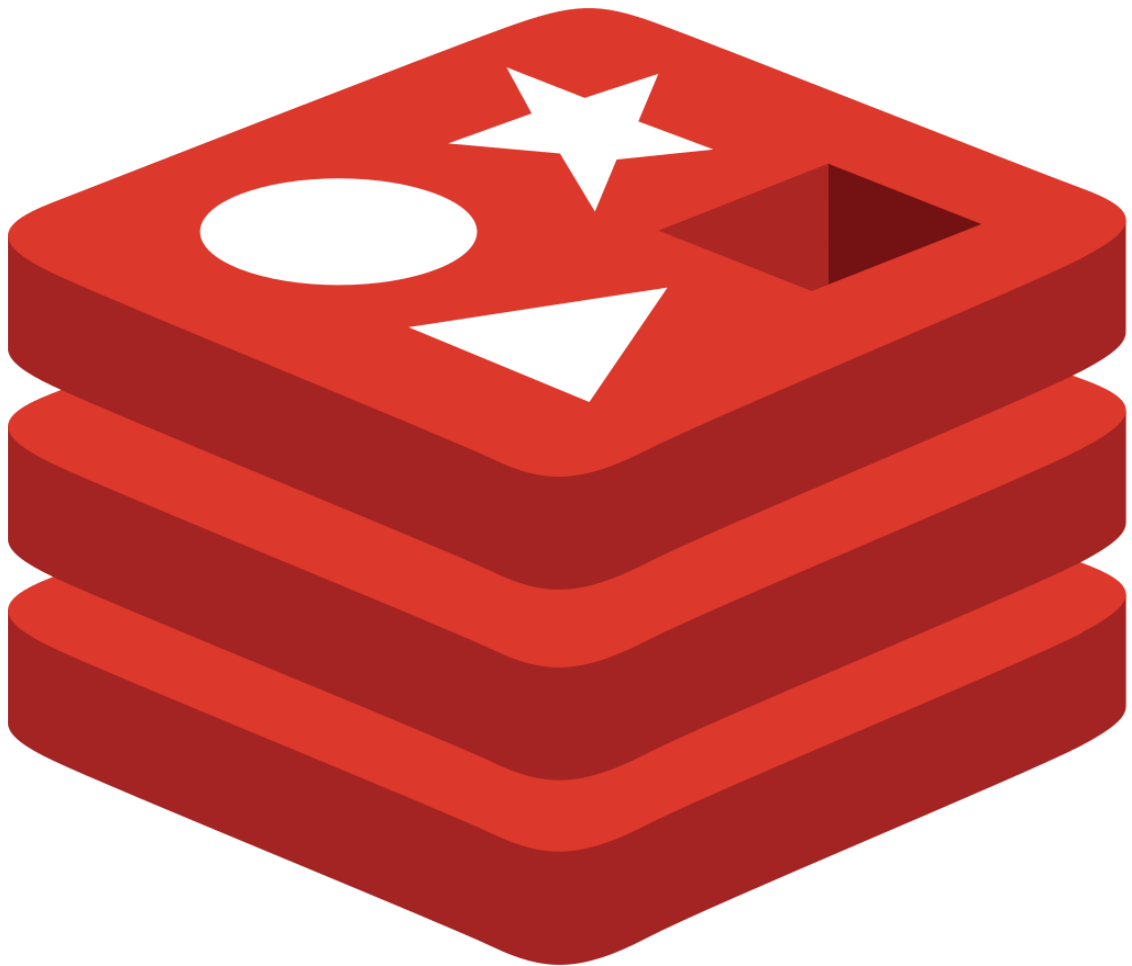


Caché

Redis

Sprint 2 Lab 3



Felipe Izquierdo Romero

Índice

<u>1</u>	<u>Descripción del problema</u>
<u>2</u>	<u>Desarrollo</u>
<u>3</u>	<u>Opcional</u>

1. Descripción del problema

Crear un sistema de notificaciones donde un profesor publica mensajes u los alumnos los visualizan.

2. Desarrollo

Para completar el ejercicio desde la consola de Docker actuará como alumno y se suscribirá al canal “TablonAnuncios” y desde una consola de comandos seremos el profesor que recibirá los mensajes.

Primero se mostrará la parte de Docker

```
redis-cli
```

Para entrar ejecutar posteriormente los comandos de redis

```
SUBSCRIBE TablonAnuncios
```

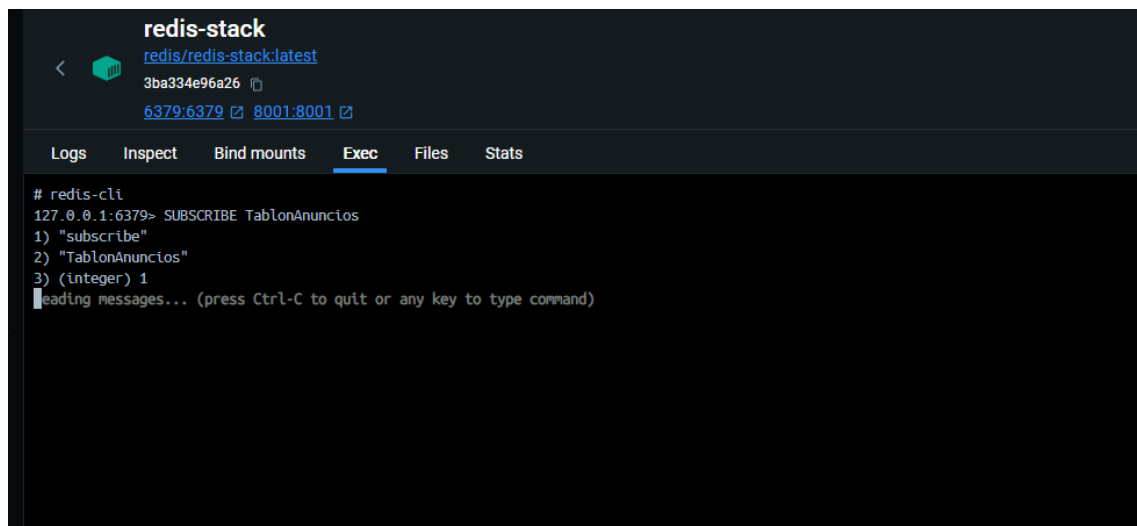


Ilustración 1- Alumno suscrito

A continuación, en una consola de comandos

```
docker ps
```

Escogemos el identificador del contenedor de Docker de redis

```
docker exec -it <id_contenedor> redis-cli
```

Ahora se ejecuta el comando PUBLISH para escribir el mensaje como profesor

```
PUBLISH TablonAnuncios 'Hola alumnos'
```

```

C:\Users\felix>docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS
3ba334e96a26   redis/redis-stack:latest  "/entrypoint.sh"    5 days ago    Up 10 minutes    0.0.0.0:6379->6379/tcp, 0.0.0.0:8001->8001/tcp
C:\Users\felix>docker exec -it 3ba334e96a26 redis-cli
127.0.0.1:6379> PUBLISH TablonAnuncios 'Hola alumnos'
(integer) 1
127.0.0.1:6379>

```

Ilustración 2- Profesor escribiendo un mensaje

Después si volvemos a Docker veremos que recibimos un mensaje del profesor

```

# redis-cli
127.0.0.1:6379> SUBSCRIBE TablonAnuncios
1) "subscribe"
2) "TablonAnuncios"
3) (integer) 1
1) "message"
2) "TablonAnuncios"
3) "Hola alumnos"
Reading messages... (press Ctrl-C to quit or any key to type command)

```

Ilustración 3- Mensaje recibido

3. Opcional

Siguiendo con el código del ejemplo de las clases he modificado algunas líneas para mostrar este funcionamiento y escribir mensajes como si fuera el profesor y visualizarlos en la consola de Docker o en el propio navegador.

Aunque estas modificaciones no distinguen la diferencia de un usuario final entre los roles de alumno o de profesor.

Primero he cambiado el canal donde se conecta el programa a recibir y/o publicar mensajes.

```

43 //-----//
44 redisSubscriber.subscribe('TablonAnuncios'); // Cambio de nombre del canal chat-channel a TablonAnuncios
45 //-----//

```

Ilustración 4- Cambio Subscripción a canal

```

redisPublisher.publish('TablonAnuncios', message); // Cambio de nombre del canal chat-channel

```

Ilustración 5- Cambio Publicaciones al canal

Después para que se carguen los mensajes que se han escrito anteriormente en el tablón he pensado en guardar los mensajes en una lista y listar los mensajes cuando te conectas.

```
redisPublisher.rpush('tablon-mensajes', message); //Cada usuario guarda el mensaje que envia en una lista
```

Ilustración 6- Guardar mensajes en una lista

```
56 // Listar los mensajes del historial del chat al conectarse
57 redisClient.lrange('tablon-mensajes', 0, -1, (err, message) => {
58     if (err) {
59         console.error('Error al obtener mensajes de Redis:', err);
60         return;
61     }
62     console.log('Mensajes guardados en Redis:');
63     message.forEach((message) => {
64         socket.emit('chat message', message);
65     });
66 });
```

Ilustración 7- Listar mensajes

Para comprobar el funcionamiento del código ejecutaremos el servidor abriendo una consola de comandos en el directorio donde se encuentra y ejecutaremos

```
node server.js
```

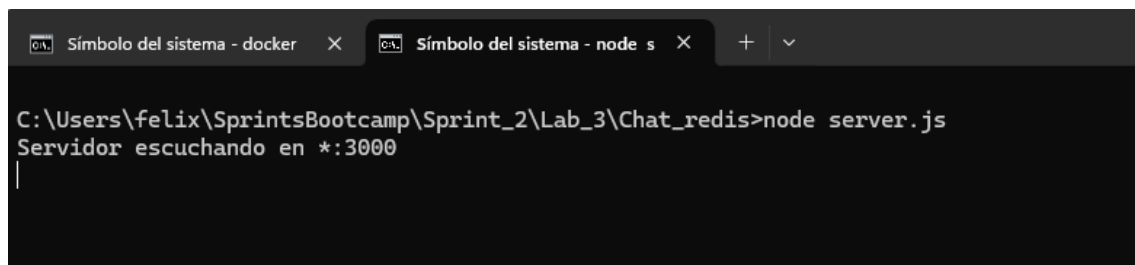


Ilustración 8- Ejecutar servidor

Al entrar en <http://localhost:3000/> Veremos que no se ha cargado el mensaje publicado anteriormente 'Hola alumnos'.

Esto es porque no hay ningún mensaje en la lista donde guardamos los mensajes al enviarlos a través de nuestro navegador.

Probaremos a poner el username de Profesor_1 y escribir un mensaje y veremos cómo se cargan en el navegador al iniciar otra sesión nueva y en la consola de Docker que esta suscrita al canal.



Ilustración 9- Mensaje desde el navegador

```
# redis-cli
127.0.0.1:6379> SUBSCRIBE TablonAnuncios
1) "subscribe"
2) "TablonAnuncios"
3) (integer) 1
1) "message"
2) "TablonAnuncios"
3) "Hola alumnos"
1) "message"
2) "TablonAnuncios"
3) "Profesor_1: Mensaje desde el navegador"
Reading messages... (press Ctrl-C to quit or any key to type command)
```

Ilustración 10- Recibido en Docker

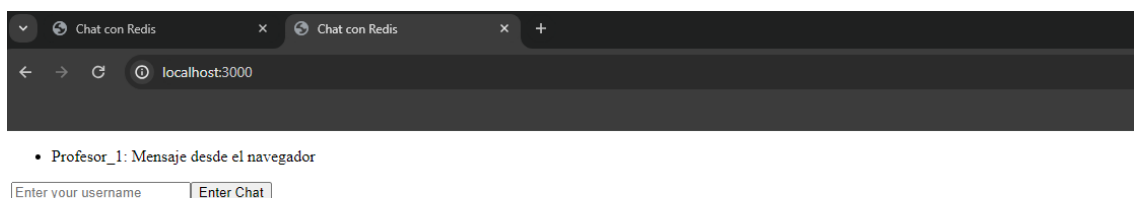


Ilustración 11- Carga de mensajes al iniciar nueva sesión