

Desplegament d'una aplicaci3 basada en Node.js



Índex

1. Introducció.....	3
2. Desenvolupament.....	3
3. Conclusions.....	6
4. Bibliografia.....	6

1. Introducció

Node.js és un entorn d'execució de JavaScript que permet executar codi del costat del servidor de manera eficient i escalable. És utilitzat per a construir aplicacions web ràpides i lleugeres, fent ús d'un model asíncron basat en esdeveniments.

En aquesta tasca, desplegarem una API creada amb Node.js en un servidor VPS. Seguirem un procés estructurat que inclou la configuració del sistema, la instal·lació de Node.js, la gestió de bases de dades i la configuració del servidor web per garantir que l'API estiga disponible en un entorn de producció.

2. Desenvolupament

1. Clonar repositori remot

Clonem el repositori per a tenir una còpia local del projecte, cosa que ens permet modificar-lo i executar-lo en el nostre entorn:

```
git clone git@github.com:corriol/stockflow-api.git
cd stockflow-api
```

2. Instal·lar Node.js

Instal·lem Node.js en la versió 22 perquè és necessari per executar aplicacions desenvolupades en aquest entorn:

```
nvm install 22 && nvm use 22
```

- Verifiquem que la instal·lació s'ha fet correctament:

```
node -v
```

3. Desplegar l'aplicació en local

Instal·lem les dependències necessàries per al funcionament de l'aplicació:

```
npm install
```

- Creem la base de dades per emmagatzemar la informació de l'API:

```
sudo mysql -u gabi -p
CREATE DATABASE `stockflow-api`
```

- Comprovem la base de dades

```
SHOW DATABASES
```

- Configurem les variables d'entorn per a indicar les credencials i configuracions necessàries dins del fitxer `.env`:

```
# Clau secreta per a generar/verificar tokens JWT
TOKEN_SECRET=tamarequanpixafacлот

# Configuració de la Base de dades
SQL_DBNAME=stockflow-api
SQL_HOST=localhost
SQL_USER=gabi
SQL_PASSWORD=gabi

#Credencials de l'aplicatiu i el port on s'executa
USER=nususa
PASSWORD=mosusa
PORT=3003
```

- Importem la base de dades dins de la configuració de la carpeta clonada anteriorment:

```
sudo mysql -u gabi -p 'stockflow-api' < config/stockflow-api.sql
```

- Executem l'aplicatiu per verificar que tot funcione correctament:

```
npm start
```

- Accedim a **localhost:3003** i comprovem el seu correcte funcionament.

4. Configurar VPS

Creem la base de dades en el VPS per emmagatzemar la informació de l'aplicació en producció i importem el fitxer **stockflow-api.sql**.

- Creem la base de dades en el VPS anant a **Dashboard > Databases > Add Database**;
- Omplim nom de la base de dades:

```
# Nom de la base de dades
stockflow-api-gabi

# Usuari
gabi

# Contrassenya (la guardarem per a després)
gabi
```

- Anem a la base de dades que acabem de crear i apremem **Import Dump**. i importem el fitxer **.sql** del nostre projecte, **stockflow-api.sql**.

5. Crear usuari FTP al VPS

- Ara creem un usuari FTP al VPS per a poder pujar fitxers utilitzant serveis de transferència com **FileZilla**.
- Anem a **Dashboard > FTP > Add an FTP Account**.
- Omplim els camps de nom de compte i contrasenya i configurem el **Home directory** per a que siga la carpeta **.httpdocs**.

6. Pujar l'aplicació web

- Utilitzem **FileZilla** per a transferir l'aplicació al servidor i garantir que els fitxers siguin accessibles.

7. Desplegar l'aplicació

- Accedim al VPS i anem a **Dashboard > Node.js**.
- Configurem el servidor VPS perquè l'aplicació s'execute correctament:

```
Node.js Version: 23.7.0
Package Manager: npm
Document Root: /httpdocs
Application Mode: production
Application URL: http://alu06.daw.iesevalorpego.es
Application Root: /httpdocs
Application Startup File: server.js
Custom environment variables: specify
```

- Fem clic en el botó de **NPM install**.
- Modifiquem el fitxer **.env** per adaptar-lo al l'entorn de producció:

```
TOKEN_SECRET=tamarequanpixafac lot
SQL_DBNAME=stockflow-api-gabi
SQL_HOST=localhost
SQL_USER=gabi
SQL_PASSWORD=gabi
USER=nususa
PASSWORD=mosusa
PORT=3003
```

8. Comprovació

- Per a comprovar que tot funcione utilitzarem l'extensió de VS Code **Thunder Client**.
- Fem clic **New Request** i introduïm la url amb el endpoint:
alu06.daw.iesevalorpego.es/api/login.
- Canviem el tipus de sol·licitud a **POST** i li donem a **Send**.
- Si ens retorna un JSON d'aquest tipus significa que tot està configurat correctament:

```
{
  "error": null,
  "token": {
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."
  }
}
```

3. Conclusions

He tingut complicacions al començar l'activitat perquè havia acceptat la sol·licitud del repositori tard i no em deixava accedir.

4. Bibliografia

- GitHub [en línia] <https://github.com/corriol/stockflow-api/> [data de consulta: 12/02/25]
- Node.js [en línia] <https://nodejs.org/> [data de consulta: 12/02/25]