**Dockerize  
Café Employee Management**  
  
**.NET 8 Application**

# Dockerfile for .NET 8 Application

# Use the .NET 8 SDK image for building the app

FROM mcr.microsoft.com/dotnet/sdk:8.0 AS build

# Set working directory

WORKDIR /app

# Copy the .NET project file and restore dependencies

COPY CafeEmployee/CafeEmployee.csproj ./

RUN dotnet restore

# Copy the rest of the .NET code

COPY CafeEmployee/ .

# Build and publish the .NET application

RUN dotnet publish -c Release -o out

# Use the .NET runtime image for running the app

FROM mcr.microsoft.com/dotnet/aspnet:8.0 AS runtime

# Set working directory

WORKDIR /app

# Copy the published .NET application from the build stage

COPY --from=build /app/out .

# Expose port 26655

EXPOSE 26655

# Start the .NET application

ENTRYPOINT ["dotnet", "CafeEmployee.dll"]

**React Application**

# Stage 1: Build the React app

FROM node:18 AS build

# Set working directory

WORKDIR /app

# Copy package.json and install dependencies

COPY cafe-employee-manage/package\*.json ./

RUN npm install

# Copy the rest of the React code

COPY cafe-employee-manage/ .

# Build the React app

RUN npm run build

# Stage 2: Serve the React app

FROM nginx:alpine

# Copy the built React app from the build stage

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

# Expose port 80 (default for Nginx)

EXPOSE 80

# Use default Nginx configuration

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

**docker-compose**

version: '3.8'

services:

  react-app:

    build:

      context: .

      dockerfile: cafe-employee-manage/Dockerfile

    ports:

      - "5173:80"

  dotnet-app:

    build:

      context: .

      dockerfile: CafeEmployee/Dockerfile

    ports:

      - "26655:26655"

**Steps to Build and Run:**

1. **Navigate to the Project Root:**

sh

Copy code

cd /path/to/project-root

1. **Build and Run Containers:**

sh

Copy code

docker-compose up --build

1. **Access Your Applications: (Note you can change your local host post according to your machine)**
   * **React Application:** [http://localhost:5173](http://localhost:5173/)
   * **.NET Application:** [http://localhost:26655](http://localhost:26655/)