**Ghid de instalare și configurare**

**(Varianta 1 – folosind codul sursă)**

**1. Clonarea repository-ului**

git clone https://github.com/cosminsilviu27/Generare\_Adeverinte\_USV.git

cd Generare\_Adeverinte\_USV

**2. Configurarea backend-ului Django**

cd backend

**2.1 Crearea și activarea unui mediu virtual**

python -m venv env

source env/bin/activate

**2.2 Instalarea dependințelor din requirements.txt**

pip install -r requirements.txt

**2.3 Configurarea variabilelor de mediu**

Creați un fișier .env în directorul backend și adăugați variabilele de mediu necesare.

CLIENT\_ID = 'your\_CLIENT\_ID

GOOGLE\_REDIRECT\_URI = 'http://localhost:8000/accounts/google/login/callback/'

GOOGLE\_FORMS\_LINK = ‘your google forms’

GOOGLE\_SERVICE\_ACCOUNT\_JSON = '{

"type": "service\_account",

"project\_id": "custom-autumn-419101",

"private\_key\_id": " your\_private\_key\_id ",

"private\_key": "your\_private\_key ",

"client\_email": "generare-adeverinte@custom-autumn-419101.iam.gserviceaccount.com",

"client\_id": "106220409631932028736",

"auth\_uri": "https://accounts.google.com/o/oauth2/auth",

"token\_uri": "https://oauth2.googleapis.com/token",

"auth\_provider\_x509\_cert\_url": "https://www.googleapis.com/oauth2/v1/certs",

"client\_x509\_cert\_url": "https://www.googleapis.com/robot/v1/metadata/x509/generare-adeverinte%40custom-autumn-419101.iam.gserviceaccount.com",

"universe\_domain": "googleapis.com"

}'

REACT\_APP\_API\_URL = 'http://localhost:8000'

redirectURI = 'http://localhost:8000/accounts/google/loginAdmin/callback/'

REACT\_APP\_GOOGLE\_FORM\_LINK= 'https://docs.google.com/forms/d/1fs1v9CXBMsCvMeR89fGW85rzmfpIu0l3g97aLXJkAjU/'

**2.4 Aplicarea migrațiilor**

python manage.py makemigrations

python manage.py migrate

python manage.py makemigrations admin\_profile

python manage.py migrate admin\_profile

**2.5 Crearea unui superuser pentru administrarea Django**

python manage.py createsuperuser

**2.6 Pornirea serverului Django**

python manage.py runserver

**3. Configurarea frontend-ului ReactJS**

**3.1 Instalarea dependințelor**

Navigați în directorul frontend:

cd ..

cd frontend

npm install

**3.2 Configurarea variabilelor de mediu**

Creați un fișier .env în directorul frontend și adăugați variabilele de mediu necesare.

REACT\_APP\_API\_URL = 'http://localhost:8000'

GOOGLE\_FORMS\_LINK = ‘your google forms’

GOOGLE\_SERVICE\_ACCOUNT\_JSON = '{

"type": "service\_account",

"project\_id": "custom-autumn-419101",

"private\_key\_id": " your\_private\_key\_id ",

"private\_key": "your\_private\_key ",

"client\_email": "generare-adeverinte@custom-autumn-419101.iam.gserviceaccount.com",

"client\_id": "106220409631932028736",

"auth\_uri": "https://accounts.google.com/o/oauth2/auth",

"token\_uri": "https://oauth2.googleapis.com/token",

"auth\_provider\_x509\_cert\_url": "https://www.googleapis.com/oauth2/v1/certs",

"client\_x509\_cert\_url": "https://www.googleapis.com/robot/v1/metadata/x509/generare-adeverinte%40custom-autumn-419101.iam.gserviceaccount.com",

"universe\_domain": "googleapis.com"

}'

REACT\_APP\_API\_URL = 'http://localhost:8000'

REACT\_APP\_GOOGLE\_CLIENT\_ID = ' yout\_REACT\_APP\_GOOGLE\_CLIENT\_ID '

**3.3 Construirea și pornirea serverului de dezvoltare ReactJS**

npm run build  
npm start

**(Varianta 2 – Folosind Docker)**

**1. Încărcarea imaginilor Docker**

Începe prin încărcarea imaginilor Docker django-image.tar și react-image.tar. Pentru asta:

cd images-directory  
docker load -i django-image.tar  
docker load -i react-image.tar

**2. Configurarea fișierului docker-compose.yml (optional)**

Actualizează fișierul docker-compose.yml astfel încât să includă serviciile backend și frontend cu setările necesare de mediu (env).

**3. Rularea aplicației folosind Docker Compose**

După ce ai încărcat imaginile și ai configurat fișierul docker-compose.yml, poți rula aplicația folosind Docker Compose.

docker-compose up

Această comandă va construi și va porni containerele definite în docker-compose.yml. Serviciile backend și frontend vor fi accesibile la adresele respective specificate în porturi (http://localhost:8000 pentru backend și http://localhost:3000 pentru frontend).