









Solar Recorder v1.1

Local screen recording platform with AI-powered transcription, translation, and PDF generation.

Built by AI | IT | Solar — A privacy-first alternative to Loom.

Features

-  **100% Local** - No cloud uploads, all data stays on your server
-  **Screen Recording** - Capture screen + audio in browser
-  **AI Transcription** - Automatic speech-to-text with Whisper
-  **Auto Language Detection** - Supports 50+ languages
-  **PDF Reports** - Generate professional transcripts
-  **Translation** - Powered by DeepSeek API
-  **Docker Ready** - One-command deployment
-  **Easy Management** - Browse, view, and delete recordings

Architecture



```
solar-recorder/
├── backend/           # FastAPI + Whisper + ReportLab
│   ├── main.py       # API server
│   ├── transcribe.py  # Whisper integration
│   ├── translate.py   # DeepSeek translation
│   ├── pdf_generator.py # PDF creation
│   └── uploads/       # Storage
├── frontend/         # Next.js 14 + Tailwind
│   ├── app/
│   │   ├── page.tsx   # Recording interface
│   │   └── records/   # Library view
│   └── components/
└── docker-compose.yml # Deployment config
```

Quick Start

Prerequisites

- Docker & Docker Compose

- (Optional) DeepSeek API key for translations

Installation

1. Clone the repository



bash

```
git clone <repository-url>
```

```
cd solar-recorder
```

2. Configure environment (backend)



bash

```
cd backend
```

```
cp .env.example .env
```

```
# Edit .env and add your DeepSeek API key (optional)
```

3. Launch with Docker



bash

```
docker compose up --build
```

4. Access the application






- Frontend: <http://localhost:3000>
- Backend API: <http://localhost:8000>
- API Docs: <http://localhost:8000/docs>

Usage

Recording

1. Open <http://localhost:3000>
2. Click "**Start Recording**"
3. Select screen/window to share
4. Record your content
5. Click "**Stop Recording**"
6. Wait for automatic transcription and PDF generation

Viewing Recordings

- 1. Navigate to "Recordings Library"
- 2. View all your recordings with metadata
- 3. Actions available:
 -  **Watch video**
 -  **Read transcript**
 -  **Translate** (requires DeepSeek API key)
 -  **Download PDF**
 -  **Delete recording**

Configuration

Backend Environment Variables

Edit backend/.env:



bash

```
# Whisper model size (affects accuracy vs speed)
WHISPER_MODEL=base # Options: tiny, base, small, medium, large

# DeepSeek API for translation
DEEPSEEK_API_KEY=your_key_here

# CORS origins (comma-separated)
ALLOWED_ORIGINS=http://localhost:3000
```

Frontend Environment Variables

Edit frontend/.env.local:



bash

```
NEXT_PUBLIC_API_URL=http://localhost:8000
```

Development

Run Backend Locally



bash

```
cd backend
pip install -r requirements.txt
uvicorn main:app --reload --host 0.0.0.0 --port 8000
```

Run Frontend Locally



bash

```
cd frontend
npm install
npm run dev
```

API Endpoints

Main Endpoints

- POST /upload - Upload video recording
- GET /files - List all recordings
- GET /files/{id} - Get recording details
- POST /translate - Translate transcript
- DELETE /files/{id} - Delete recording
- GET /download/{id}/pdf - Download PDF report
- GET /health - Health check

See full API documentation at <http://localhost:8000/docs>

Troubleshooting

Video Upload Fails

- Check backend logs: `docker compose logs backend`
- Ensure uploads directory has write permissions
- Verify ffmpeg is installed in container

Transcription Stuck

- Large files take time to process (background task)
- Check Whisper model is downloaded

- View logs for errors

PDF with Broken Characters

- DejaVu fonts should be installed automatically
- Verify fonts in container: `/usr/share/fonts/truetype/dejavu/`

Translation Not Working

- Verify `DEEPSEEK_API_KEY` is set in `.env`
- Check API key is valid
- Review backend logs for API errors

Roadmap

v1.1 (Current)

- Background processing
- Metadata management
- Translation support
- Delete recordings

v1.2 (Planned)

- Celery task queue
- Multiple Whisper models selection
- Batch processing
- User authentication

v1.3 (Future)

- Video editing
- Custom branding
- Export formats (SRT, VTT)
- Cloud storage integration (optional)

Contributing

This is a private project for **AI | IT | Solar**.
For feature requests or bugs, contact the development team.

License

Proprietary - AI | IT | Solar © 2025

Team

- **Leanid** - Architect
 - **Dashka** - Senior Assistant
 - **Claude** - AI Implementation Lead
-

Links

- [Whisper Documentation](#)
- [FastAPI](#)
- [Next.js](#)
- [DeepSeek API](#)

Built with  by Solar Team