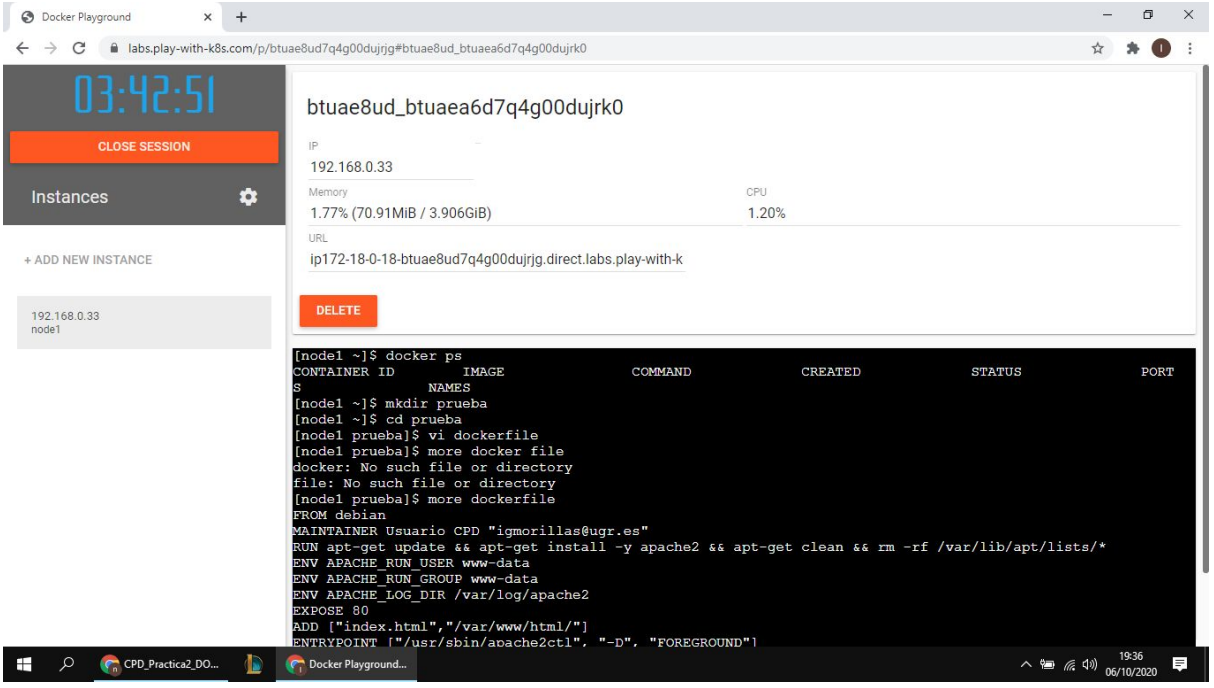
Nombre: Ignacio Morillas Padial

CPD_Practica_2. Docker II

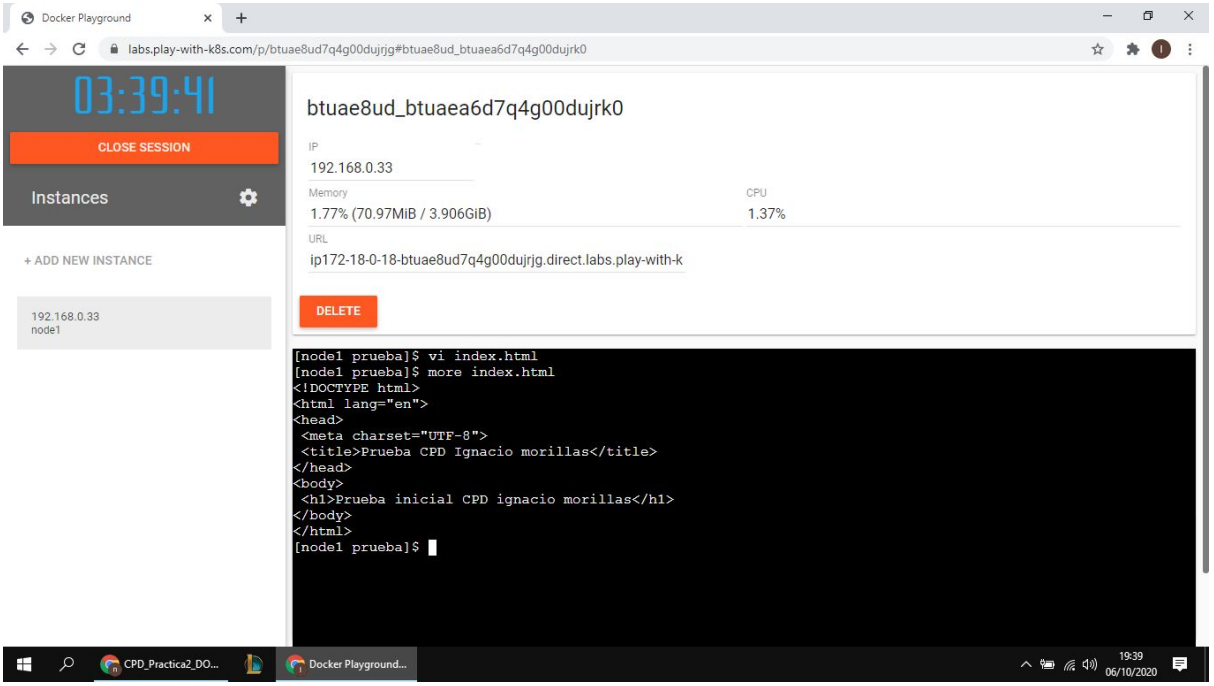
Crear una imagen personalizada de Docker con Apache de forma que cuando se acceda a http://localhost:8888 aparezca vuestro nombre (editando el index.html) (según apartado I).

1. **Creamos el directorio pruebacpd con mkdir.**

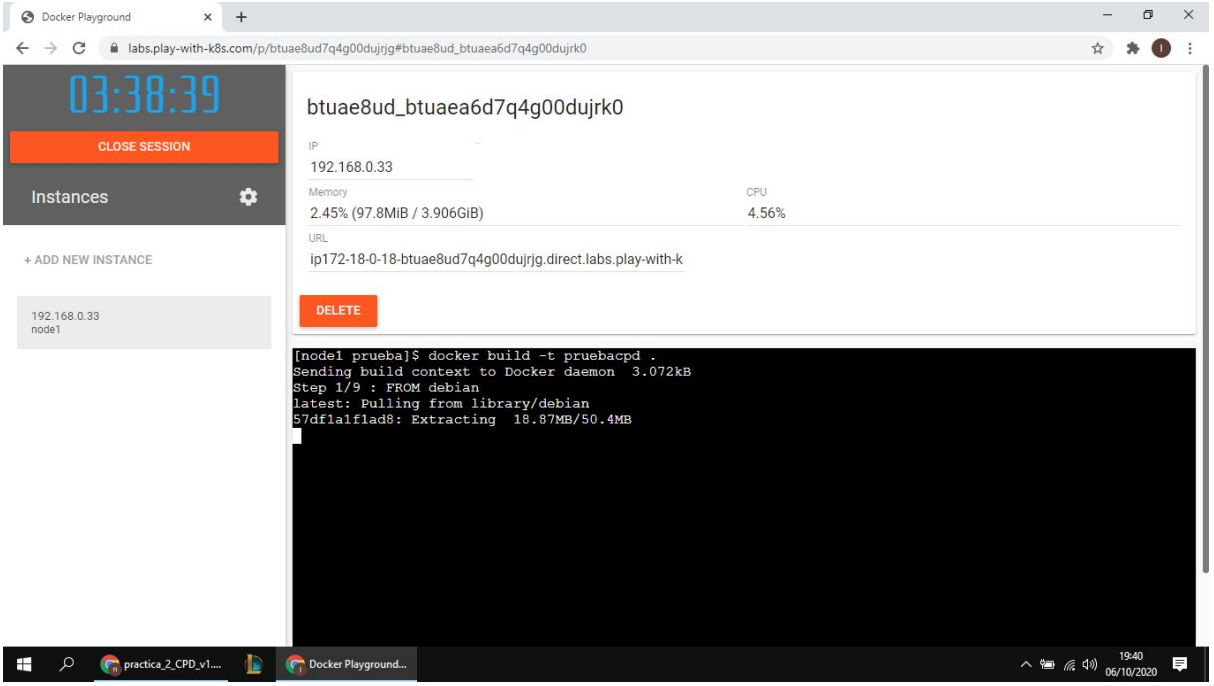
2. **creamos el fichero docker file dentro de pruebacpd:**



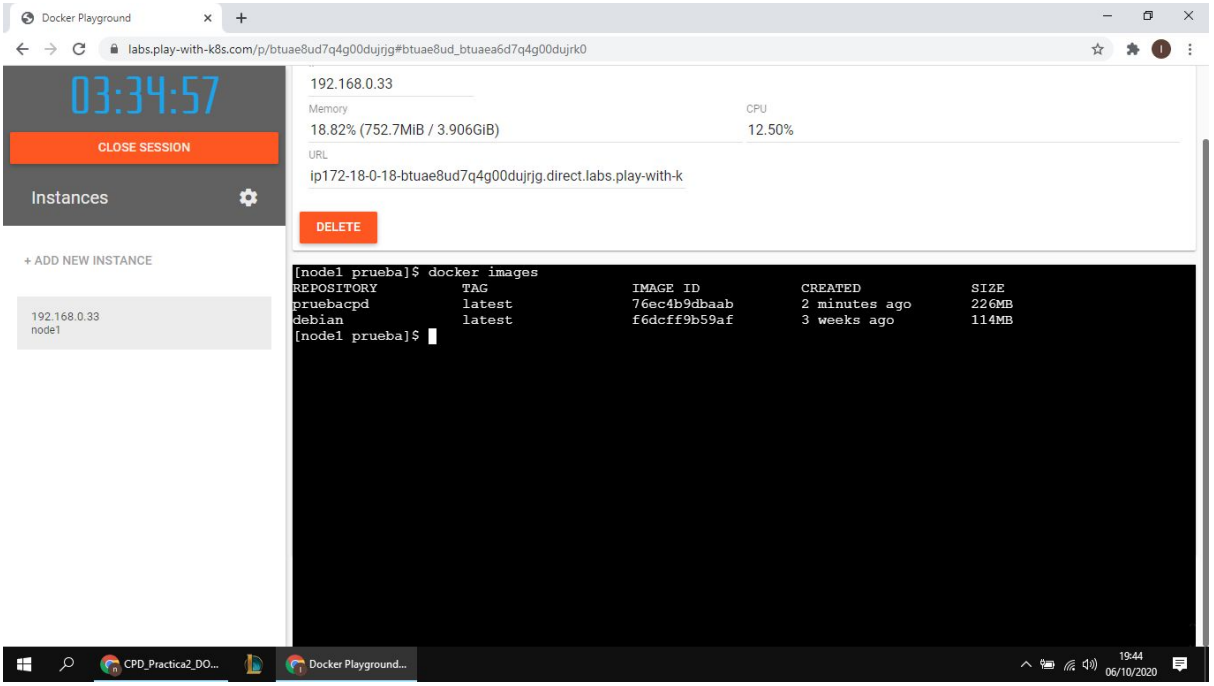
3. Creamos un fichero index.html:



4. Creamos nuestro contenedor:



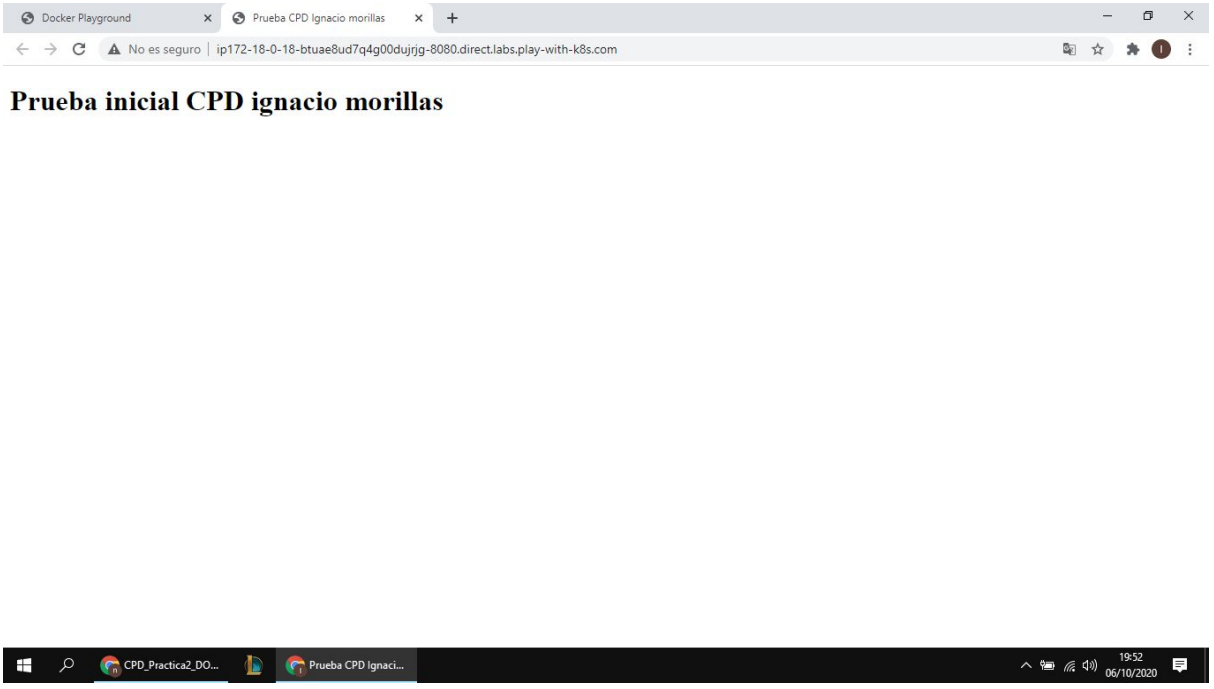
Vemos que se ha hecho correctamente



5. **Ejecutamos el docker:**

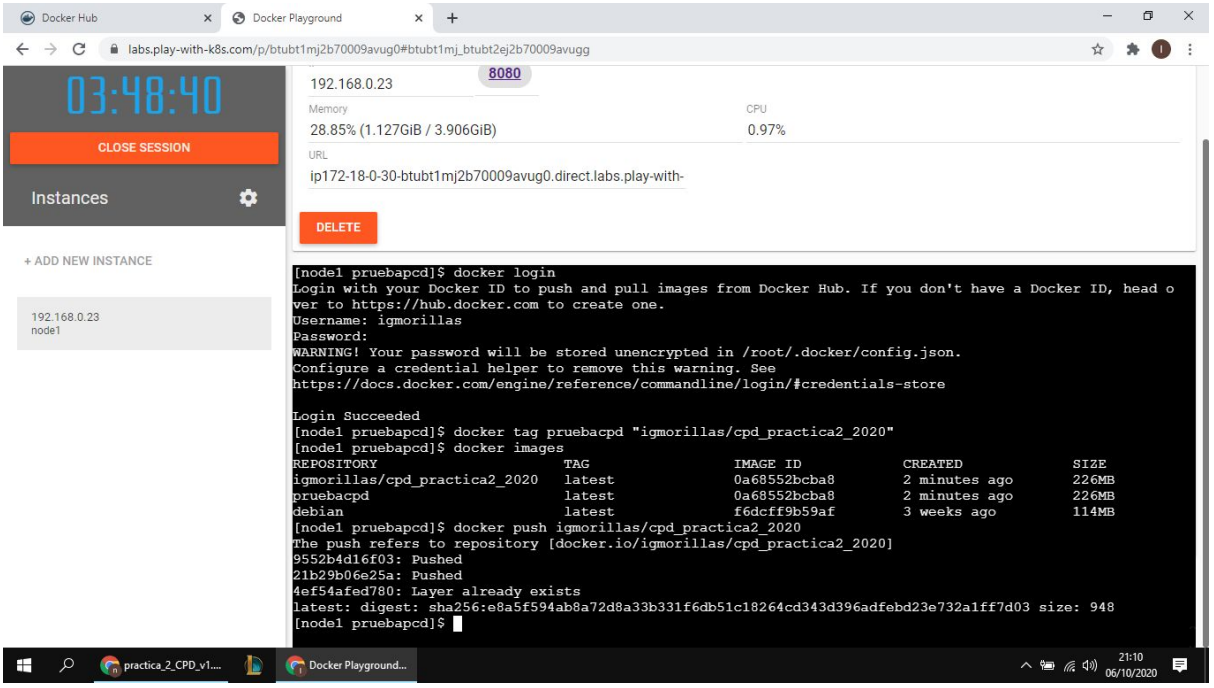
   docker run --name x1 -p8080:80 -d pruebacpd

   Accedemos a 8080 y vemos que se ha modificado correctamente index.html y que se ha creado todo correcamente.

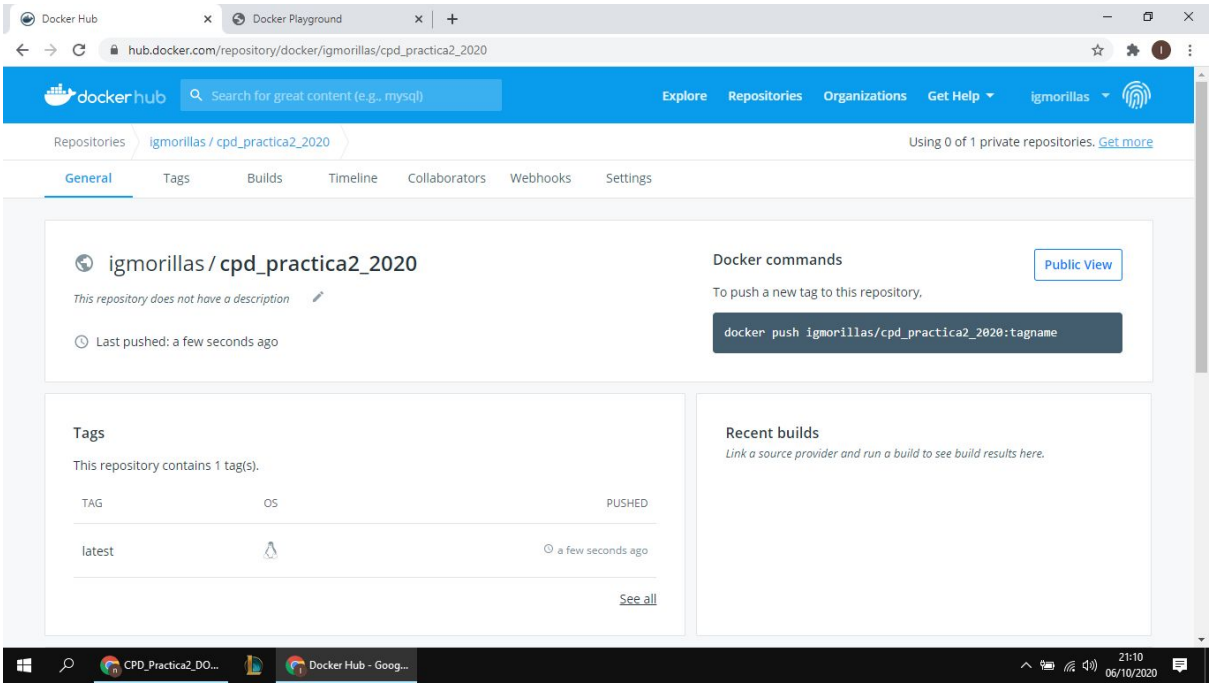


A partir de los pasos explicados en el apartado II subir la imagen a hub.docker.com e indicar el nombre de la imagen creada.

1. Nos logueamos con docker login
2.Con tag le ponemos un alias al contenedor acorde al repositorio en la web
3.subimos la imagen con docker push “igmorillas/cpd_practica2_2020”

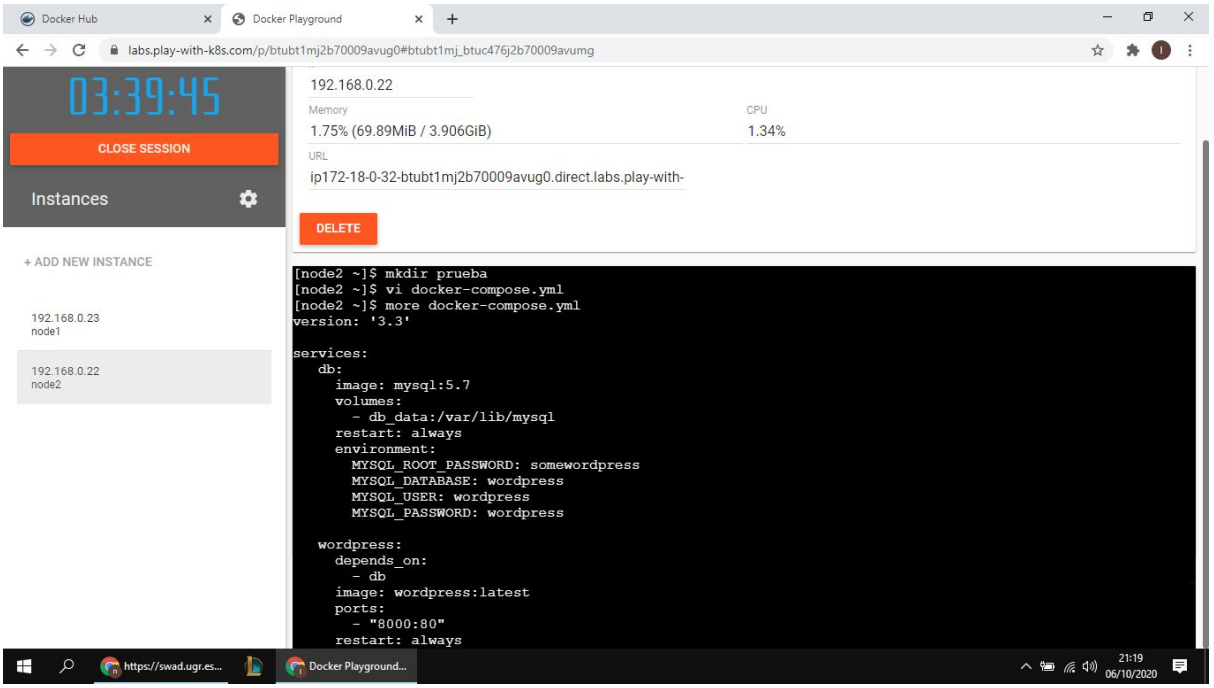


4.Vemos que se ha producido una actualización del repositorio

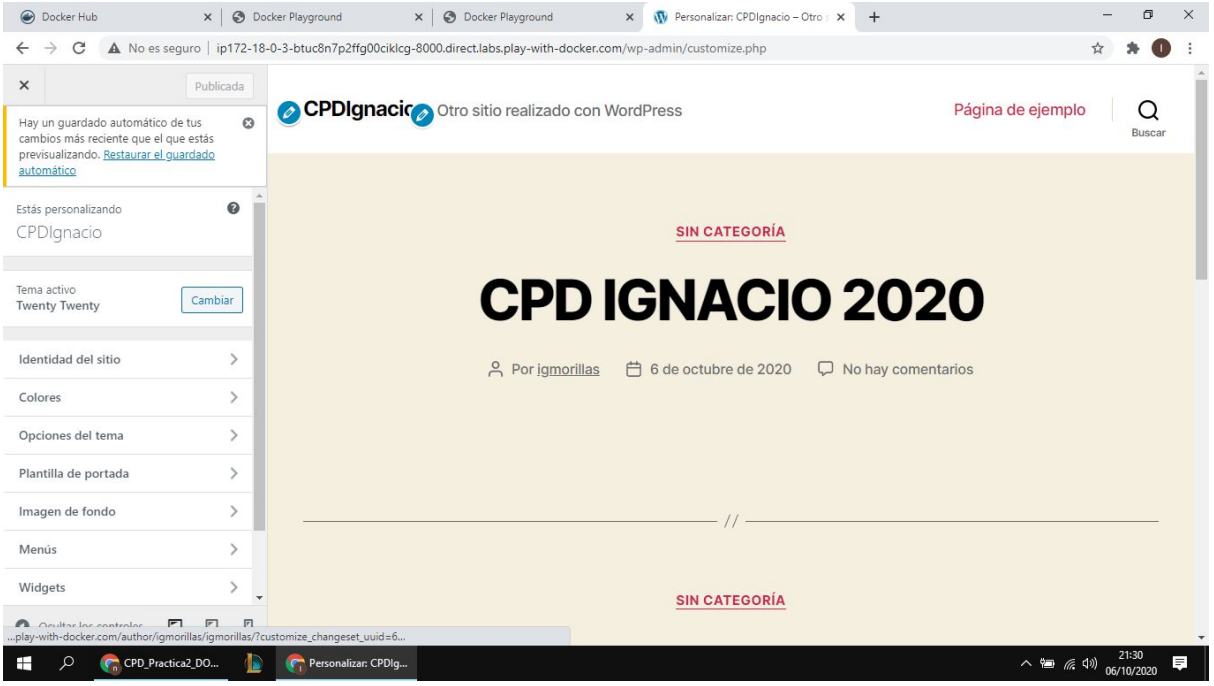


Según el apartado III, una vez desplegado el servidor Wordpress, editar la página principal para que aparezca el nombre del usuario y realizar una captura de pantalla.

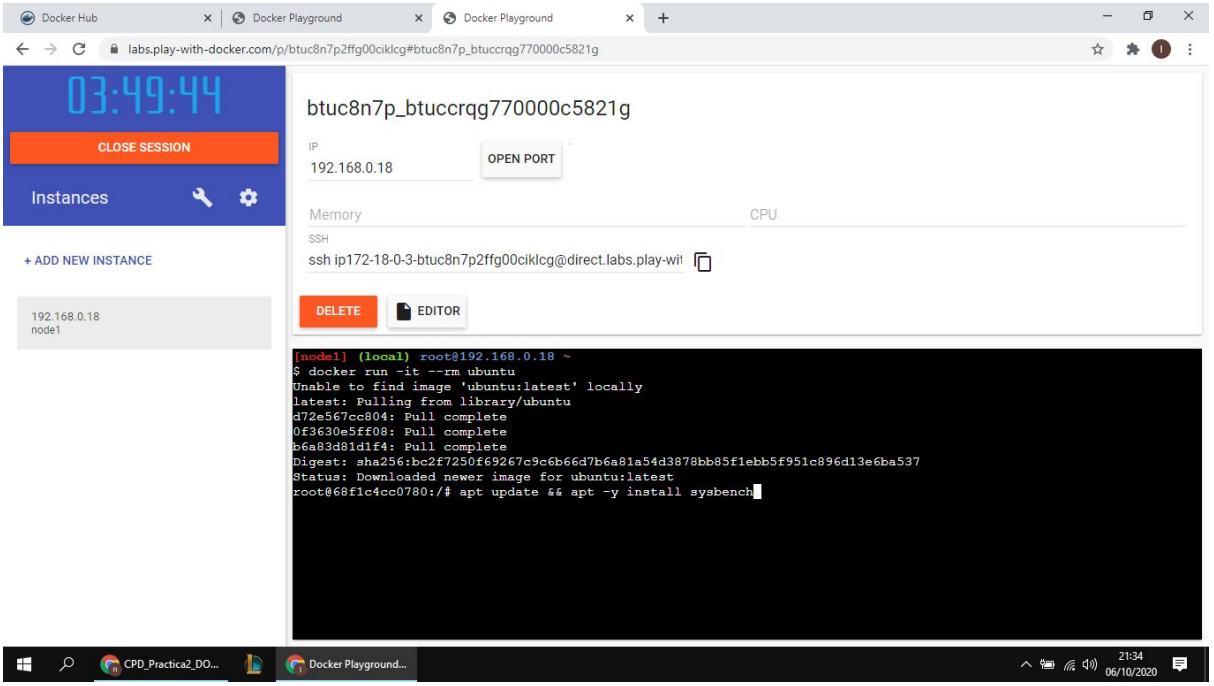
1. Creamos el archivo .yml



2. Accediendo a 8080 y tras registrarnos y demás finalmente llegamos a la parte final ya con la alteración hecha



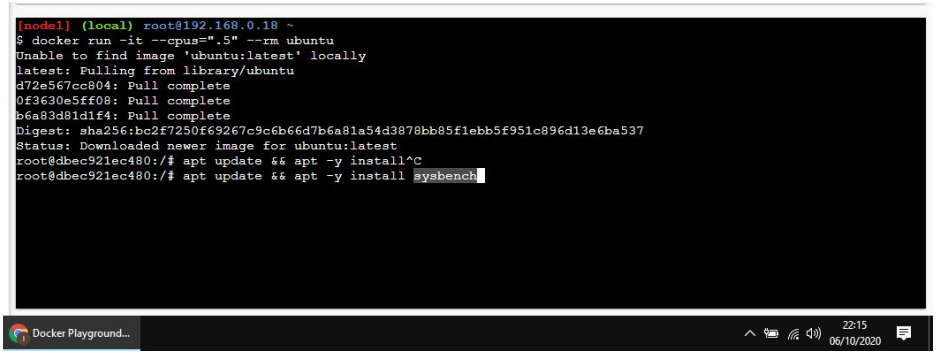
Siguiendo los pasos del apartado IV, ejecute el test que permita evaluar el tiempo de ejecución de un benchmark, y comprobar cómo podemos aumentar o reducir la CPU dedicada y por tanto el tiempo de ejecución. Prepare un contenedor con dicho experimento, súbalo a hub.docker.com e indique en el documento los pasos que realiza para el experimento y los tiempos de ejecución obtenidos.



PASOS:

1. **docker run -it --cpus=".<numerocpu>" --rm ubuntu (indicamos el numero de cpu)**
2. **apt update && apt -y install sysbench(actualizamos y instalamos el benchmark)**
3. **sysbench --test=cpu --cpu-max-prime=20000 run (Ejecutamos el benchmark)**

EJEMPLO CON 5



Obtenemos dicho resultado

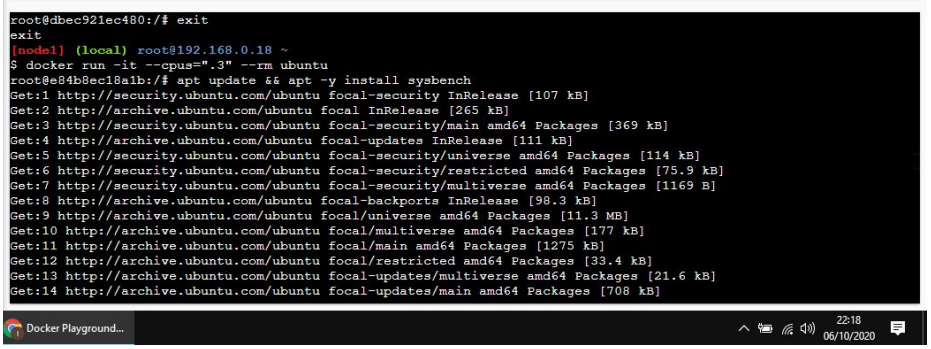
```
CPU speed:
    events per second:   150.24

General statistics:
    total time:                          10.0016s
    total number of events:              1503

Latency (ms):
         min:                                    2.59
         avg:                                    6.62
         max:                                   62.13
         95th percentile:                       51.02
         sum:                                 9943.54

Threads fairness:
    events (avg/stddev):           1503.0000/0.00
    execution time (avg/stddev):   9.9435/0.00
```

EJEMPLO CON 3 CPU



Obtenemos dicho resultado

```
CPU speed:
    events per second:    85.03

General statistics:
    total time:                          10.0405s
    total number of events:              854

Latency (ms):
         min:                                    2.64
         avg:                                   11.75
         max:                                   78.83
         95th percentile:                       74.46
         sum:                                10036.04

Threads fairness:
    events (avg/stddev):           854.0000/0.00
    execution time (avg/stddev):   10.0360/0.00
```

Conclusión

Cuanto más cpus contamos más eventos se podrán realizar en un periodo de tiempo