
Guia d'arrencada del sistema de gestió de la fatiga i somnolència

Aquest document explica com posar en funcionament tant els contenidors del sistema com els models de detecció, incloent les instruccions per preparar l'entorn Python i solucionar problemes freqüents de ports o dependències.

ARRENCADA DEL SISTEMA PRINCIPAL (CONTENIDORS DOCKER)

Components llançats

- Backend (Spring Boot – port 8080)
- Frontend (React/Vite – port 5173)
- PostgreSQL (base de dades – port 33333)
- pgAdmin (gestor gràfic de base de dades – port 5050)

Credencials d'Accés

A continuació es mostren les credencials per accedir a la plataforma segons el rol:

- **Administrador**
 - Correu: admin@tfg.com
 - Contrasenya: 12345678
- **Conductor**
 - Correu: driver@tfg.com
 - Contrasenya: 12345678

Des de la carpeta arrel TFG_FLEET_GUARD

Executa:

```
docker compose up -d
```

PROBLEMES HABITUALS

El port 8080 està ocupat

El backend fa ús exclusiu del port 8080. Si aquest port està ocupat, el backend no podrà arrencar i apareixeran errors. Es pot finalitzar la tasca directament des del administrador de tasques o seguir els següents passos.

1. **Identifica el procés que ocupa el port 8080:**
2. `netstat -ano | findstr :8080`
3. **Atura'l si és segur:**
4. `taskkill /PID <PID> /F`

Canviar el port del backend pot provocar inconsistències amb el front-end, ja que caldria modificar tant les rutes de petició com els valors al codi del back-end.

ARRENCADA DELS MODELS DE DETECCIÓ (PYTHON)

Recomanació inicial

És altament recomanable utilitzar un IDE com Visual Studio Code o PyCharm, ja que gestionen correctament les dependències i entorns virtuals, evitant errors de versions de paquets.

Possible fallada de la connexió amb postgres dels models:

Modifica el fitxer `database_connection.py` segons l'entorn:

Ordinador de sobretaula (accedint a la BBDD dins Docker):

```
engine =  
sqlalchemy.create_engine('postgresql://postgres:admin123@host.docker.  
internal:33333/FLEET_GUARD')
```

Portàtil o sistema local (accés directe):

```
engine =  
sqlalchemy.create_engine('postgresql://postgres:admin123@localhost:33  
333/FLEET_GUARD')
```

PREPARAR L'ENTORN PYTHON

Eliminar entorns virtuals previs

```
rmdir /S /Q env
```

Crear un entorn virtual net amb Python 3.10

```
py -3.10 -m venv env
```

Activar l'entorn (PowerShell amb permisos)

```
Set-ExecutionPolicy -Scope Process -ExecutionPolicy RemoteSigned
```

```
.\env\Scripts\Activate.ps1
```

Instal·lació de les dependències necessàries

Amb l'entorn virtual activat, executa:

```
pip install --upgrade pip
```

```
pip install flask joblib numpy opencv-python dlib imutils pygame  
scipy pandas sqlalchemy pycpg2 scikit-learn keras tensorflow  
markupsafe
```

Des del Explorador de fitxers, obre i executa:

camera_based_detection\cmake-3.31.5-windows-x86_64.msi

EXECUCIÓ DEL SISTEMA PYTHON

Flask server

```
python flask_server.py
```

COMPATIBILITAT AMB LA CÀMERA I DEPENDÈNCIES

Assegura't que la càmera connectada és **compatible amb OpenCV i Dlib**. Algunes càmeres integrades de portàtil poden no ser reconegudes correctament per aquestes biblioteques, especialment si no ofereixen accés directe via cv2.VideoCapture.

Per garantir el correcte funcionament del sistema, és **imprescindible instal·lar totes les dependències amb les versions indicades**.

Si en executar algun model apareixen errors per dependències no trobades, pots instal·lar-les directament mitjançant el fitxer requirements.txt, disponible a la carpeta fatigue_drowsiness_detection_models.

Executa la comanda següent amb l'entorn virtual activat:

```
pip install -r requirements.txt
```

Aquesta acció completarà o actualitzarà totes les biblioteques necessàries per garantir l'estabilitat dels models.

SELECCIÓ DE LA CÀMERA EN EL MODEL CAMERA_BASED_DETECTION

El fitxer main.py del mòdul camera_based_detection està configurat per defecte per fer servir la càmera principal del sistema (habitualment la integrada del portàtil), mitjançant el paràmetre següent:

```
ap.add_argument("-w", "--webcam", type=int, default=0,  
help="Index of webcam on system")
```

Si es desitja utilitzar una càmera externa, cal assegurar-se que, aquesta està connectada i reconeguda pel sistema.

Es modifica el paràmetre --webcam a 1 (o a l'índex corresponent al dispositiu detectat).

En cas de conflicte, es pot desactivar temporalment la càmera integrada des del Gestor de dispositius de Windows.