

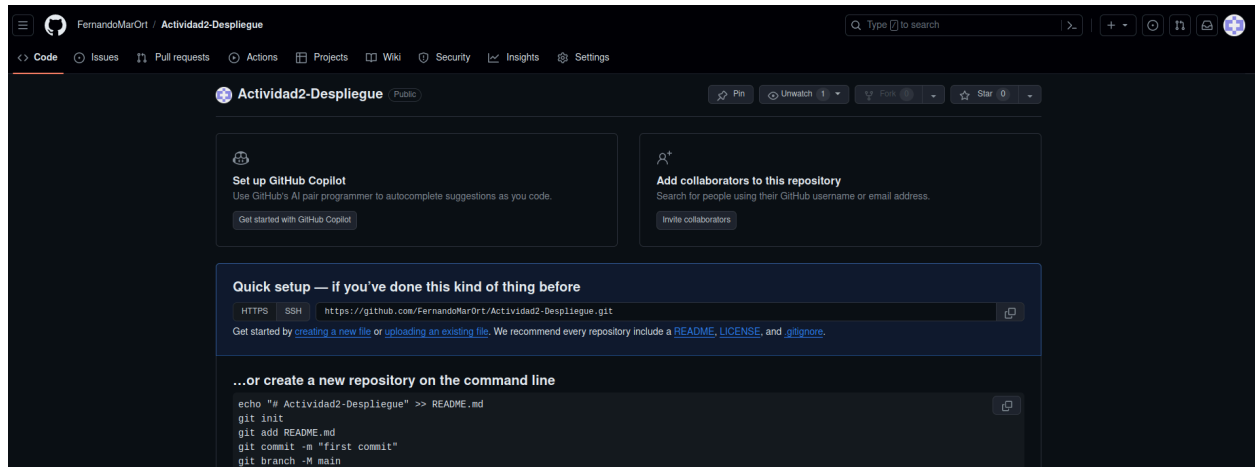
Actividad 2 - Despliegue de una Aplicación React en un Contenedor Docker con Integración de GitHub

Fernando Martinez Ortiz

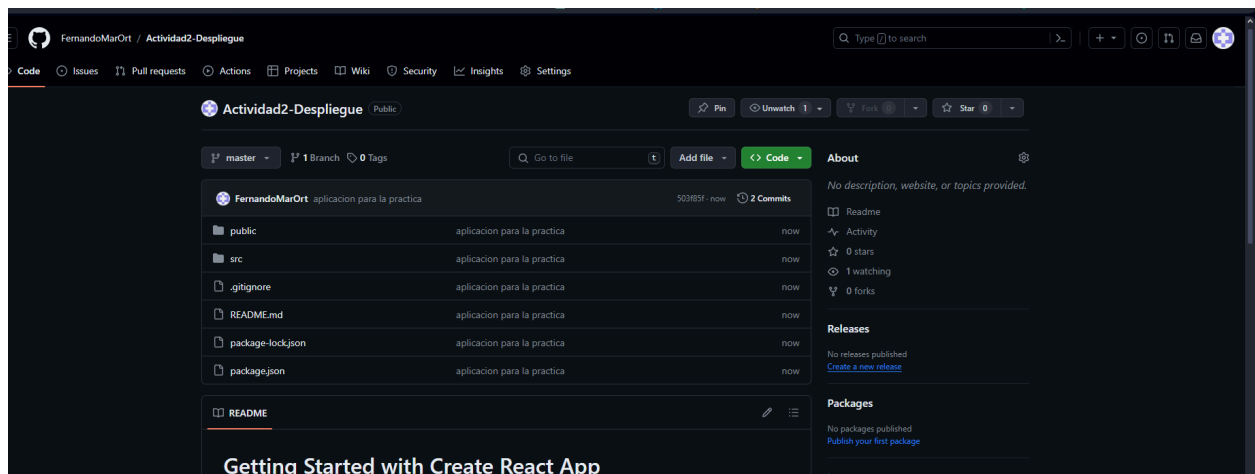
2º DAW 2023/24

Paso 1: Creación de repositorio

Creo el repositorio de github donde subire la aplicación de React



Clono el repositorio proporcionado en esta práctica



Paso 2: Creación del Dockerfile

Utilizo de base la imagen de node:alpine

Creo el directorio de trabajo /app

Instalo git y otras dependencias que voy a necesitar

Clono el repositorio de mi github

Copio los archivos de configuración (package.json y package-lock.json).

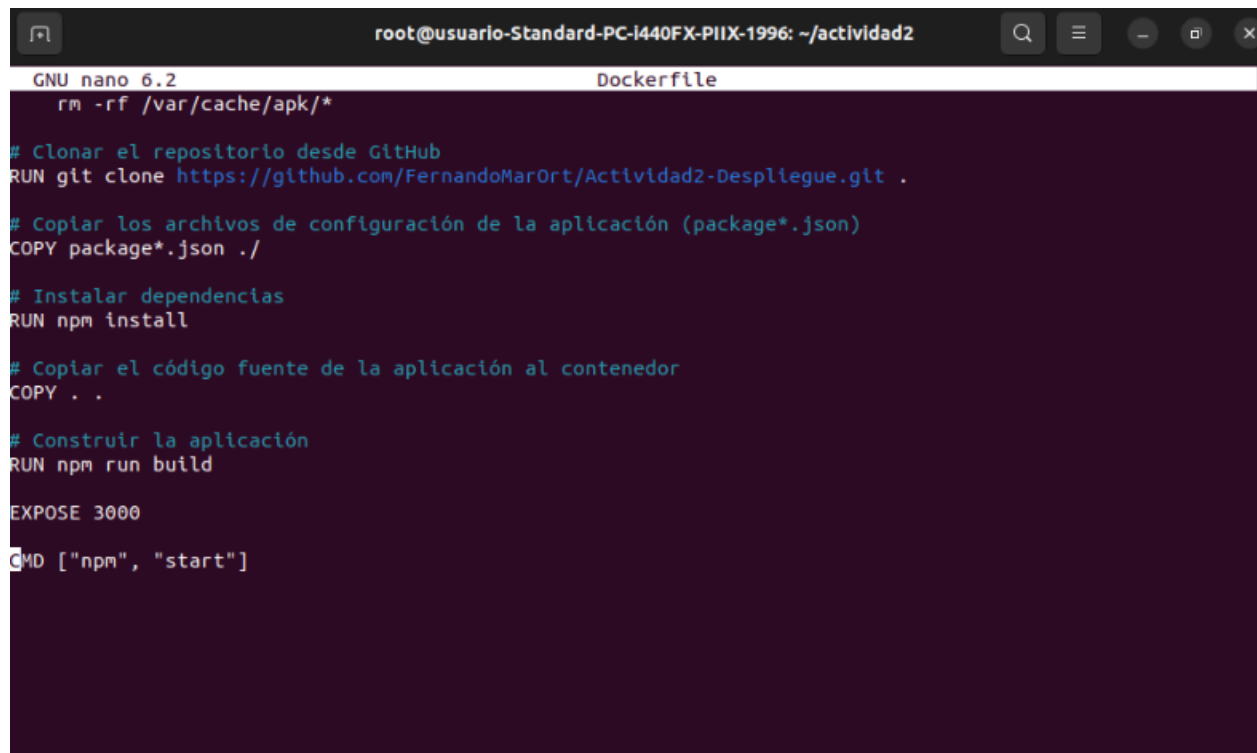
Instalo las dependencias necesarias de Node.js.

Copio el código fuente de la aplicación.

Construyo la aplicación para luego ejecutarla en el navegador

Expongo el puerto 3000 por el cual acceder a la aplicación

Defino el comando para iniciar la aplicación.

A screenshot of a terminal window titled "root@usuario-Standard-PC-I440FX-PIIX-1996: ~/actividad2". The terminal shows the content of a Dockerfile being edited with GNU nano 6.2. The Dockerfile instructions are:

```
rm -rf /var/cache/apk/*  
  
# Clonar el repositorio desde GitHub  
RUN git clone https://github.com/FernandoMarOrt/Actividad2-Despliegue.git .  
  
# Copiar los archivos de configuración de la aplicación (package*.json)  
COPY package*.json ./  
  
# Instalar dependencias  
RUN npm install  
  
# Copiar el código fuente de la aplicación al contenedor  
COPY . .  
  
# Construir la aplicación  
RUN npm run build  
  
EXPOSE 3000  
  
CMD ["npm", "start"]
```

Paso 3: Construcción del Contenedor Docker

Construyo la imagen para el contenedor que usare

```
root@usuario-Standard-PC-i440FX-PIIX-1996: ~/actividad2
successfully: exit code: 127
root@usuario-Standard-PC-i440FX-PIIX-1996:~/actividad2# nano Dockerfile
root@usuario-Standard-PC-i440FX-PIIX-1996:~/actividad2# docker build -t imagenreact .
[+] Building 180.6s (13/13) FINISHED
=> [internal] load build definition from Dockerfile
=> == transferring dockerfile: 600B
=> [internal] load metadata for docker.io/library/node:alpine
=> [internal] load .dockerignore
=> == transferring context: 2B
=> [1/8] FROM docker.io/library/node:alpine@sha256:d3271e4bd89eec4d97087
=> [internal] load build context
=> == transferring context: 600B
=> CACHED [2/8] WORKDIR /app
=> [3/8] RUN apk update && apk add --no-cache git && rm -rf /var
=> [4/8] RUN git clone https://github.com/FernandoMarOrt/Actividad2-Desp
=> [5/8] COPY package*.json ./
=> [6/8] RUN npm install
=> [7/8] COPY . .
=> [8/8] RUN npm run build
=> exporting to image
=> == exporting layers
=> == writing image sha256:032be852d82964cb4b007c3db8ccf1b0d73827f05aa60
=> == naming to docker.io/library/imagenreact
```

Paso 4: Ejecución de la Aplicación en Docker

Pongo a funcionar el contenedor de docker creado con la imagen anterior en el puerto 3001 y compruebo que se ejecuta la aplicación de react correctamente en el navegador

```
root@usuario-Standard-PC-i440FX-PIIX-1996:~/actividad2# docker run -d -p 3001:3000 imagenreact
786ff5d10d62ef649c5640ba1f619292f3286c9779da767a4b22881ab56e2e46
```



AVERIGUA TU FOTOTIPO

1 - ¿Cuál es el color natural de tu piel cuando no está bronceada?

Rojiza, blanca

Blanca-beige

Beige

Marrón clara

```
root@usuario-Standard-PC-i440FX-PIIX-1996: ~/actividad2
00 imagenreact
786ff5d10d62ef649c5640ba1f619292f3286c9779da767a4b22881ab56e2e46
root@usuario-Standard-PC-i440FX-PIIX-1996:~/actividad2# nano Dockerfile
root@usuario-Standard-PC-i440FX-PIIX-1996:~/actividad2# docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS
PORTS
786ff5d10d62   imagenreact    "docker-entrypoint.s..." 3 minutes ago  Up 3 minut
es           0.0.0.0:3001->3000/tcp, :::3001->3000/tcp
ecstatic_sande
rson
dc4806b5f108   imagenjen      "/usr/bin/tini -- /u..." 2 days ago    Up 2 days
0.0.0.0:3000->3000/tcp, :::3000->3000/tcp, 0.0.0.0:5000->5000/tcp, :::5000-
>5000/tcp, 50000/tcp, 0.0.0.0:8084->8080/tcp, :::8084->8080/tcp   jenkins
```

Enlace al repositorio de github

<https://github.com/FernandoMarOrt/Actividad2-Despliegue.git>