

**Ingeniería en Desarrollo de Tecnología y Software**

**Sistemas Multidimensionales**

**por Mtro. Alfredo Bolio**

**Séptimo Semestre**

**Parcial 2**

**Reporte Segundo Parcial**

**Adrián L. Gorocica C.**

**Instrucciones**

Se deberá elaborar un documento por persona, de la realización de cada una de las actividades mencionados con anterioridad con imagen de captura y descripción de actividad realizada, siguiendo la siguiente estructura de contenido:

1. Objetivo del documento

2. Alcance del documento

3. Procesos

3.1 Repositorios

3.1.1. Creación de Repositorios

3.1.2. Clonación de Repositorios en Local

3.1.3. Creación Contenedor

3.1.3. Carga de código fuente incial

3.1.4. Creación de ramas, Master y Develop

3.2. Nube

3.2.1. Creación de Instancias LightSail

3.2.2. Instalación Docker en Instancias

3.2.3. Clonación de repositorio

3.2.4. Montaje de Contenedor

3.2.5. Configuración de aplicaciones

3.3. Seguridad y Redundancia

3.3.1. Configuración de firewall (Puertos e Ip's)

3.3.2. Creación de Instantánea de Web

3.3.3. Creación de segunda instancia de Web

3.3.4. Creación de subdominio para Web

3.3.5. Creación de Balanceador de Carga

3.3.6. Asignación de instancias a balanceador de carga

3.3.7. Creación de certificado SSL

4. Resumen

Objetivo

Tener un registro del flujo del desarrollo de un proyecto web haciendo uso de las tecnologías modernas. El flujo de desarrollo cubrirá las etapas de repositorio, nube y finalmente seguridad y redundancia. Las herramientas por usar serán: Github, para la creación de repositorios; Docker, para el montaje y manejo de instancias, y LightSail, para la gestión y mantenimiento de la web.

Alcance

Los puntos que se seguirán en la etapa de repositorios serán: la creación de un repositorio en Github que servirá para el almacenamiento de todo el proyecto, clonación del repositorio en local, creación de un contenedor, carga inicial de código y finalmente la creación de diferentes ramas para el correcto desarrollo; para la segunda etapa, de montaje en la nube de nuestro proyecto, se utilizará LightSail para crear las instancias, instalar Docker en la instancia creada, clonar el repositorio, montar el contenedor creado en la etapa anterior y configurar las aplicaciones; finalmente, en la última etapa, nos dedicaremos a la seguridad y redundancia al configurar el firewall, crear un subdominio, asignar un balanceador de cargas y terminaremos creando un certificado SSL.

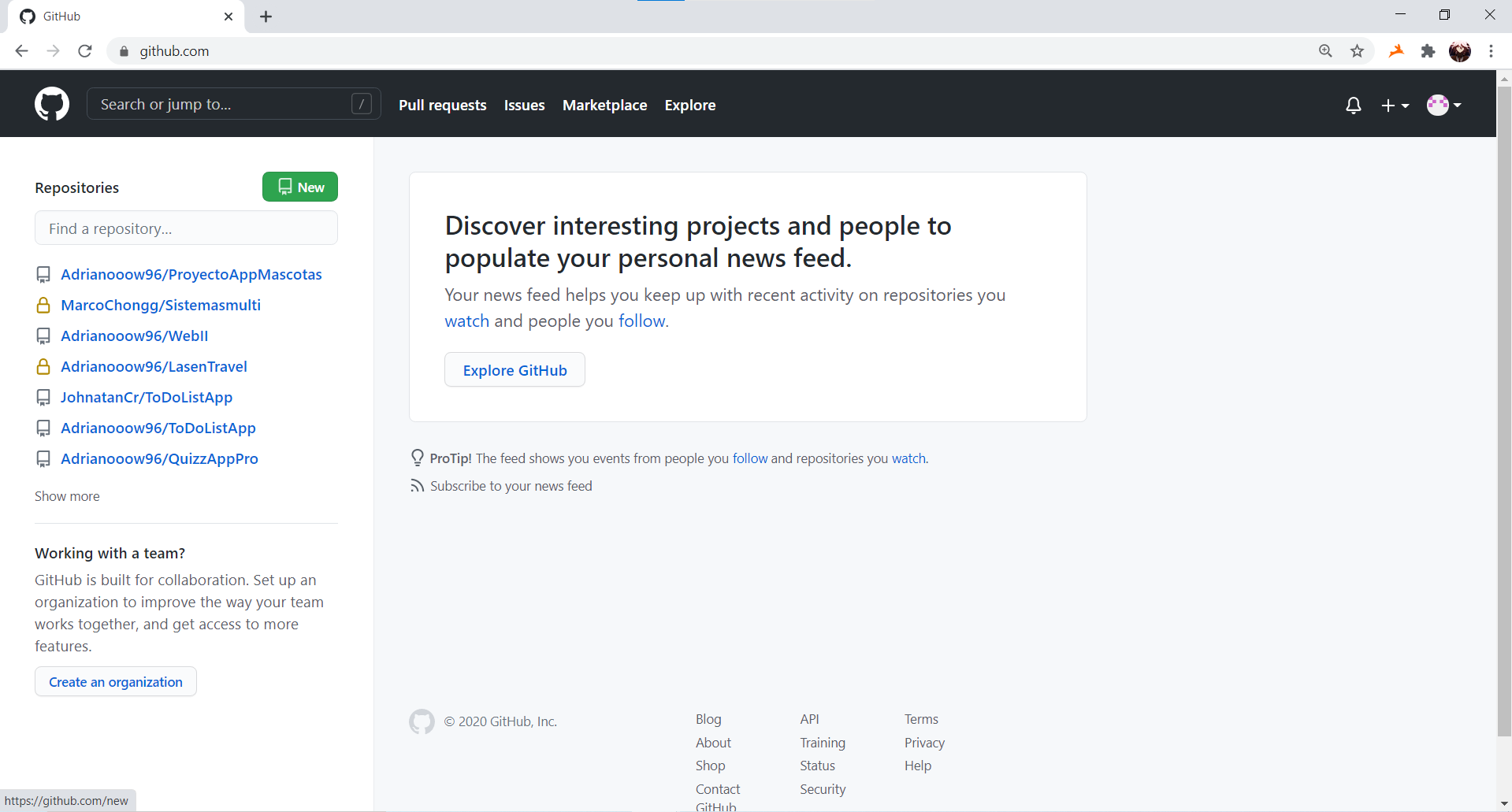
Procesos

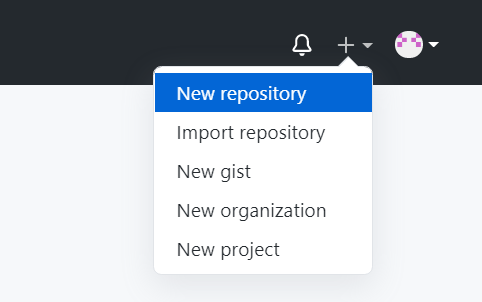
Repositorios

Como primer paso vamos a establecer el entorno de trabajo con un flujo de trabajo con Git, para poder tener un correcto manejo de las versiones de nuestro proyecto y un fácil acceso a nuestros archivos.

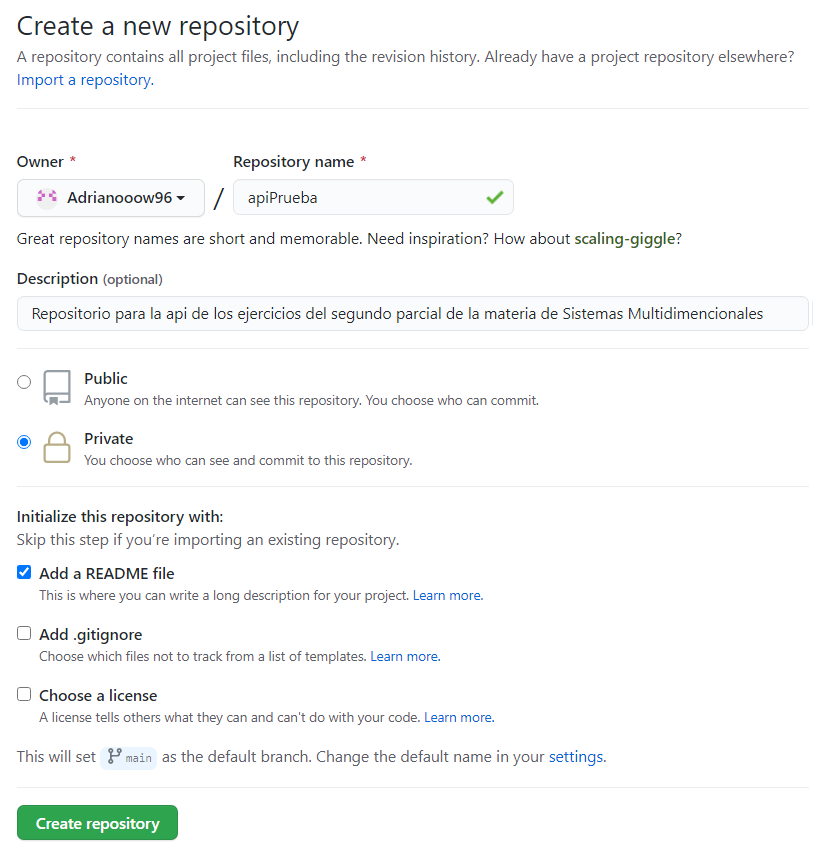
Creación de repositorios

Lo primero que deberemos hacer será crear los repositorios para el sitio web, la API y la base de datos. Para ello usaremos la plataforma llamada Github.

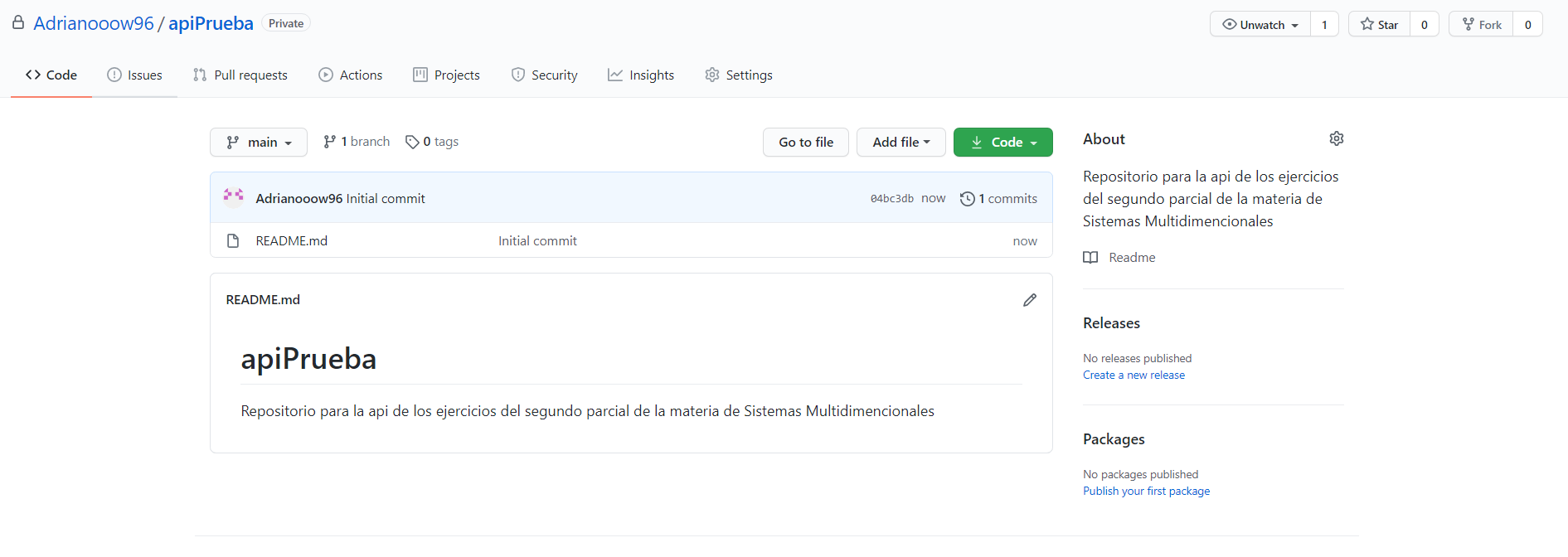
Con una cuenta de Github vamos a Nuevo Repositorio y esperamos a que nos cargue la siguiente pantalla.



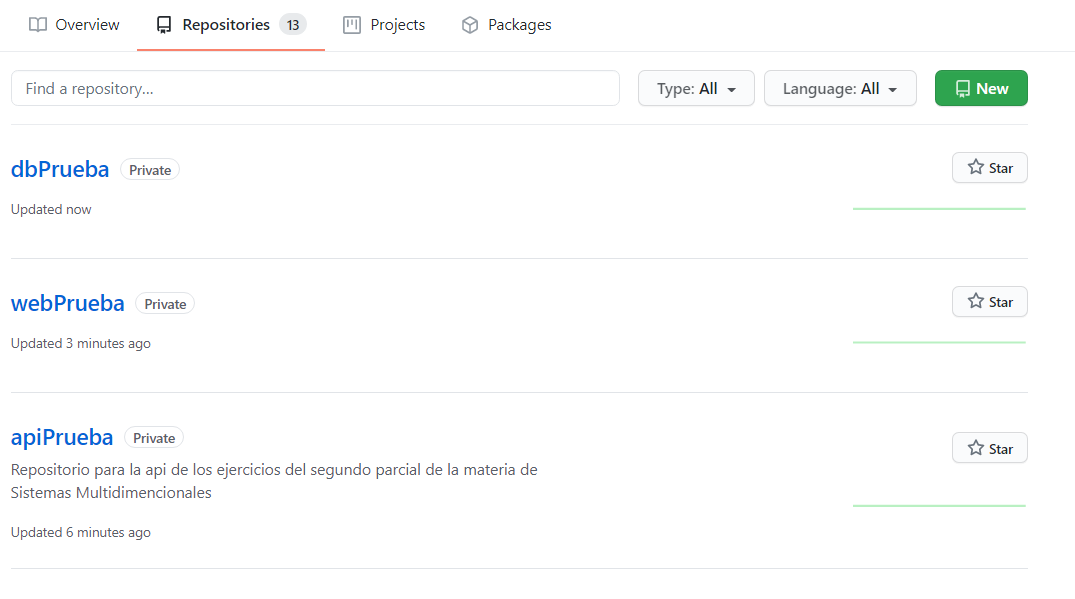
En la nueva pantalla llenaremos los datos solicitados, brindando un nombre al repositorio, una descripción opcional y otros detalles de preferencias:



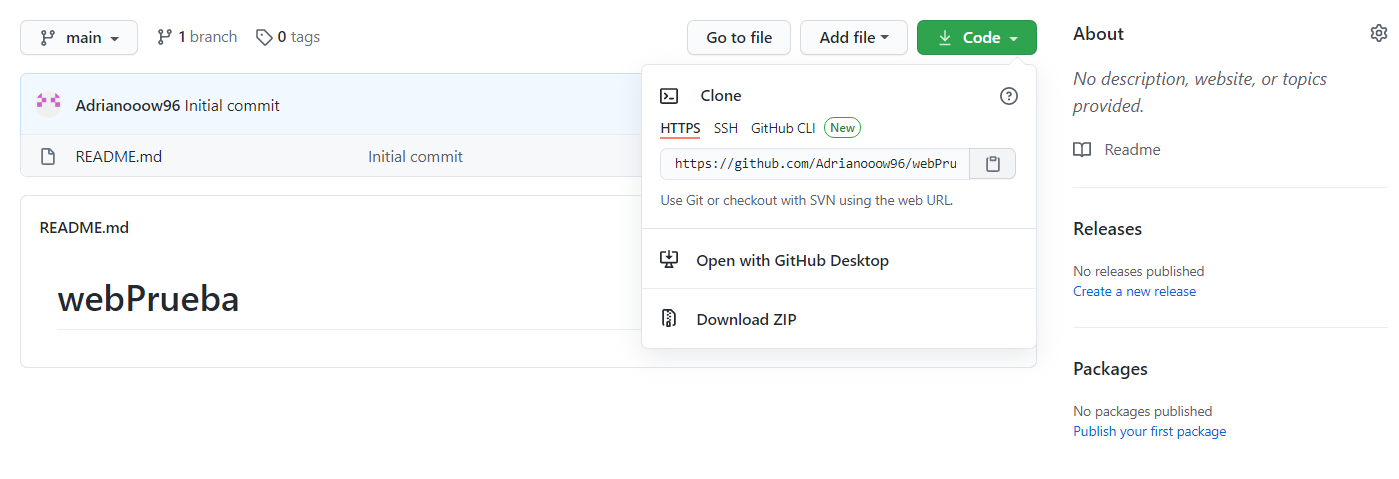
Finalmente damos a Create Repository para crear el repositorio con la configuración brindada. Si todo salió correctamente veremos la siguiente pantalla:

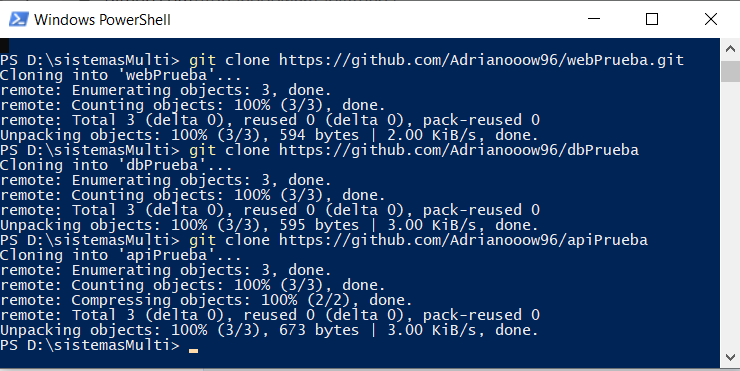


Repetiremos los pasos anteriores para el repositorio de base de datos y el de los archivos de la web.

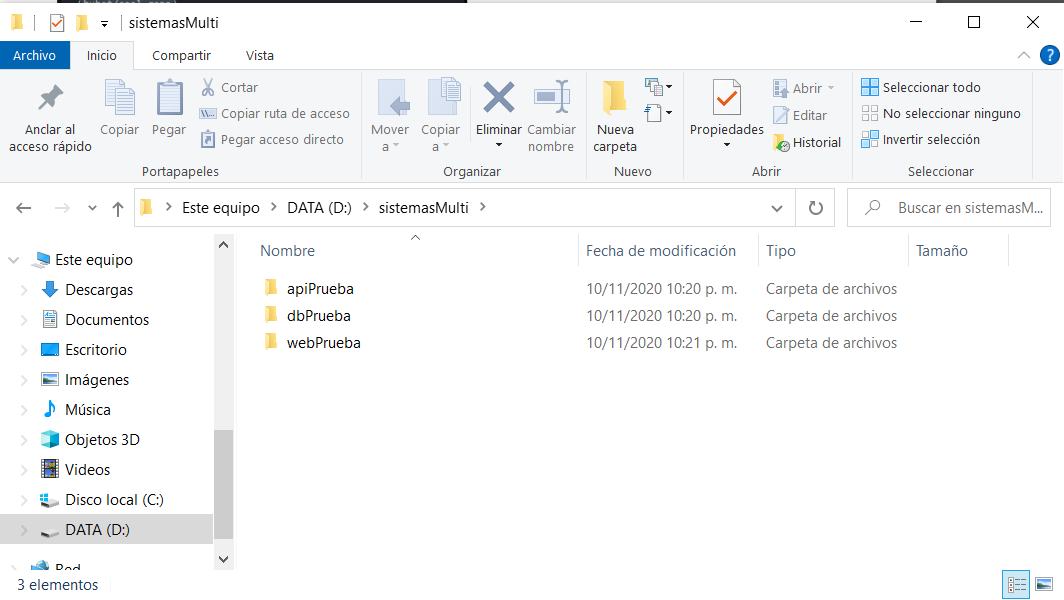


Clonación de repositorios en local

Una vez con los repositorios creados, procederemos a clonarlos para tener acceso y gestión de ellos desde nuestro dispositivo. Para ello ejecutaremos el comando git clone en la consola, ubicados en el directorio que deseemos. El enlace necesario es otorgado por cada repositorio en la siguiente sección: 

En la siguiente captura mostramos la ejecución exitosa de los comandos necesarios: 

Como resultado tendremos los directorios de los repositorios clonados, como podemos ver en la siguiente captura:

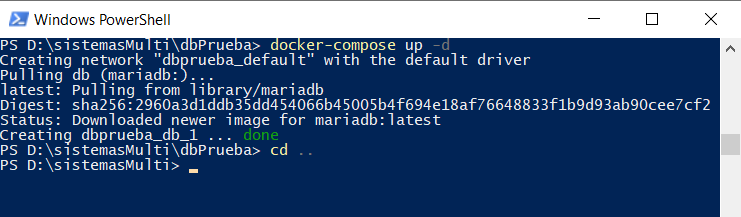


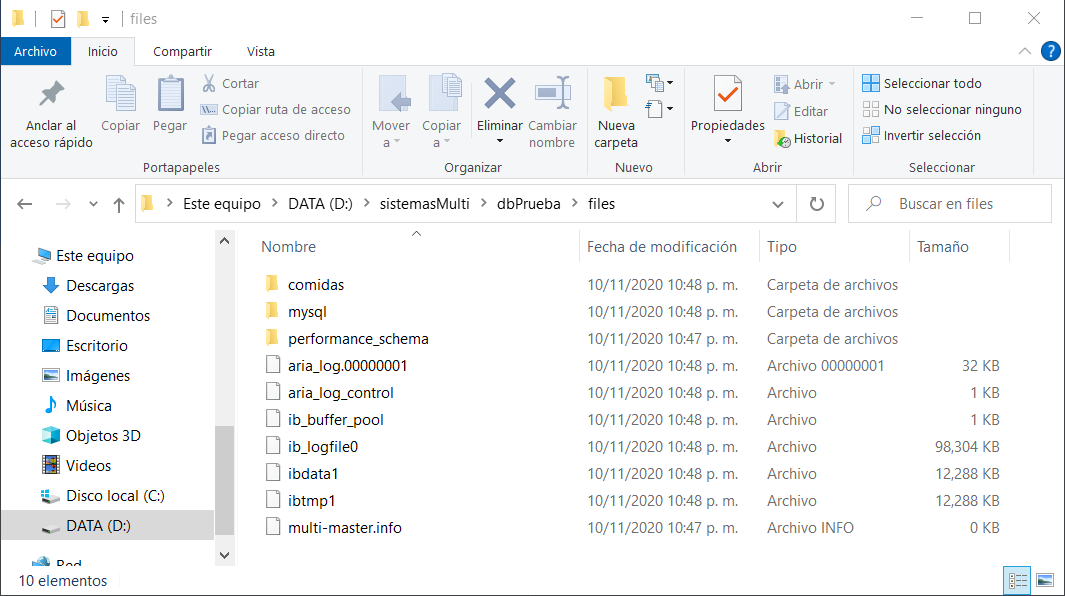
Creación de contenedores

Como siguiente paso, crearemos un archivo docker-compose que nos servirá para ambientar nuestros contenedores. Una vez creados los archivos en cada carpeta, procederemos a correr el siguiente comando en cada una de nuestras carpetas:

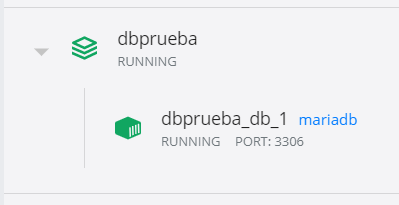
docker-compose up

Y una vez finalizado podremos visualizar los cambios en el directorio.

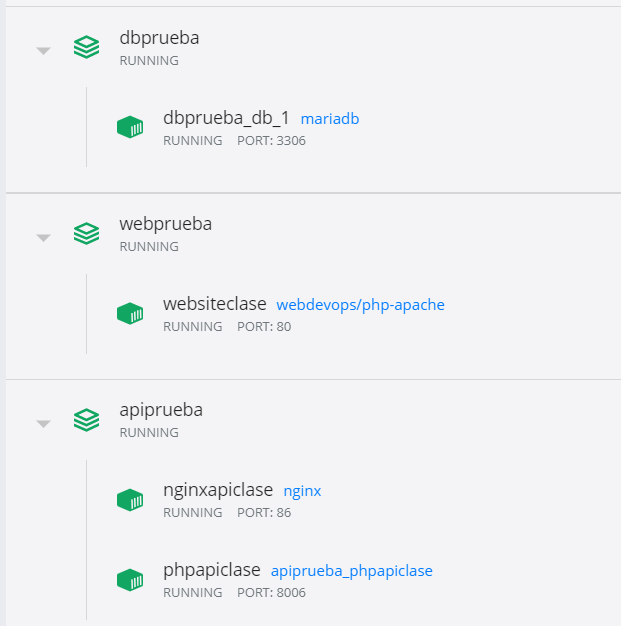




Y también podemos visualizar desde Docker que nuestro contenedor está funcionando:

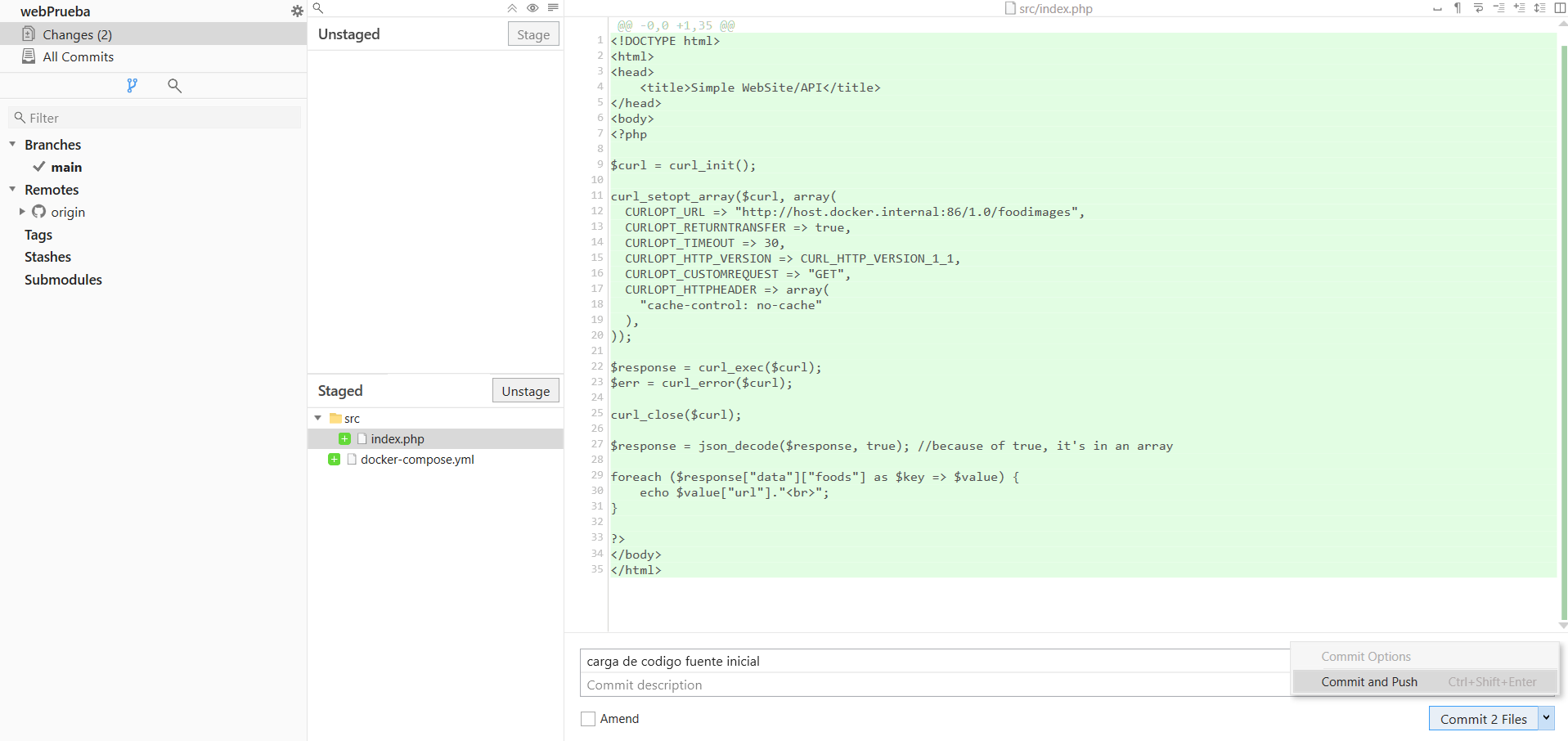


Haremos lo mismo en los otros repositorios para tener todos los contenedores listos.

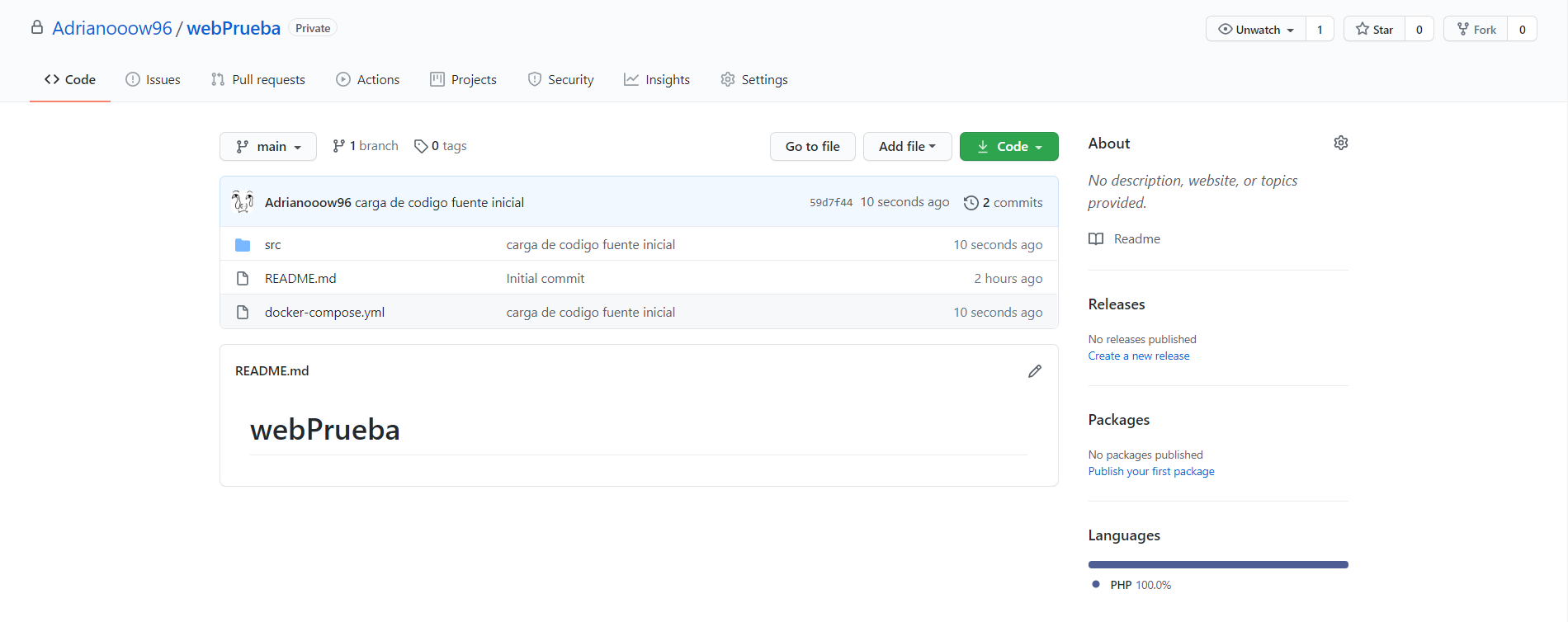


Carga de código fuente inicial

Seguido de ello vamos a utilizar una herramienta llamada Fork para hacer manejo de nuestros repositorios y cargar todos los archivos que hemos creado.

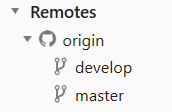


Y podemos ver que nuestros repositorios son actualizados con el código inicial:



Creación de ramas master y develop

Como último paso en esta etapa, vamos a crear una rama develop, que es en la que trabajaremos y una rama master para lanzar nuestro código cuando esté finalizado.



Nube

Una vez que tengamos todo el proyecto terminado, comienza la parte de pasarlo a la nube. Para ello vamos a utilizar LightSail, una plataforma web que nos servirá durante esta etapa y la siguiente.

Creación de instancias LightSail

Como primer paso, vamos a crear nuestras instancias para cada uno de nuestros contenedores. Este proceso ya lo hemos documentado en el archivo Tarea 1 – Parcial 2 – Gorocica Coral Adrian.

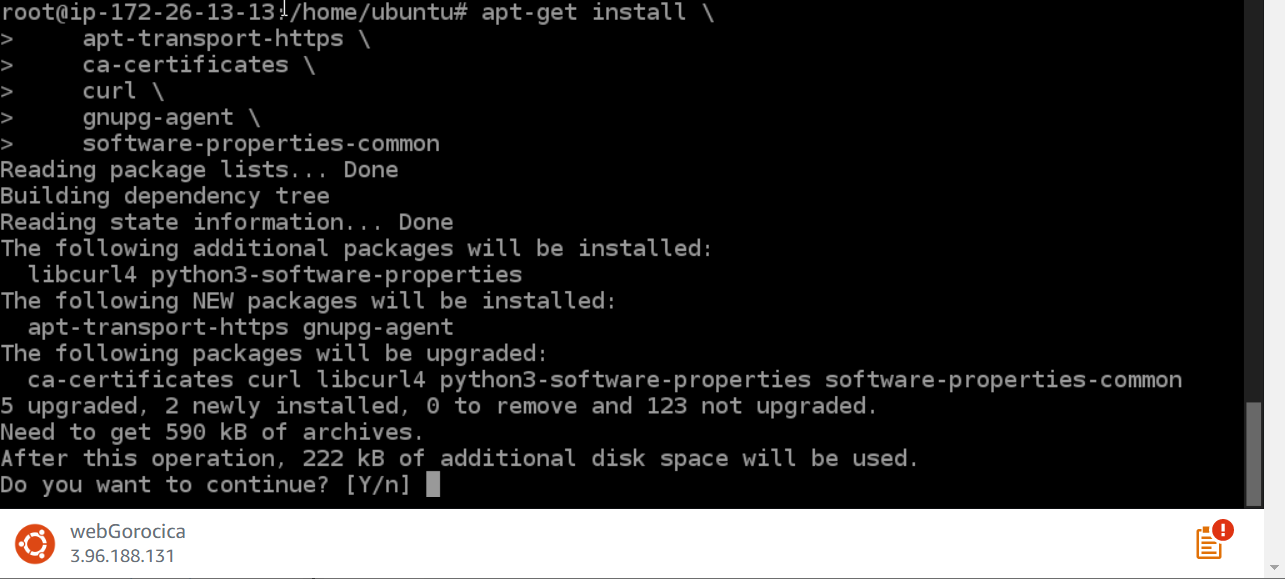
Al finalizar el proceso debemos poder observar las instancias:



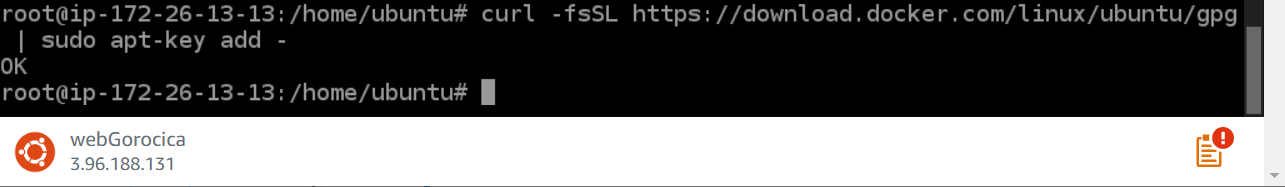
Instalación Docker en instancias

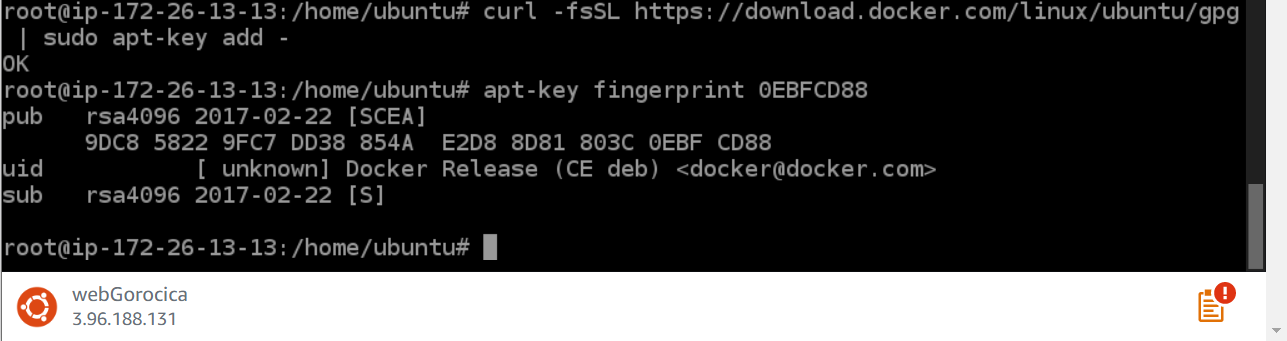
Como siguiente paso vamos a instalar Docker en las instancias creadas para poder tener las instancias ambientadas en la nube.

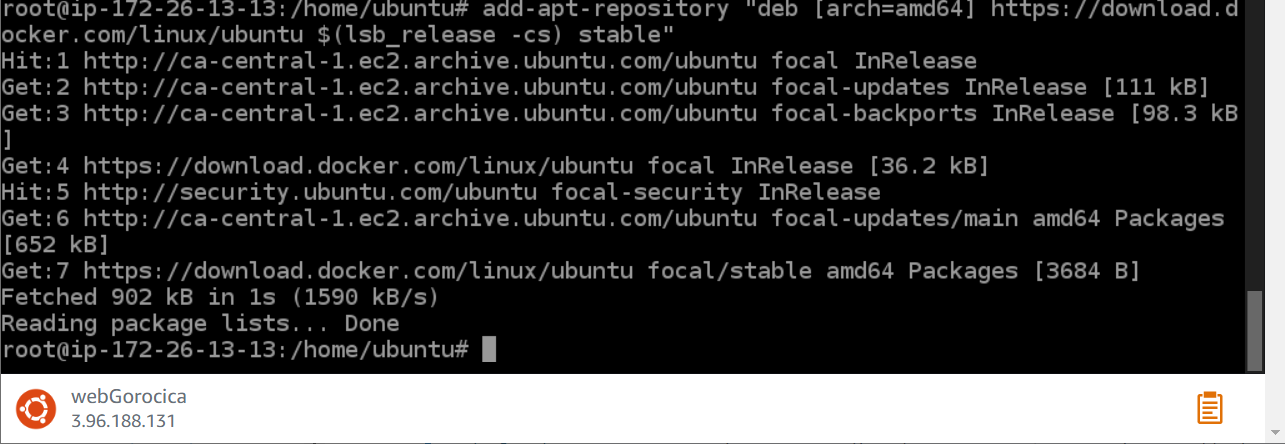
Para ello accederemos al siguiente enlace <https://docs.docker.com/engine/install/ubuntu/> y ejecutaremos lo siguiente en la consola de las instancias:

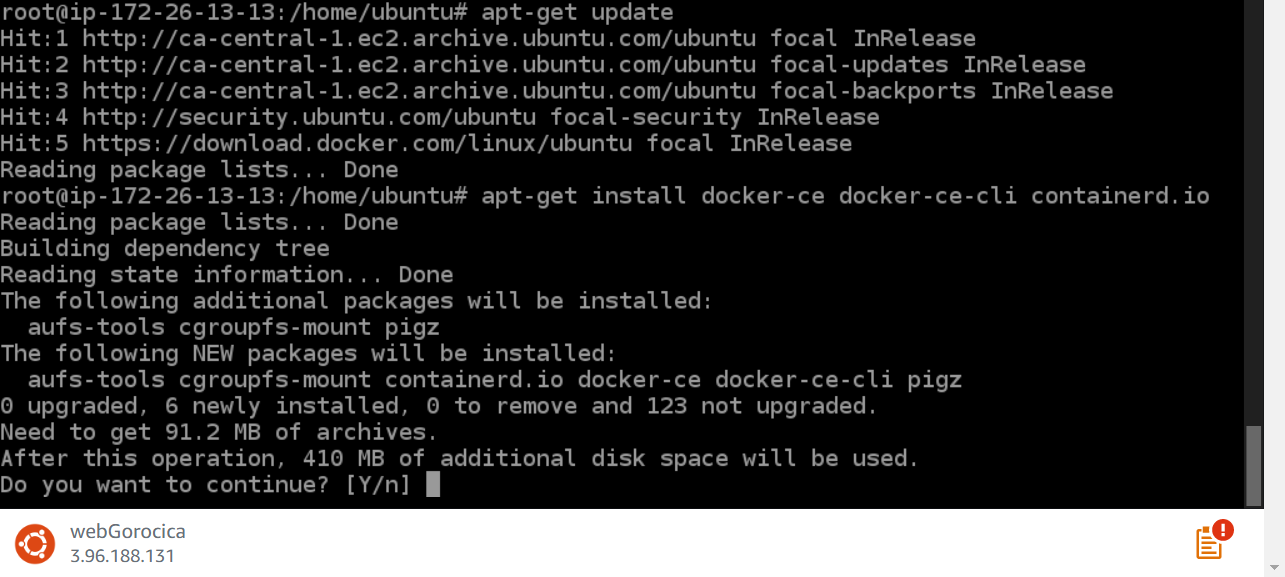


Aceptamos y dejamos que la instalación continúe.



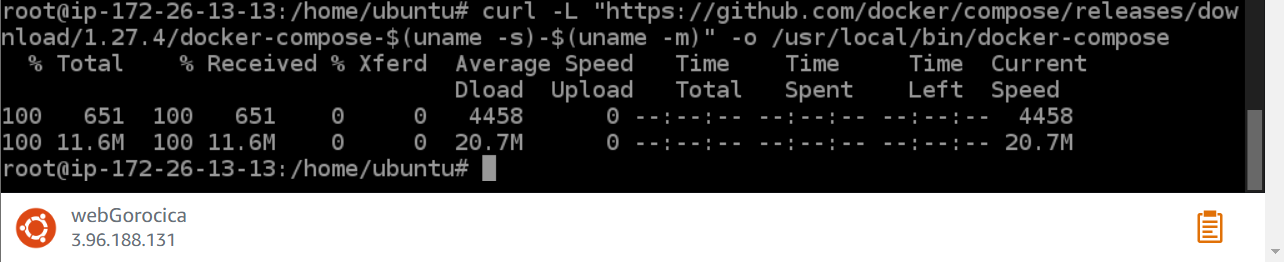






Aceptamos una vez más para finalizar.

Seguido de ello realizaremos la instalación de Docker Compose desde el enlace https://docs.docker.com/compose/install/ .

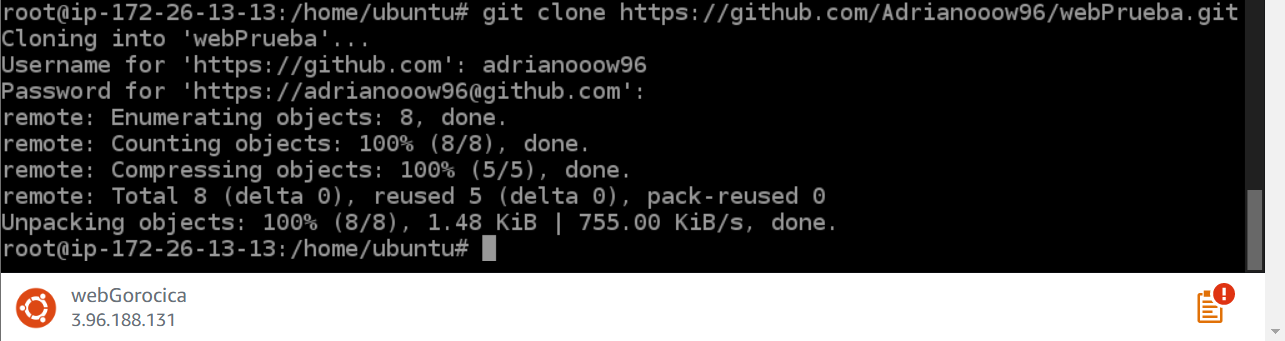




Podemos verificar la instalación escribiendo docker-compose:

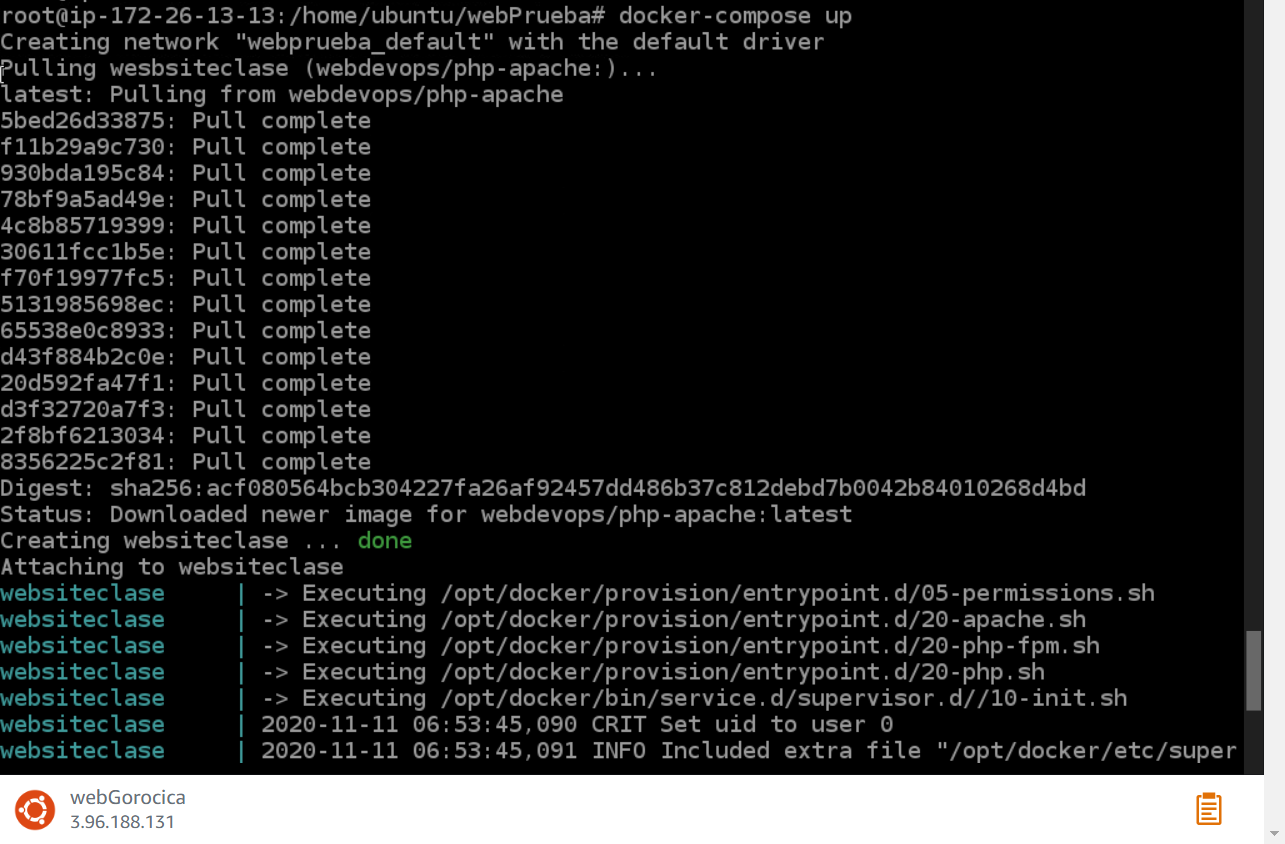


Clonación de repositorio

Una vez que esté instalado docker en nuestras instancias, procederemos a clonar nuestros repositorios de la misma forma que hicimos en el punto 3.1.2 pero ahora en la consola de nuestras instancias. 

Montaje en contenedor

Ya con nuestros repositorios clonados podemos continuar con el montaje de nuestros contenedores al igual que en el punto 3.1.3, pero ahora en la consola de nuestras instancias:



Configuración de aplicaciones

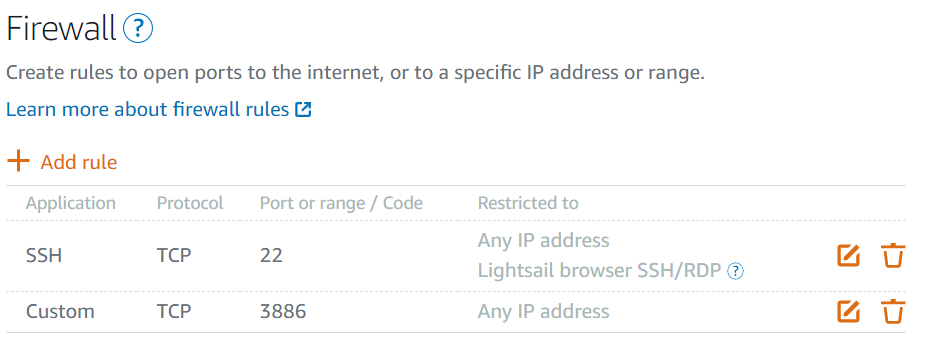
Finalmente terminamos de ambientar nuestros contenedores como lo hicimos de manera local con los archivos que no se cargan al repositorio. Y con eso tendremos todo nuestro proyecto en la nube listo para usar.

Seguridad y redundancia

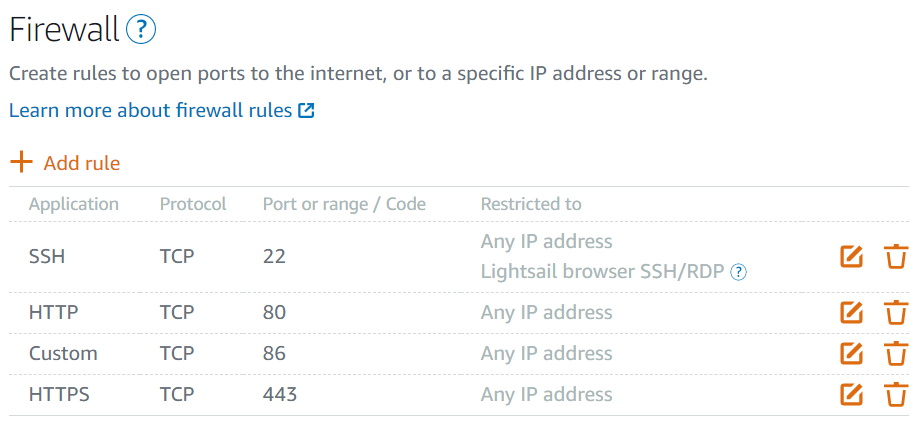
Después de tener nuestra aplicación montada en la nube, hace falta configurar la seguridad y redundancia.

Configuración de firewall

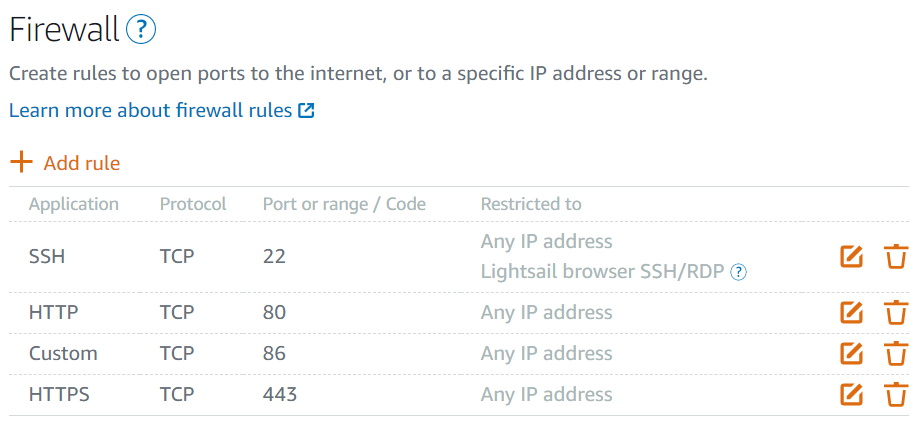
Primero configuraremos nuestras reglas de firewall para que sólo exista acceso a lo que realmente necesitamos de cada instancia.

Para la instancia de la base de datos quedaría de la siguiente forma:

Para la instancia de la API quedaría lo siguiente:

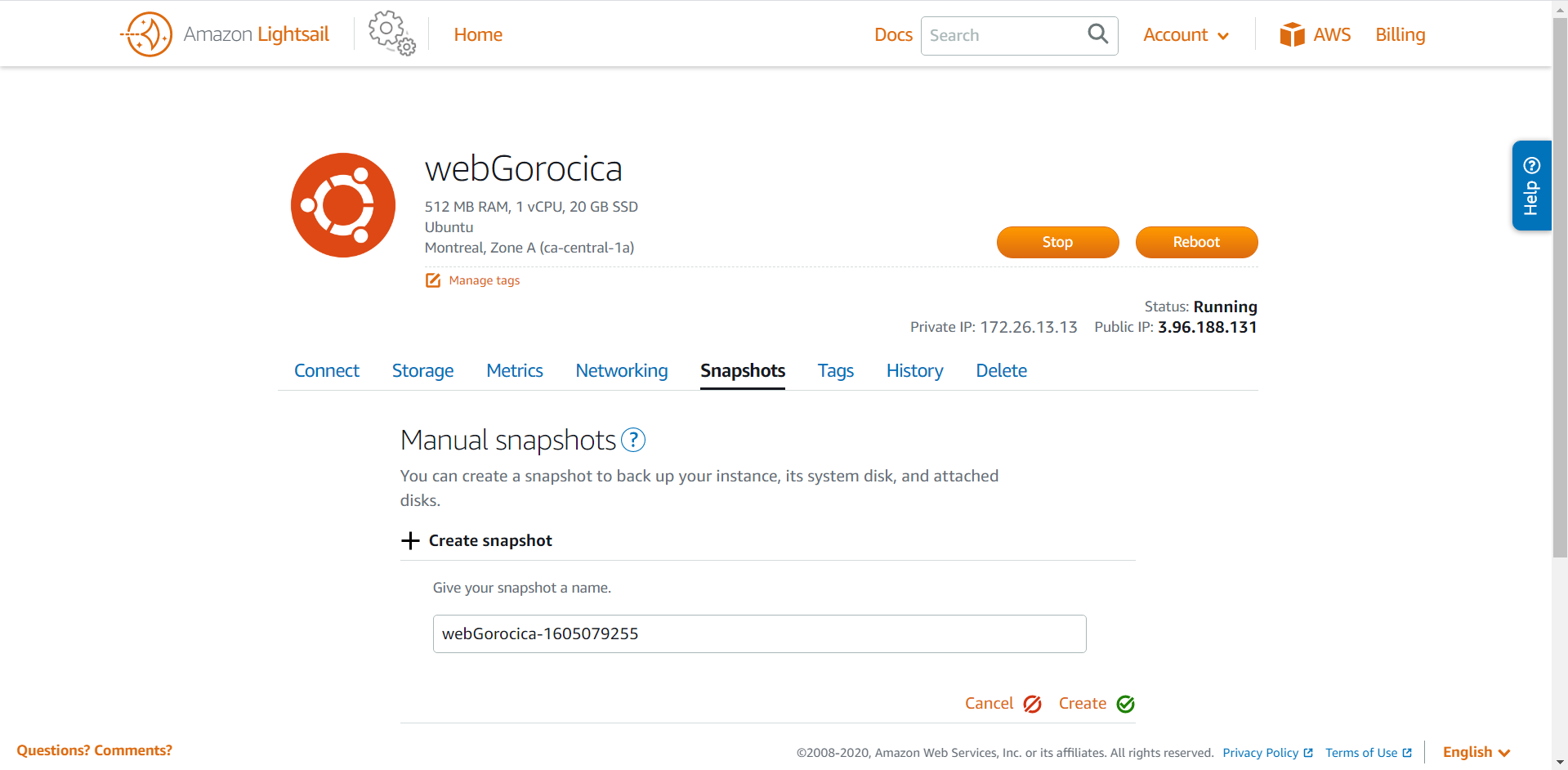


Y para la instancia de la página web



Creación de instantánea de web

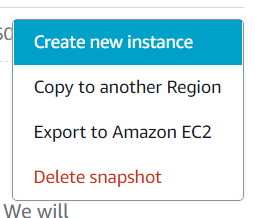
Para poder hacer redundante nuestro sistema, necesitamos copiar la instancia que tenemos de nuestra instancia web. Para ello iremos al apartado Snapshot ubicado dentro de la gestión de nuestra instancia:



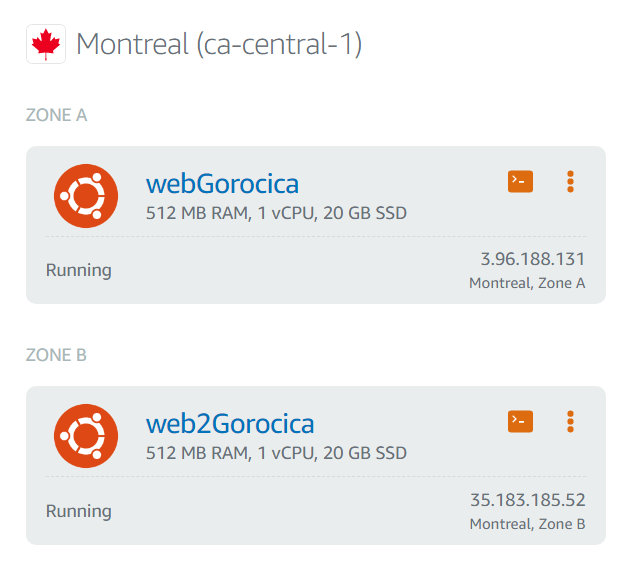
Esperamos a que se termine de crear el snapshot para poder avanzar al siguiente punto.

Creación de segunda instancia de web

Una vez lista nuestra instantánea, podemos ir a sus opciones para encontra la opción Create new instance, que nos servirá para crear una instancia basada en la instantánea, lo que nos es práctico ya que necesitamos una copia de nuestra instancia original.



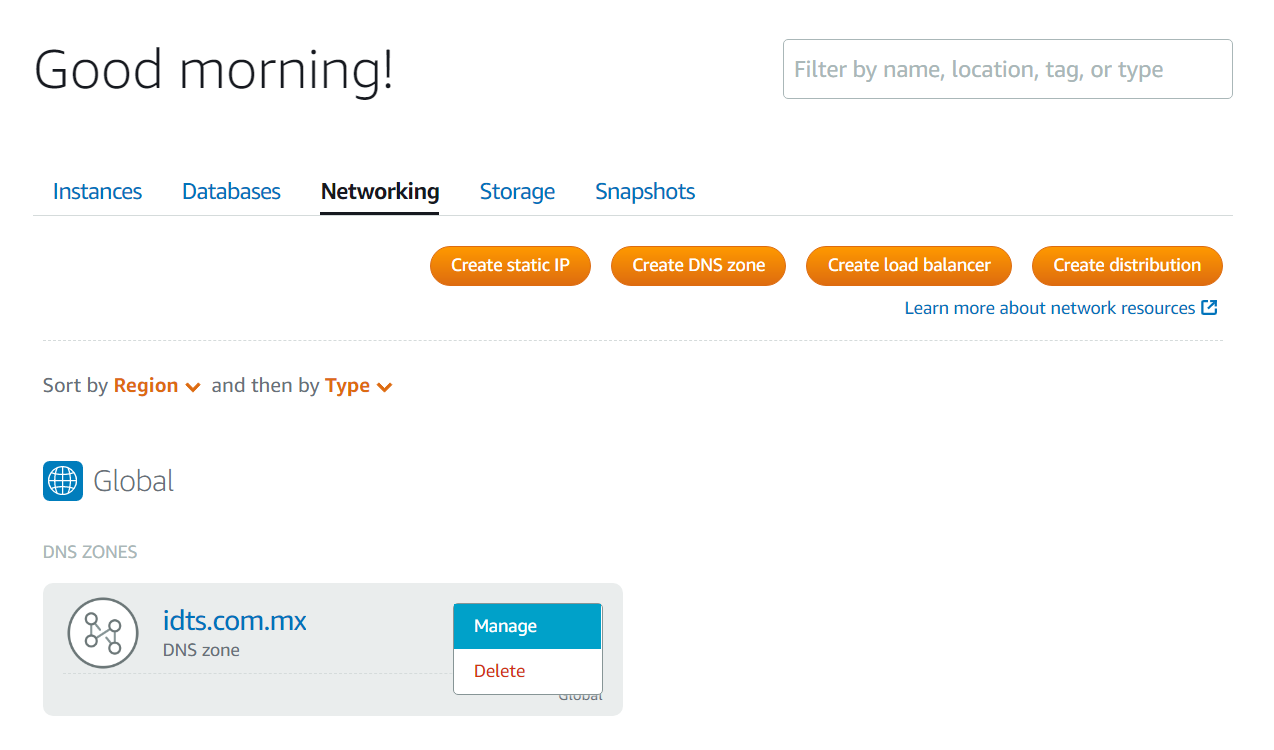
Al finalizar veremos las dos snapshots coexistiendo como en la siguiente captura:



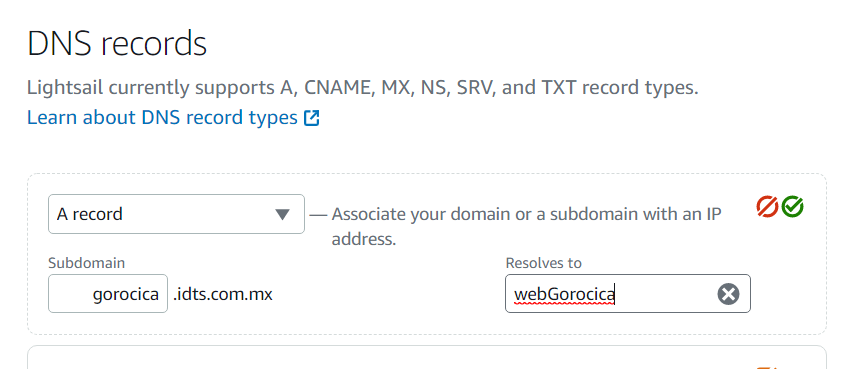
Creación de subdominio para web

Como siguiente paso vamos a crear un subdominio para acceder a la web que estamos desarrollando.

Para ello accedemos al apartado de Networking dentro de LightSail, accedemos a la configuración de la DNS a la que le vamos a crear un subdominio:

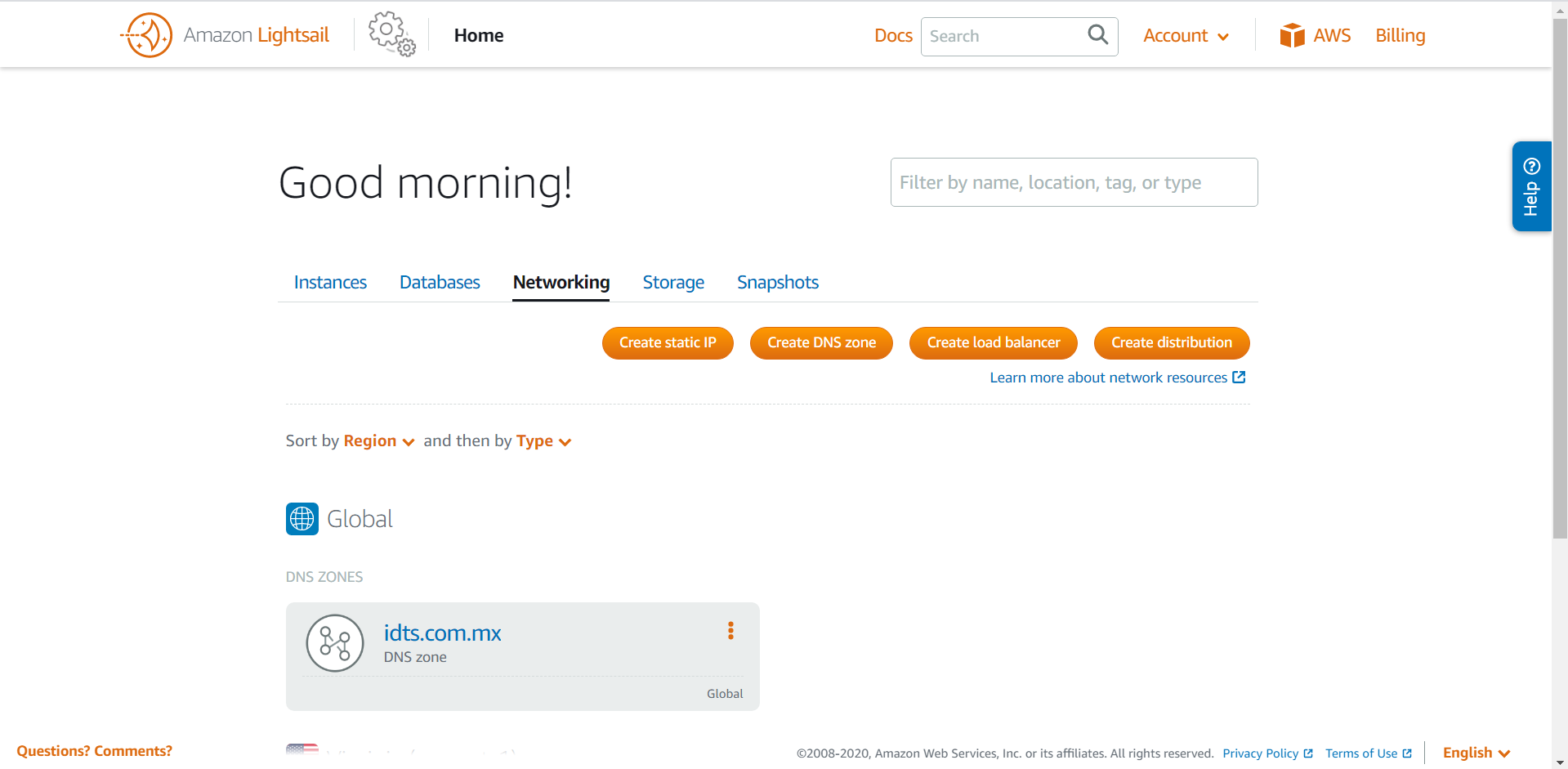


Y creamos un nuevo registro con el subdominio que deseemos:

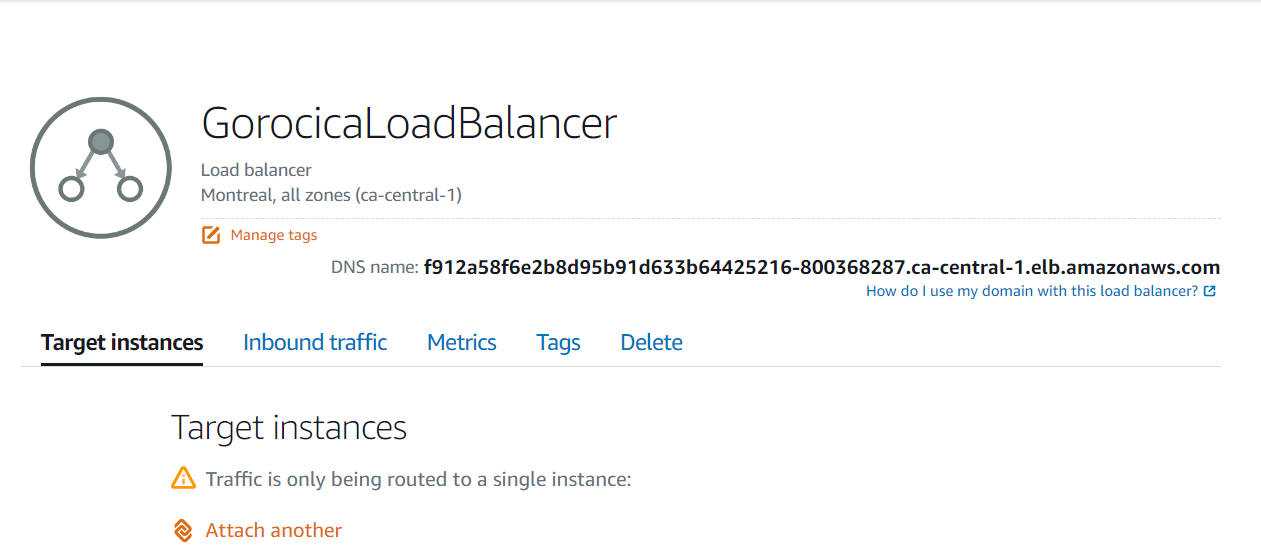


Creación de balanceador de carga

Para uso de la redundancia y el buen flujo de nuestra web, vamos a crear un balanceador de carga que nos permita que nuestra web no se sature. Para ello nos ubicamos en el mismo apartado de Networking ubicado en nuestra pantalla principal de LightSail:

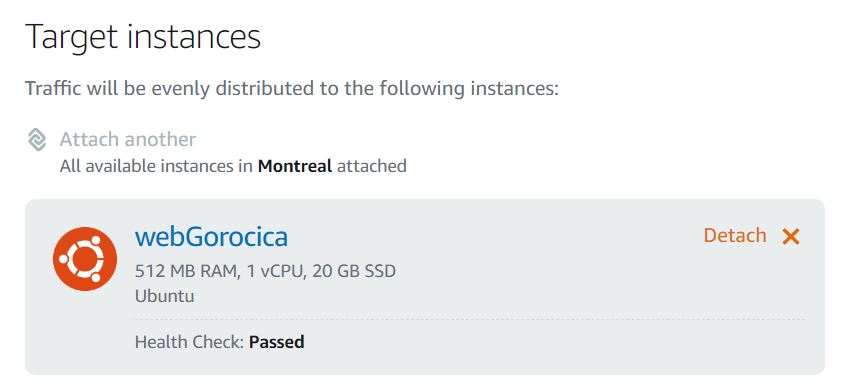


Y nos vamos a Create load balancer, le asignamos un nombre y la región en la que se encuentre nuestra web, que en este caso es Montreal, y la siguiente página que nos muestre será la configuración del balanceador de carga.

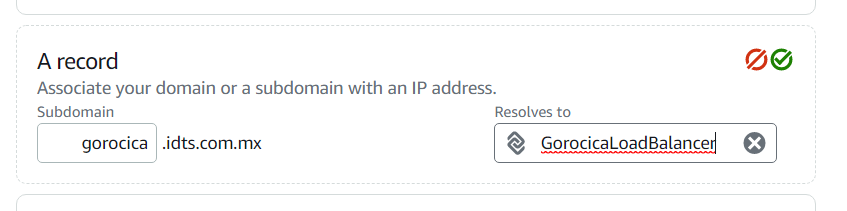


Asignación de instancias a balanceador de carga

En la pantalla de configuración del balanceador de cargas, podremos asignar instancias a las que redirigirá, ahí seleccionamos nuestra instancia de web y la segunda instancia, para que el balanceador se encargue de redirigir a la que considere apropiada sin problemas de saturación a alguna de las dos.

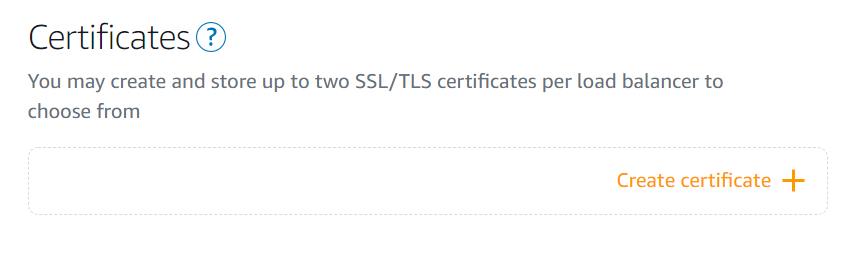


Finalmente regresamos a nuestro subdominio para cambiar su dirección de nuestro webGorocica para el balanceador de carga y todo esté correctamente configurado:

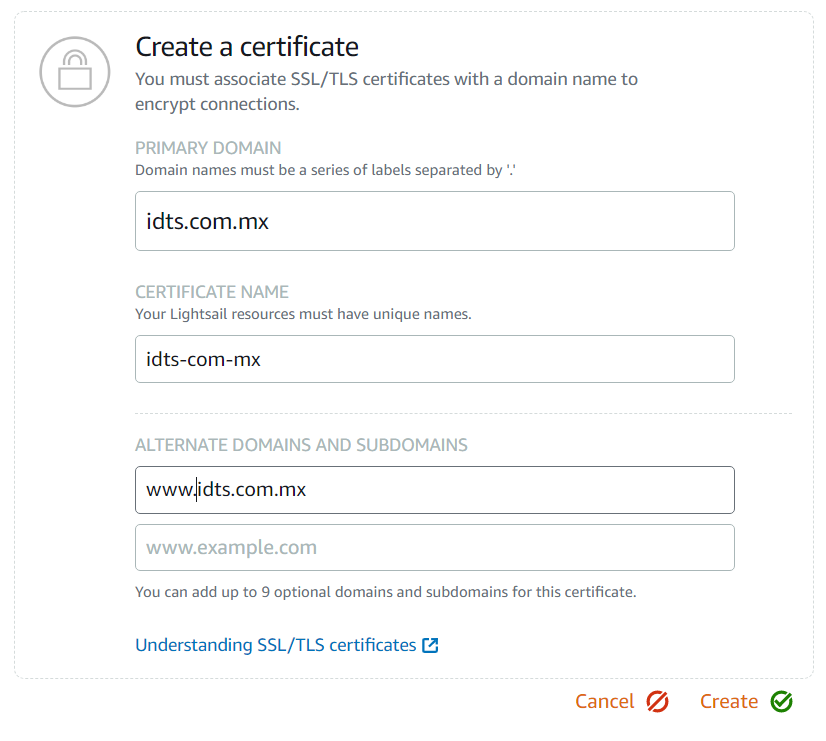


Creación de certificado SSL

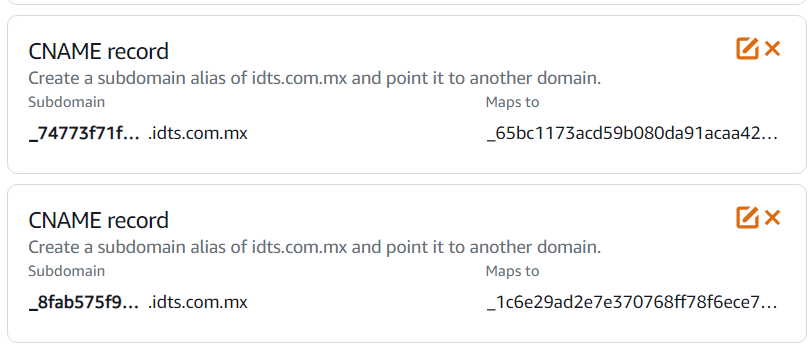
Como último paso en la seguridad, se debe contar con un certificado SSL en nuestro balanceador de carga. Para ello accedemos una vez más a la configuración de nuestro balanceador de carga y nos dirigimos a la sección de Inbound Traffic y en el apartado de Certificates accedemos a Create certificate:



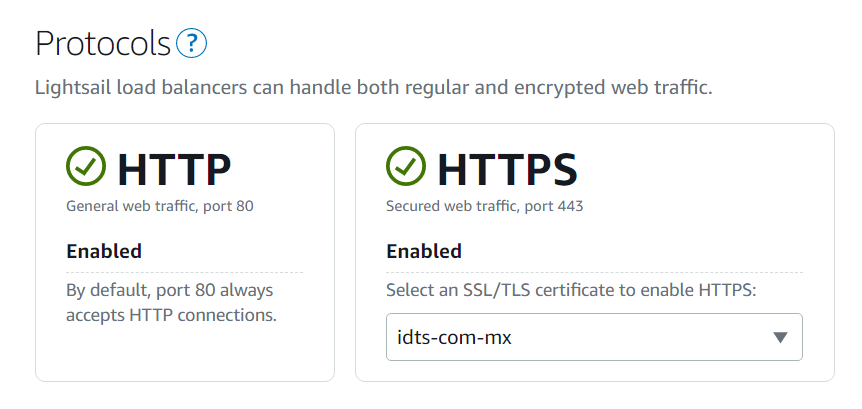
Escribimos el dominio principal para crear el certificado:



Seguido de ello tendremos que crear registros CNAME para hacer válido el certificado desde la configuración del dominio de la misma forma que creamos un subdominio:



Si todo salió bien, el protocolo HTTPS nos debería dejar seleccionar el certificado creado para que quede en uso:



Resumen

Con esta práctica pudimos observar lo sencillo que es utilizar las herramientas modernas para el desarrollo seguro de aplicaciones web así como su mantenimiento, basándonos en el diagramado realizado con anterioridad. De no haber sido por el diagrama o por las herramientas usadas, el proceso hubiera sido más complejo, más tardado y más propenso a fallos.