Sprawozdanie:

Kod app.js

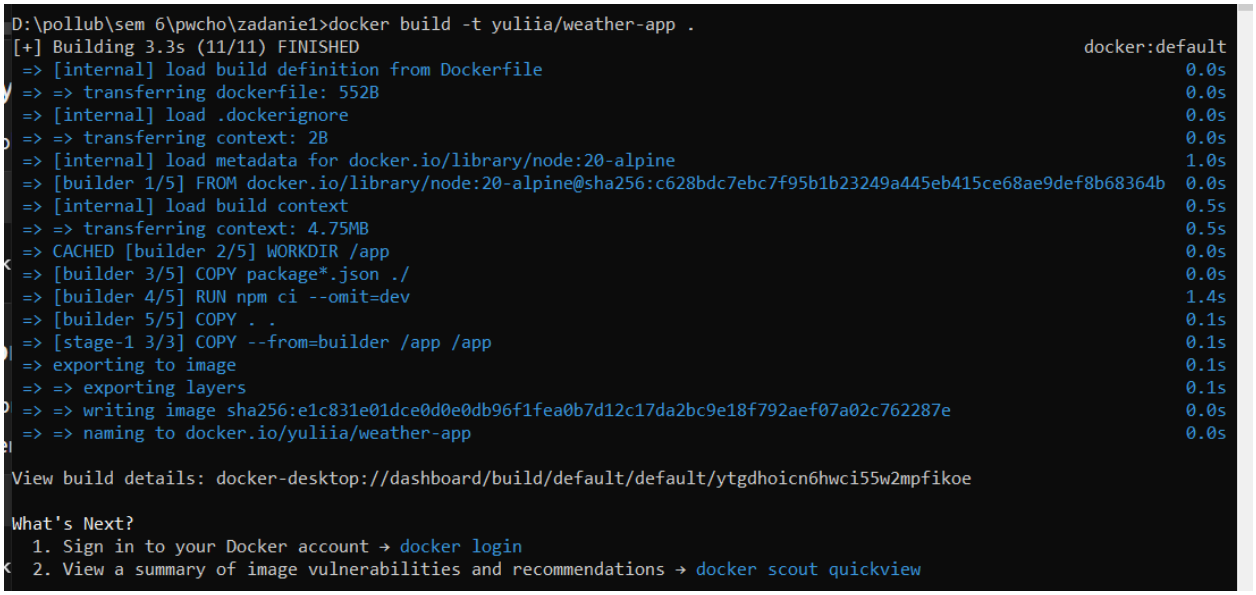
```
const express = require('express'); // Import biblioteki Express, ktora ulatwia tworzenie serwera HTTP w Node.js.
const axios = require('axios'); // Import biblioteki Axios, która sluzy do wykonywania zapytan HTTP (np. do API pogodowego).
const dotenv = require('dotenv');// Import biblioteki dotenv, ktora pozwala wczytywac zmienne srodowiskowe z pliku `.env`.
const app = express(); //Tworzy nowa app Express
dotenv.config();//Wczytuje zmienne z pliku `.env`


//port i klucz api pobierane z pliku .env
const PORT = process.env.PORT || 3000;
const API_KEY = process.env.API_KEY;

//predefiniowana lista krajow i miast
const cities = {
  Poland: ['Warsaw', 'Gdańsk', 'Lublin',],
  Germany: ['Berlin', 'Munich']
};

//uzycie folderu do statycznych plikow (index.html)
app.use(express.static('public'));
app.use(express.urlencoded({ extended: true }));

app.post('/weather', async (req, res) => {
  const { country, city } = req.body; //pobiera dane z formularza
  const url = `https://api.openweathermap.org/data/2.5/weather?q=${city},${country}&appid=${API_KEY}&units=metric`;
  try {
    const response = await axios.get(url);// Wysyła zapytanie GET do API i czeka na odpowiedź.
    const weather = response.data;//zapis danych z odp
    res.send(`<h1>Weather in ${city}, ${country}</h1><p>${weather.weather[0].description}, temp: ${weather.main.temp}°C</p>`);
     // Wyświetla prosty HTML z opisem pogody i temperaturą.
  } catch {
    res.send(`<p>Error getting weather for ${city}</p>`);
  }
});

app.listen(PORT, () => { //uruchamia serwer i nasluchuje na okreslonym porcie
  console.log(`App started on port ${PORT}`);
  console.log(`Author: Yuliia Kozlova`);
  console.log(`Start date: ${new Date().toISOString()}`);
});
```

Plik Dockerfile

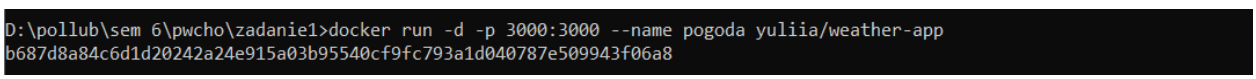
```
#etap 1: Budowanie
FROM node:20-alpine AS builder
WORKDIR /app
COPY package*.json ./

#instalacja tylko zaleznosci produkcyjnych
RUN npm ci --omit=dev
COPY . .

#etap 2: Minimalny obraz
FROM node:20-alpine
LABEL org.opencontainers.image.authors="Yuliia Kozlova"
WORKDIR /app
COPY --from=builder /app /app
ENV NODE_ENV=production
EXPOSE 3000
HEALTHCHECK --interval=30s --timeout=10s --start-period=5s --retries=3 \
 CMD wget --spider http://localhost:3000 || exit 1
CMD ["node", "app.js"]
```

Zrzuty :

a) Zbudowania opracowanego obrazu kontenera



b) uruchomienia kontenera na podstawie zbudowanego obrazu



c) sposobu uzyskania informacji z logów, które wygenerowałą opracowana aplikacja podczas uruchamiana kontenera

```
D:\pollub\sem 6\pwcho\zadanie1>docker logs pogoda
App started on port 3000
Author: Yuliia Kozlova
Start date: 2025-05-07T17:35:04.533Z
```

```
D:\pollub\sem 6\pwcho\zadanie1>docker image inspect yuliia/weather-app --format='{{.RootFS.Layers}}'
'[sha256:08000c18d16dadf9553d747a58cf44023423a9ab010aab96cf263d2216b8b350 sha256:75228a052fed2baf07b8652c14c8aba95480f15
6a027bf287f1ad44237d4ca9a sha256:2c9cabe102898883f748afd7f07319db1bcfeed1124c768394ba01d122cd526f sha256:b8185dcc9be5529
abfdea9590093f7474f53a9cbc91a3adf3fbebb3a42a83ec4 sha256:9e3b4e3e7cbe2da94d86fd543d92c4775d8cea3e35b3045b676662eedab48d1
f sha256:0fb2e2f690752c86d36284dc06be15f4c007948e44def06c65ceea6dc064d76f]'

D:\pollub\sem 6\pwcho\zadanie1>docker image ls yuliia/weather-app
REPOSITORY          TAG       IMAGE ID       CREATED         SIZE
yuliia/weather-app  latest    e1c831e01dce   4 minutes ago   139MB

D:\pollub\sem 6\pwcho\zadanie1>
```