



# Recuerda marcar tu asistencia



Asistencia en LMS



Semana 1

# Microservicios Contenedores

Ing Edwin Garcia



El futuro digital  
es de todos

MinTIC

# Hechos

QUE

CONECTAN

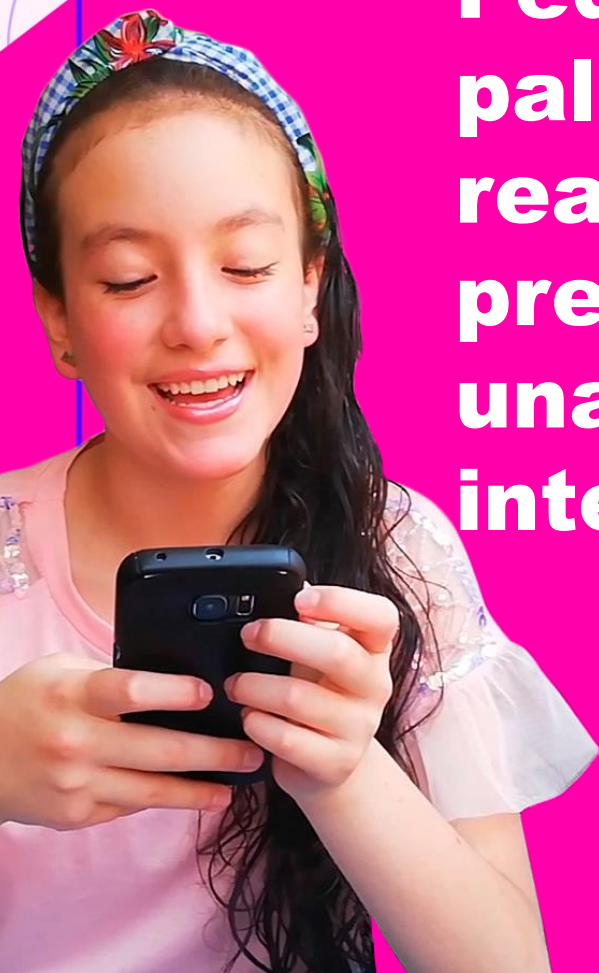
Misión  
TIC 2022

unab  
VIGILADA MINEDUCACIÓN

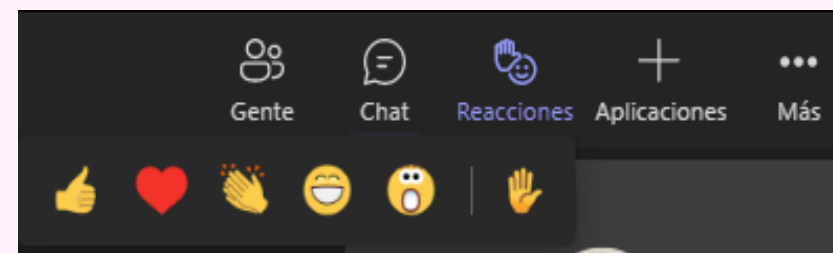
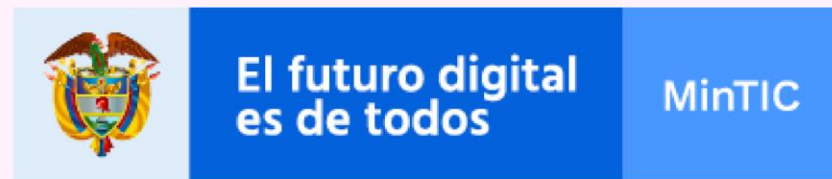
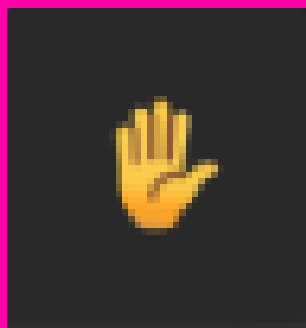


## Recuerden marcar su asistencia

Asistencia en LMS



**Pedir la  
palabra para  
realizar una  
pregunta o  
una  
intervención**







## Desarrollo de Aplicaciones Web



El futuro digital  
es de todos

MinTIC

Mision  
TIC 2022



Hechos  
QUE CONECTAN ✓



# ¿Qué son los microservicios?

Los microservicios son un enfoque arquitectónico y organizativo para el desarrollo de software donde el software está compuesto por pequeños servicios independientes que se comunican a través de API bien definidas. Los propietarios de estos servicios son equipos pequeños independientes.



# ¿Qué son los microservicios?

Las arquitecturas de microservicios hacen que las aplicaciones sean más fáciles de escalar y más rápidas de desarrollar. Esto permite la innovación y acelera el tiempo de comercialización de las nuevas características.



## Arquitectura monolítica en comparación con la arquitectura de microservicios

Con las arquitecturas monolíticas, todos los procesos están estrechamente asociados y se ejecutan como un solo servicio. Esto significa que, si un proceso de una aplicación experimenta un pico de demanda, se debe escalar toda la arquitectura. Agregar o mejorar las características de una aplicación monolítica se vuelve más complejo a medida que crece la base de código.





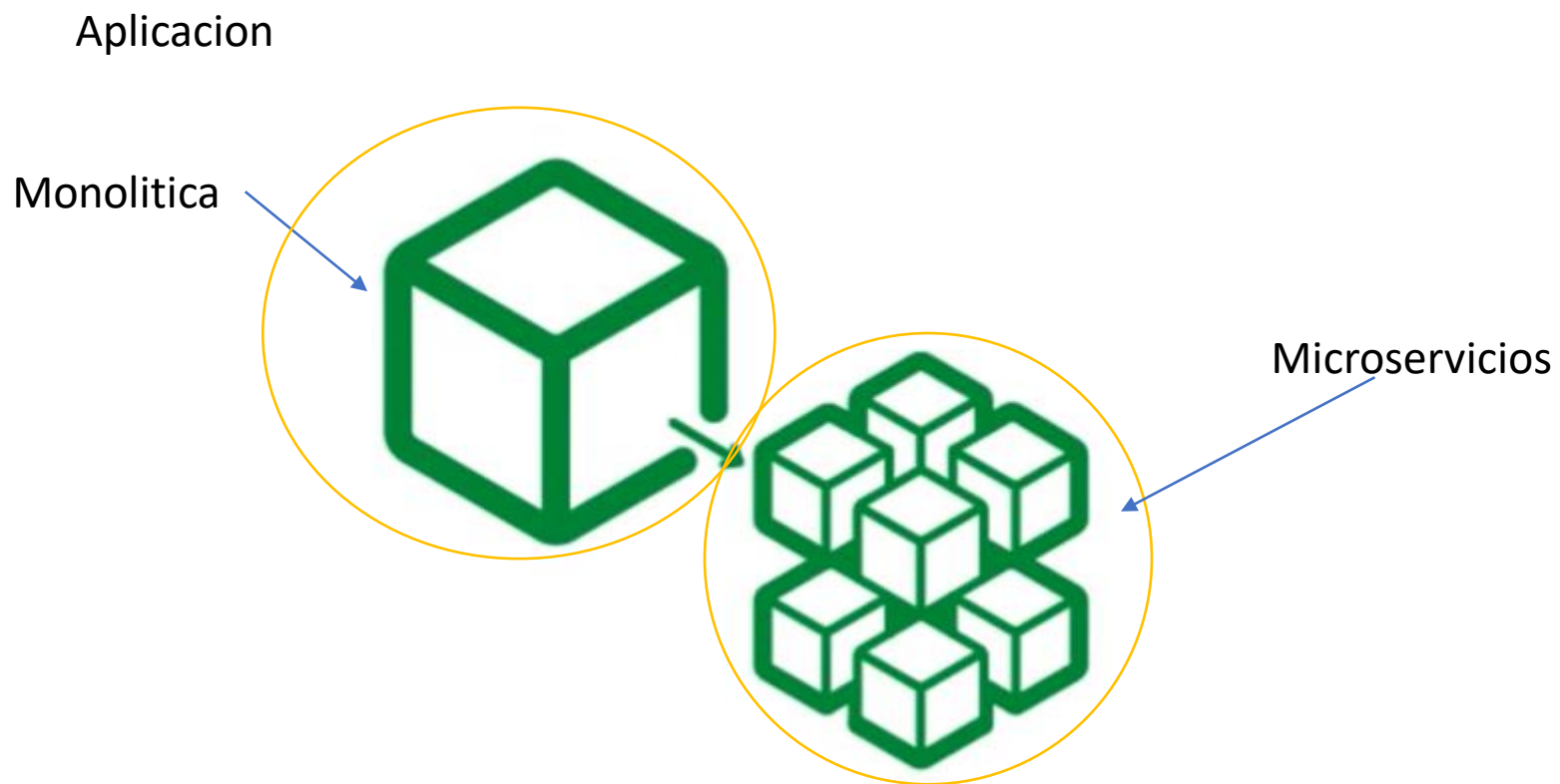
Esta complejidad limita la experimentación y dificulta la implementación de nuevas ideas. Las arquitecturas monolíticas aumentan el riesgo de la disponibilidad de la aplicación porque muchos procesos dependientes y estrechamente vinculados aumentan el impacto del error de un proceso



Con una arquitectura de microservicios, una aplicación se crea con componentes independientes que ejecutan cada proceso de la aplicación como un servicio. Estos servicios se comunican a través de una interfaz bien definida mediante API ligeras.

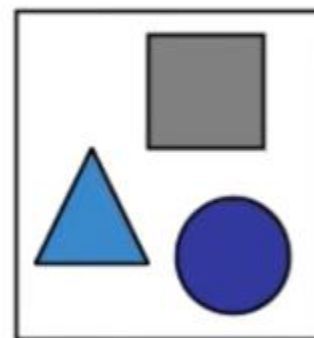


Los servicios se crean para las capacidades empresariales y cada servicio desempeña una sola función. Debido a que se ejecutan de forma independiente, cada servicio se puede actualizar, implementar y escalar para satisfacer la demanda de funciones específicas de una aplicación.



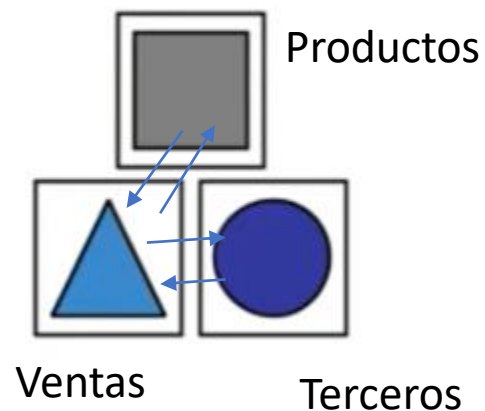


## Monolith



□ Process

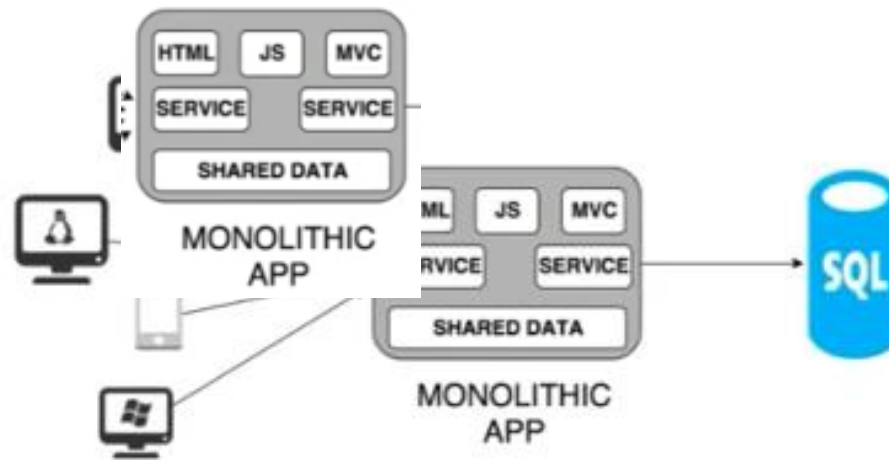
## Microservices





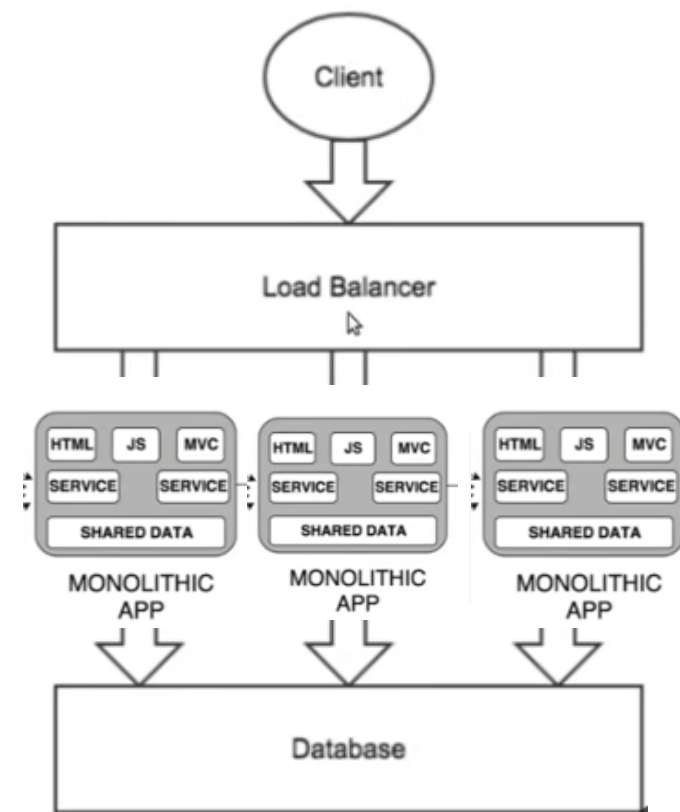


# Arquitectura Monolítica



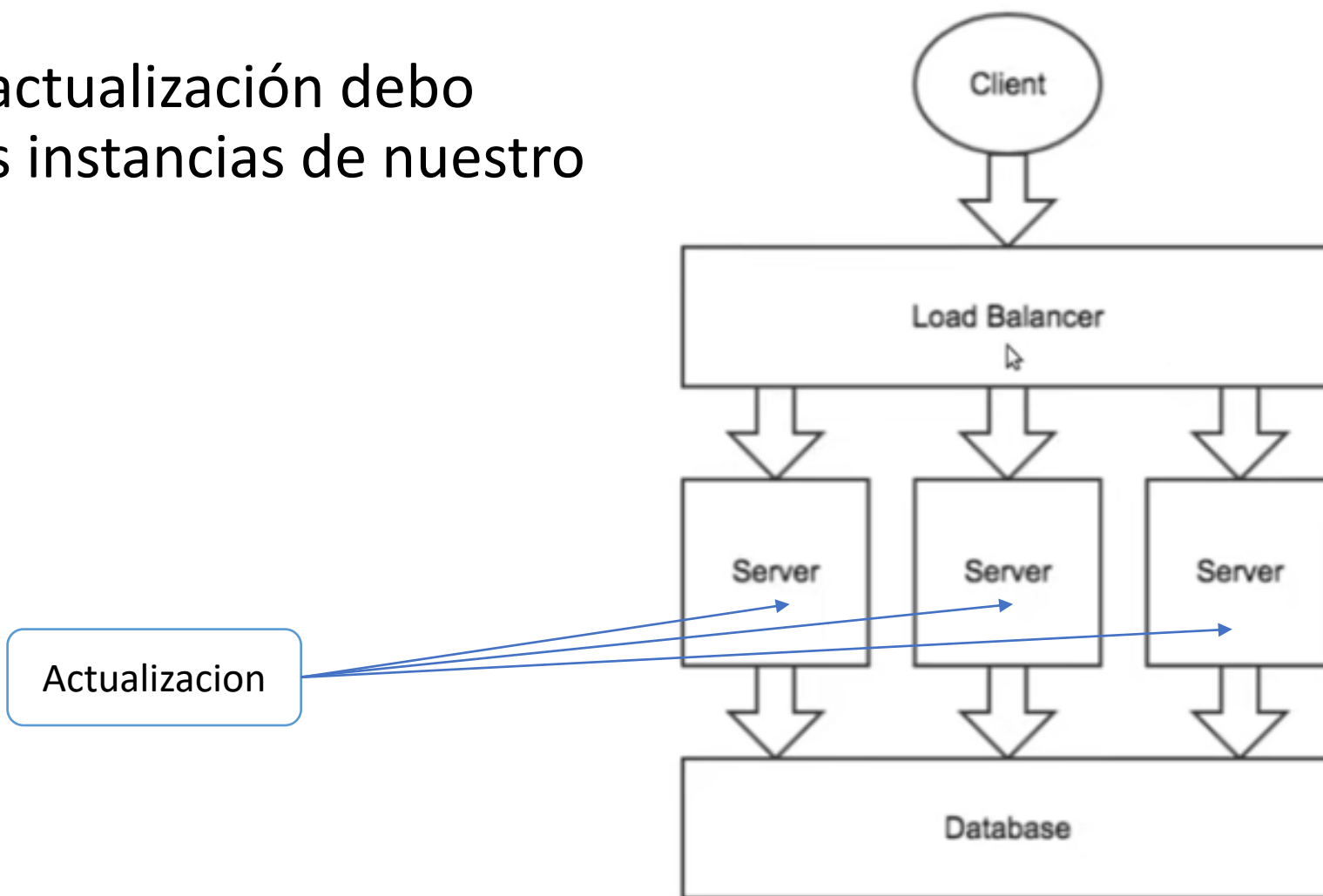


Cuando nuestras aplicaciones crezcan podemos replicarlas y ponerlas en sincronización bajo un balanceador de cargas..





Pero si existe una actualización debo actualizar todas las instancias de nuestro sistema.



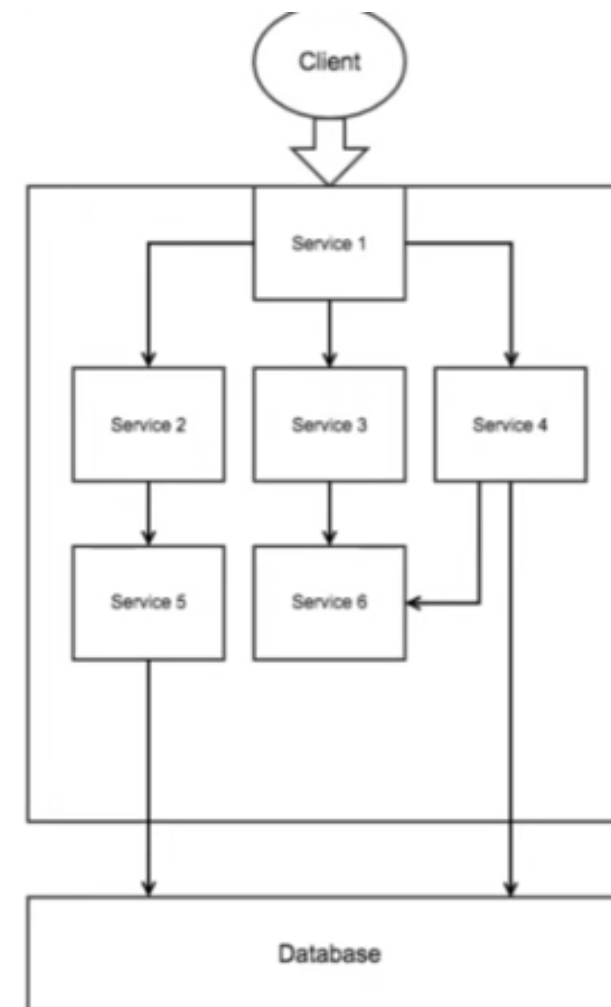


Y si se agregan mas funcionalidades?





Cada servicio puede comunicarse para dar una solución pero cada una de ellos puede tener ser desarrollado con lenguajes independientes...







MONOLITHIC / LAYERED



MICRO SERVICES



# Características de los microservicios



# Autónomos

Cada servicio componente en una arquitectura de microservicios se puede desarrollar, implementar, operar y escalar sin afectar el funcionamiento de otros servicios. Los servicios no necesitan compartir ninguno de sus códigos o implementaciones con otros servicios. Cualquier comunicación entre componentes individuales ocurre a través de API bien definidas.



# Especializados

Cada servicio está diseñado para un conjunto de capacidades y se enfoca en resolver un problema específico. Si los desarrolladores aportan más código a un servicio a lo largo del tiempo y el servicio se vuelve complejo, se puede dividir en servicios más pequeños.



# Beneficios de los microservicios





# Agilidad

Los microservicios fomentan una organización de equipos pequeños e independientes que se apropian de los servicios. Los equipos actúan en un contexto pequeño y bien comprendido, y están facultados para trabajar de forma más independiente y más rápida. Esto acorta los tiempos del ciclo de desarrollo. Usted se beneficia significativamente del aumento de rendimiento de la organización.



# Escalado flexible

Los microservicios permiten que cada servicio se escale de forma independiente para satisfacer la demanda de la característica de la aplicación que respalda. Esto permite a los equipos adecuarse a las necesidades de la infraestructura, medir con precisión el costo de una característica y mantener la disponibilidad si un servicio experimenta un aumento en la demanda.



# Implementación sencilla

Los microservicios permiten la integración y la entrega continuas, lo que facilita probar nuevas ideas y revertirlas si algo no funciona. El bajo costo de los errores permite experimentar, facilita la actualización del código y acelera el tiempo de comercialización de las nuevas características.



# Libertad tecnológica

Las arquitecturas de microservicios no siguen un enfoque de "diseño único". Los equipos tienen la libertad de elegir la mejor herramienta para resolver sus problemas específicos. Como consecuencia, los equipos que crean microservicios pueden elegir la mejor herramienta para cada trabajo.

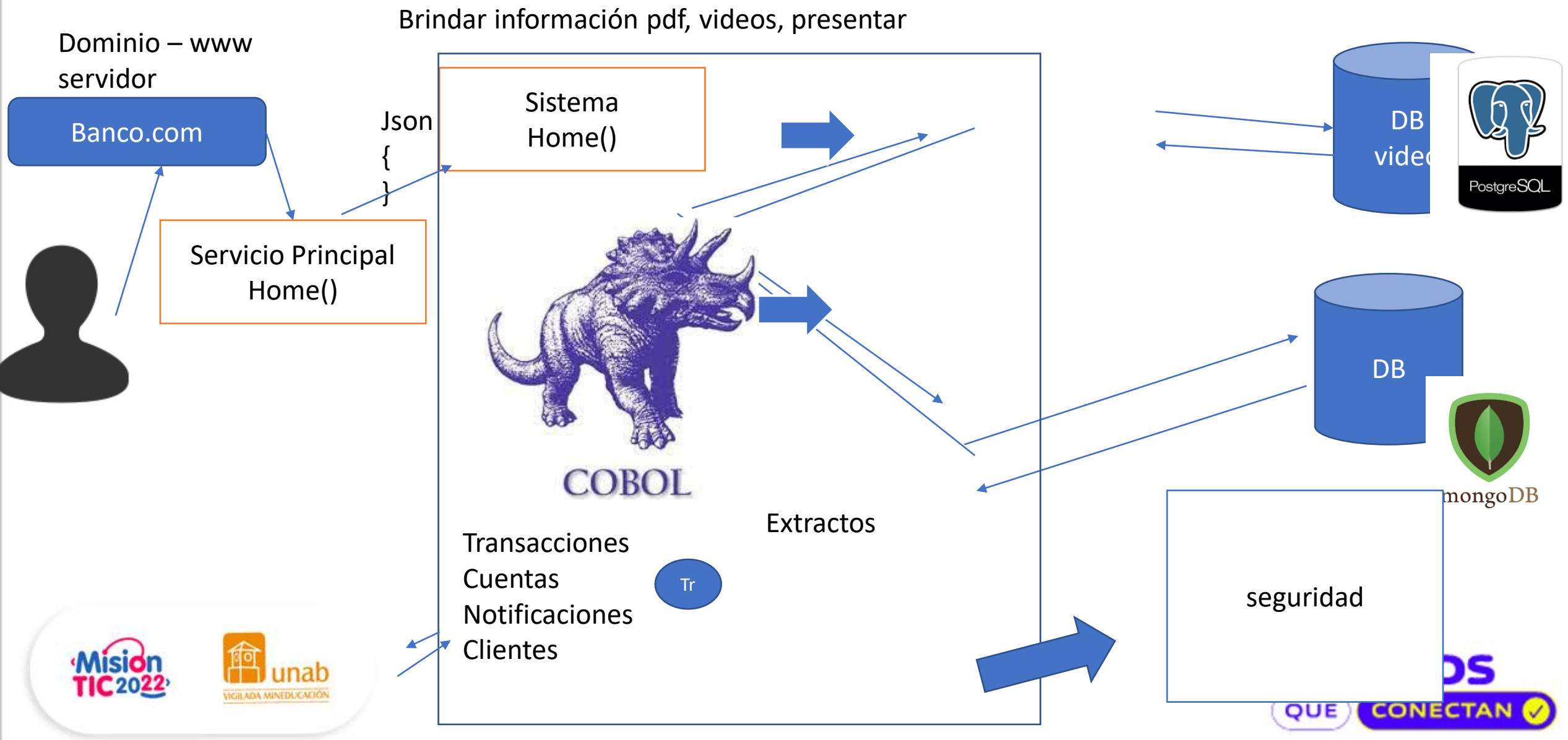
La división del software en módulos pequeños y bien definidos les permite a los equipos usar funciones para diferentes propósitos. Un servicio escrito para una determinada función se puede usar como un componente básico para otra característica. Esto permite que una aplicación arranque por sí sola, ya que los desarrolladores pueden crear nuevas capacidades sin tener que escribir código desde cero.





# Resistencia

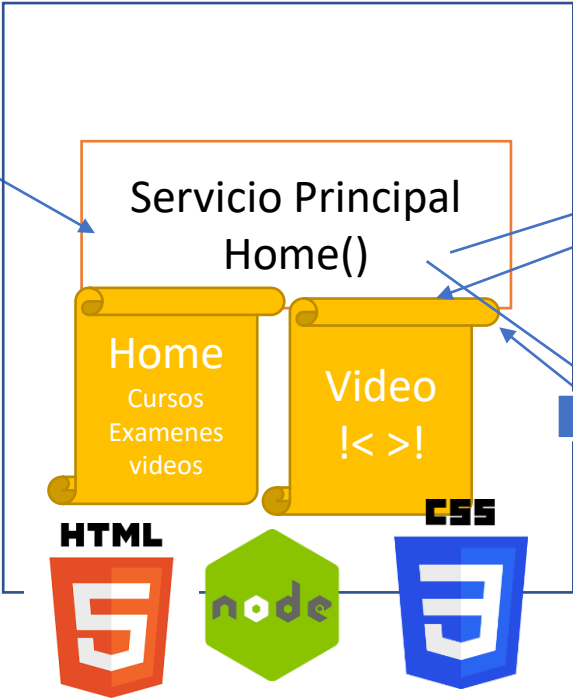
La independencia del servicio aumenta la resistencia de una aplicación a los errores. En una arquitectura monolítica, un error en un solo componente, puede provocar un error en toda la aplicación. Con los microservicios, si hay un error en todo el servicio, las aplicaciones lo manejan degradando la funcionalidad sin bloquear toda la aplicación.



Dominio – www  
servidor

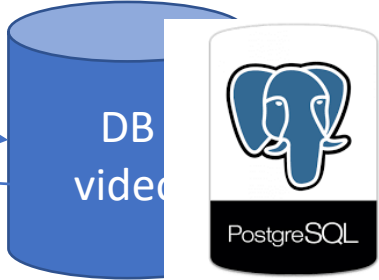
Bano.com

Brindar información pdf, videos, presentar

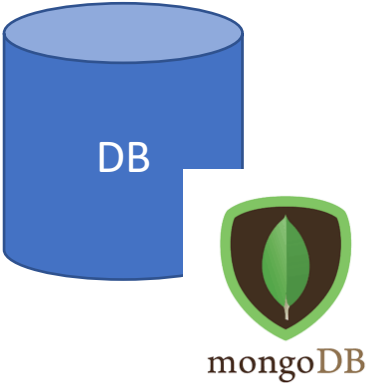
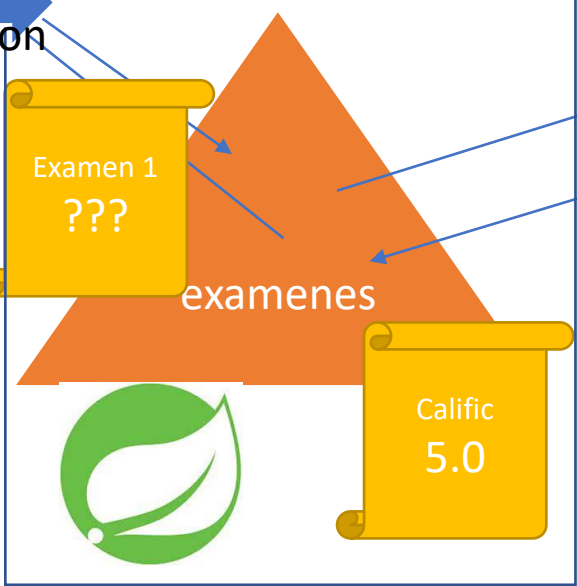


Json

Plataforma video



Json



Videos streaming

Ciclo5.com

Servicio Principal



## Desarrollo de Aplicaciones Web



El futuro digital  
es de todos

MinTIC

Misión  
TIC 2022



Hechos  
QUE CONECTAN 



# Contenedores



Los contenedores son una forma de virtualización del sistema operativo. Un solo contenedor se puede usar para ejecutar cualquier cosa, desde un microservicio o un proceso de software a una aplicación de mayor tamaño. Dentro de un contenedor se encuentran todos los ejecutables, el código binario, las bibliotecas y los archivos de configuración necesarios.





Sin embargo, en comparación con los métodos de virtualización de máquinas o servidores, los contenedores no contienen imágenes del sistema operativo. Esto los hace más ligeros y portátiles, con una sobrecarga significativamente menor. En implementaciones de aplicaciones de mayor tamaño, se pueden poner en marcha varios contenedores como uno o varios clústeres de contenedores. Estos clústeres se pueden gestionar mediante un orquestador de contenedores, como Kubernetes.

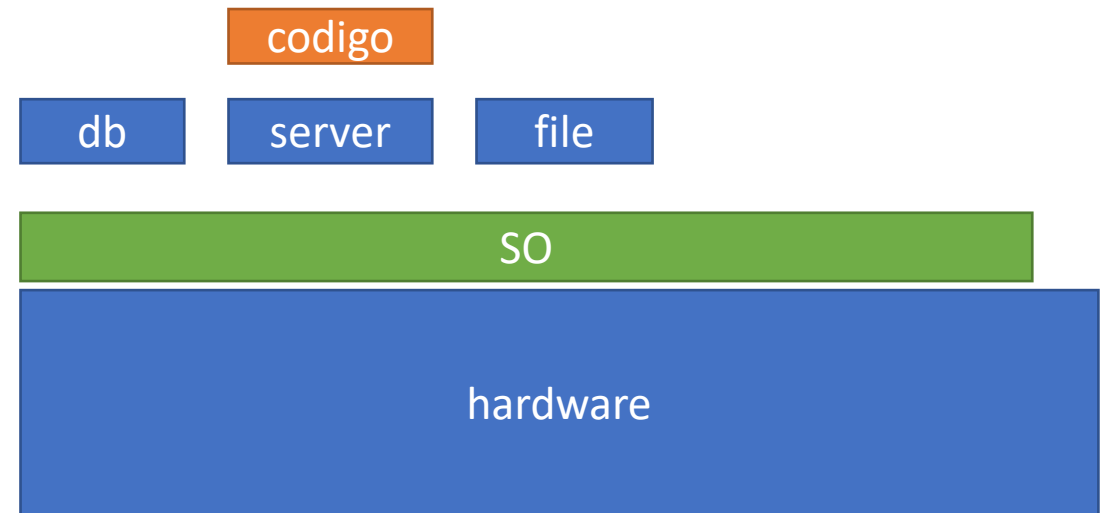
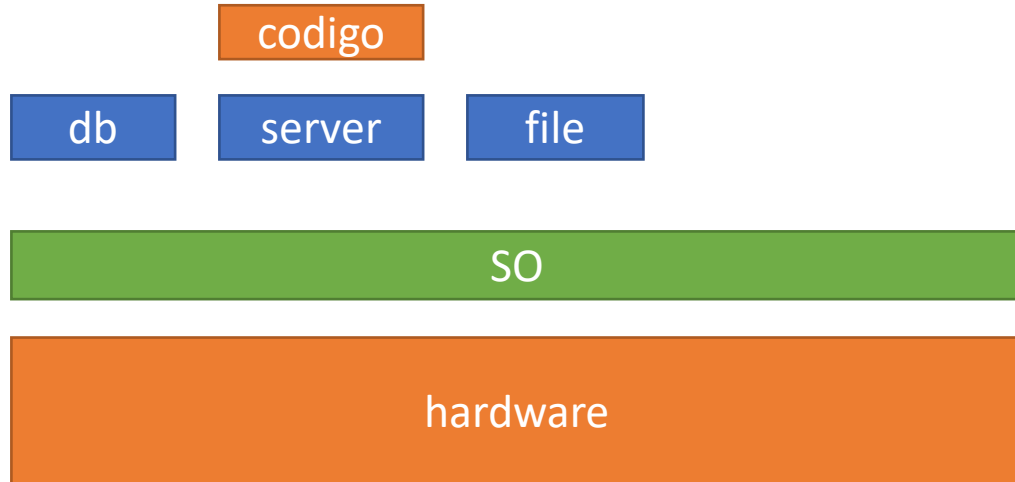


## Desarrollo de Aplicaciones Web



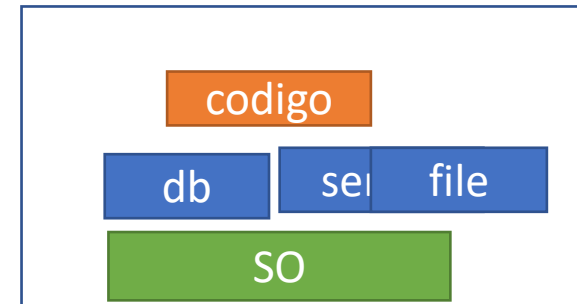
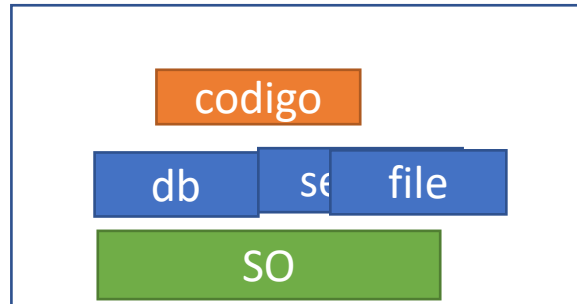
El futuro digital  
es de todos

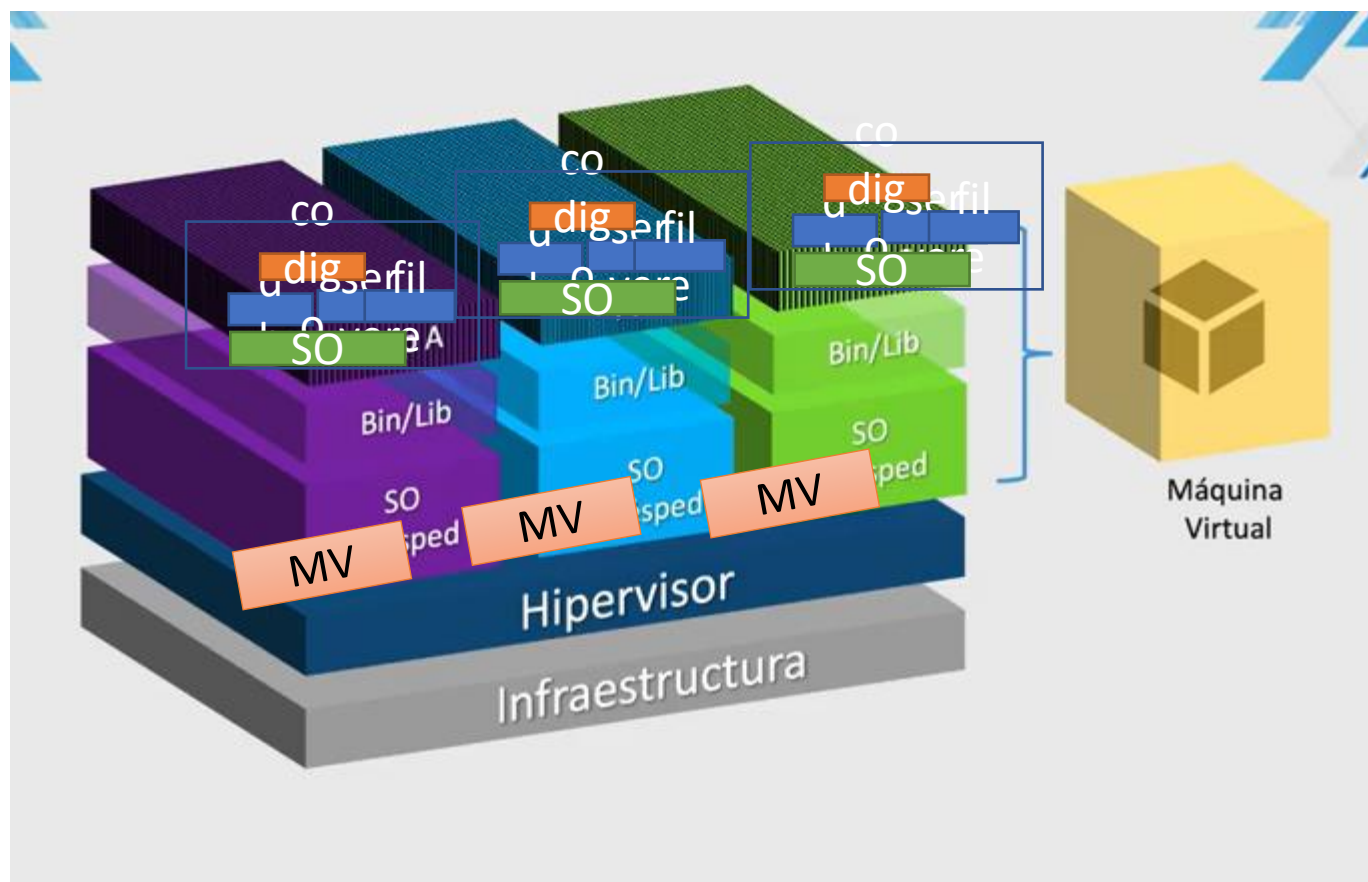
MinTIC

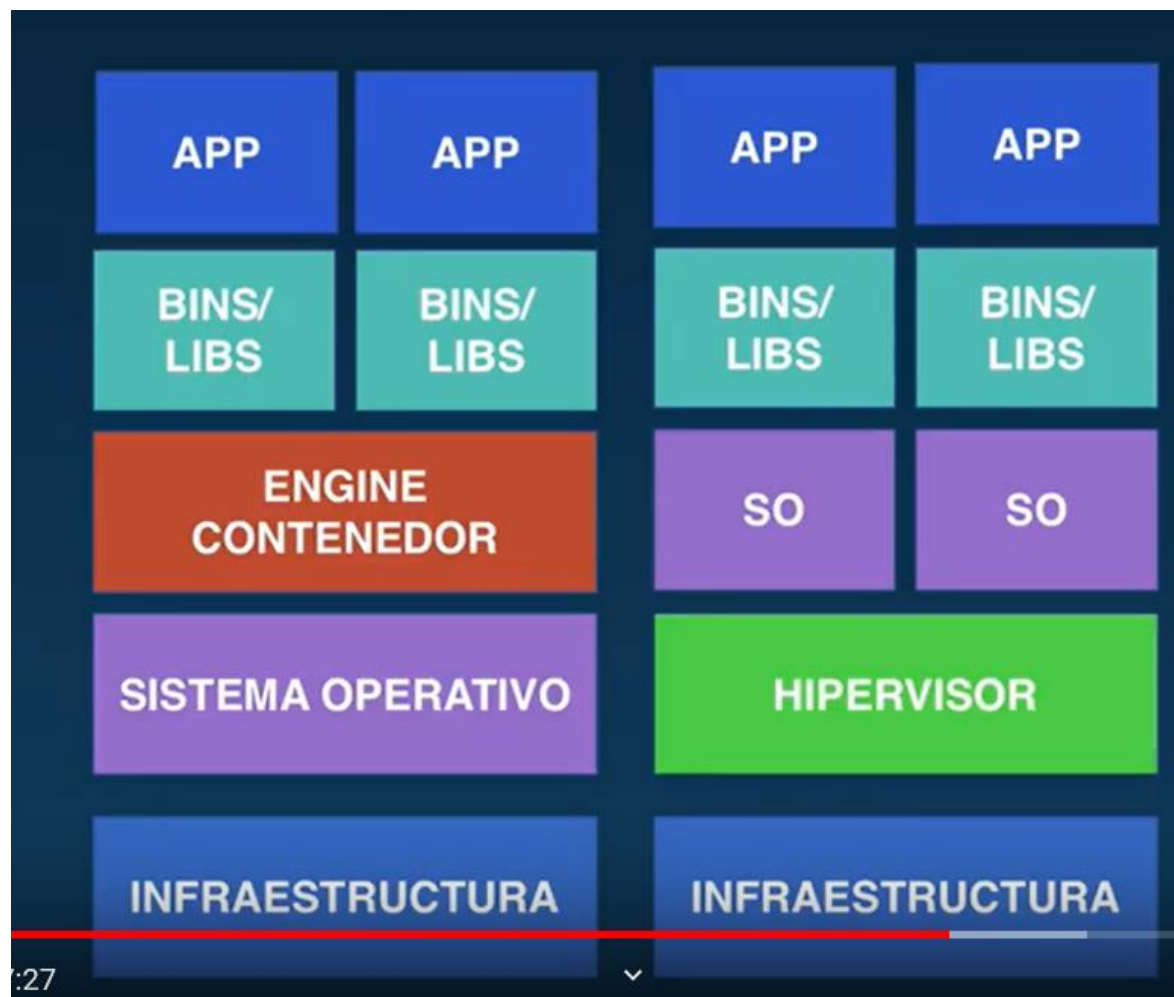




## Virtualización









# crear contenedor personalizado





# Beneficios de los contenedores



# Beneficios de los contenedores

Los contenedores son una forma optimizada de crear, probar, poner en marcha y volver a poner en marcha aplicaciones en varios entornos, desde un portátil local de un desarrollador hasta un centro de datos on-premises e incluso en la nube.

Algunos de los beneficios de los contenedores son:



# Menos sobrecarga

Los contenedores requieren menos recursos del sistema que los entornos de máquinas virtuales tradicionales o de hardware porque no incluyen imágenes del sistema operativo.



# Mayor portabilidad

Las aplicaciones que se ejecutan en contenedores se pueden poner en marcha fácilmente en sistemas operativos y plataformas de hardware diferentes.



# Funcionamiento más constante

Los equipos de DevOps saben que las aplicaciones en contenedores van a ejecutarse igual, independientemente de dónde se pongan en marcha.



# Mayor eficiencia

Los contenedores permiten poner en marcha, aplicar parches o escalar las aplicaciones con mayor rapidez.



# Mejor desarrollo de aplicaciones

Los contenedores respaldan los esfuerzos ágiles y de DevOps para acelerar los ciclos de desarrollo, prueba y producción.



# Contenedor



Contenedor ???

SO



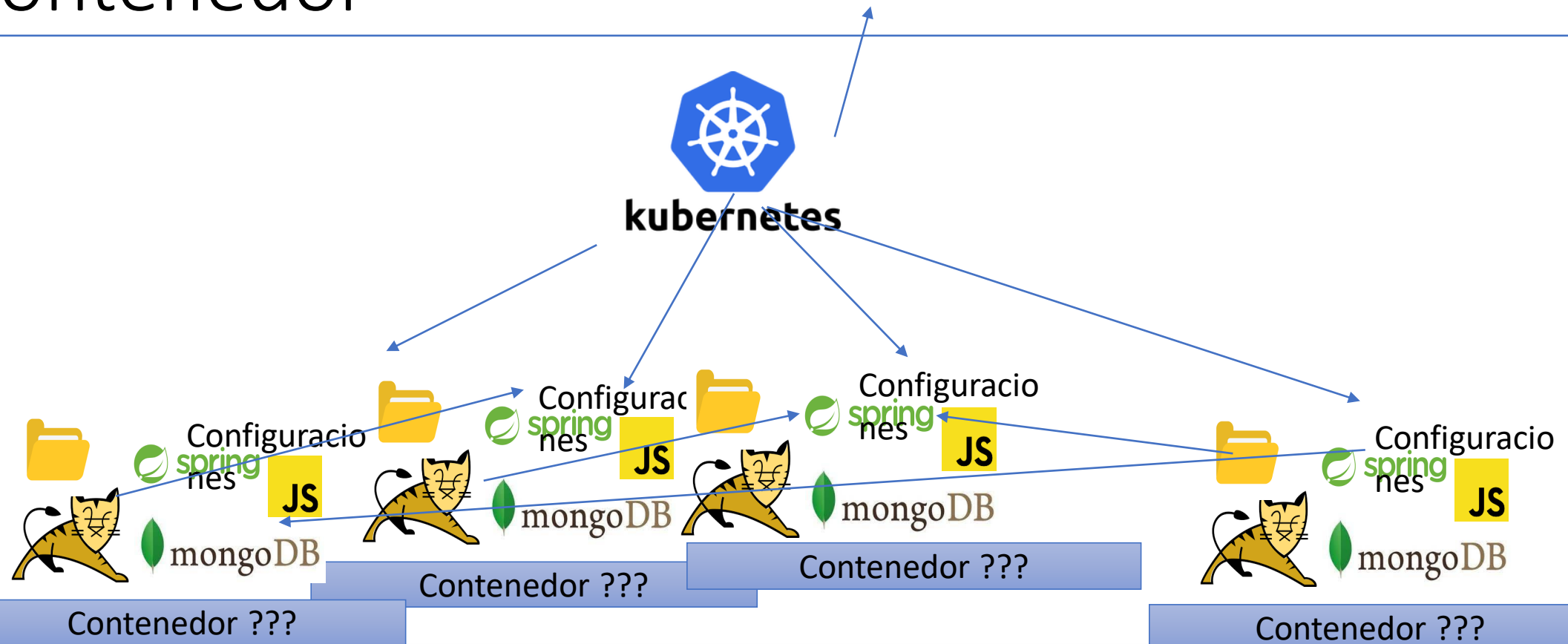
Contenedor ???

SO



# Contenedor







## Desarrollo de Aplicaciones Web



El futuro digital  
es de todos

MinTIC



Windows 10

Mision  
TIC 2022



Hechos  
QUE CONECTAN



🔍 [Todo](#)

🖼️ [Imágenes](#)

📺 [Vídeos](#)

📰 [Noticias](#)

📍 [Maps](#)

⋮ [Más](#)

[Herramientas](#)

Cerca de 48.400.000 resultados (0,42 segundos)

<https://www.docker.com> > products ▾ [Traducir esta página](#)

## Docker Desktop

**Docker Desktop** is an application for MacOS and Windows machines for the building and sharing of containerized applications and microservices. **Docker Desktop** ...

[Comprehensive Overview Guide](#) · [Docker Extensions](#) · [Latest blog](#)

## Docker Desktop

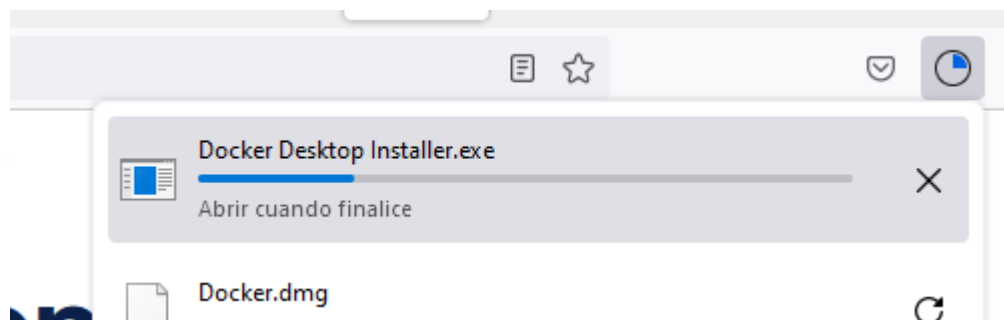
Install Docker Desktop – the fastest way to containerize applications.

Mac with Intel Chip

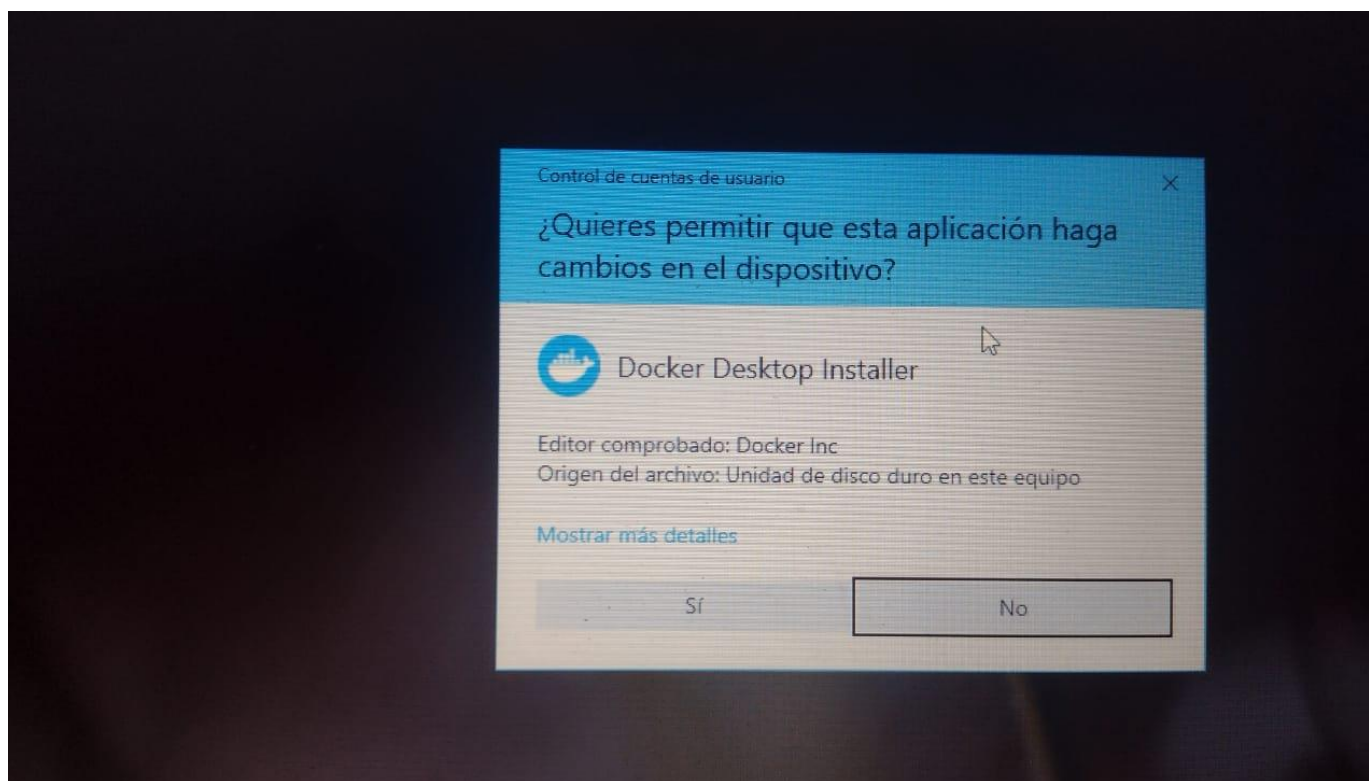
Mac with Apple Chip

MOST COMMON

Also available for [Windows](#) and [Linux](#)







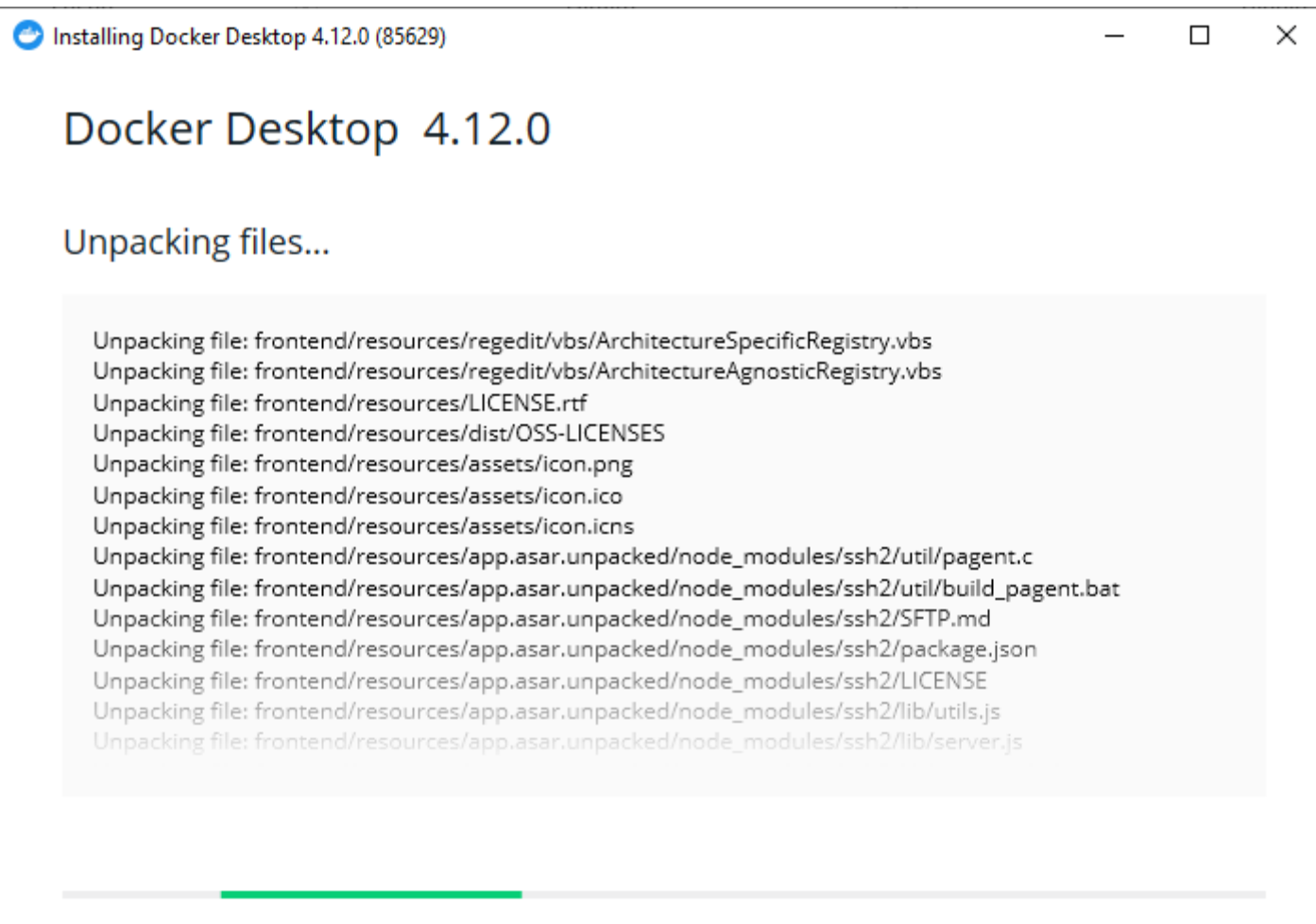


Installing Docker Desktop 4.12.0 (85629)

## Configuration

☒ Add shortcut to desktop

Ok





Installing Docker Desktop 4.12.0 (85629)



## Docker Desktop 4.12.0

Installation succeeded

You must restart Windows to complete installation.


Close and restart



## Docker Subscription Service Agreement

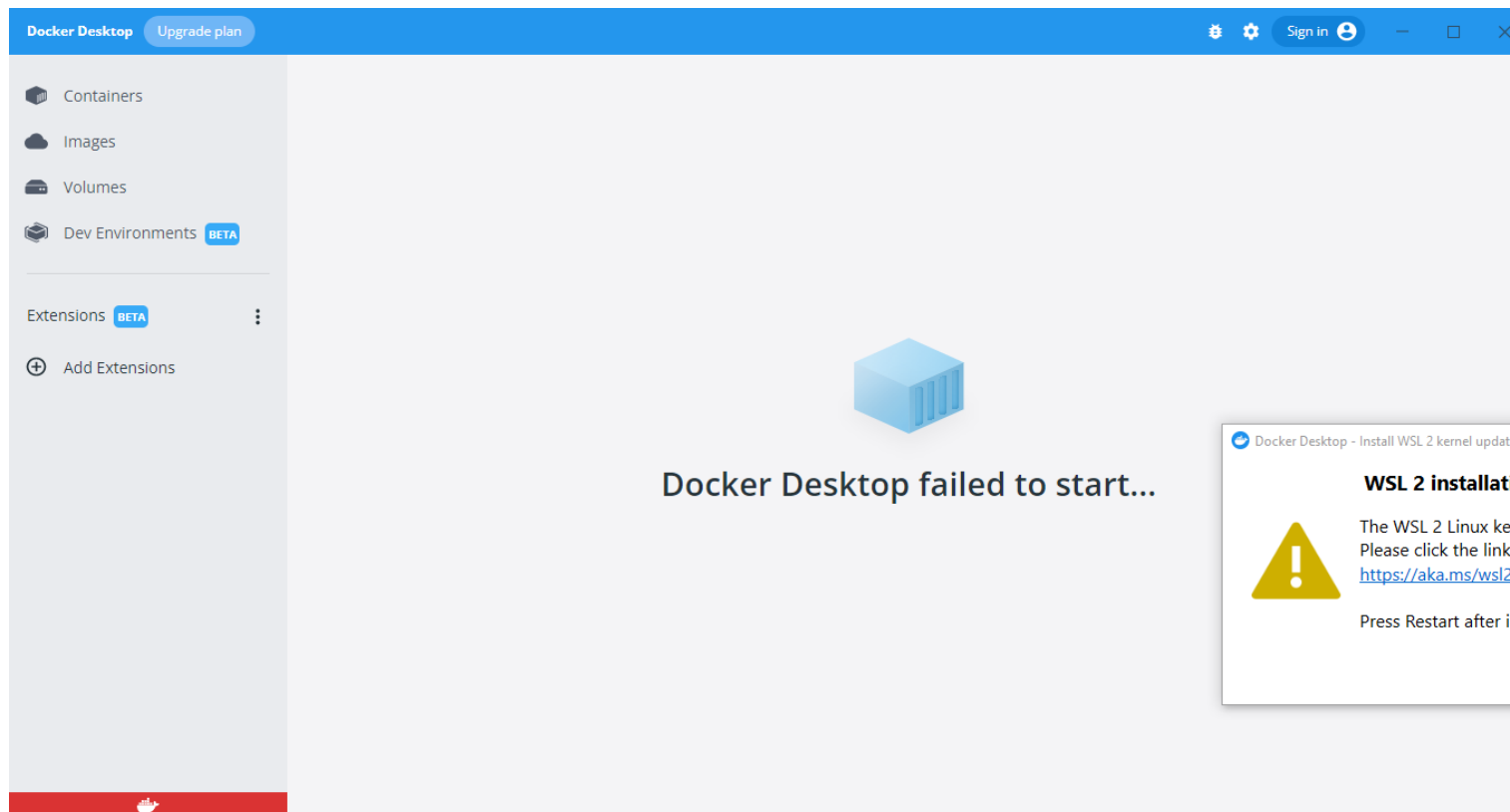
By selecting **accept**, you agree to the [Subscription Service Agreement](#), the [Docker Data Processing Agreement](#), and the [Data Privacy Policy](#).

**Note:** Docker Desktop is free for small businesses (fewer than 250 employees AND less than \$10 million in annual revenue), personal use, education, and non-commercial open source projects. Otherwise, it requires a paid subscription for professional use. Paid subscriptions are also required for government entities. [Read the FAQ to learn more.](#)

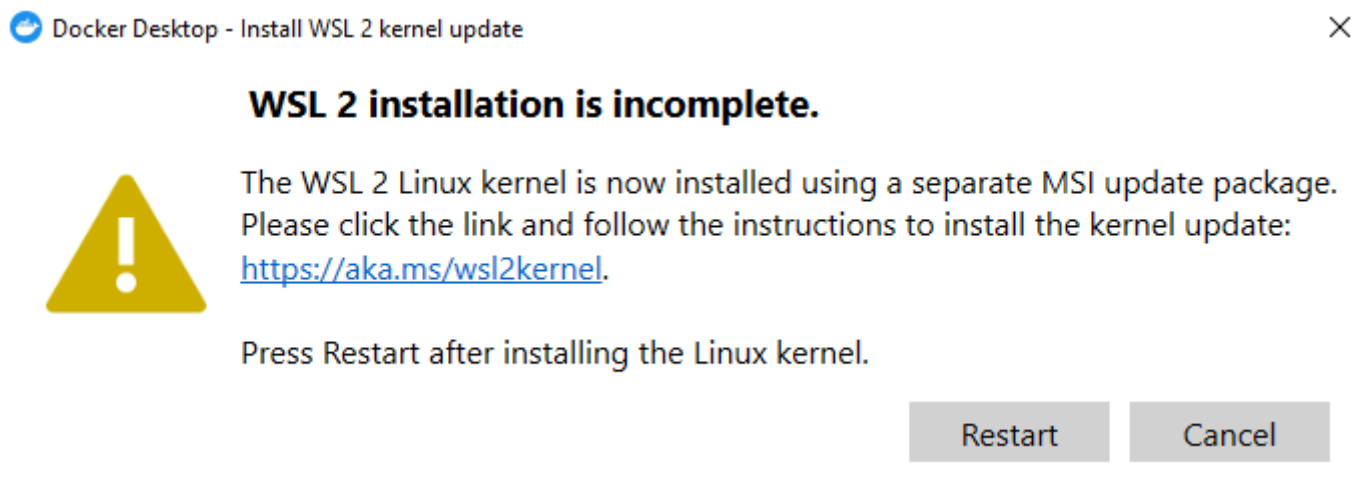
[View Full Terms](#) 

Accept

Close











# Paso 4: Descarga del paquete de actualización del kernel de Linux

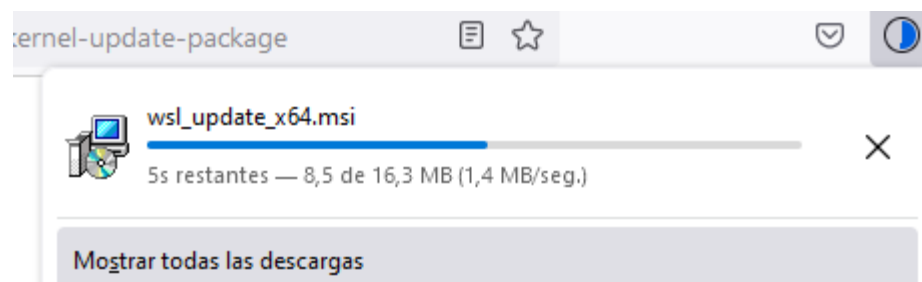
1. Descargue la versión más reciente:

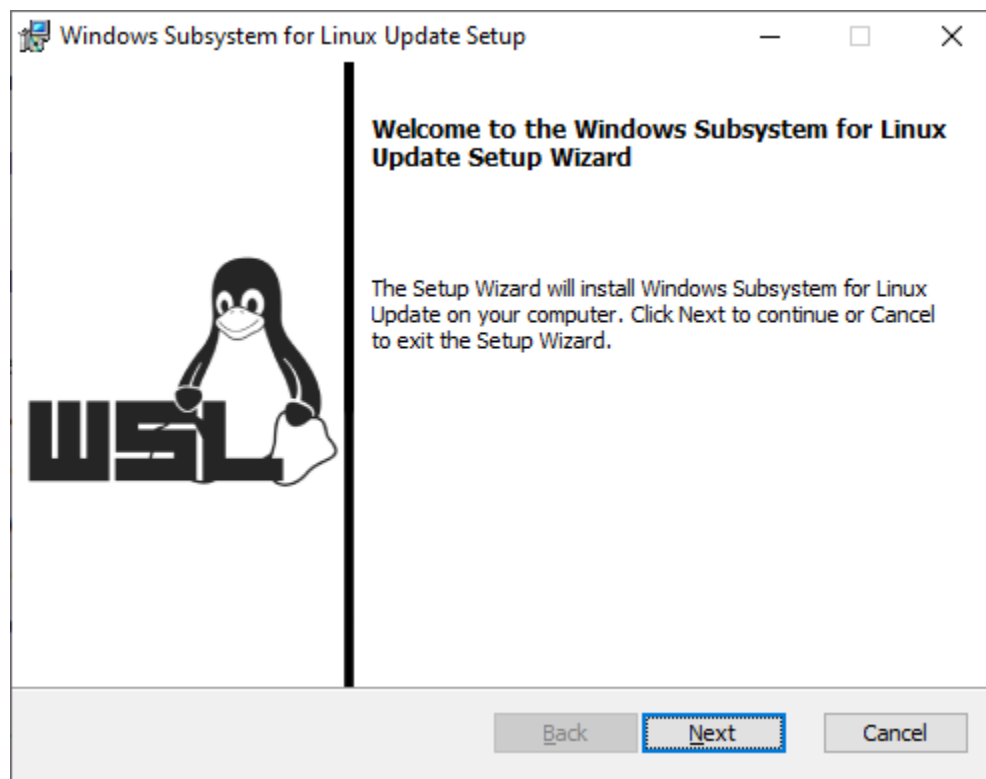
- [Paquete de actualización del kernel de Linux en WSL 2 para máquinas x64](#)

### Nota

Si estás usando una máquina ARM64, descarga el [paquete ARM64](#) en su lugar. Si no está seguro de qué tipo de máquina tiene, abra el símbolo del sistema o PowerShell y escriba: `systeminfo | find "System Type"`.

**Advertencia:** En versiones de Windows que no están en inglés, es posible que tenga que modificar el texto de búsqueda, traduciendo la cadena "System Type" (Tipo de sistema). Es posible que también tenga que escapar las comillas del comando find. Por ejemplo, en alemán, `systeminfo | find`







## Desarrollo de Aplicaciones Web



El futuro digital  
es de todos

MinTIC

Misión  
TIC 2022



Hechos  
QUE CONECTAN 



```
PS C:\Windows\system32> dism.exe /online /enable-feature /featurename:VirtualMachinePlatform /all /norestart

Herramienta Administración y mantenimiento de imágenes de implementación
Versión: 10.0.19041.844

Versión de imagen: 10.0.19043.1889

Habilitando características
[=====100.0%=====]
La operación se completó correctamente.
PS C:\Windows\system32>
```



## Desarrollo de Aplicaciones Web



El futuro digital  
es de todos

MinTIC

```
PS C:\Users\LENOVO> wsl --set-default-version 2
Para información sobre las diferencias clave con WSL 2, visita https://aka.ms/ws12
La operación se completó correctamente.
PS C:\Users\LENOVO>
```



[Inicio](#) / [Herramientas de desarrollo](#) / Ubuntu 18.04 on Windows



## Ubuntu 18.04 on Windows

Canonical Group Limited

★★★★☆ 8 | [Herramientas de desarrollo](#)

Obtener en la aplicación de Microsoft Store

Gratis



Microsoft Store



### Ubuntu 18.04 on Windows

Canonical Group Limited

4,5 ★ [Términos y privacidad](#) [Más detalles](#)

 **TODOS**  
Clasificación por edades: ESRB. TODOS

Obtener

Detalles

[Disponible en 1 idioma](#)

Publicada por



Microsoft Store



## Ubuntu 18.04 on Windows

Canonical Group Limited

4,5 ★



Términos y privacidad



Más detalles



TODOS

Clasificación por edades: ESRB. TODOS

Instalar



Microsoft Store



## Ubuntu 18.04 on Windows

Canonical Group Limited

4,5 ★



Términos y privacidad



Más detalles



TODOS

Clasificación por edades: ESRB. TODOS

Abrir







```
Ubuntu 18.04.5 on Windows
Installing, this may take a few minutes...
WslRegisterDistribution failed with error: 0x800701bc
Error: 0x800701bc WSL 2 requiere una actualizaci n de su componente de kernel. Para obtener informaci n, visita https://aka.ms/wsl2kernel
Press any key to continue...
```



## Desarrollo de Aplicaciones Web



El futuro digital  
es de todos

MinTIC

```
Enter new UNIX username: edwing  
Enter new UNIX password:  
Retype new UNIX password:
```

Se debe reiniciar la computadora





Docker Desktop

Aplicación

## Configuración

🔍 Cambiar la vista de la Lupa >

## Buscar en Internet

🔍 dock - Ver resultados web >

# Get started with Docker in a few easy steps!

🕒 ESTIMATED TIME: 2 minutes

Start

[Skip tutorial](#)

We send usage statistics. Check your [privacy settings](#).



## Desarrollo de Aplicaciones Web



El futuro digital  
es de todos

MinTIC

```
PS C:\Users\LENOVO> docker run --name repo alpine/git clone https://github.com
/d
ocker/getting-started.git
Unable to find image 'alpine/git:latest' locally
213ec9aee27d: Pull complete
fd2d85b6e65f: Pull complete
72ec27a2c987: Pull complete
Digest: sha256:7b9c96bea8268579e925b17170ae208788e90d508ec334421384913ba61cd61
0
```





## Now, build the image

A Docker image is a private file system just for your container.  
It provides all the files and code your container needs.

```
cd getting-started
docker build -t docker101tutorial . >>
```

```
s
=> [app-zip-creator 3/4] COPY app/src ./src 0.4
s
=> [app-zip-creator 4/4] RUN apk add zip && zip -r /app.zip /app 3.3
s
=> [stage-6 2/3] COPY --from=app-zip-creator /app.zip /usr/share/nginx/h 0.2
s
=> [build 1/2] COPY . . 0.3
s
=> [build 2/2] RUN mdocs build 1.8
s
=> [stage-6 3/3] COPY --from=build /app/site /usr/share/nginx/html 0.2
s
=> exporting to image 0.2
s
=> => exporting layers 0.2
s
=> => writing image sha256:cb8844fa29bf51d3cbcf7370449303adcd0378e3b89 0.0
s
=> => naming to docker.io/library/docker101tutorial 0.0
s

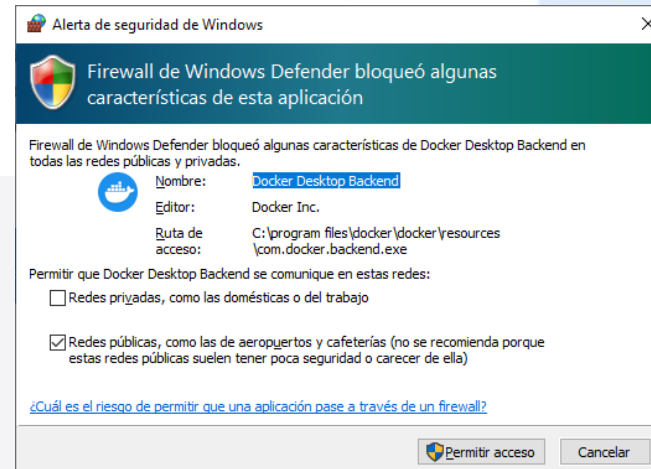
Use 'docker scan' to run Snyk tests against images to find vulnerabilities and
1
earn how to fix them
PS C:\Users\LENOVO\getting-started>
```



## Run your first container

Start a container based on the image you built in the previous step. Running a container launches your application with private resources, securely isolated from the rest of your machine.

```
docker run -d -p 80:80 \
  --name docker-tutorial docker101tutorial
```





## Now save and share your image

You must be signed in to Docker Hub to share your image.

[Sign in here.](#)

Save and share your image on Docker Hub to enable other users to easily download and run the image on any destination.

## Create a Docker Account.

Already have an account? [Sign In](#)

edwing

edwin\_garcia5@hotmail.com

••••••••••



- ☐ Send me occasional product updates and announcements.
- ☐ I agree to the [Subscription Service Agreement](#), [Privacy Policy](#) and [Data Processing Terms](#).

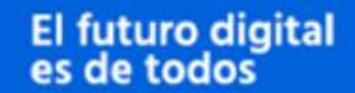
✓ No soy un robot



reCAPTCHA  
Privacidad • Términos

Sign Up





MinTIC

Thanks for creating a Docker ID. Please visit the button below.

## Verify email address

Containers

Images

Volumes

Dev Environments BETA

Extensions BETA

Add Extensions

## Containers

[Give Feedback](#)

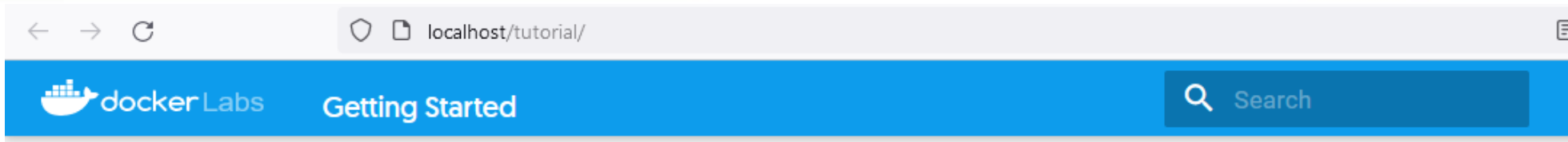
A container packages up code and its dependencies so the application runs quickly and reliably from one computing environment to another. [Learn more](#)

Showing 2 items

	NAME	IMAGE	STATUS	PORT(S)	STARTED	ACTIONS
<input type="checkbox"/>	<div>repo</div> <div>2e651dc02268</div>	<a href="#">alpine/git:latest</a>	Exited	-		<div>▶</div> <div>⋮</div> <div>🗑️</div>
<input type="checkbox"/>	<div>docker-tutorial</div> <div>789963ef30cb</div>	<a href="#">docker101tutorial:latest</a>	Running	80	5 minutes ago	<div>■</div> <div>⋮</div> <div>🗑️</div>



## Hechos QUE CONECTAN



## Getting Started

[Getting Started](#)

[Our Application](#)

[Updating our App](#)

[Sharing our App](#)

[Persisting our DB](#)

[Using Bind Mounts](#)

[Multi-Container Apps](#)

[Using Docker Compose](#)

[Image Building Best Practices](#)

[What Next?](#)

# Getting Started

## The command you just ran

Congratulations! You have started the container for this tutorial! Let's first explain the command that you just ran. In case you forgot, here's the command:

```
docker run -d -p 80:80 docker/getting-started
```

You'll notice a few flags being used. Here's some more info on them:

- `-d` - run the container in detached mode (in the background)
- `-p 80:80` - map port 80 of the host to port 80 in the container



## Desarrollo de Aplicaciones Web

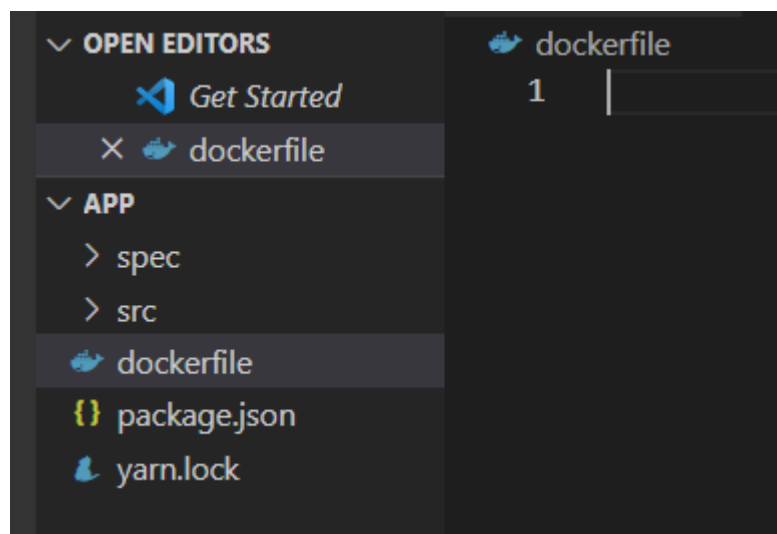


El futuro digital  
es de todos

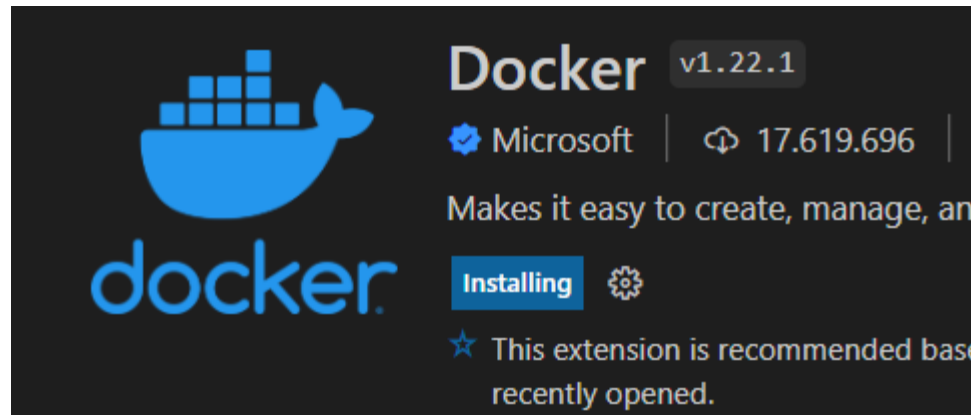
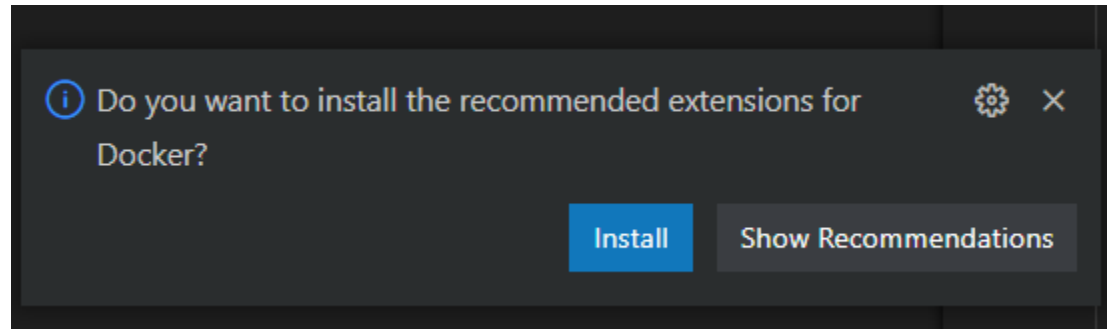
MinTIC

1. [Download the ZIP](#). Open the ZIP
2. Once extracted, use your favorite





```
dockerfile
1 FROM node:12-alpine
2 # Adding build tools to make yarn install work
3 RUN apk add --no-cache python2 g++ make
4 WORKDIR /app
5 COPY . .
6 RUN yarn install --production
7 CMD ["node", "src/index.js"]
```



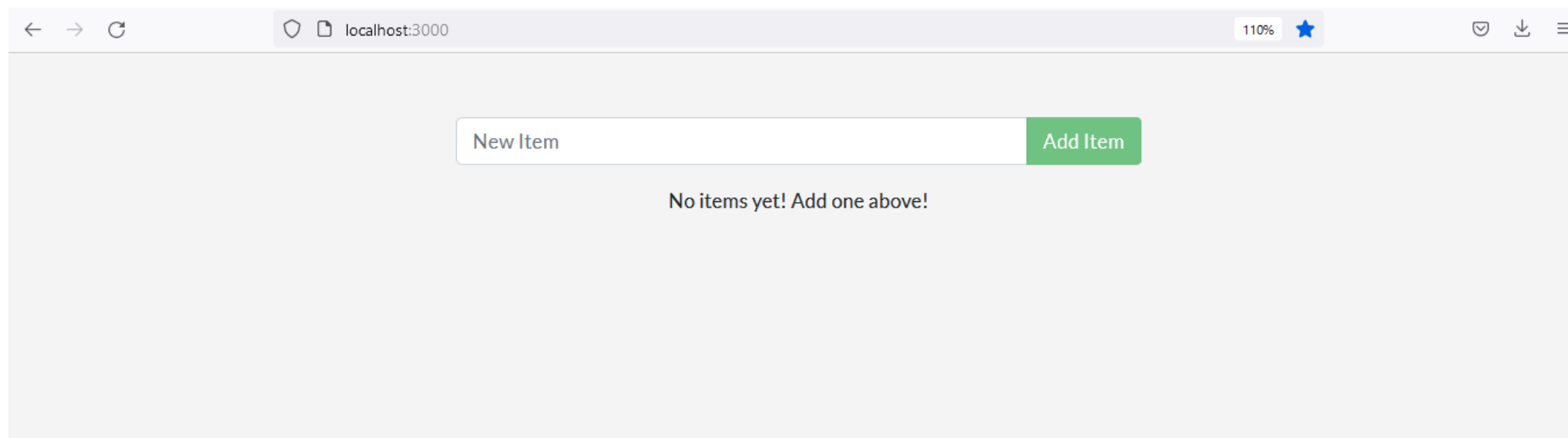


```
> docker build -t getting-started .
```

```
PS D:\UNAB\2022\Ciclo4\docker\app\app> docker build -t getting-started .  
[+] Building 4.7s (2/3)  
=> [internal] load build definition from Dockerfile  
=> => transferring dockerfile: 266B  
=> [internal] load .dockerignore  
=> => transferring context: 2B  
=> [internal] load metadata for docker.io/library/python:3.9-slim
```



```
PS D:\UNAB\2022\Ciclo4\docker\app\app> docker run -dp 3000:3000 getting-started  
49ebee8a102343b1af52f22d5bb4c7d957e53c75565e8d77982b419e6ddd7b20  
PS D:\UNAB\2022\Ciclo4\docker\app\app>
```







## Desarrollo de Aplicaciones Web



El futuro digital  
es de todos

MinTIC



festive\_galois

49ebee8a1023

[getting-started:latest](#)

Running

3000

2 minutes ago



Stop

Mision  
TIC 2022



Hechos  
QUE CONECTAN



- <https://www.youtube.com/watch?v=oD09MVpLZ-s>
- <https://sii-concatel.com/microservicios-contenedores-y-kubernetes/>
- <https://www.youtube.com/watch?v=-ksmE3KoX9U>
- <https://www.youtube.com/watch?v=9R2hFwIPGnQ>
- <https://www.netapp.com/es/devops-solu>
- <https://www.youtube.com/watch?v=kkfZs0vJFyUtions/what-are-containers/>
- <https://aws.amazon.com/es/getting-started/deep-dive-containers/>

# Bibliografía



El futuro digital  
es de todos

MinTIC

**Practicar los  
temas tratados**

Mision  
TIC2022



Hechos  
QUE CONECTAN

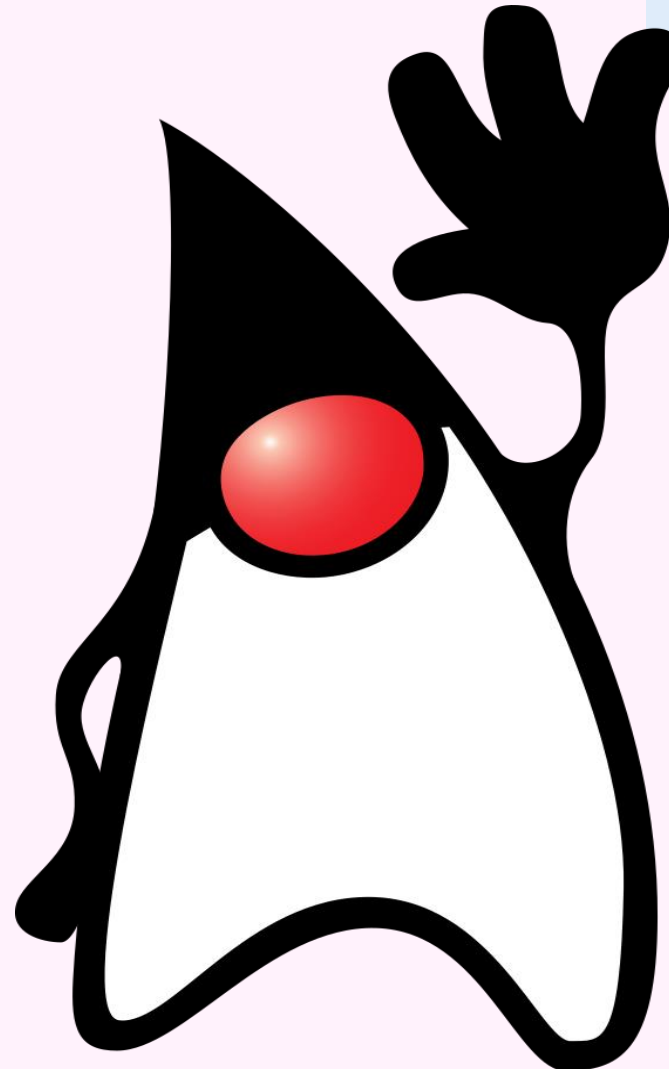
**Nos vemos en la siguiente sesión**



**El futuro digital  
es de todos**

**MinTIC**

**Muchas  
Gracias**



**[www.mintic.gov.co](http://www.mintic.gov.co)**

**Misión  
TIC2022**

**unab**  
VIGILADA MINEDUCACIÓN