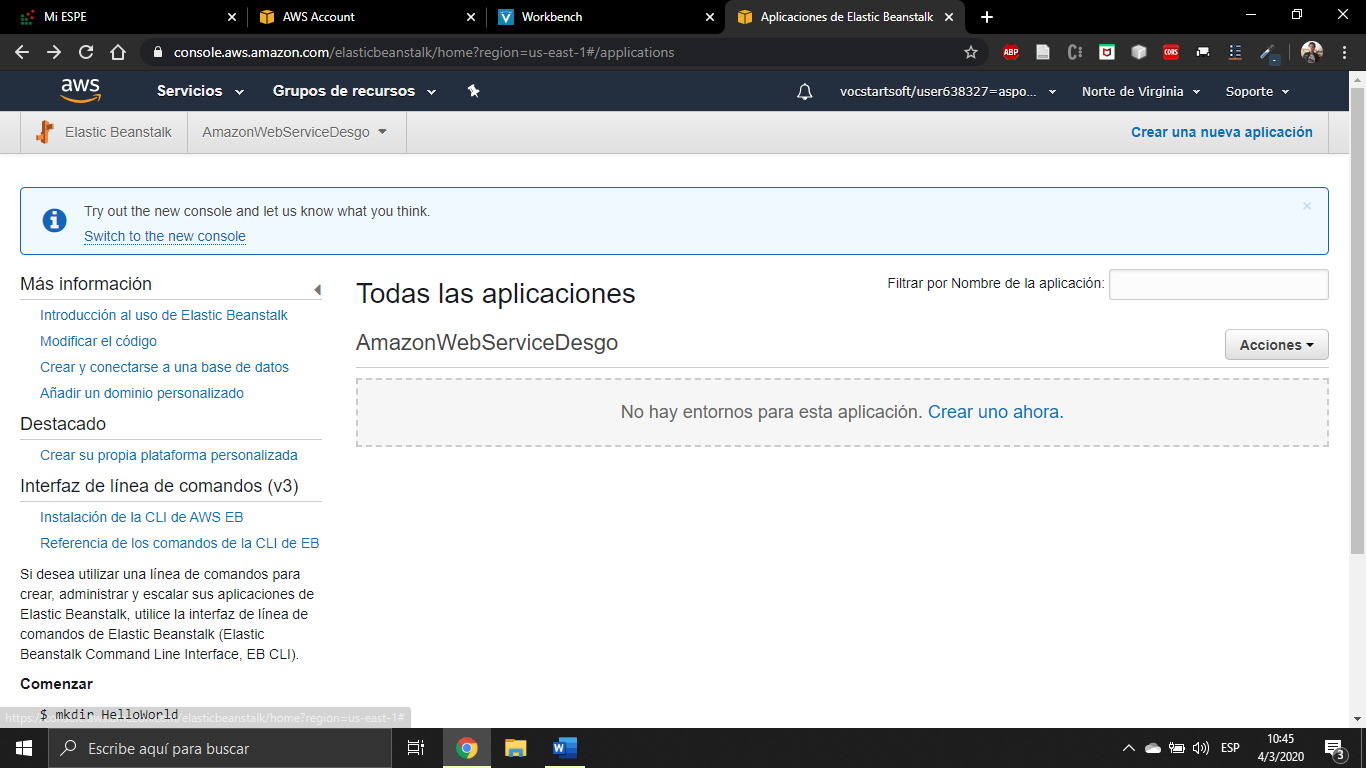
Se procede a crear un servicio ***Elastic Beanstalk (es un PaaS servicio de Amazon Web Services que permite crear aplicaciones y desplegarlas a un conjunto definido de Servicios de AWS)***

Seleccionar la opción ***Empezar***

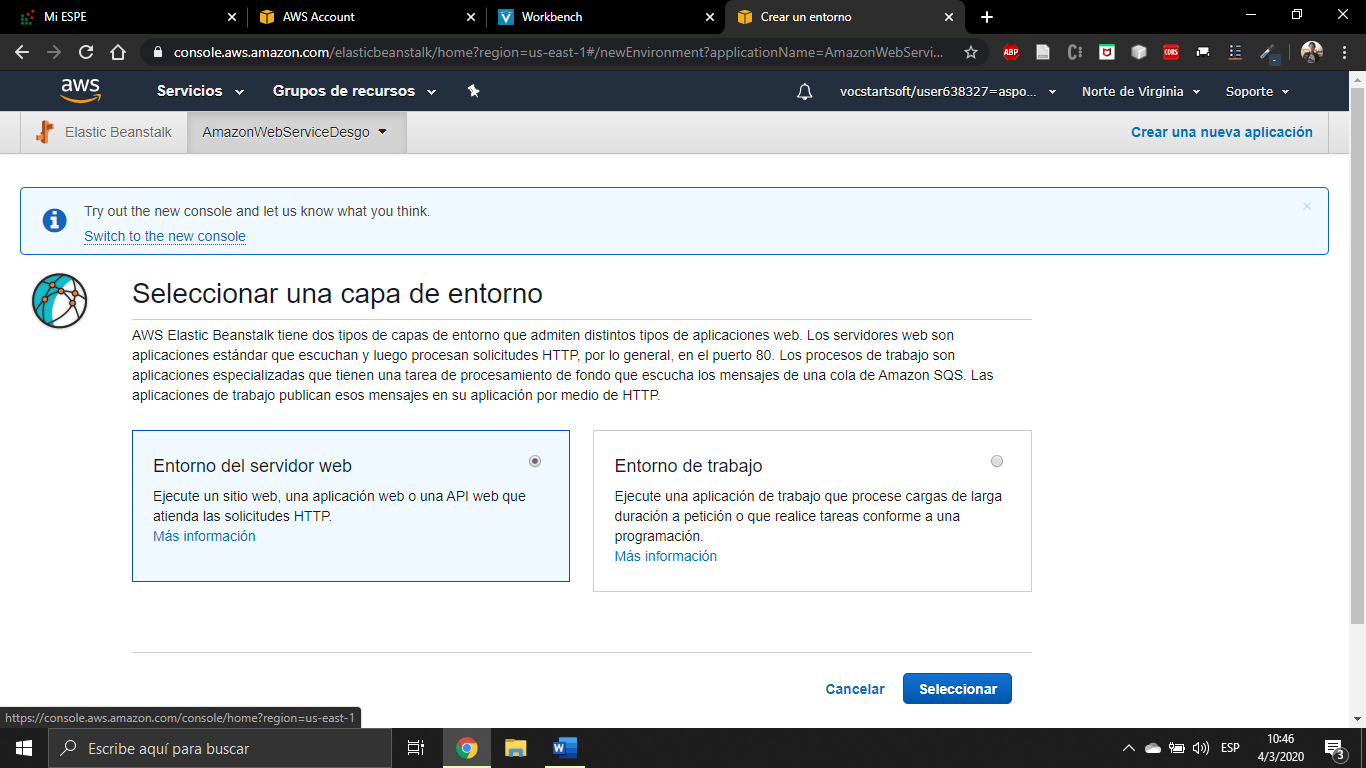


Ingresar los siguientes datos, crear la aplicación.

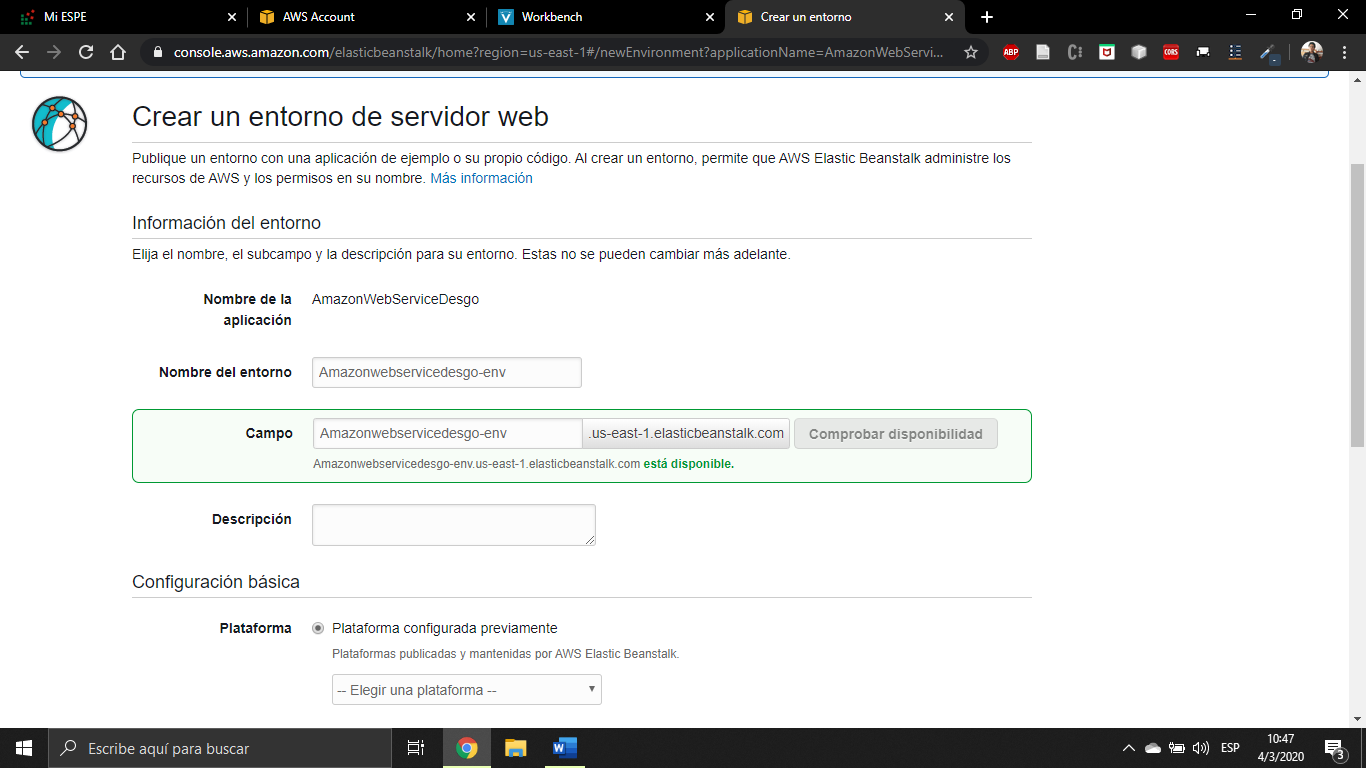




Crear un entorno, seleccionar ***crear uno ahora***

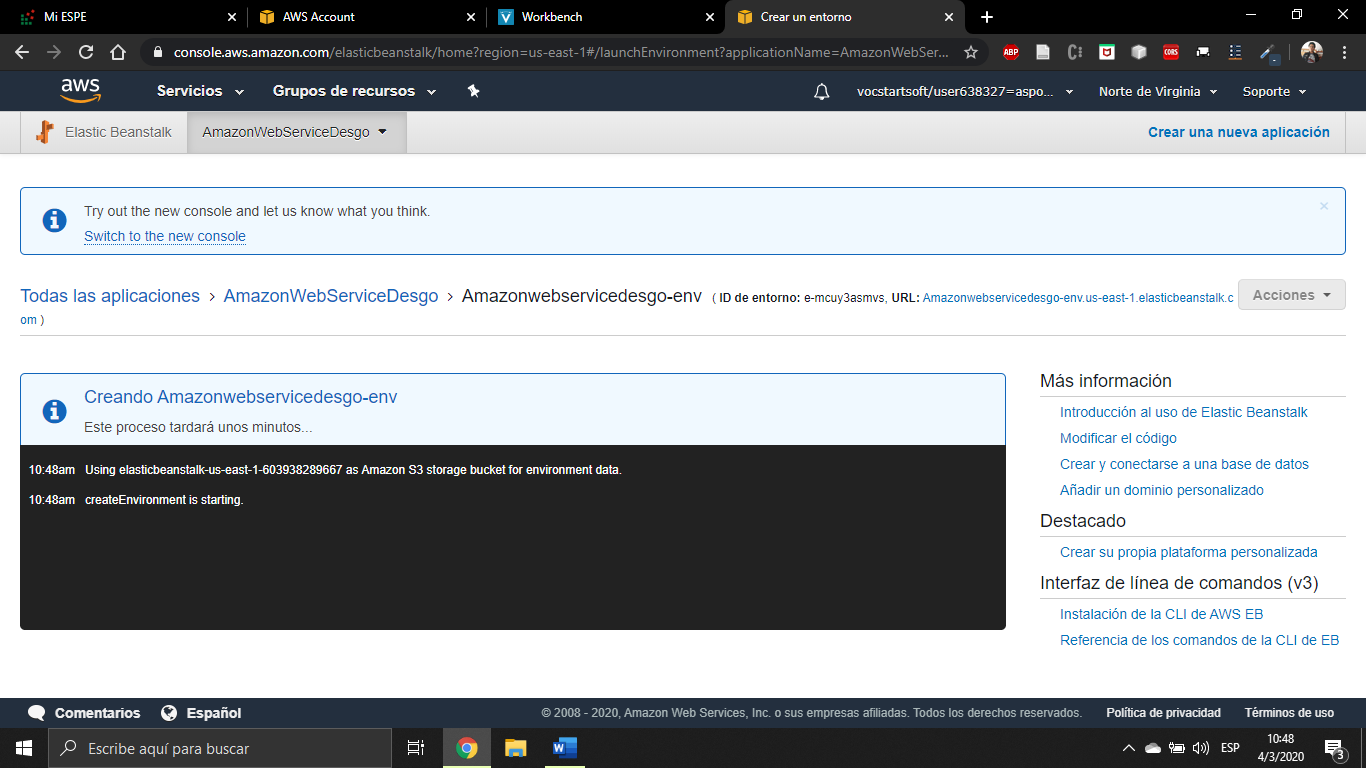


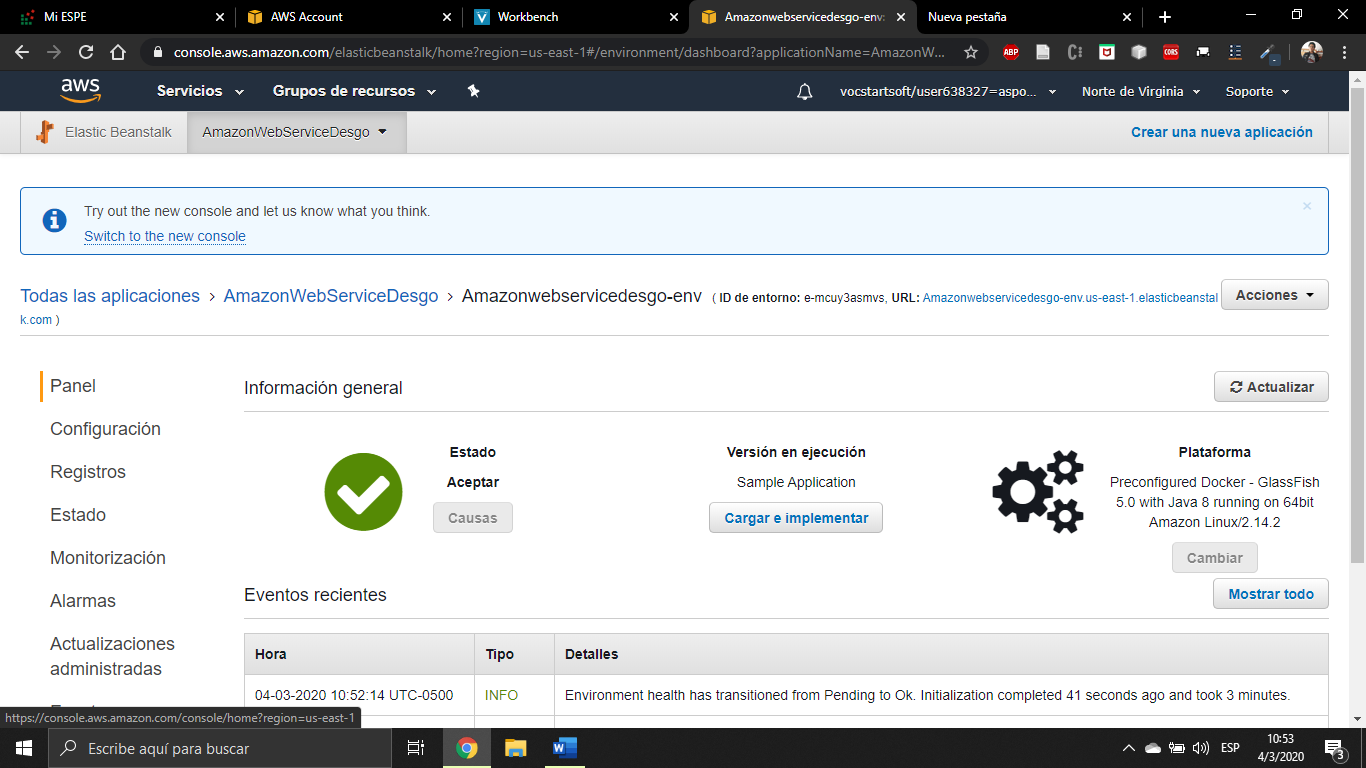
Ingresar los datos como se muestran en las imágenes



Nota: en plataforma escoger el servidor de aplicaciones (tomcat, glassfish) el cual fue desarrollado la app java web.

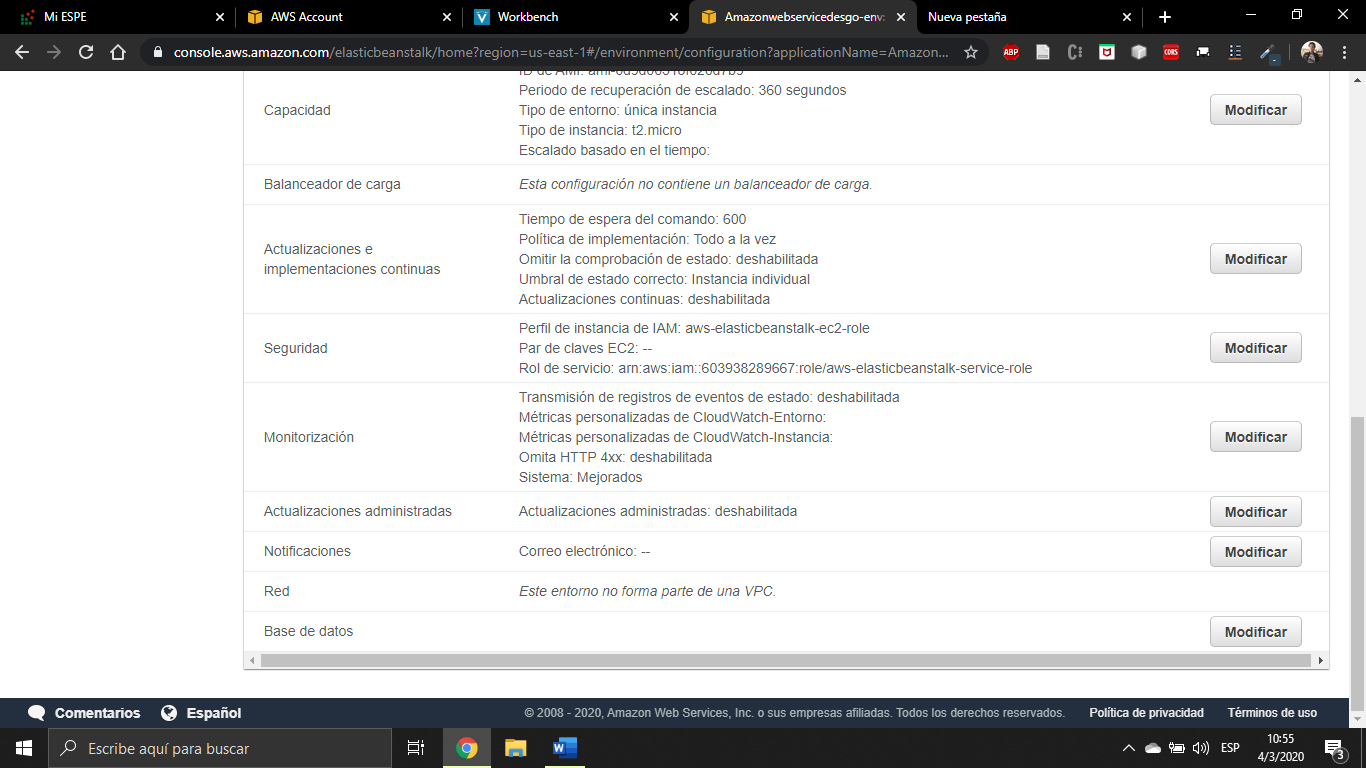
Esperar hasta que se cree el entorno:





Configurar la base de datos, seleccionar la opción ***Configuración*** (menú lateral izquierdo)

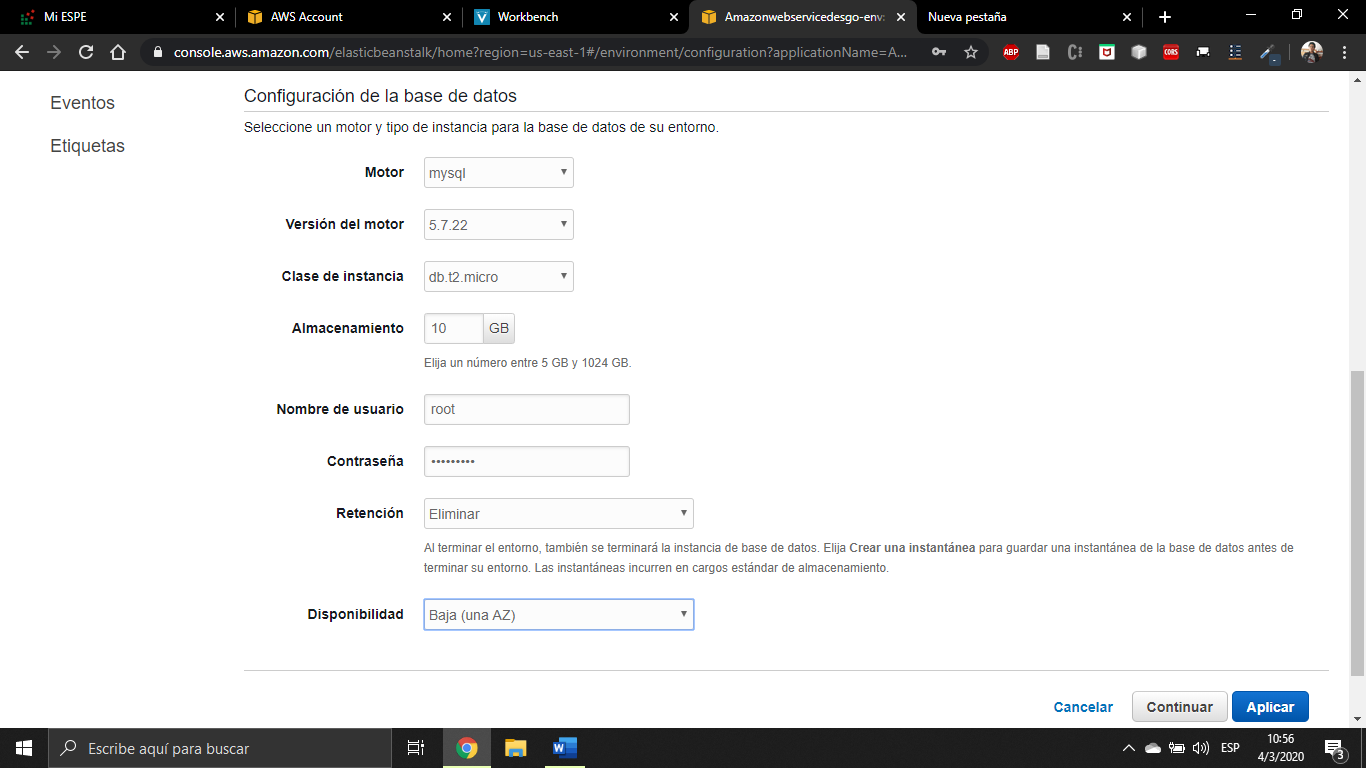
Seleccionar base de datos



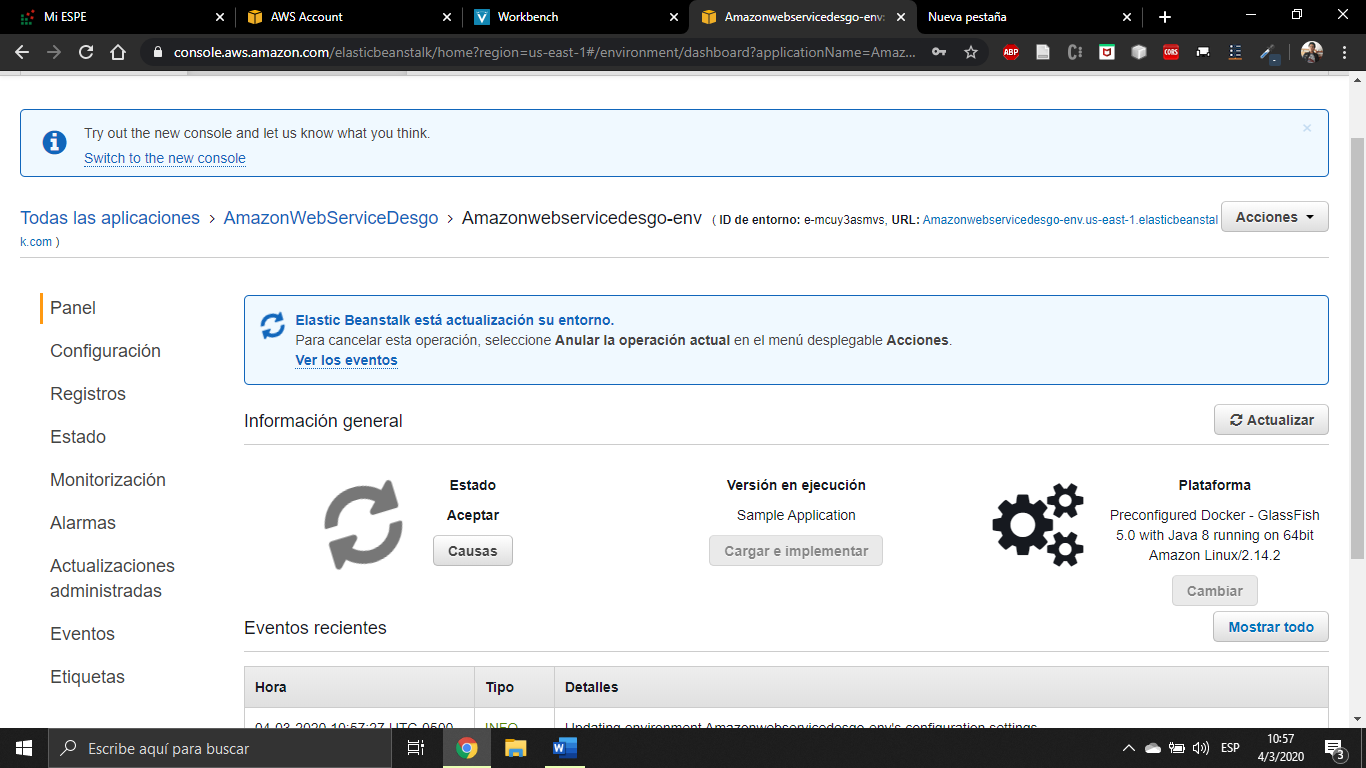
Ingresar las siguientes configuraciones

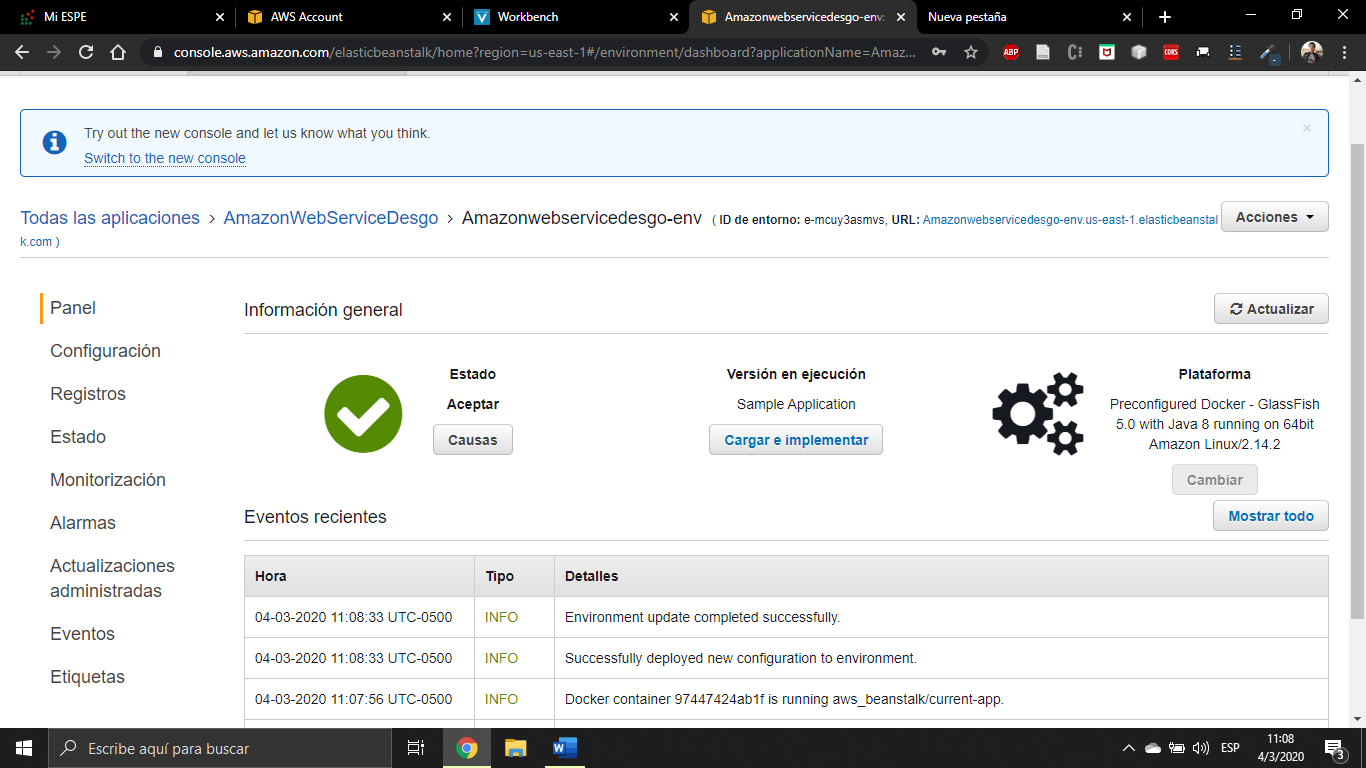
Nombre usuario: root

Contraseña: rootdesgo



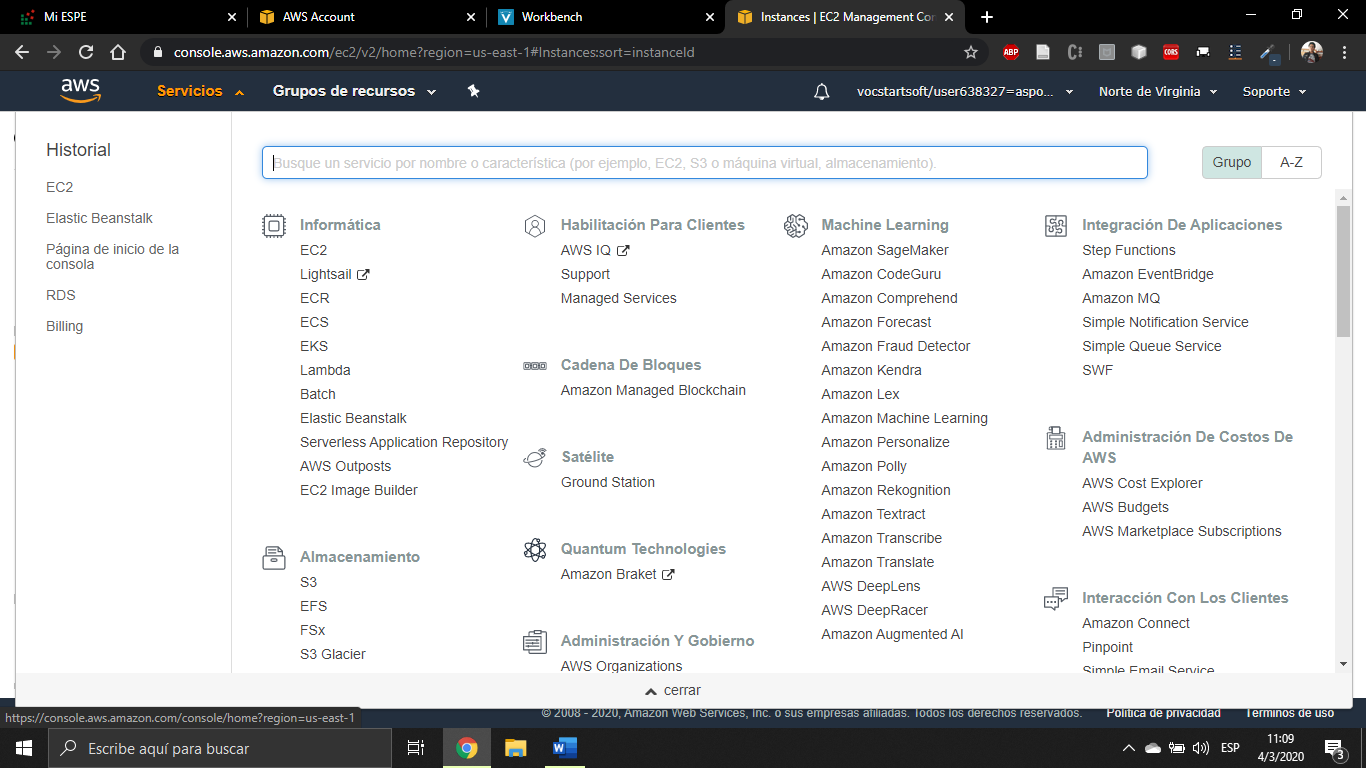
Aplicar cambios



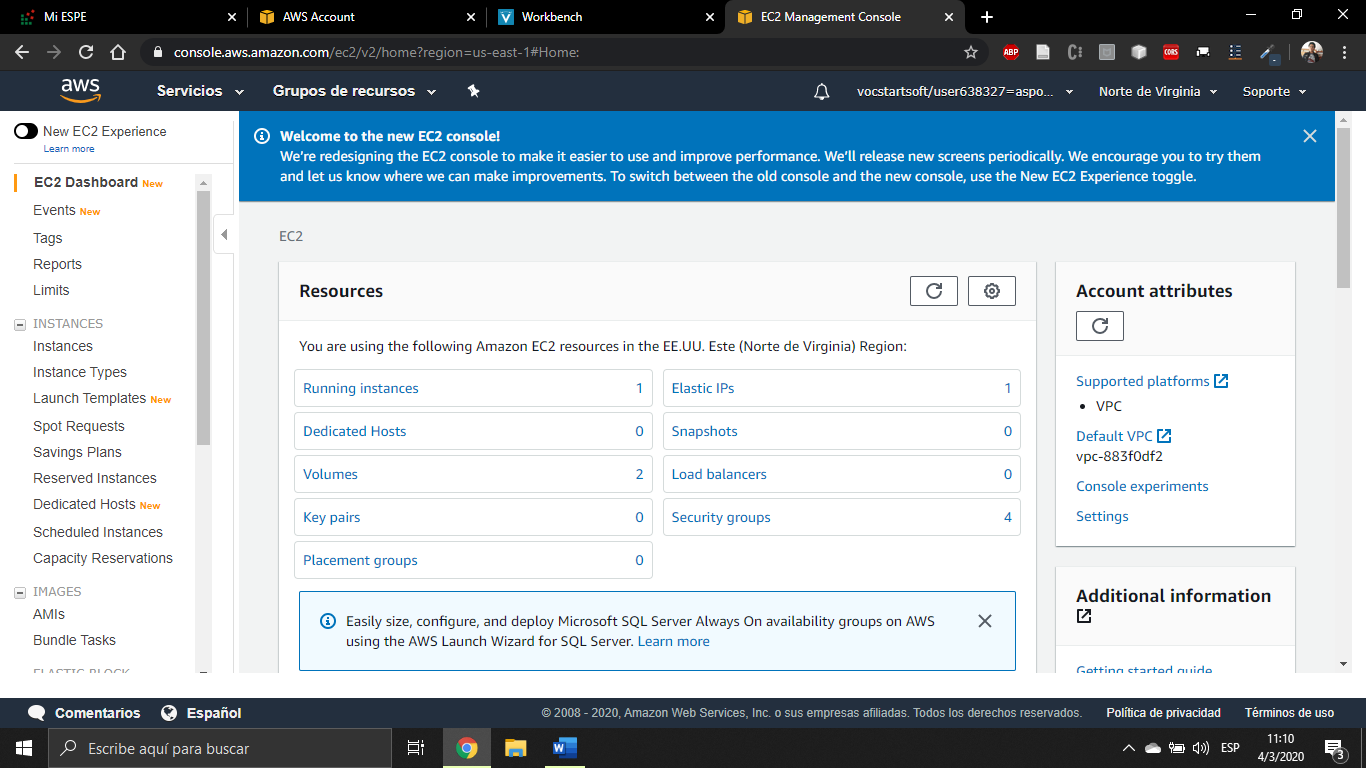


Configurar los puertos

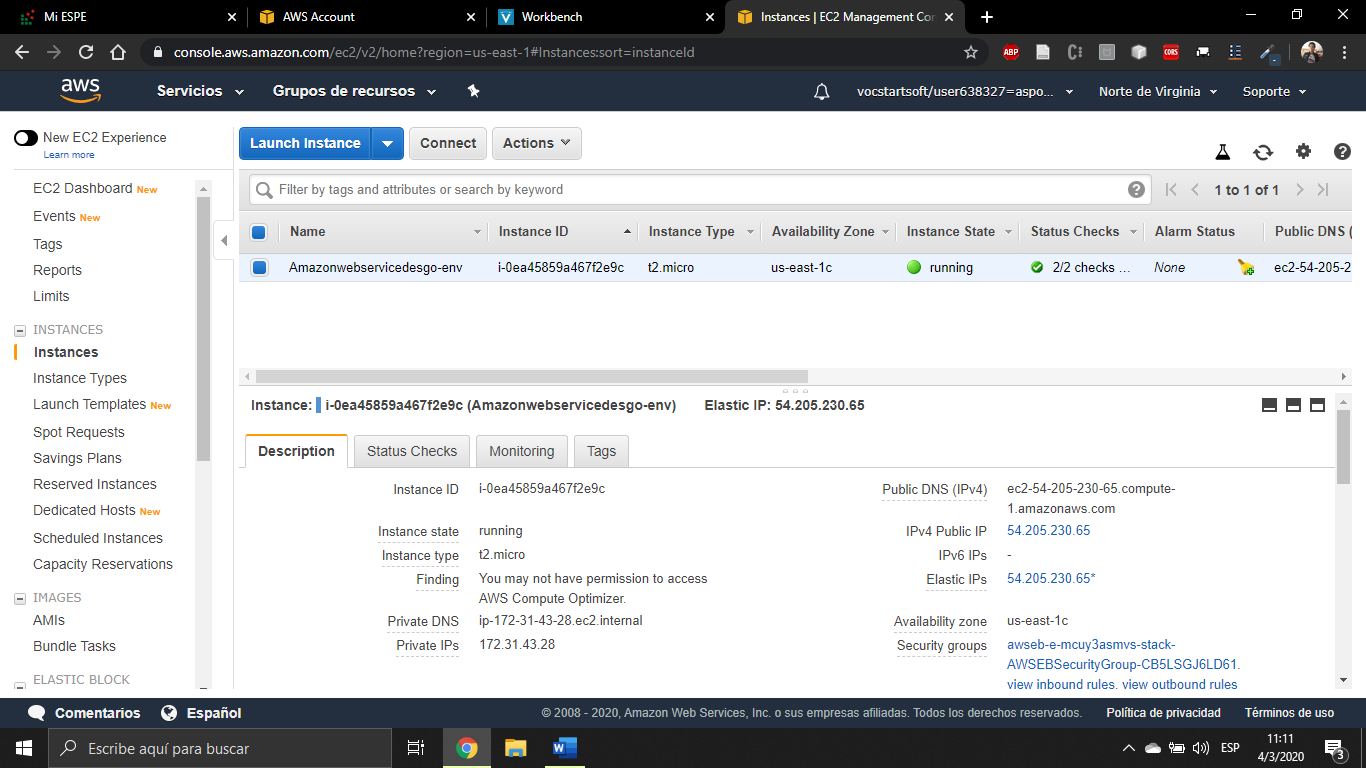
Seleccionar ***servicios***

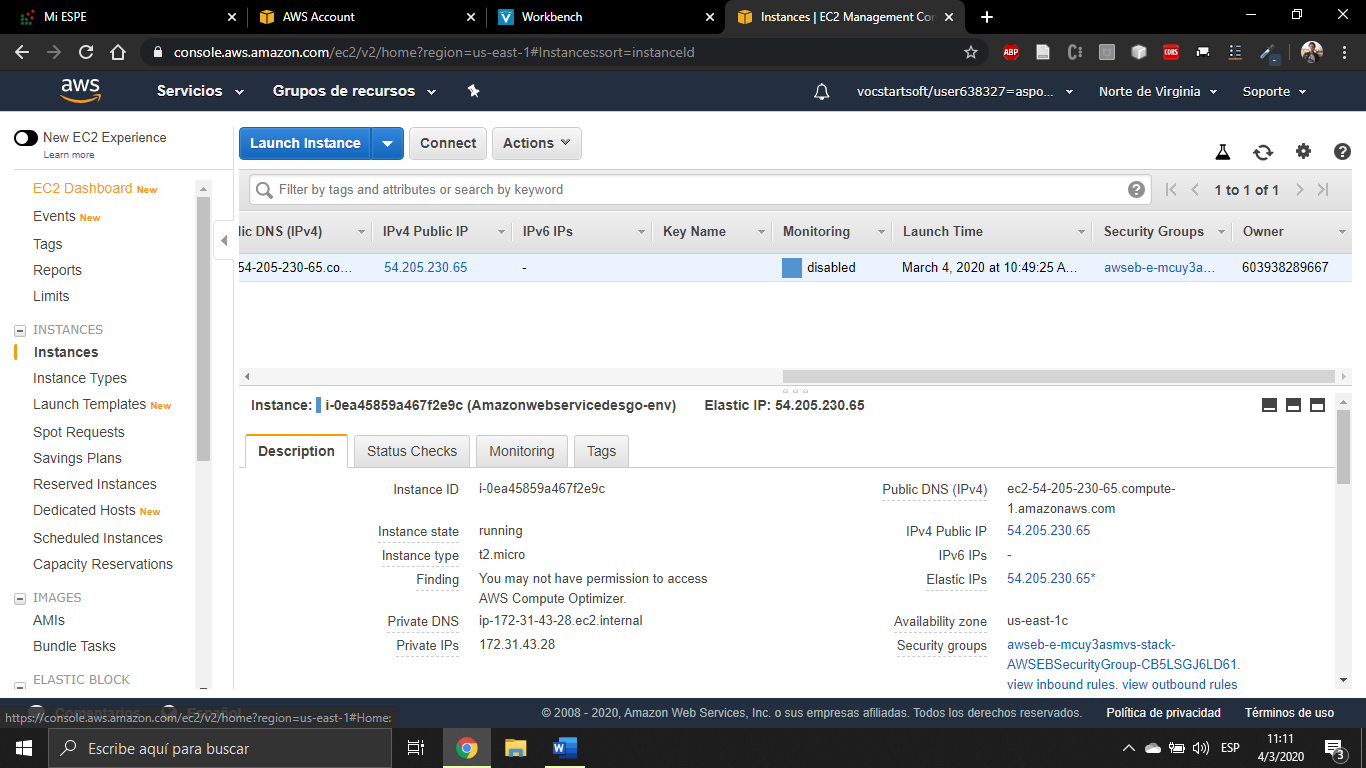


Seleccionar ***EC2***



Seleccionar ***Instances***

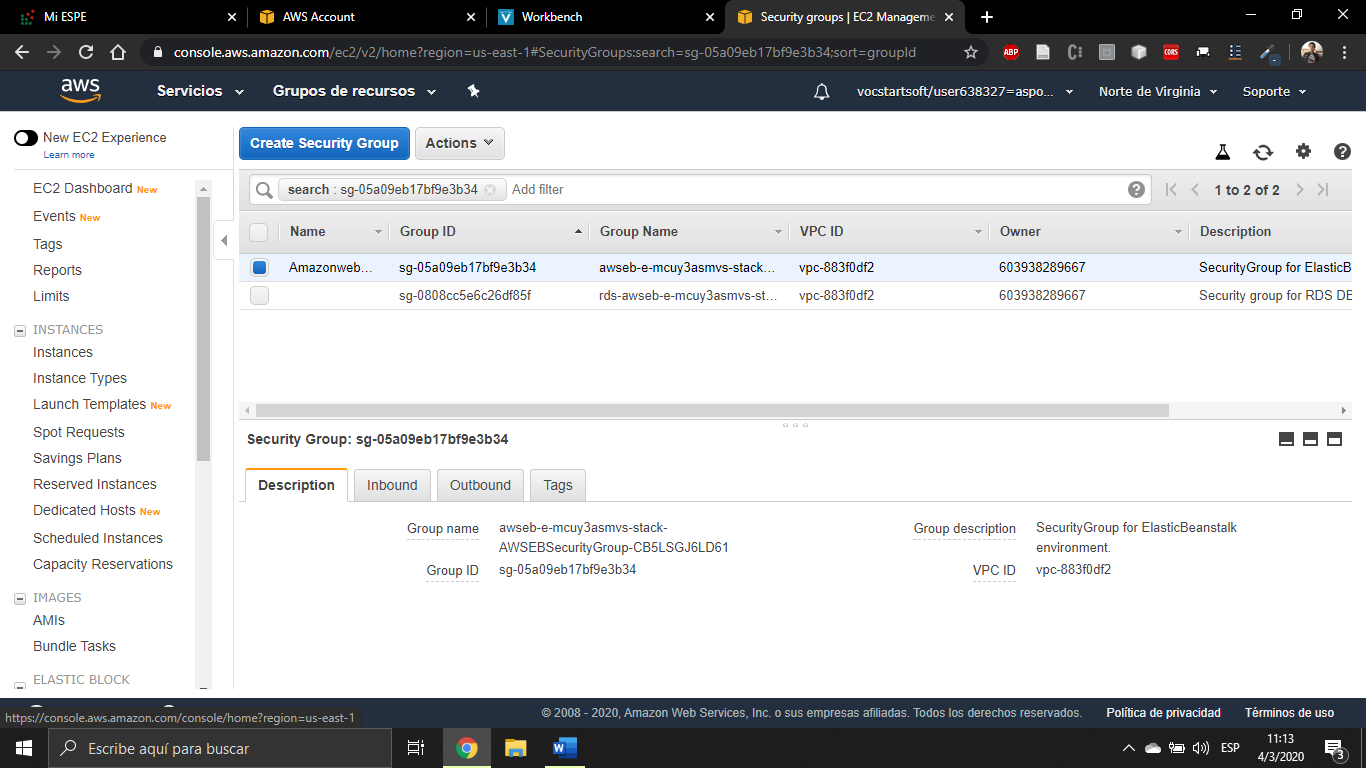


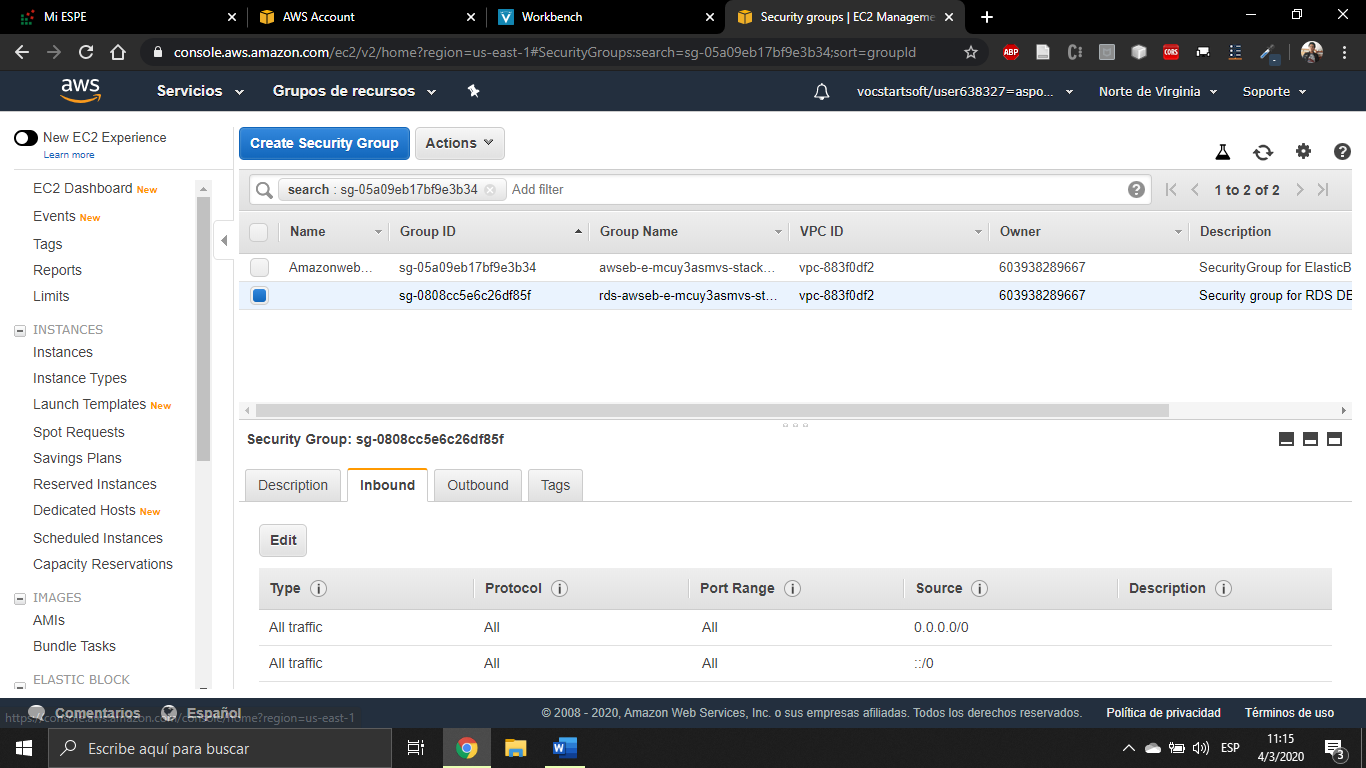
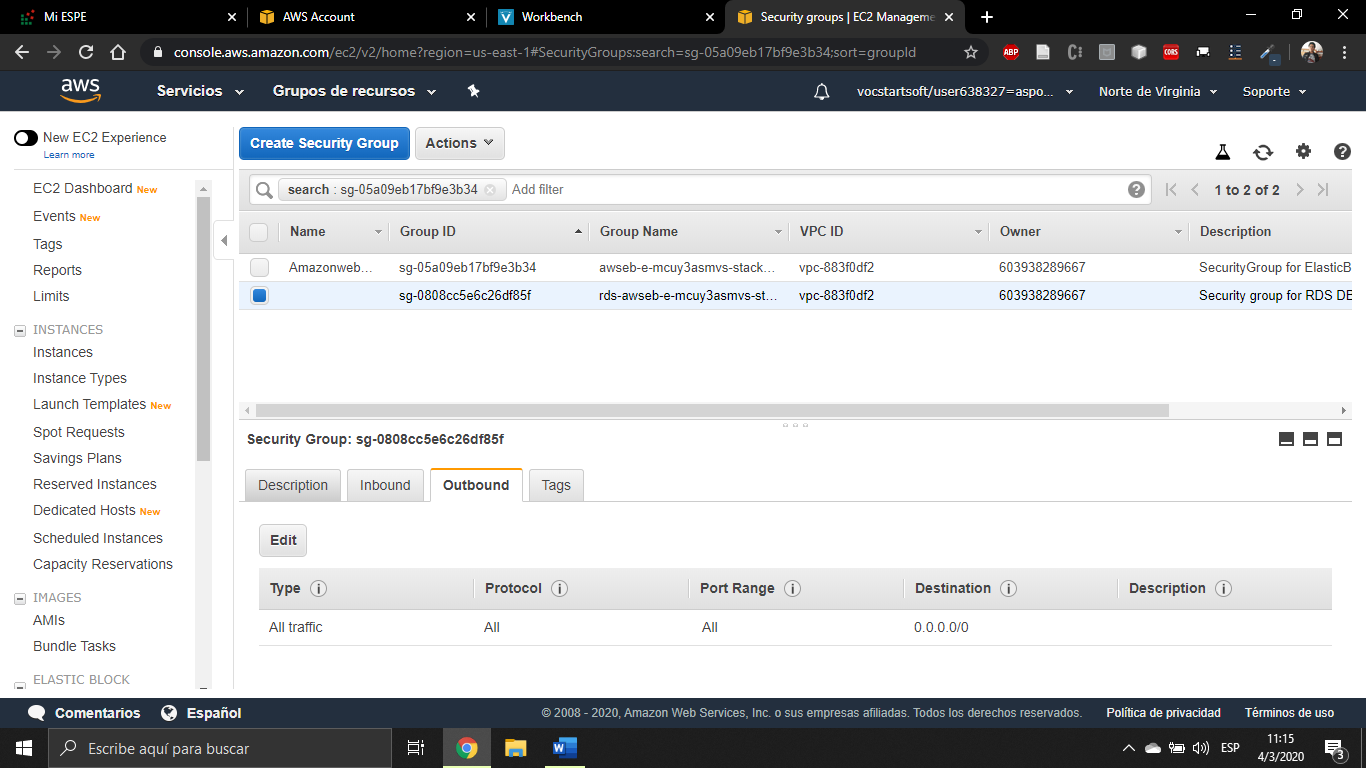


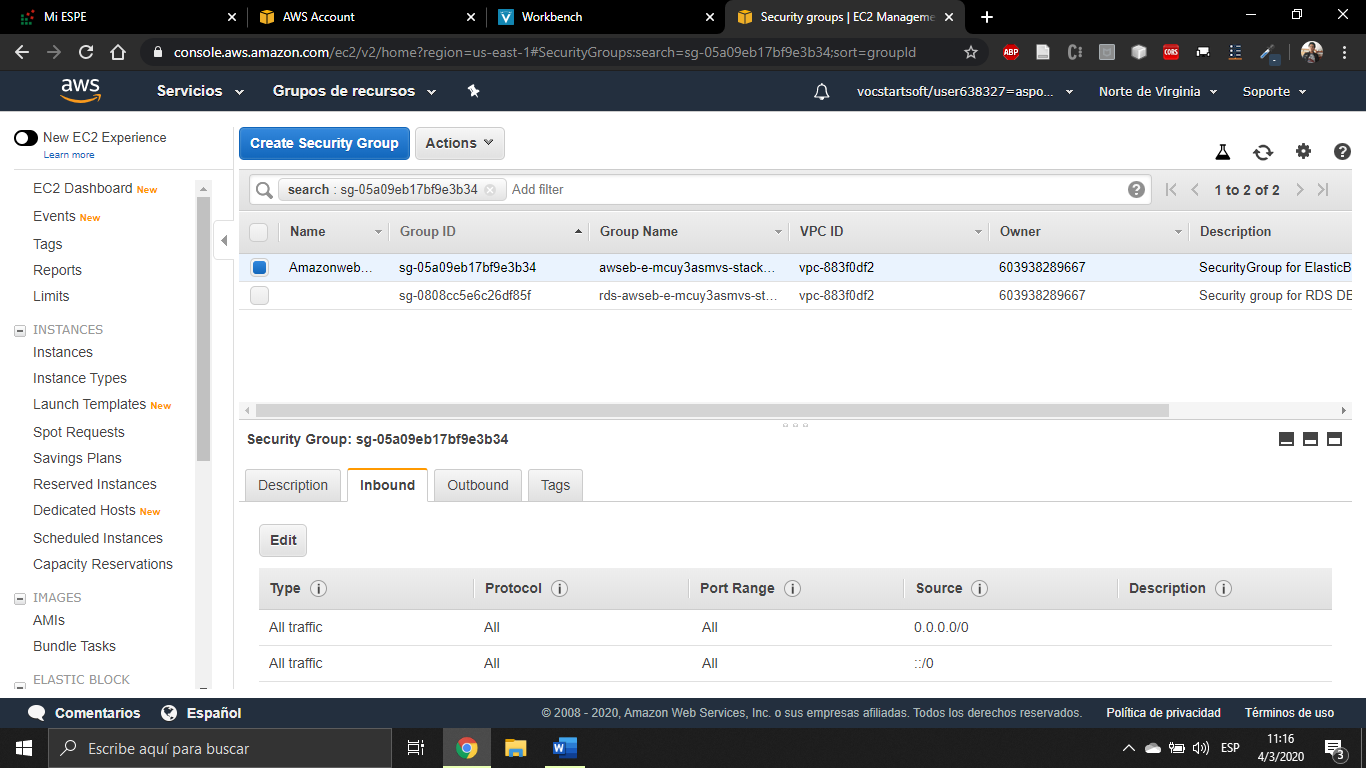
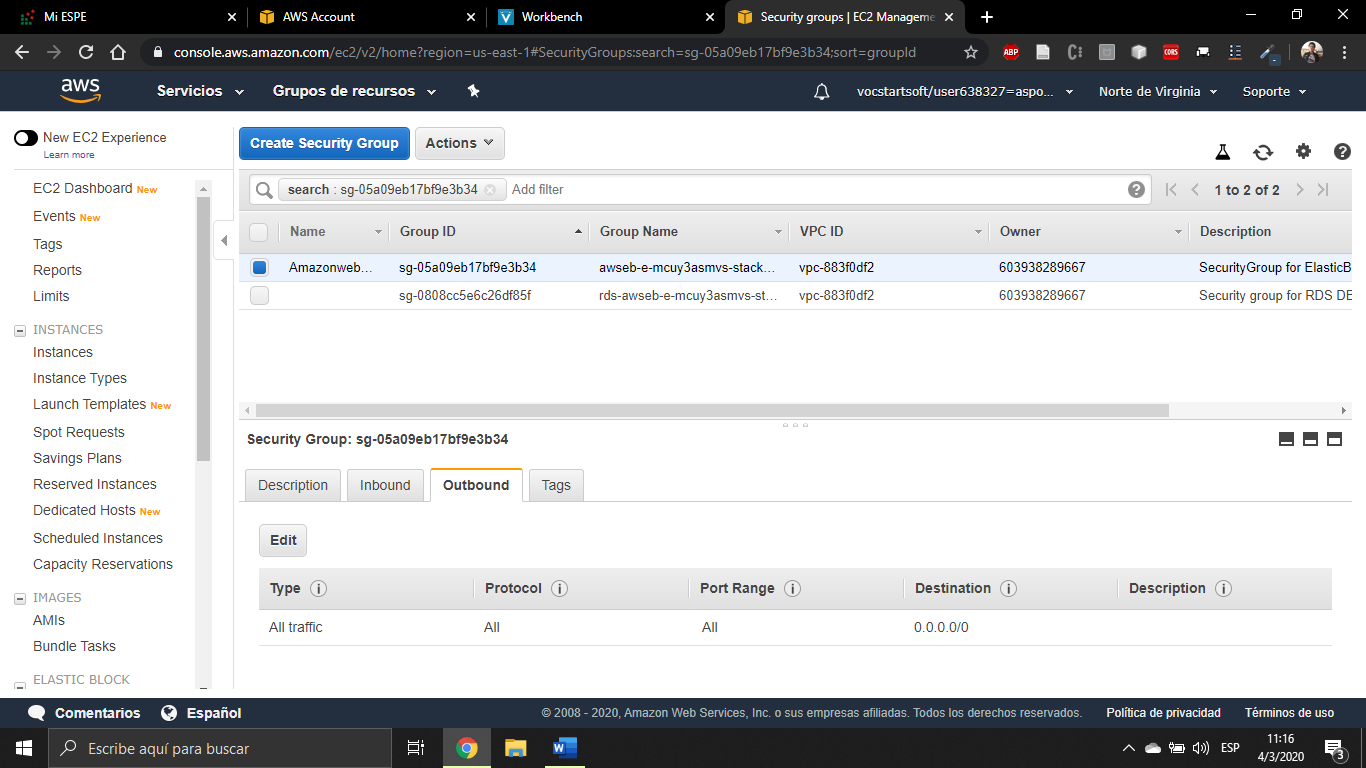
Seleccionar la opción ***Security group***

Configurar los Inbound y outbound de los dos security group

***All traffic***

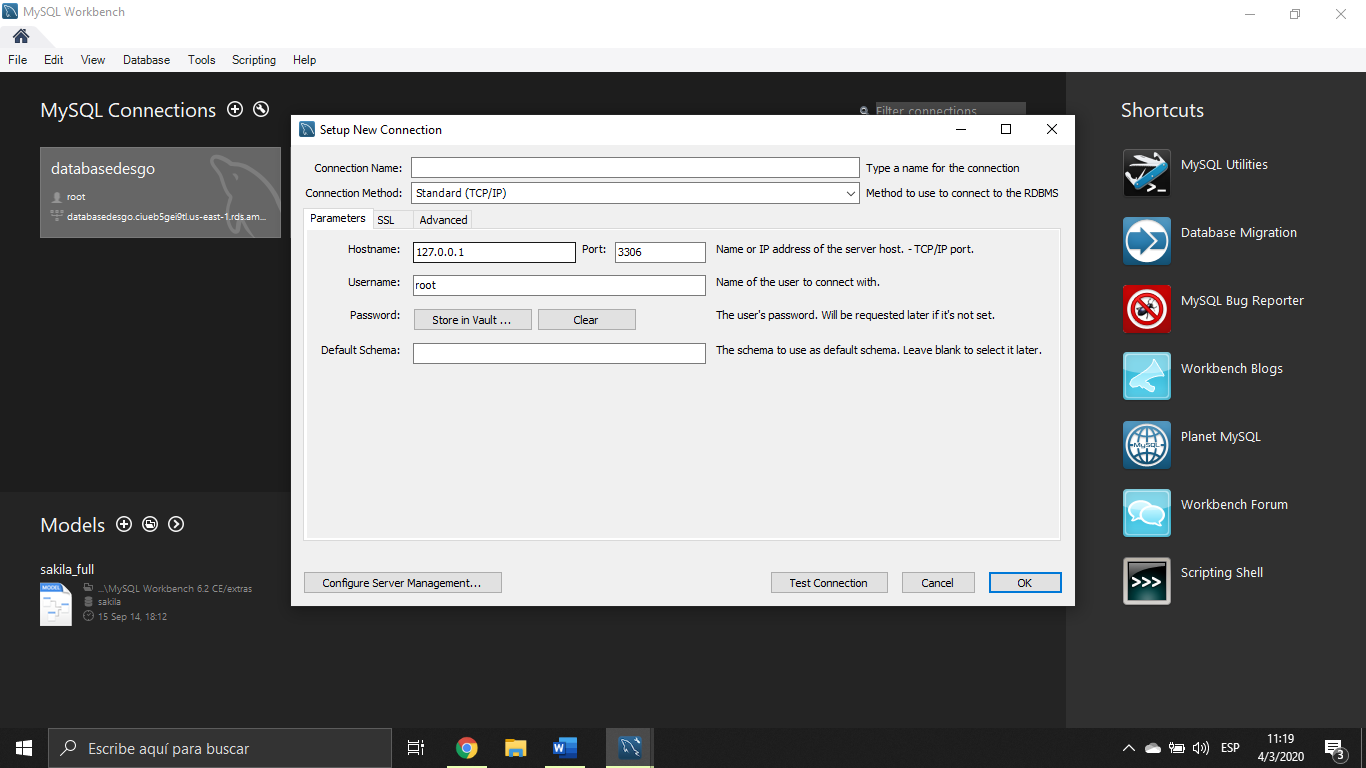


Cargar la base de datos

Abrir la aplicación MySQL Workbench para establecer la conexión



Ingresar los siguientes datos:

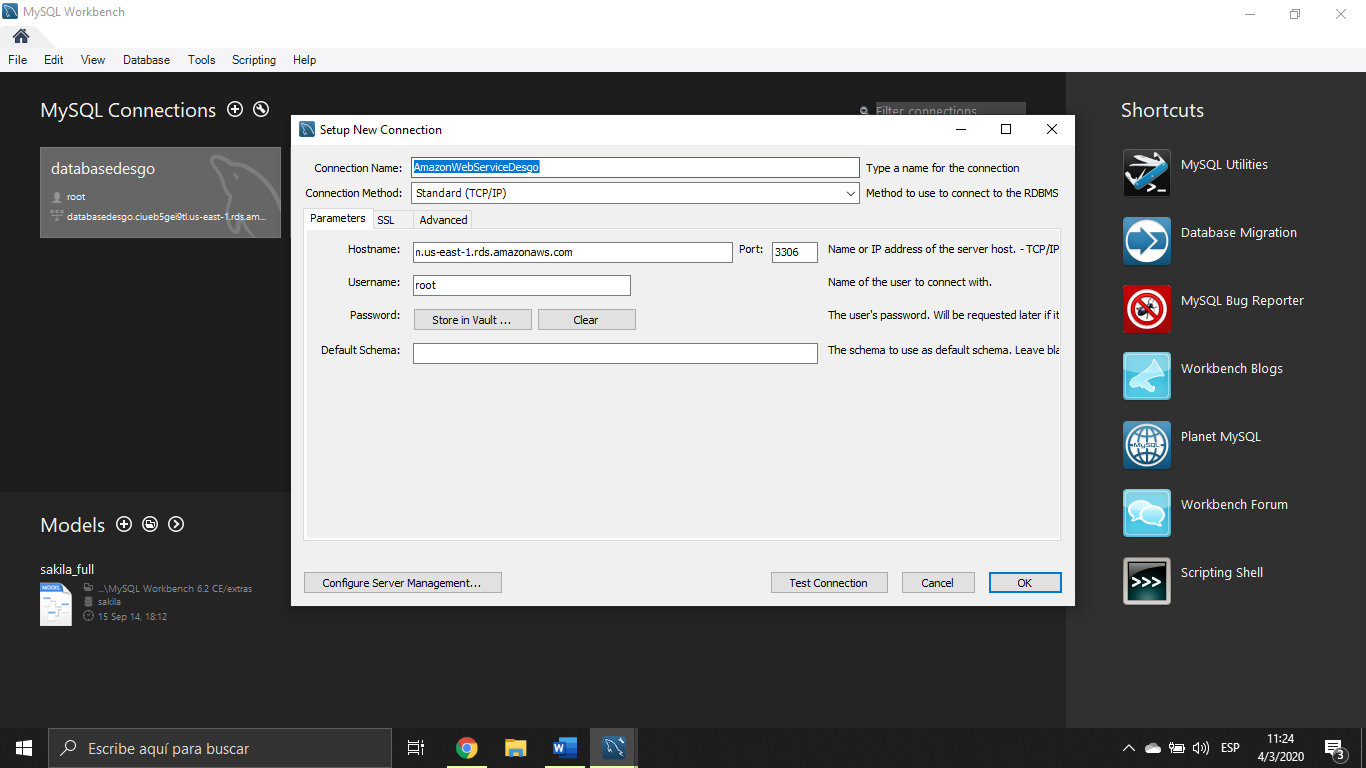
Nota: la información requerida se encuentra en las configuraciones/ sección Base de datos / del Elastic B.



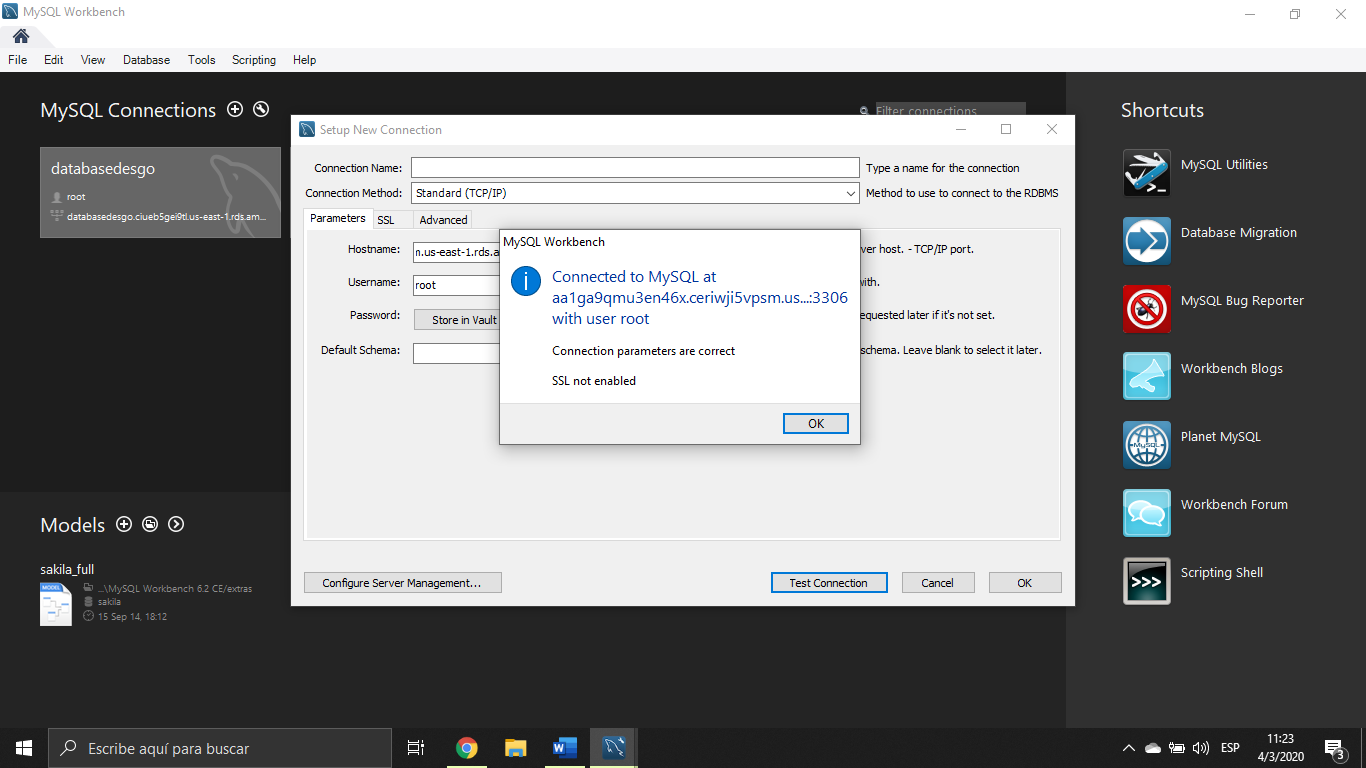
Hostname:  [aa1ga9qmu3en46x.ceriwji5vpsm.us-east-1.rds.amazonaws.com](https://console.aws.amazon.com/rds/home?region=us-east-1#dbinstances:id=aa1ga9qmu3en46x)

Username: root

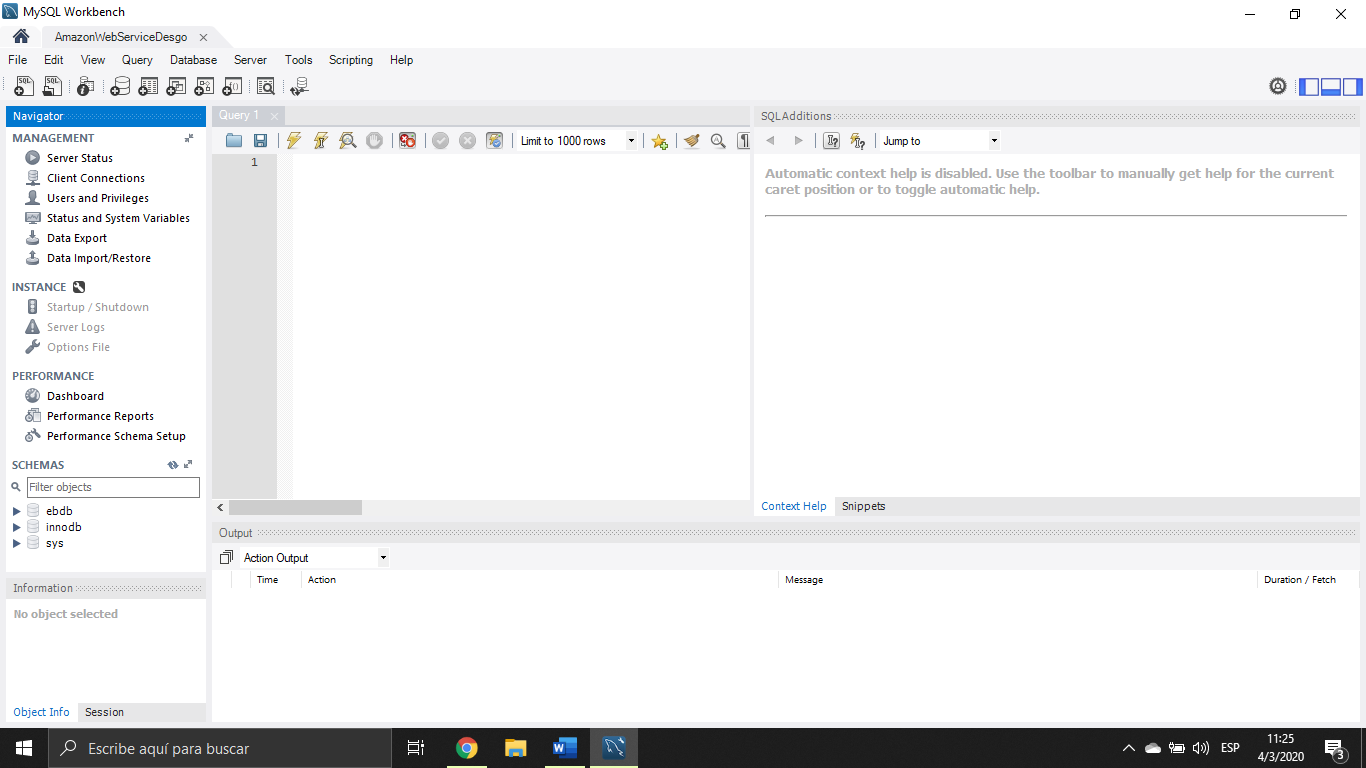
Password: rootdesgo



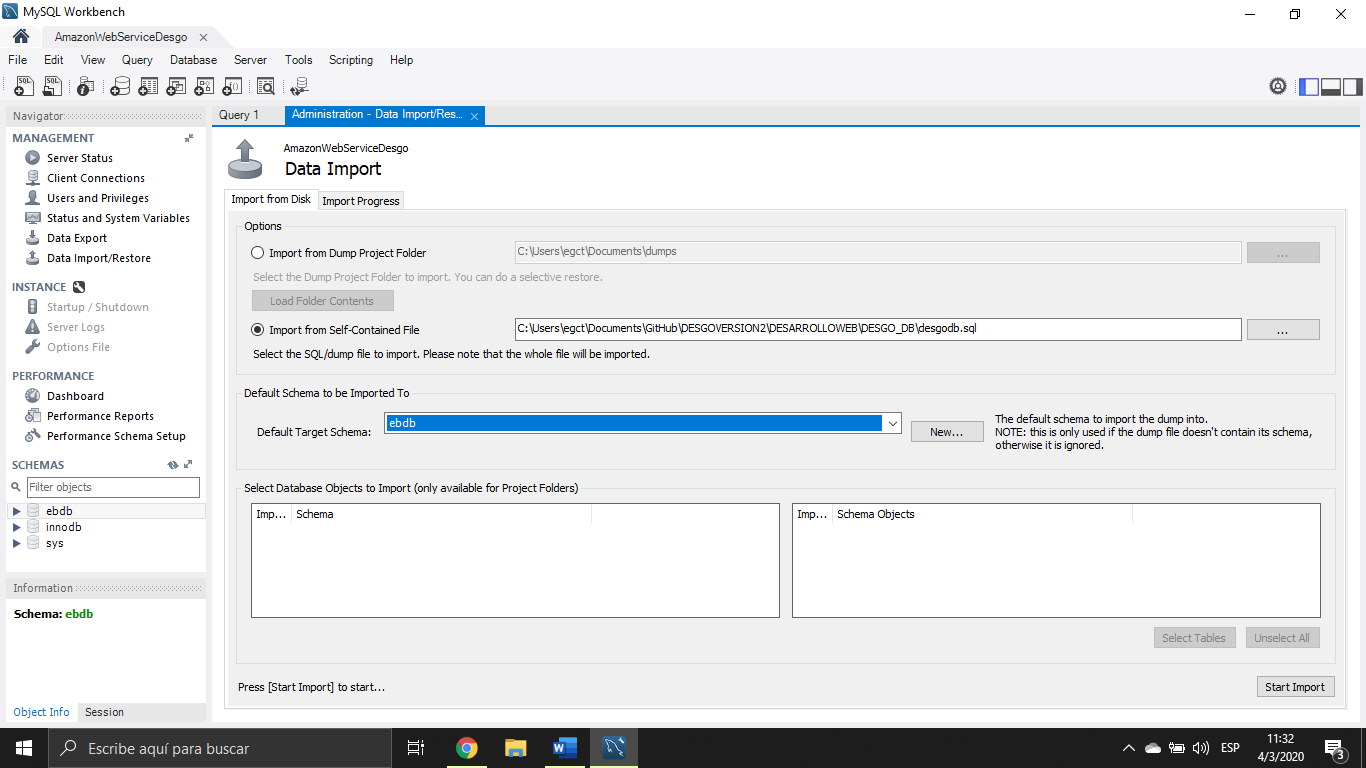
Verificamos conexión

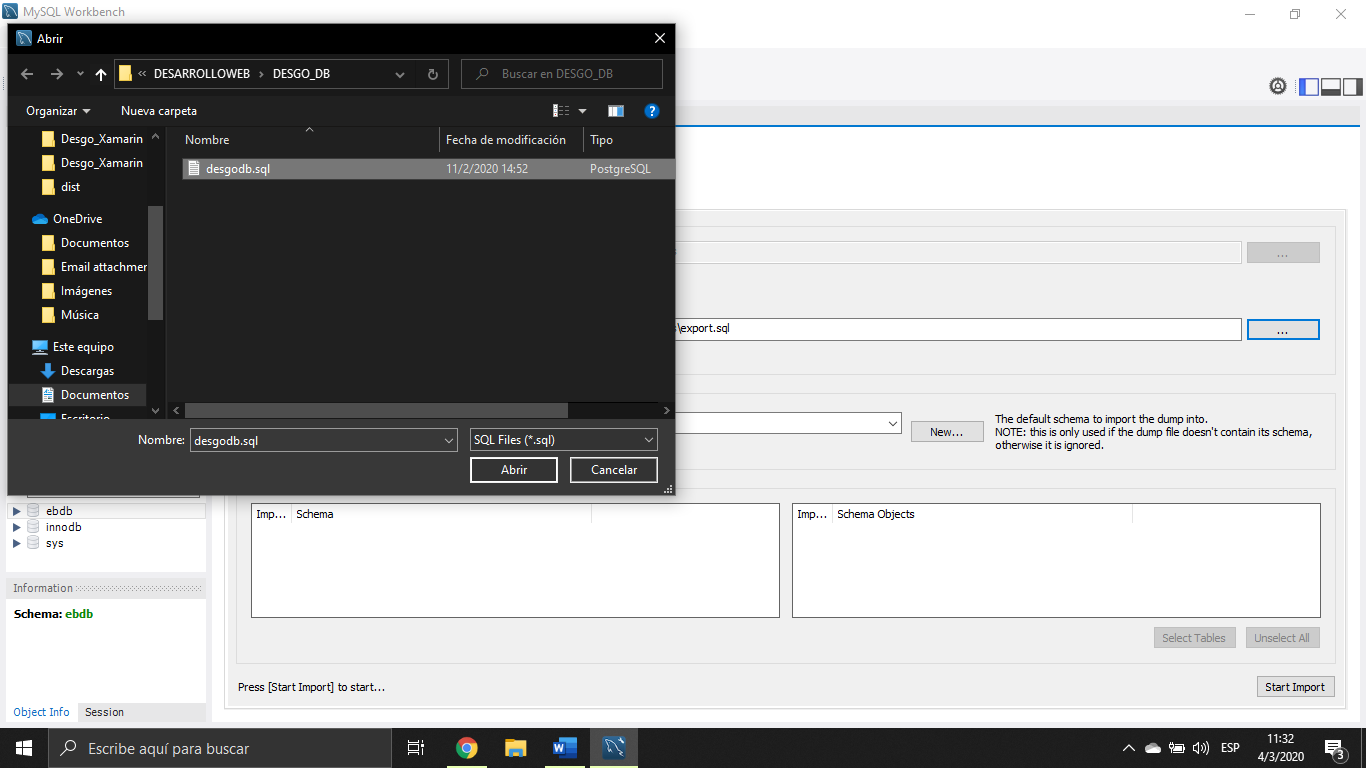


Ingresamos al entorno para modificar la base de datos

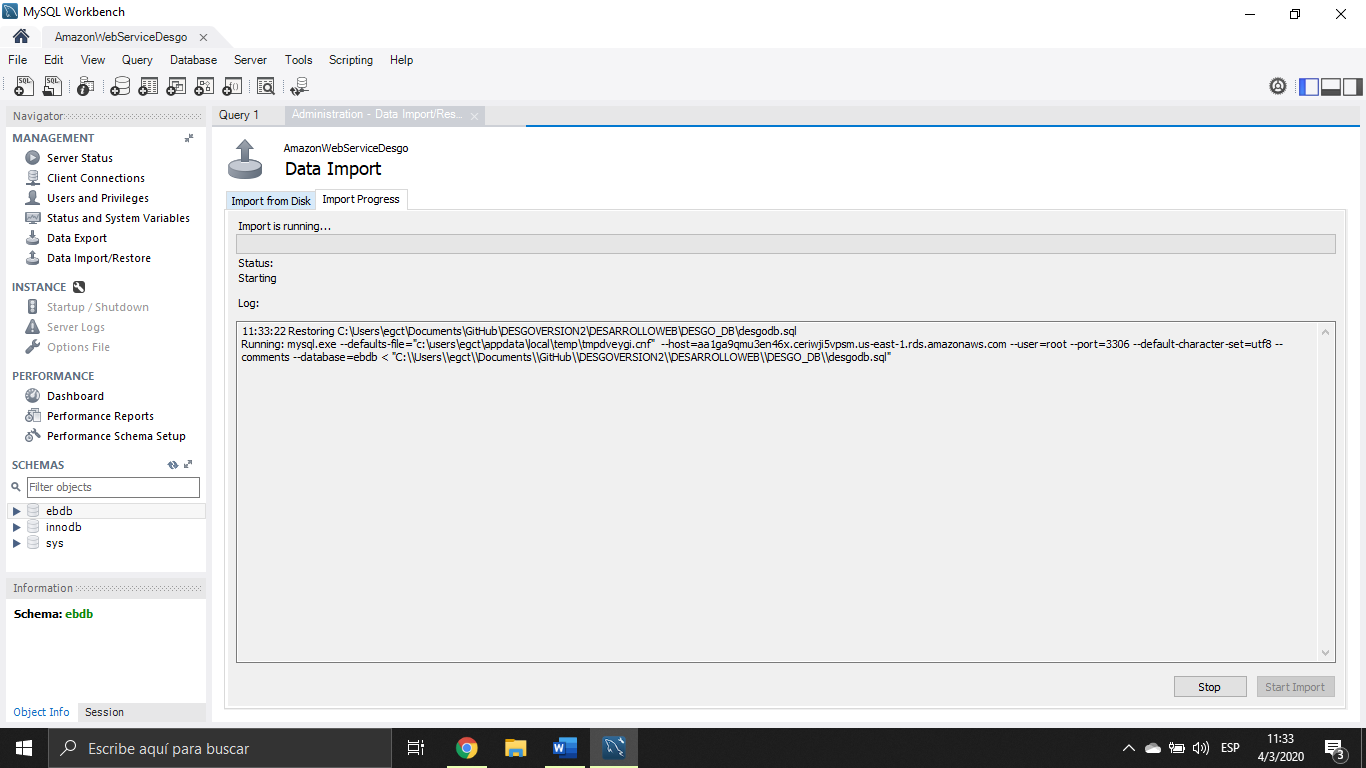


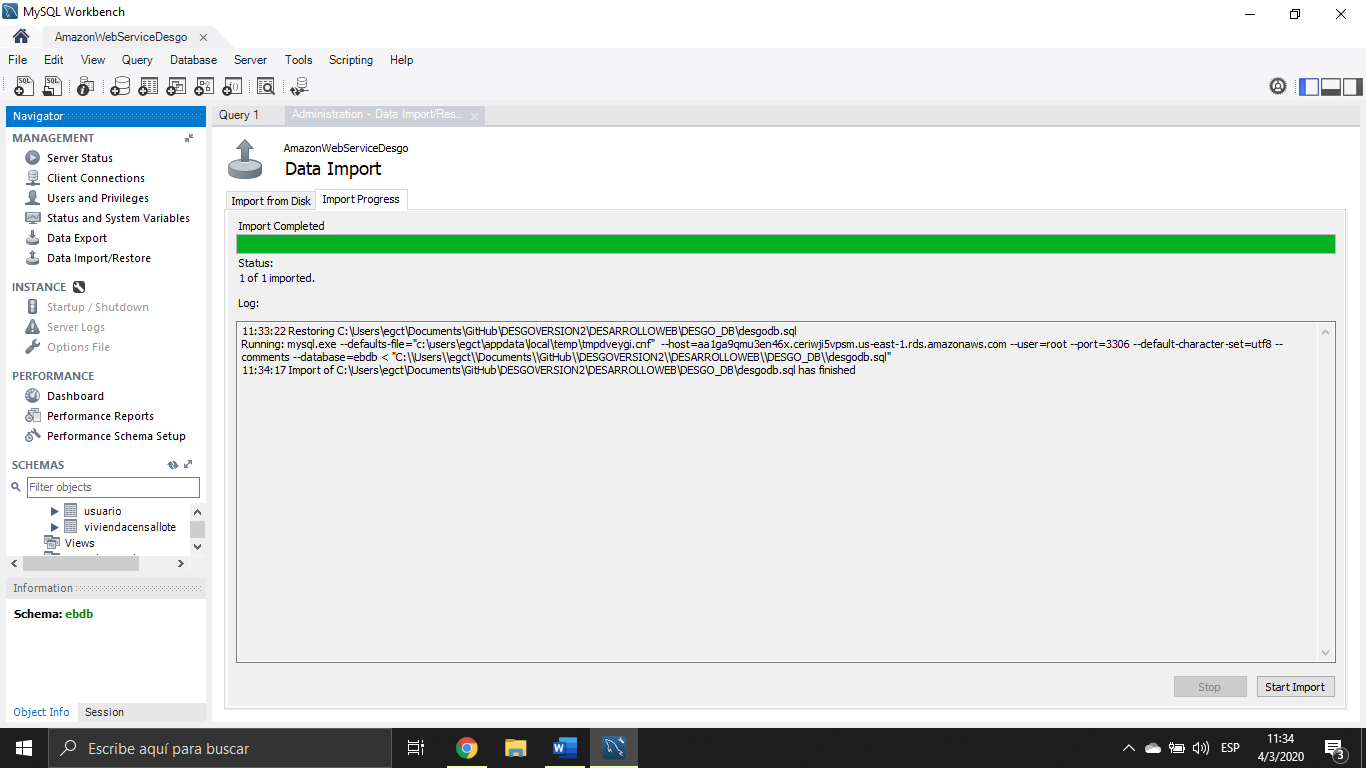
Importar la base de datos que se tiene local en el servidor



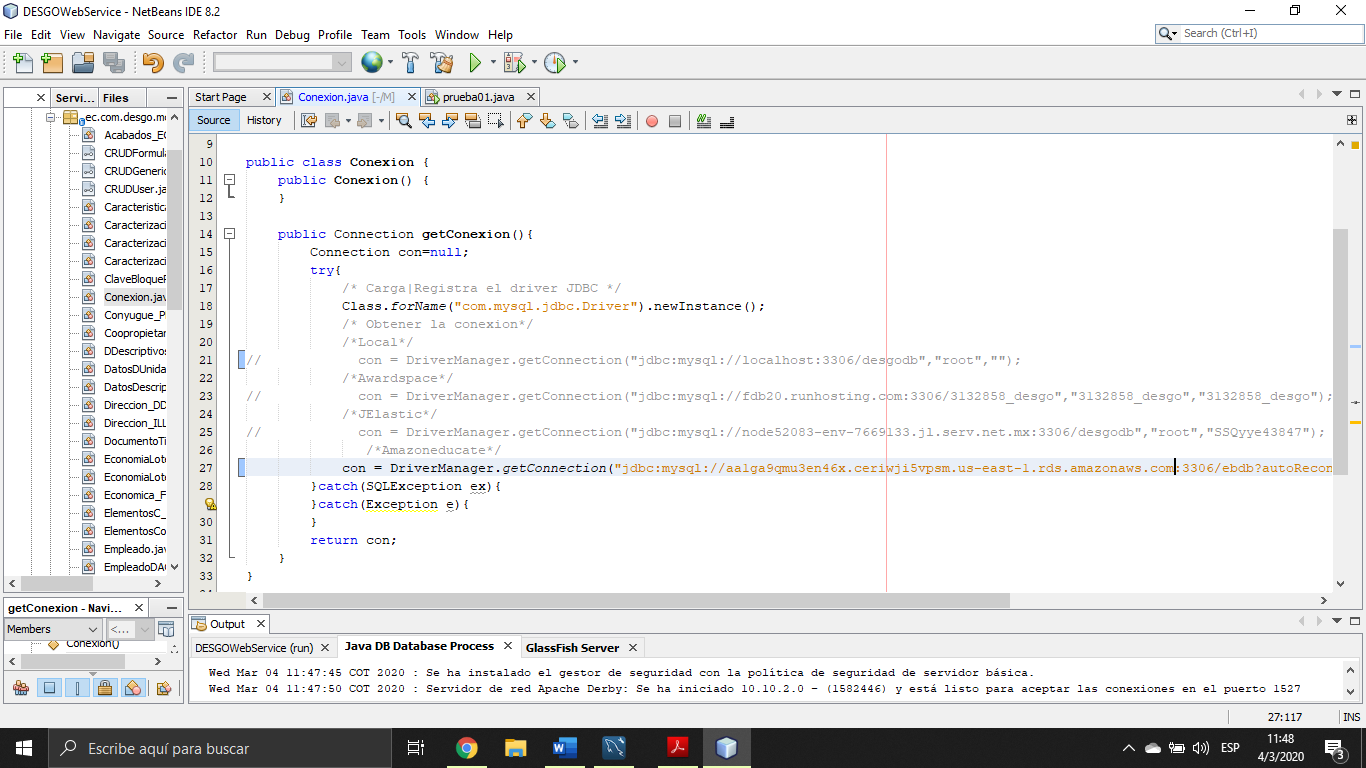


Start import





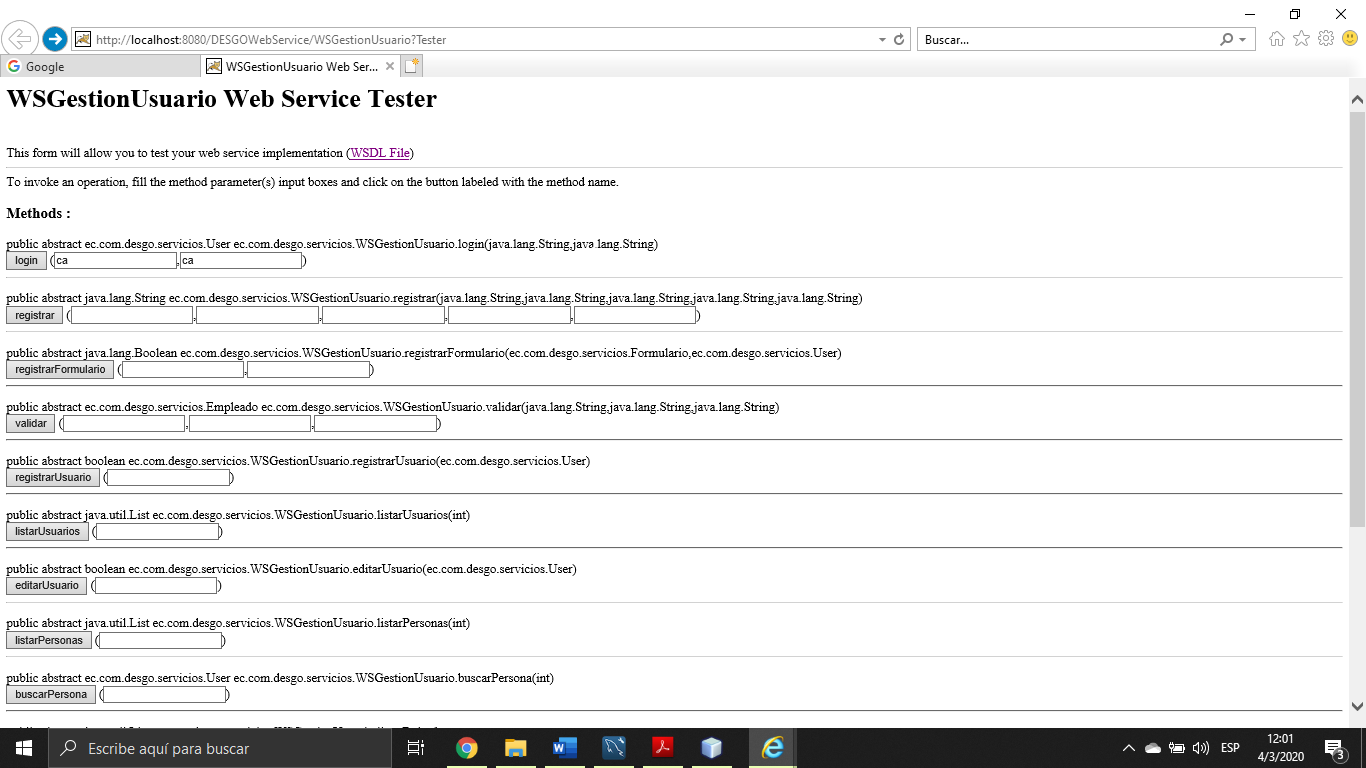
Modificar el proyecto “DesgowebService” cola nueva uri de conexión

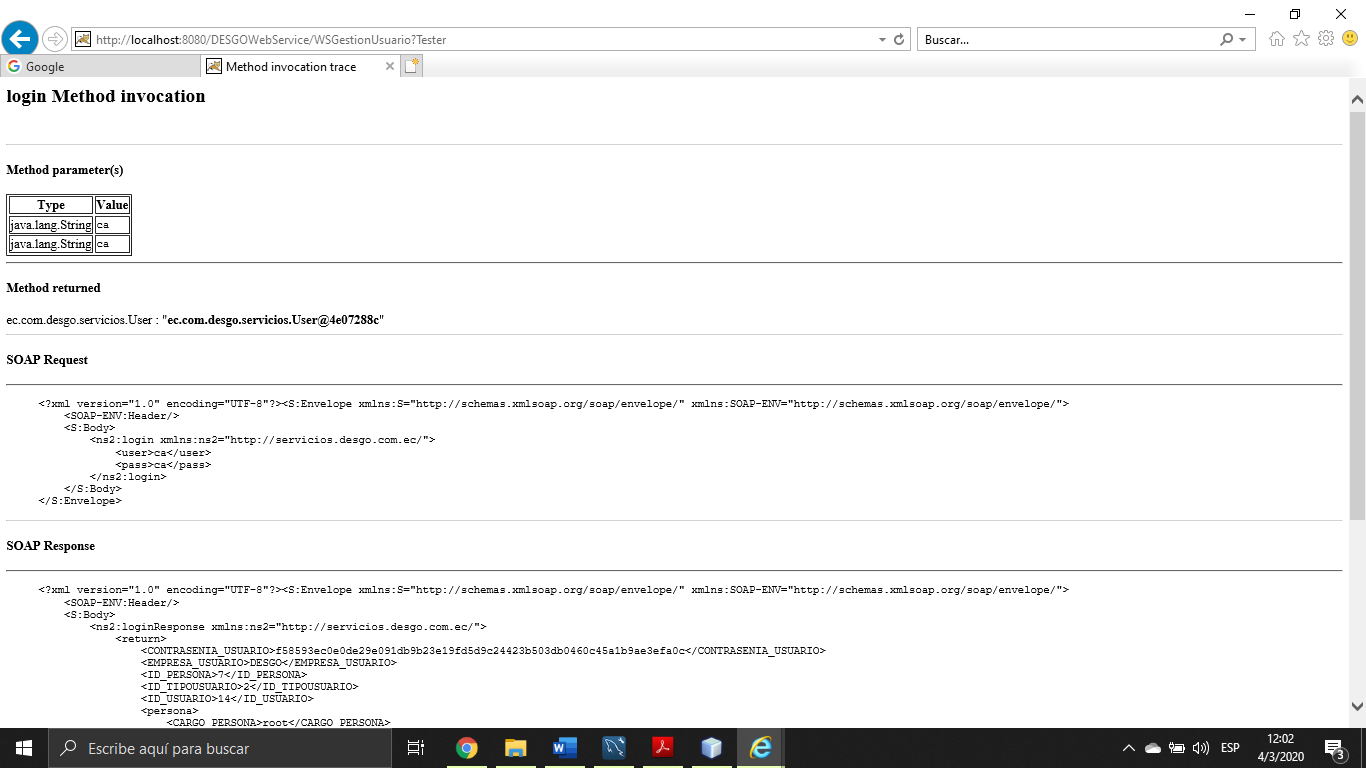


con = DriverManager.getConnection("jdbc:mysql://aa1ga9qmu3en46x.ceriwji5vpsm.us-east-1.rds.amazonaws.com:3306/ebdb?autoReconnect=true&useSSL=false","root","rootdesgo");

Proceder a limpiar, ejecutar y deployar el proyecto java web service para que se reconfiguren los cambios.

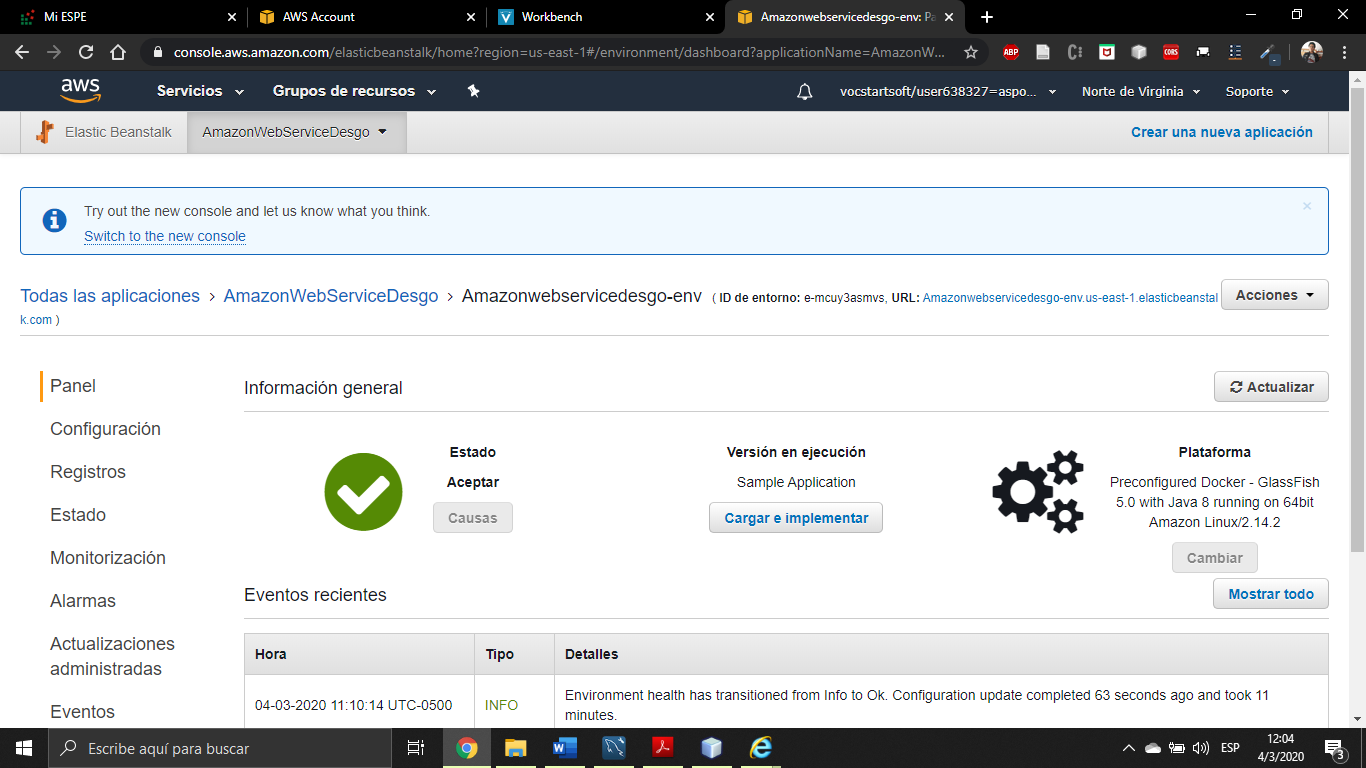
Verificar que el servicio ejecute correctamente con la nueva base de datos en AWS





Ahora, subir el punto .war (aplicación web service) al servidor AWS para que se pueda acceder desde cualquier parte del mundo.

En esta sección, clic en cargar e implementar:

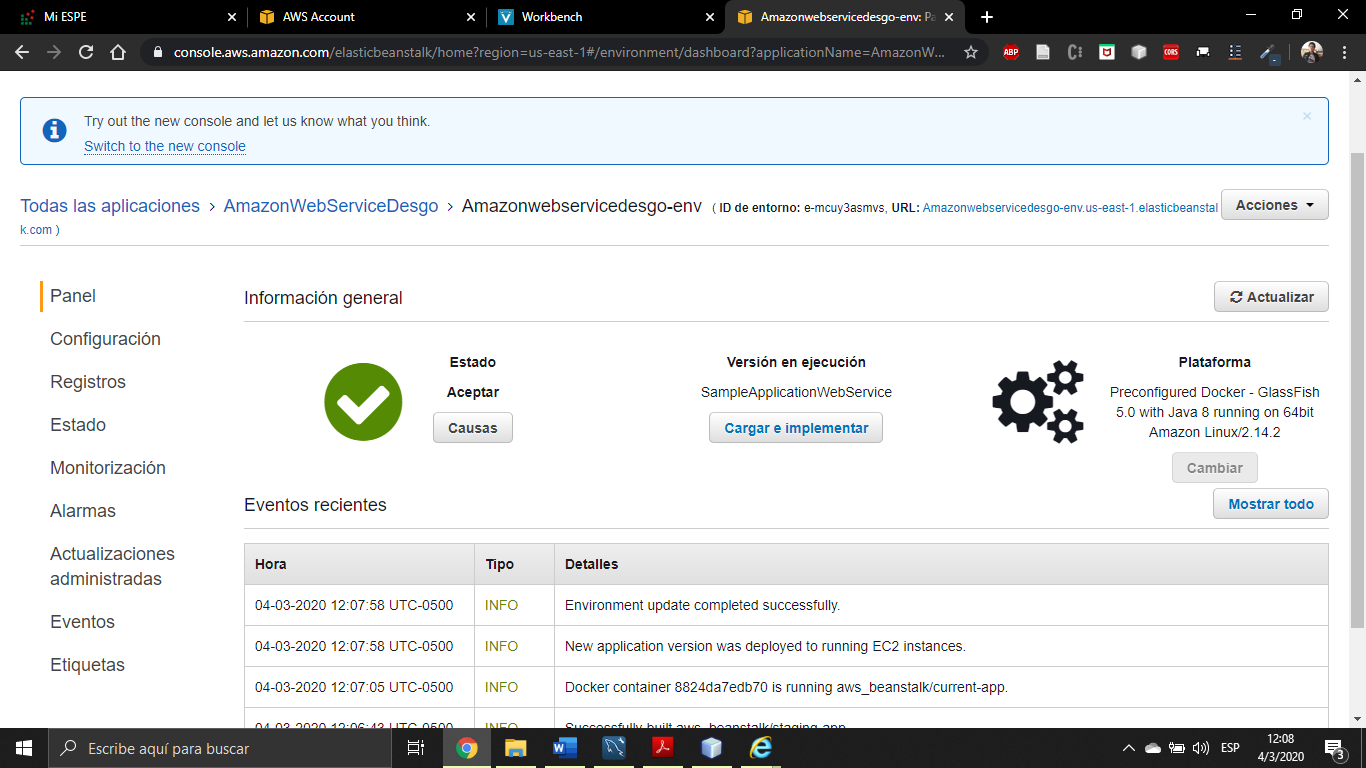


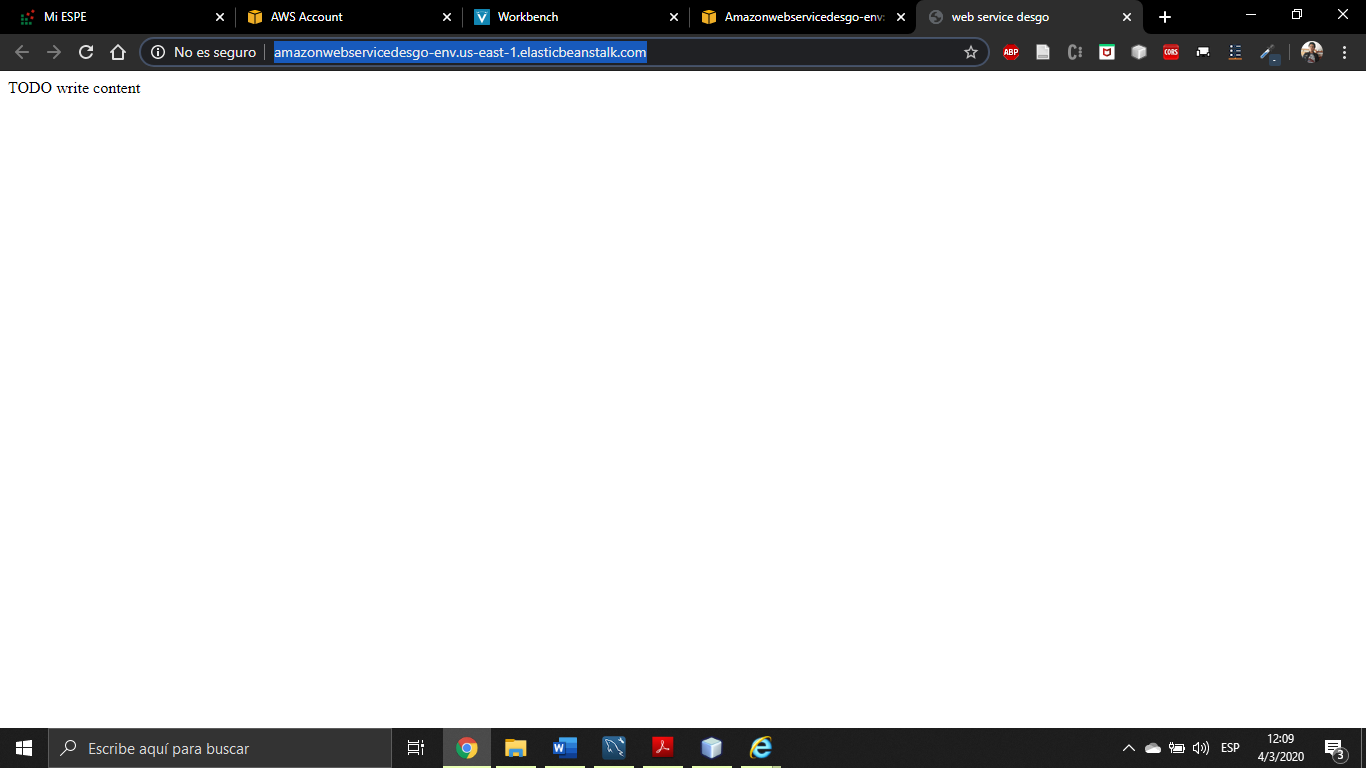
Seleccionar el archivo .war y clic en implementar

Verificamos si está disponible el servicio en amazon:

Clic en URL:

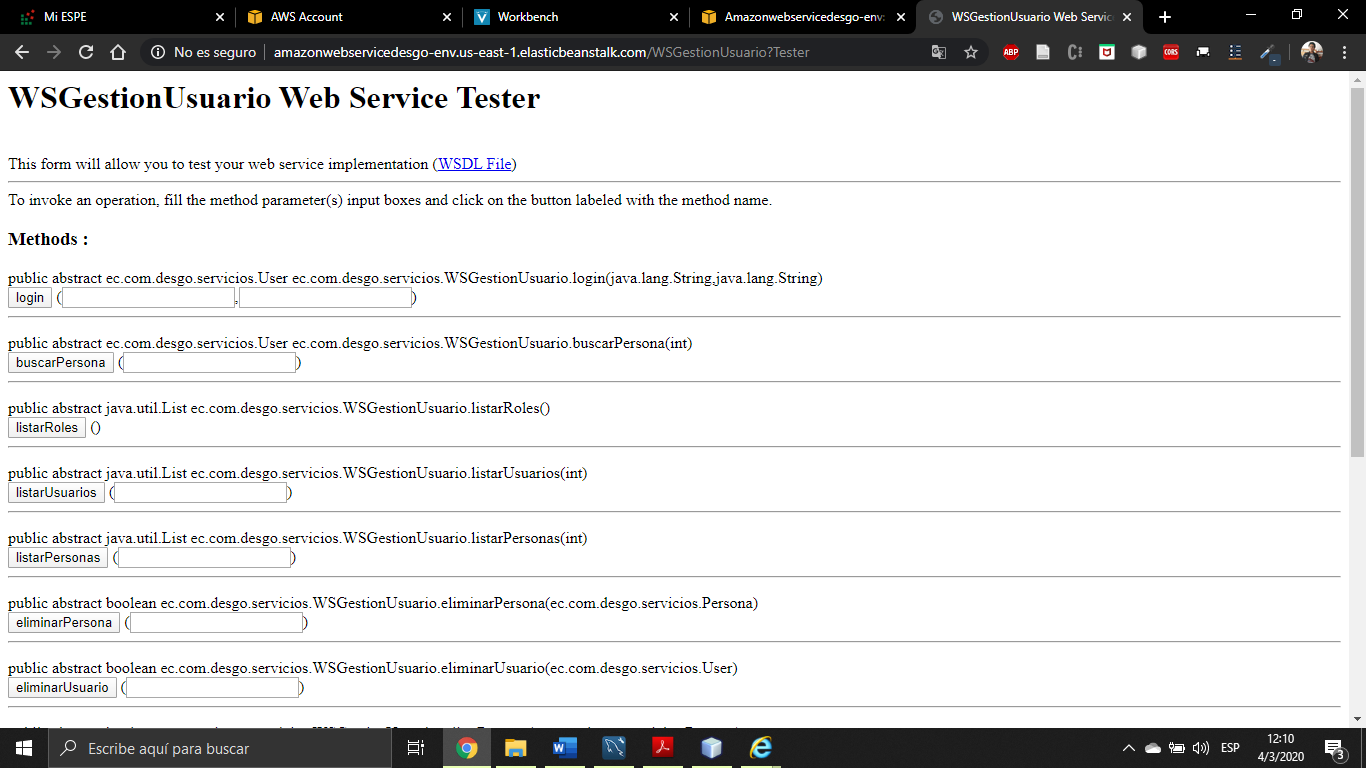
<http://amazonwebservicedesgo-env.us-east-1.elasticbeanstalk.com/>

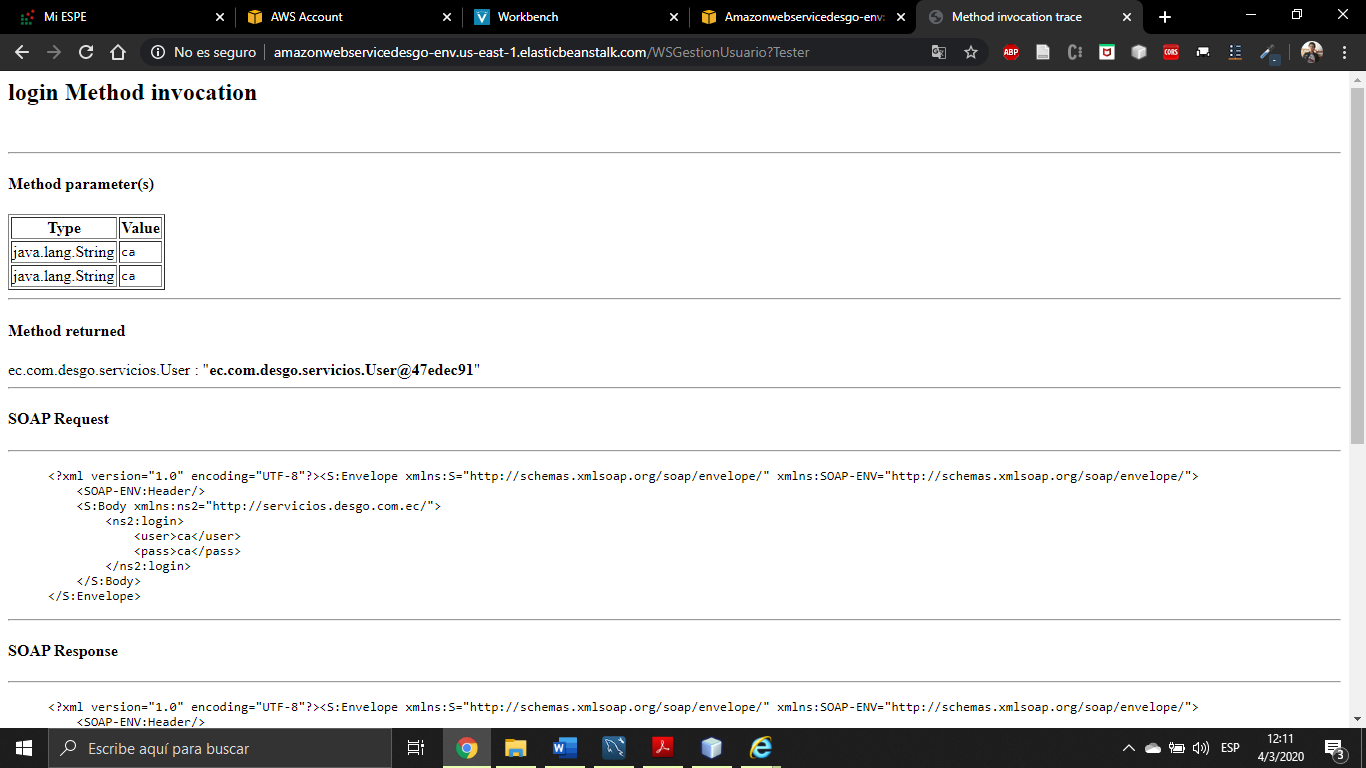




Ahora con los wsdl:

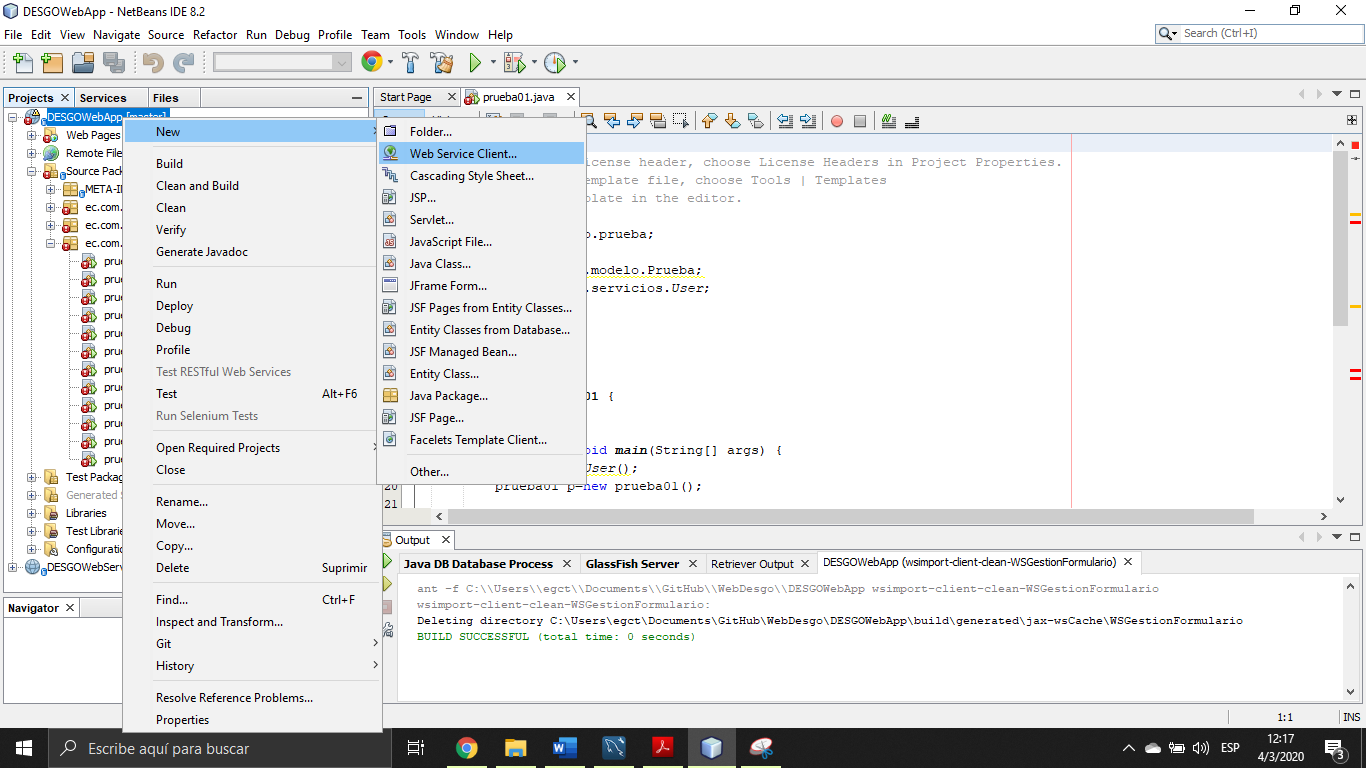
Efectivamente funciona

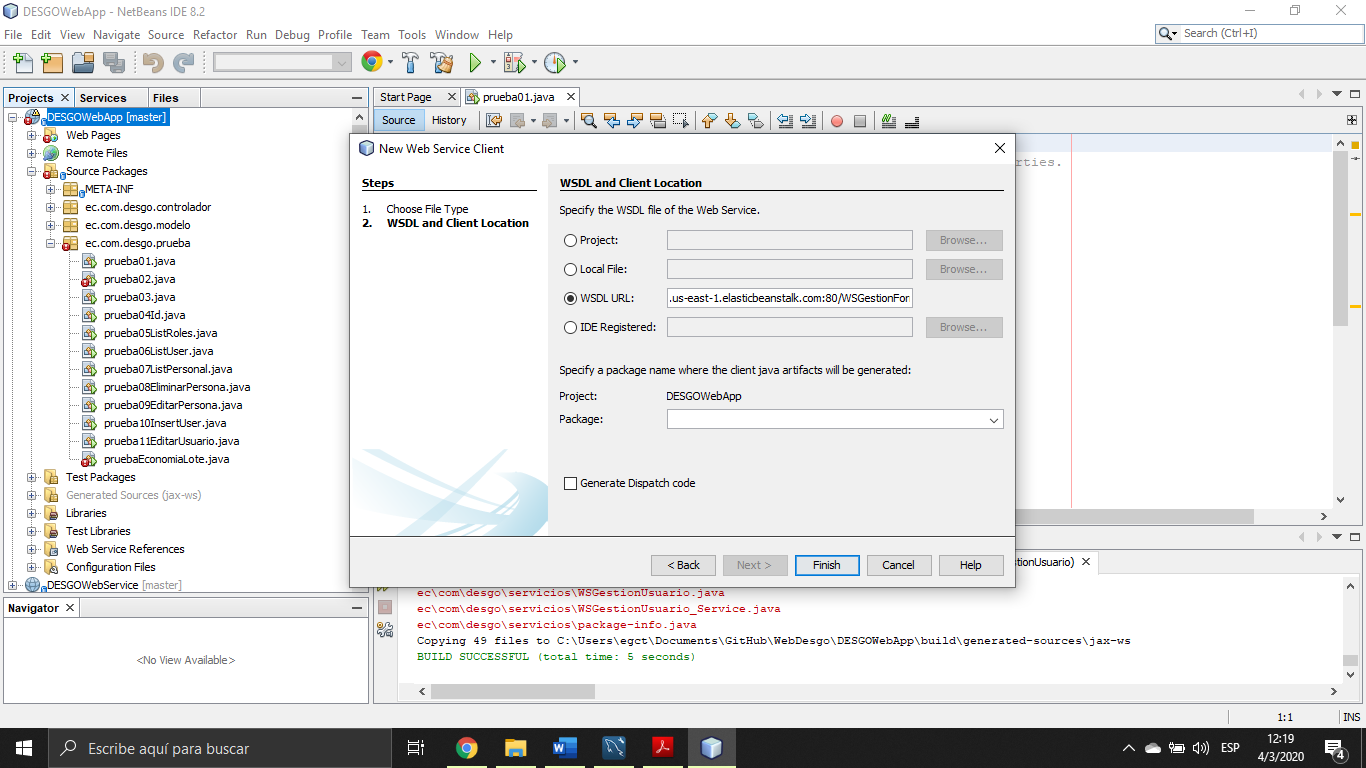




Consumir los servicios en la app web desgo

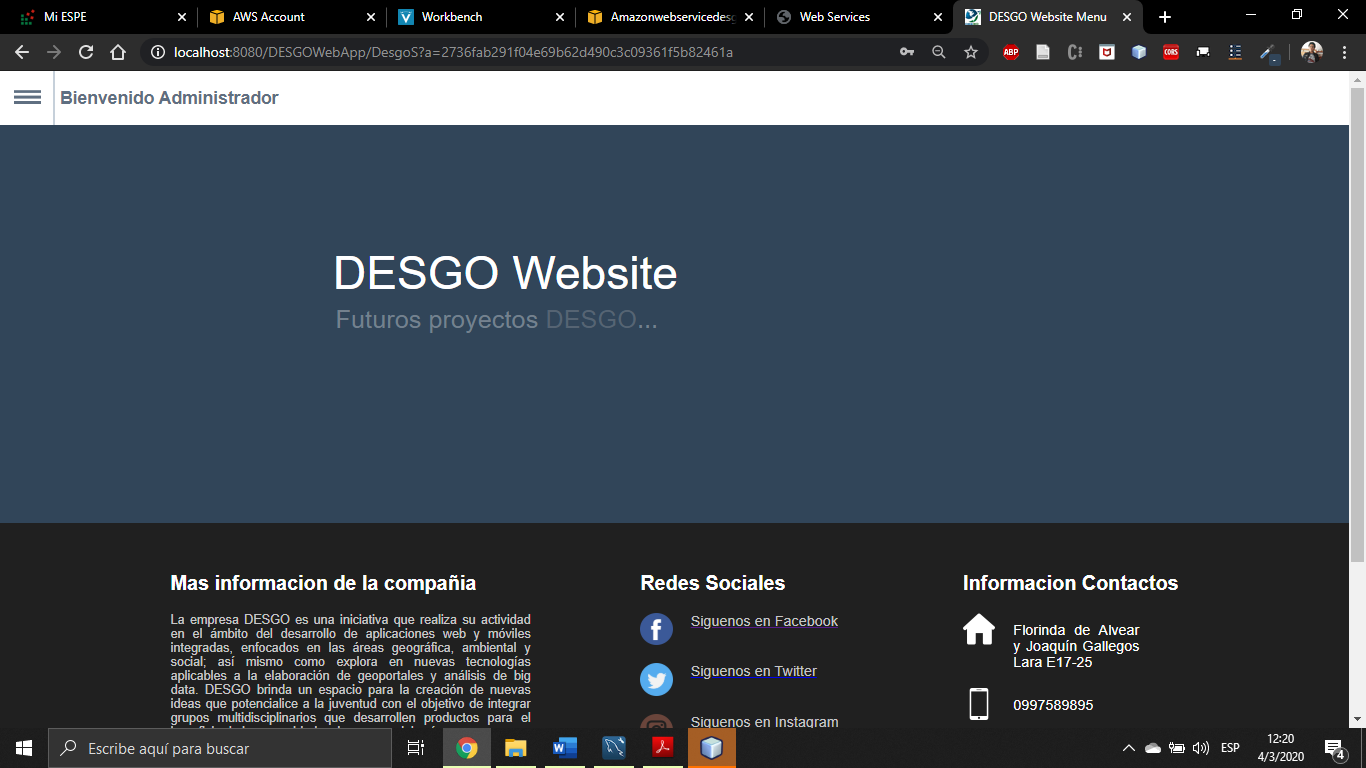
Crear un web service client, tanto para los usuarios como para el formulario





Procedemos a comprobar si funciona

Con desgo web local funciona



Ahora proceder a subir el proyecto desgo web local a la nube para que se pueda acceder desde cualquier parte.