# Creación de aplicativos en contendores con diferentes lenguajes de programación

# Creación de una contenedor con Ruby

### Aplicación básica

Para la creación de esta imagen primero crearemos el directorio donde se encontrarán y ejecutarán los archivos y comandos necesarios.

Nombre: ruby-app

Luego se creará el archivo de la aplicación básica que sólo mostrará un saludo.

Nombre: app.rb

Este será un servidor web mínimo que usa Sinatra, un framework simple en Ruby.

En el caso de esta aplicación de Ruby también necesitamos el archivo Gemfile el cual le dice a Ruby qué dependencias usar.

Nombre: Gemfile

### Dockerfile

Para la creación de la imagen será necesario un Dockerfile para poder construir la imagen paso por paso.

Nombre: Dockerfile

Dentro del archivo ser utiliza la versión ruby:3.2, se hace un cambio al directorio de trabajo /app y después se hace una copia de todos los archivos del directorio local hacia el WORKDIR definido en el paso anterior, luego se instalan las dependencias necesarias del Gemfile, se expone el puerto 4567 y se ejecuta el comando ruby app.rb para poder iniciar la aplicación.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

## Creación de la imagen Ruby

Para iniciar el proceso de construcción de la imagen es necesario usar el comando docker build, pero en este caso también queremos asignar una tag a la imagen, esto con el fin de poder hacer uso de mi repositorio de Docker Hub el comando final se vería así:

ruby-app> docker build -t lemendezc/unir-lemc:ruby .

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.



## Ejecución de la imagen Ruby

Para correr un contenedor basado en mi imagen podemos usar el comando docker run, para no quedar “bloquedos” ejecutaremos el comando en modo detached (en segundo plano), y finalmente mapearemos el puerto local al del contenedor. El comando se vería así:

ruby-app> docker run -d -p 4567:4567 lemendezc/unir-lemc:ruby

Para comprobar que funciona podemos usar el navegador que queramos (Chrome en mi caso) y acceder a la siguiente url http://localhost:4567

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

## Publicación de la imagen Ruby

Para subir la imagen en Docker Hub usaremos el comando docker push, con el nombre exacto de la imagen que subiremos. (Es importante iniciar session en la aplicación de Docker Desktop)

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

El comando para publicar la imagen es el siguiente:

ruby-app> docker push lemendezc/unir-lemc:ruby

Texto

El contenido generado por IA puede ser incorrecto.

# Creación de un contenedor con PHP

## Aplicación básica

Nombre del directorio: php-app

Aplicaicón básica que solo mostrará un saludo: index.php

## Dockerfile

Dentro del archivo se utiliza la imagen oficial de PHP versión 8.2-cli, después se copia el archivo index.php al directorio /var/www.html del contenedor. Luego se establece ese mismo directorio como WORKDIR, se expone el puerto 8081 y se ejecuta el servidor embebido de PHP para atender peticiones en todas las interfaces (0.0.0.0).

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

## Creación de la imagen PHP

php-app> docker build -t lemendezc/unir-lemc:php .

Texto

El contenido generado por IA puede ser incorrecto.

## Ejecución de la imagen PHP

php-app> docker run -d -p 8081:8081 lemendezc/unir-lemc:php



Comprobación de que funciona correctamente abriremos un navegador y accederemos a la siguiente dirección <http://localhost:8081>

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

## Publicación de la imagen PHP

php-app> docker push lemendezc/unir-lemc:php

# Captura de pantalla de computadora El contenido generado por IA puede ser incorrecto.Creación de un contenedor con Perl

## Aplicación básica

Nombre del directorio: perl-app

Aplicación básica que solo mostrará un saludo: app.pl

## Dockerfile

Dentro del archivo se utiliza la imagen oficial de Perl. Luego se instalan los módulos necesarios mediante cpan y cpanm, en especial Dancer2. Posteriormente, se copia el archivo de la aplicación al directorio /app del contenedor, se expone el puerto 3000 y finalmente se ejecuta el archivo app.pl como punto de entrada de la aplicación.

Pantalla de un video juego

El contenido generado por IA puede ser incorrecto.

## Creación de la imagen Perl

perl-app> docker build -t lemendezc/unir-lemc:perl .

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

## Ejecución de la imagen Perl

perl-app> docker run -d -p 3000:3000 lemendezc/unir-lemc:perl



Comprobación de que funciona correctamente abriremos un navegador y accederemos a la siguiente dirección <http://localhost:3000>

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

## Publicación de la imagen Perl

perl-app> docker push lemendezc/unir-lemc:perl

Texto

El contenido generado por IA puede ser incorrecto.

# Creación de un contenedor con Elixir

## Aplicación básica

Nombre del directorio: elixir-app

Aplicación básica que solo mostrará un saludo: hello\_web.ex

Archivos extra necesarios: mis.exs, /lib/hello\_web/router

## Dockerfile

En este contenedor se ha desarrollado una aplicación web básica utilizando Plug y Cowboy, dos bibliotecas ligeras y ampliamente utilizadas dentro del ecosistema Elixir. Se expondrá su ejecución en el puerto 4000.

## Creación de la imagen Elixir

elixir-app> docker build -t lemendezc/unir-lemc:elixir .

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

## Ejecución de la imagen Elixir

elixir-app> docker run -d -p 4000:4000 lemendezc/unir-lemc:elixir



Comprobación de que funciona correctamente abriremos un navegador y accederemos a la siguiente dirección <http://localhost:4000>

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

## Publicación de la imagen Elixir

elixir-app> docker push lemendezc/unir-lemc:elixir

Texto

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.