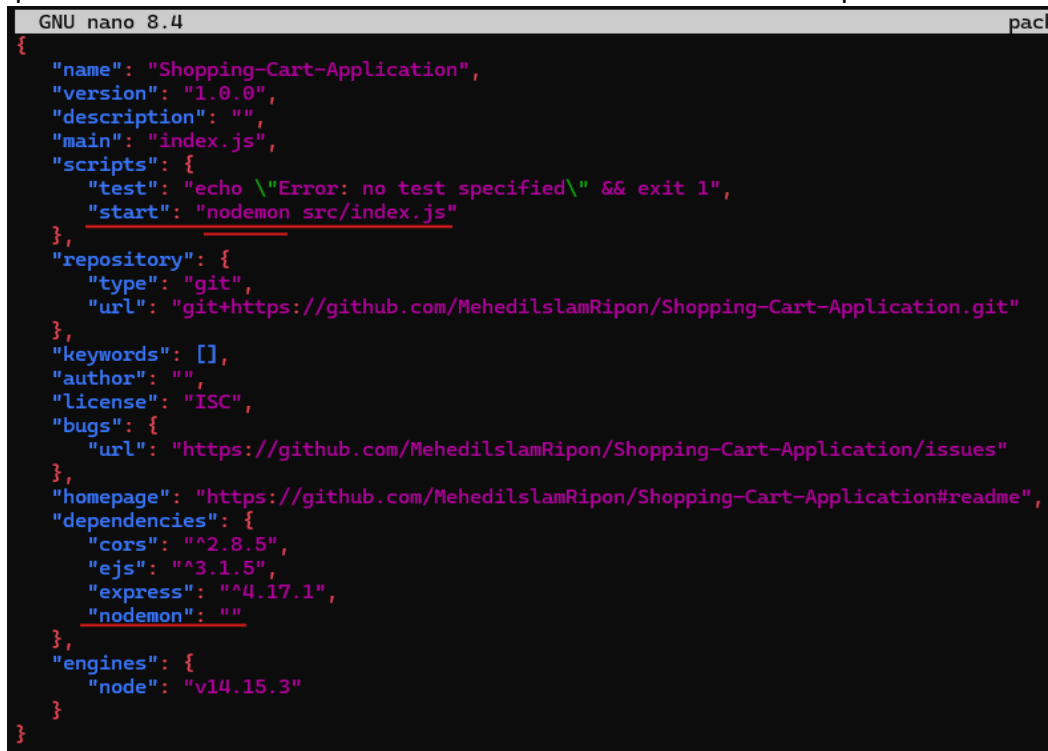


Actividades

- 1 Buscad cómo solucionar este problema y, tras ello, iniciad la aplicación sin problemas.

¿Qué comando habéis usado para solucionar el fallo anterior? ¿Cuál es su cometido? ¿Qué archivo se ha modificado al ejecutarlo? Documentadlo todo en el informe de la práctica.

Resolví este problema añadiendo la herramienta «nodemon» a la lista de dependencias.



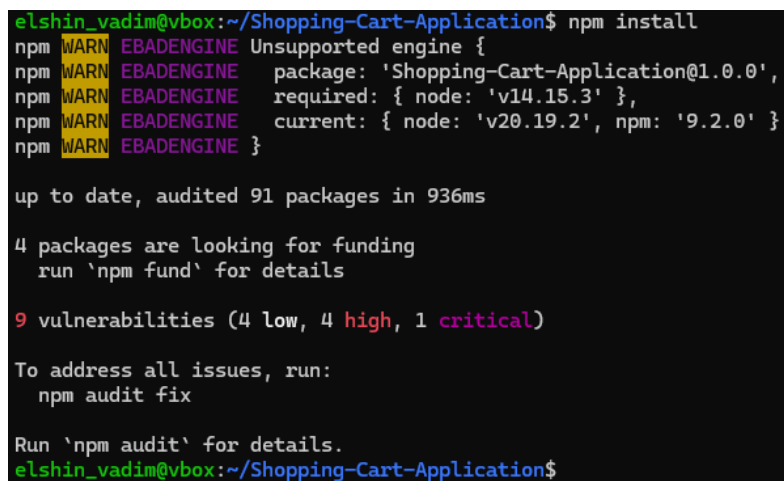
```
GNU nano 8.4 package.json
{
  "name": "Shopping-Cart-Application",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1",
    "start": "nodemon src/index.js"
  },
  "repository": {
    "type": "git",
    "url": "git+https://github.com/MehediIslamRipon/Shopping-Cart-Application.git"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "bugs": {
    "url": "https://github.com/MehediIslamRipon/Shopping-Cart-Application/issues"
  },
  "homepage": "https://github.com/MehediIslamRipon/Shopping-Cart-Application#readme",
  "dependencies": {
    "cors": "^2.8.5",
    "ejs": "^3.1.5",
    "express": "^4.17.1",
    "nodemon": ""
  },
  "engines": {
    "node": "v14.15.3"
  }
}
```

El problema se debía a que el script «start» utilizaba nodemon, que no estaba instalado.

Nodemon es una herramienta de línea de comandos que facilita el desarrollo rápido de aplicaciones Node.js. Supervisa el directorio del proyecto y reinicia automáticamente la aplicación Node cuando detecta algún cambio.

Utilizando nuevamente el comando npm install, instalé nodemon y pude ejecutar el script "start".

- 2 Documenta, incluyendo capturas de pantallas, el proceso que has seguido para realizar el despliegue de esta nueva aplicación, así como el resultado final.



```
elshin_vadim@vbox:~/Shopping-Cart-Application$ npm install
npm WARN EBADENGINE Unsupported engine {
  package: 'Shopping-Cart-Application@1.0.0',
  required: { node: 'v14.15.3' },
  current: { node: 'v20.19.2', npm: '9.2.0' }
}

up to date, audited 91 packages in 936ms

4 packages are looking for funding
  run `npm fund` for details

9 vulnerabilities (4 low, 4 high, 1 critical)

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
elshin_vadim@vbox:~/Shopping-Cart-Application$
```

```

elshin_vadim@vbox:~/Shopping-Cart-Application$ npm run start
> Shopping-Cart-Application@1.0.0 start
> nodemon src/index.js

[nodemon] 3.1.10
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node src/index.js`
server is running on PORT: 3000

```



3 Cuando ejecutáis el comando `npm run start`, lo que estáis haciendo es ejecutar un script:

- ¿Dónde podemos ver que script se está ejecutando?

Dentro del archivo `package.json`

```

GNU nano 8.4      ../package.json
{
  "name": "Shopping-Cart-Application",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "nodemon src/index.js"
  },
}

```

- ¿Qué comando está ejecutando?

Nodeman `src/index.js`

```

elshin_vadim@vbox: ~/Shopp  ×  +  ▾
GNU nano 8.4      index.js
// dependencies
const express = require("express");
const cors = require("cors");
const path = require("path");
const router = require("./routes");

const app = express();

app.use(cors());
app.set("view engine", "ejs");
app.set("views", path.join(__dirname, "views"));
app.use(express.static(path.join(__dirname, "../", "public")));

// server PORT
const PORT = 3000;

// basic routes
router(app);
app.get("/", (req, res) => {
  res.send("<h1> Hello World </h1>");
});

// middleware
app.use((req, res, next) => {
  const error = new Error();
  error.status = 404;
  next(error);
});

app.use((error, req, res, next) => {
  if (error.status === 404) {
    res.status(404);
    return res.render("errors/404");
  }

  res.status(500);
  res.render("error/500");
});

// start server
app.listen(PORT, () => {
  console.log(`server is running on PORT: ${PORT}`);
});

```