رائع! ✅ نواصل مع \*\*الجزء 3: ملف `README.md` + تعليمات التشغيل\*\*

---

# 📄 الجزء 3: ملف `README.md`

قم بإنشاء ملف باسم:

```

README.md

```

ووضع النص التالي بداخله:

---

# 🎤 Vocal Separator Web App

\*\*AI-Powered Vocal Remover using React.js and FastAPI with Demucs Model.\*\*

This is a full-stack web application that allows users to separate vocals from music tracks using AI (Demucs). It supports \*\*Arabic and English\*\* languages, has waveform visualization using `wavesurfer.js`, and uses Firebase for file storage.

---

## 🧰 Features

✅ Drag-and-drop audio upload

✅ Waveform visualization (Purple for vocals, Green for music)

✅ Multi-language support (RTL layout for Arabic)

✅ Dark mode UI

✅ Secure file handling with FastAPI backend

✅ Firebase Cloud Storage integration

✅ AI-powered vocal separation via Demucs

---

## 🛠️ Tech Stack

- \*\*Frontend:\*\* React.js + Tailwind CSS + Wavesurfer.js

- \*\*Backend:\*\* FastAPI (Python)

- \*\*AI Model:\*\* Demucs (PyTorch)

- \*\*Storage:\*\* Firebase Cloud Storage

- \*\*Languages:\*\* Arabic / English (i18n)

---

## 📦 Installation & Setup

### 1. Install dependencies

#### Frontend:

```bash

cd frontend

npm install

```

#### Backend:

```bash

cd backend

pip install -r requirements.txt

```

---

### 2. Firebase Setup

- Go to [Firebase Console](https://console.firebase.google.com/)

- Create a new project.

- Enable \*\*Firebase Storage\*\*.

- Download the service account key as `serviceAccountKey.json`.

- Place it inside `/backend`.

Update the bucket name in `firebase\_setup.py`:

```python

firebase\_admin.initialize\_app(cred, {

'storageBucket': 'your-firebase-bucket-name.appspot.com'

})

```

---

### 3. Run the app

#### Start frontend:

```bash

cd frontend

npm start

```

#### Start backend:

```bash

cd backend

uvicorn main:app --reload

```

Open your browser at:

👉 `http://localhost:3000`

---

## 🚀 Deployment

### Frontend:

- Deploy on [Vercel](https://vercel.com/):

```bash

vercel

```

### Backend:

- Deploy on AWS EC2 or use a containerized solution (Docker).

- Make sure to have:

- GPU support if running Demucs locally.

- Or use pre-separated files during testing.

---

## 🌍 Language Support

- Default language: \*\*Arabic\*\*

- Toggle between Arabic/English using built-in i18n logic.

RTL layout is automatically applied when switching to Arabic.

---

## 📁 Project Structure

```

/vocal-separator-app

│

├── /frontend # React.js UI

│ ├── public/

│ ├── src/

│ │ ├── components/

│ │ ├── i18n.js

│ │ └── App.js

│ └── package.json

│

├── /backend # FastAPI Server

│ ├── main.py

│ ├── demucs\_utils.py

│ ├── firebase\_setup.py

│ └── requirements.txt

│

└── README.md

```

---

## 📝 Notes

- Make sure you have `ffmpeg` installed globally for audio processing.

- Demucs requires PyTorch and optionally CUDA support.

- For production: consider moving Demucs to a batch-processing system or using pre-trained APIs.

---

## 💬 Feedback

If you found this template useful, feel free to contribute or share it!

---

# 📁 سيتم متابعة باقي التعليمات في الرسالة التالية 👇

هل تريد أن أرسل لك تعليمات ضغط المشروع كـ ZIP؟

(نعم / لا)