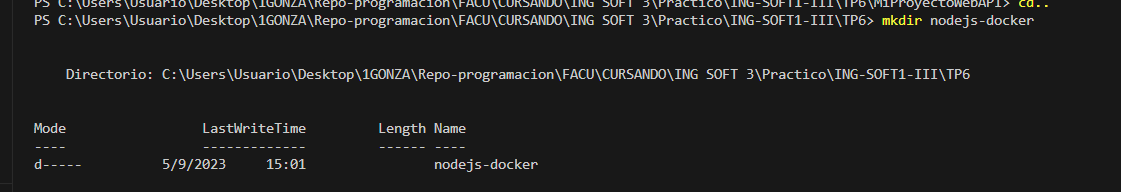
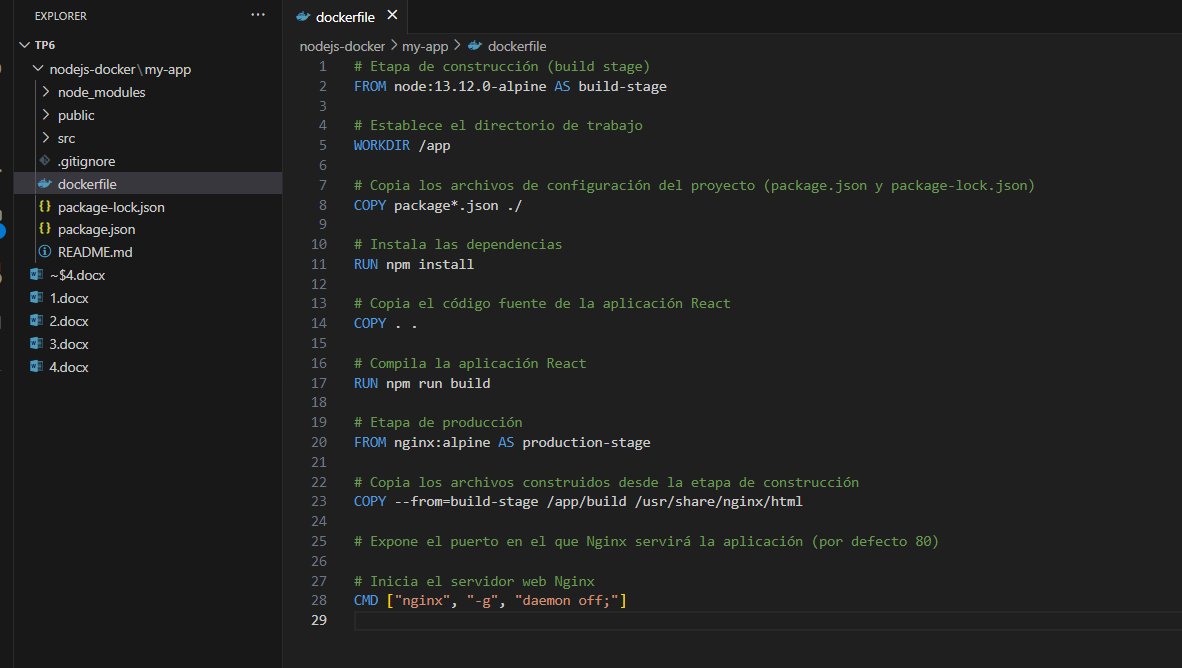
**4- Imagen para aplicación web en Nodejs**

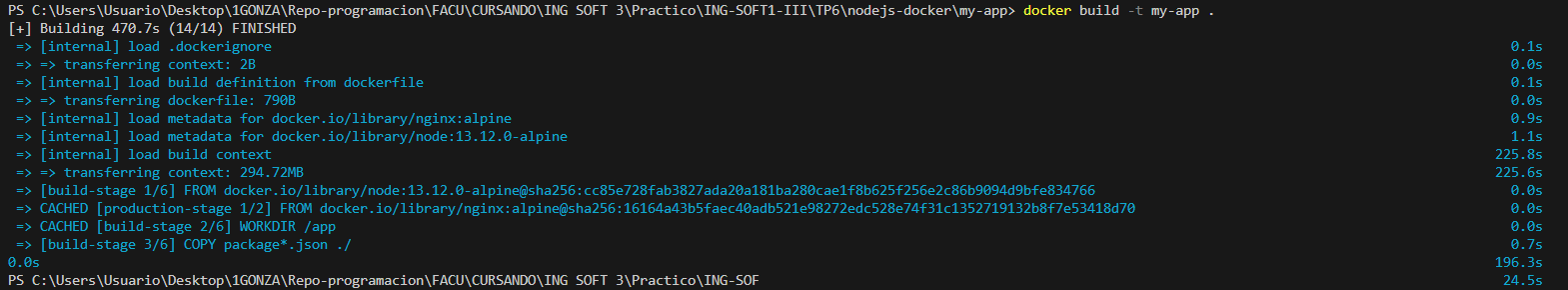
* Crear una la carpeta trabajo-practico-06/nodejs-docker



* Generar un proyecto siguiendo los pasos descriptos en el trabajo práctico 5 para Nodejs
* Escribir un Dockerfile para ejecutar la aplicación web localizada en ese directorio
  + Idealmente que sea multistage, con una imagen de build y otra de producción.
  + Usar como imagen base **node:13.12.0-alpine**
  + Ejecutar **npm install** dentro durante el build.
  + Exponer el puerto 3000



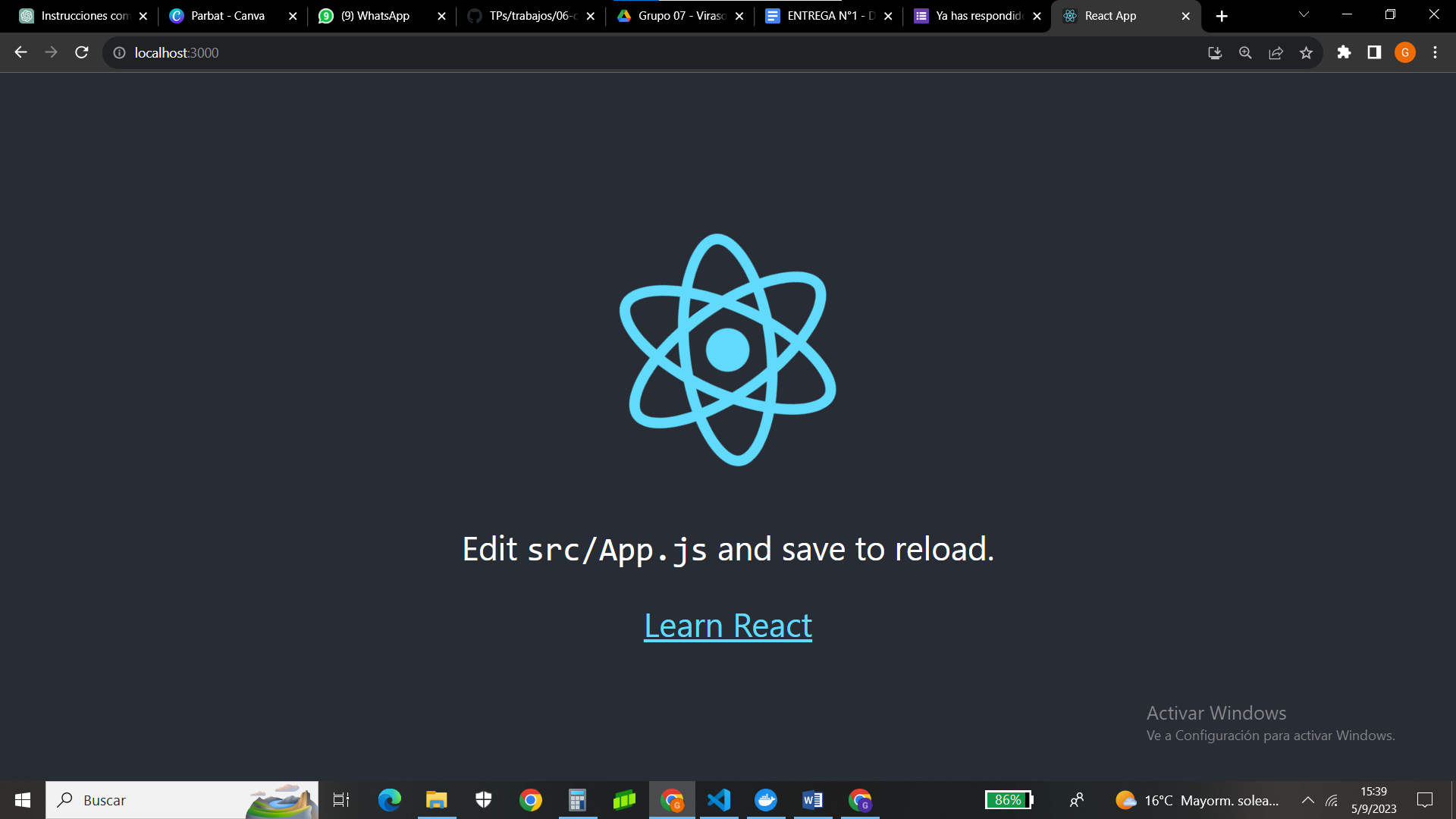
* Hacer un build de la imagen, nombrar la imagen **test-node**.



* Ejecutar la imagen **test-node** publicando el puerto 3000.



* Verificar en [http://localhost:3000](http://localhost:3000/) que la aplicación está funcionando.



* Proveer el Dockerfile y los comandos ejecutados como resultado de este ejercicio.

Dockerfile:  
  
# Etapa de construcción (build stage)

FROM node:13.12.0-alpine AS build-stage

# Establece el directorio de trabajo

WORKDIR /app

# Copia los archivos de configuración del proyecto (package.json y package-lock.json)

COPY package\*.json ./

# Instala las dependencias

RUN npm install

# Copia el código fuente de la aplicación React

COPY . .

# Compila la aplicación React

RUN npm run build

# Etapa de producción

FROM nginx:alpine AS production-stage

# Copia los archivos construidos desde la etapa de construcción

COPY --from=build-stage /app/build /usr/share/nginx/html

# Expone el puerto en el que Nginx servirá la aplicación (por defecto 80)

# Inicia el servidor web Nginx

CMD ["nginx", "-g", "daemon off;"]