

Reporte de Proyecto final de Redes

WORLDFEED

(Diciembre de 2022)

Irving Yarón García Pérez **212163**, Ángel Gabriel Galindo López **212276**, Isabel Abigail Bermudez Navarro **175623**, Monica lizeeth Lozano Velázquez **275470**

EL modo de emplear las conexiones en modo cliente- servidor nos ayudan a entender de mejor forma cómo funcionan las redes distribuidas, entender como se relacionan unas con otras, las transferencias de datos mediante texto plano al hacer uso de las APIs, las cuales han revolucionado y facilitado el compartir información en las diferentes tecnologías web que utilizamos días con día. Temas que iniciamos a ver el semestre pasado con la materia de Tecnologías Web, para el desarrollo de páginas y aplicaciones de manera general, descubrimos las interesantes formas que podemos agregar hasta multimedia como lo son imágenes o graficas.

Entender los sistemas distribuidos mediante algún ejercicio o en este caso proyecto hace que desempeñemos de nuestras capacidades ante la investigación, comprensión y forma de entender conforme se presentaba en el semestre.

miento en el mercado. Mostraremos cada ejemplo de las marcas usadas al igual de la API y “Servicios” que nosotros hemos creado.

Aquí es donde se usan en los archivos Stock.js y news.js

```
let res = await fetch(
  'https://newsapi.org/v2/everything?q=${this.props.newsName}&language=es&pageSize=4&apiKey=d3222134946243268fd7e6263cfb632'
);
let data = await res.json();
console.log(data, typeof data.articles, data.articles, typeof ar);
```

```
let API_Call = 'https://www.alphavantage.co/query?function=TIME_SERIES_DAILY_ADJUSTED&symbol=${StockSymbol}&outputsize=compact&apikey=${API_KEY}';
let API_Call = 'https://www.alphavantage.co/query?function=TIME_SERIES_DAILY_ADJUSTED&symbol=APC&outputsize=compact&apikey=PMWZK1ZJFCCUMK';
let stockChartValuesFunction = {};
fetch(API_Call)
  .then(
    function(response){
      return response.json();
    }
  )
  .then(
    function(data){
```

A. Apple

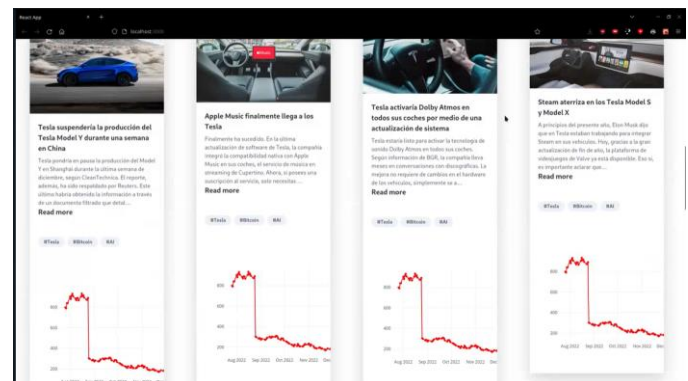
Índice de Términos – Sistema distribuido Servicios Web, Obtención de APIs, Implementación de un Server.

I. INTRODUCCION

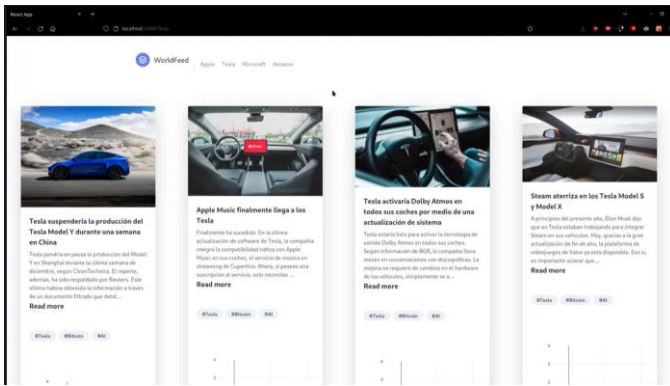
Este documento presentaremos finalmente nuestro proyecto, queremos brindar a la persona promedio un mejor entendimiento de los acontecimientos que suceden en el mundo y el impacto que provocan, mostrándoles una imagen más completa de las noticias. Se busca generar una conciencia social de los eventos internacionales y su efecto en la economía de varias entidades como empresas, divisas u otros indicadores económicos.

II. INTEGRACIÓN DE AL MENOS DOS SERVICIOS WEB

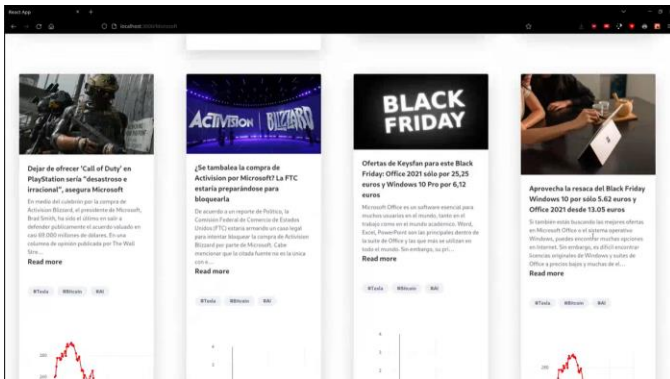
En nuestra demostración de servicios requeridos de manera externa, los uniremos mediante una aplicación en React a manera de Local Host, nos decidimos entre 2 Alpha Vantage y NewsAPI, las cual por un lado uno deja ver noticias de artículos del mercado y por el otro, gráficamente su avanza



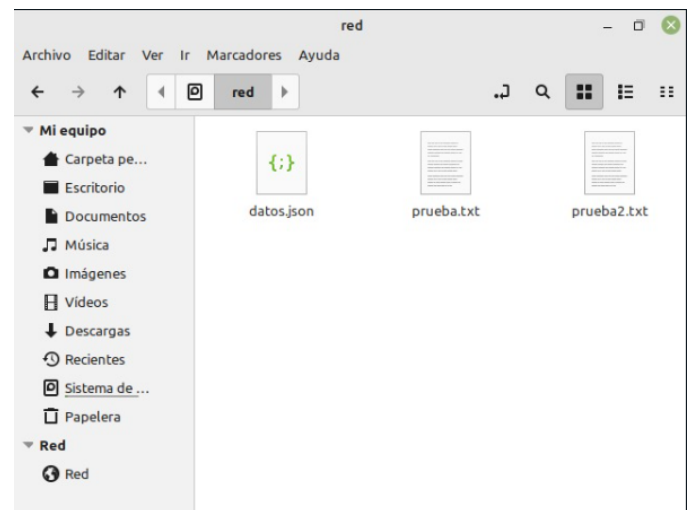
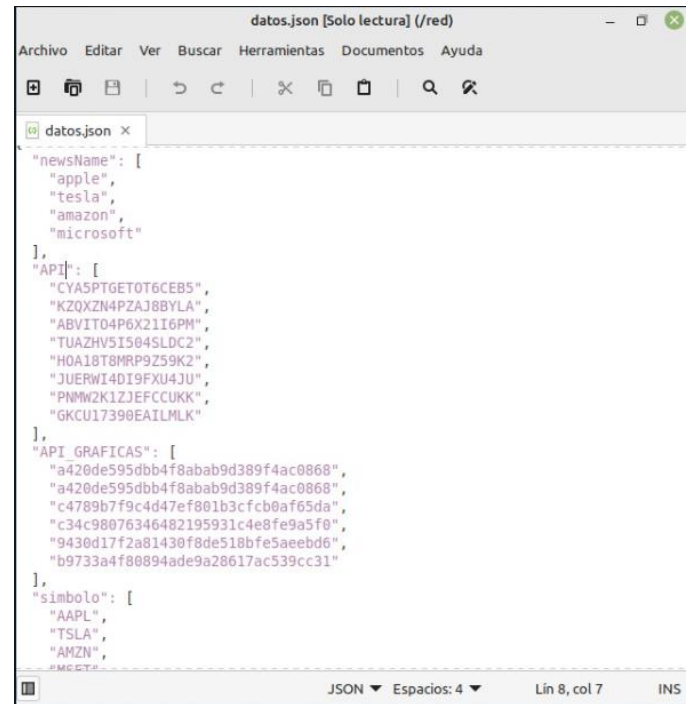
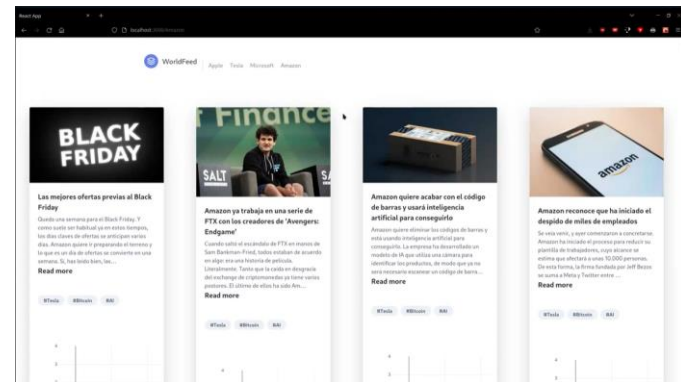
B. Tesla



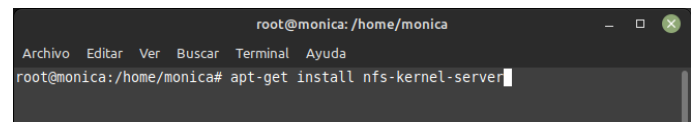
C. Figuras.



D. Amazon



2) Replicación de almacenamiento. Instala servicio de nfs



Crea una nueva carpeta

```
root@monica:~# mkdir /red
root@monica:~#
```

La Configuración de la red se deja manual, en este caso para identificar que ip se tiene se escribe lo siguiente

```
root@monica:/etc# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.47.80 netmask 255.255.255.0 broadcast 192.168.47.255
```

Editar el archivo nano dentro de etc

III. PROPORCIONAR UN SERVICIO WEB BÁSICO

1) Estructura de almacenamiento

Se almacenarán palabras clave que se almacenaran en una tabla de etiquetas, en otra tabla se almacenara el símbolo con el que cotiza en bolsa una empresa, símbolo de varias divisas y un conjunto de tags para identificar palabras clave de la noticia y en otra tabla se almacenara la información de la noticia, en otra podrían almacenarse nombres de recursos multimedia que podríamos necesitar, estas tablas se almacenaran en archivos formato .txt. Para la replicación del almacenamiento a través de un sistema de archivos distribuidos usaremos NFS por su uso en sistemas Linux ya que este será el sistema que usaremos para el servidor.

```
root@monica:~# cd /etc
root@monica:/etc# nano exports
```

Se edita lo siguiente

```
GNU nano 6.2 exports
# /etc/exports: the access control list for filesystems which may be exported
# to NFS clients. See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4 gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
# red 192.168.47.80/24(rw,sync,no_root_squash)
```

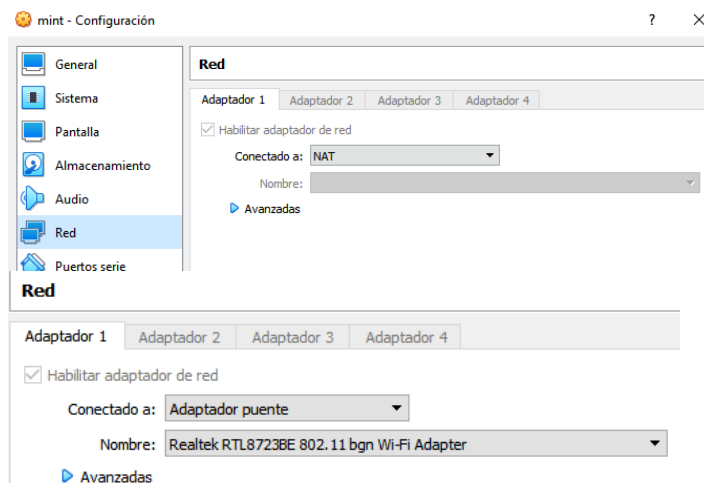
Se agrega el nombre de la carpeta a compartir, la dirección ip del host que estará compartiendo la carpeta.

```
GNU nano 6.2 exports
# /etc/exports: the access control list for filesystems which may be exported
# to NFS clients. See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4 gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
# red 192.168.47.80/24(rw,sync,no_root_squash)
```

Inicia el servicio de nfs

```
root@monica:/etc# /etc/init.d/nfs-kernel-server start
Starting nfs-kernel-server (via systemctl): nfs-kernel-server.service.
```

De NAT se cambia a adaptador puente



Actualiza los últimos cambios realizados

```
root@monica:/etc# systemctl restart nfs-kernel-server.service
root@monica:/etc#
```

Restablecer el rpcbind y portmap

```
root@monica:/etc# systemctl restart rpcbind
root@monica:/etc# cd init.d
root@monica:/etc/init.d# systemctl restart portmap
root@monica:/etc/init.d#
```

Se monta la carpeta en la red

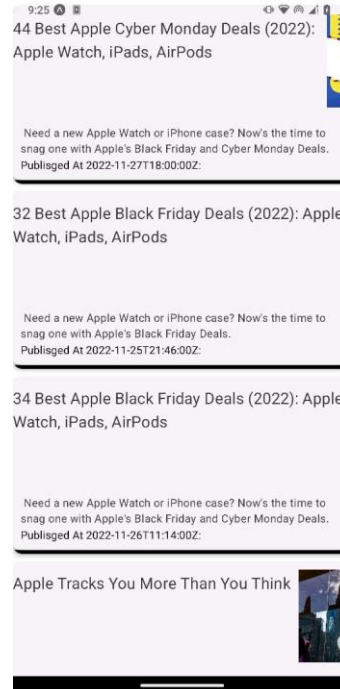
```
root@monica:/etc/init.d# mount 192.168.47.193:/red /home/yo/Red
```

Ahora solo queda agregar los archivos a la carpeta compartida para que se muestren simultáneamente en ambas máquinas.

3) Multiplataforma.

Realizamos una aplicación web desarrollada en React y una adaptación a Android probablemente usando lo que aprendemos en la materia de Programación de

dispositivos móviles con kotlin ó tentativamente también se podría desarrollar con React.



IX. CONCLUSIÓN

Queremos reconocer que el encargado del proyecto de equipo, nuestro compañero Irving, nos ayudo a organizarnos y ayudarnos a indicar en donde podíamos investigar las partes que no teníamos claras de que era o que teníamos que hacer. El hacer funcionar las APIs de manera correcta a como queríamos que funcionaran, la decisión de usarlas creo que fue la mejor e innovadora ya que no fue una tienda en línea como nos dijo el profesor. Esperamos con ansias seguir creando mas funciones de todas las tecnologías que hemos aprendido en las diferentes materias al pasar de los semestres.

REPOSITORIO EN GITHUB

Nuestro link directo a nuestro repositorio:

-Aplicación Web

<https://github.com/1RV1NG-Y/WorldFeed>

-App movil

<https://github.com/1RV1NG-Y/WorldFeedApp>

RECONOCIMIENTO

Queremos agradecer al profesor que nos brindo la mejor información para impartir el curso de manera que supo liderar con un grupo tan grande como al de nosotros, al igual que agradecer al departamento y función de mantener acutalizados los programas con las tecnologías mas recientes para mantener nuestro nivel fresco para nuevos aprendizajes .

REFERENCES

- [1] G. O. Young, “Synthetic structure of industrial plastics (Book style with paper title and editor),” in *Plastics*, 2nd ed. vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15–64.
- [2] W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.
- [3] H. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1985, ch. 4.

q