Transcode Nexus – Cloud-Based Video Converter

Anju Lakshmi

2025

Project Overview

Transcode Nexus is a complete, Dockerized, cloud-native video conversion platform built as a DevOps capstone project. Users can upload videos, convert them into formats like MP4, AVI, MKV, and WEBM, and receive a secure download link via email.

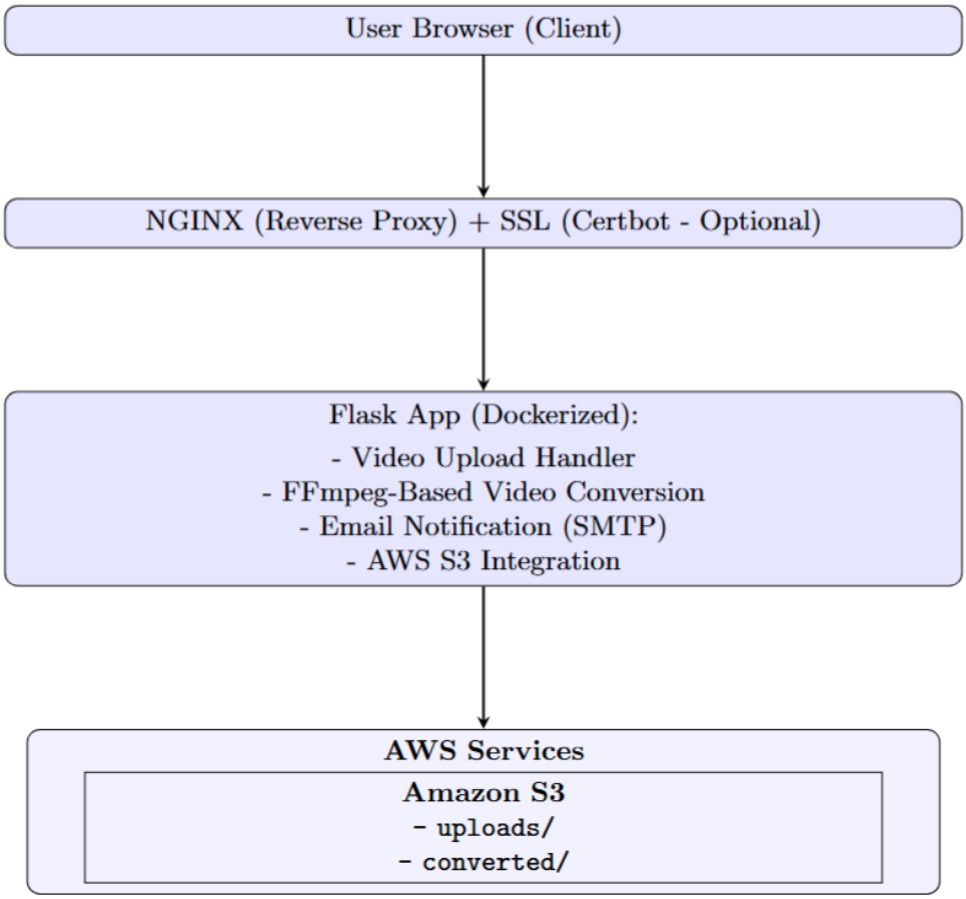
This project was designed to practice key DevOps and cloud skills such as containerization, automation, asynchronous task processing, cloud storage, and reverse proxy deployment.

- Upload and convert videos using a Flask-powered web interface
- Process conversions asynchronously using Celery and Redis
- Store original and converted videos in AWS S3 with automatic cleanup
- Send presigned S3 download links via email
- Track real-time upload and conversion progress
- Deploy with Docker and Docker Compose on AWS EC2
- Integrated NGINX reverse proxy and Certbot SSL

Learning Outcomes

- Deployed and managed Docker containers on a cloud VM.
- Integrated cloud services (AWS S3, EC2) in a real-world web application.
- Applied asynchronous processing using Celery and Redis.
- Implemented secure file upload, storage, and email delivery.

Project Architecture



Tech Stack

- **Frontend:** HTML, CSS, JavaScript
- **Backend:** Python, Flask
- **Video Processing:** FFmpeg
- **Asynchronous Tasks:** Celery + Redis
- **Cloud:** AWS EC2, S3
- **Email Notifications:** SMTP (e.g., Gmail)
- **Deployment:** Docker, Docker Compose, NGINX (optional), Certbot (optional SSL)

Features

- Upload and convert videos in multiple formats

- Asynchronous background processing
- Secure file storage in AWS S3
- Email notifications with download links
- Presigned S3 URLs (valid for 1 hour)
- Auto-delete files after a set duration (optional)
- Real-time progress feedback (upload and conversion)
- Fully containerized with Docker and Docker Compose

Folder Structure

```
transcode-nexus/
          app.py
          tasks.py
          templates/
                index.html
          Dockerfile
          docker-compose.yml
          requirements.txt
          .env (not included - see below)
          README.md
          static/
```

Environment Variables (.env)

Create a file named `.env` in the root directory and add:

```
AWS_ACCESS_KEY_ID=YOUR_ACCESS_KEY
AWS_SECRET_ACCESS_KEY=YOUR_SECRET_KEY
AWS_DEFAULT_REGION=ap-south-1
EMAIL_ADDRESS=youremail@example.com
EMAIL_PASSWORD=yourpassword
```

Note: Do NOT commit this file to GitHub. It should be listed in `.gitignore`.

Deployment Instructions (Amazon EC2)

1. Launch EC2 Instance

- OS: Amazon Linux 2 or RHEL-based
- Open ports: 22 (SSH), 80 (HTTP), 443 (HTTPS), 5000 (Flask)

2. Install Docker and Docker Compose

```
yum update -y
yum install docker -y
service docker start
usermod -aG docker ec2-user
curl -L https://github.com/docker/compose/releases/download/v2.20.2/docker
   -compose-$(uname -s)-$(uname -m) -o /usr/local/bin/docker-compose
chmod +x /usr/local/bin/docker-compose
docker-compose --version
```

3. Clone the Project and Set Up

```
git clone https://github.com/Anju-Lakshmi81/transcode-nexus.git
cd transcode-nexus
nano .env    # Add your secrets
```

4. Run the App

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

Access the app at: `http://your-ec2-ip:5000`

Screenshots

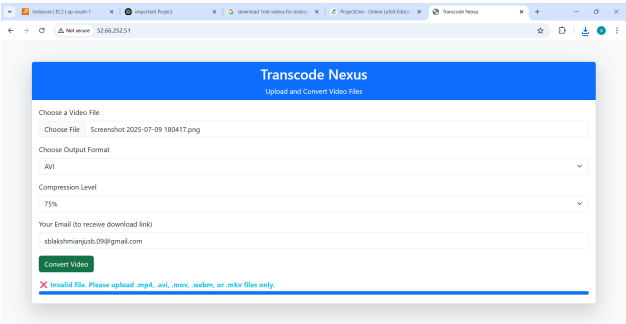


Figure 1: Output http://(public-ip-address) Error message if file is invalid

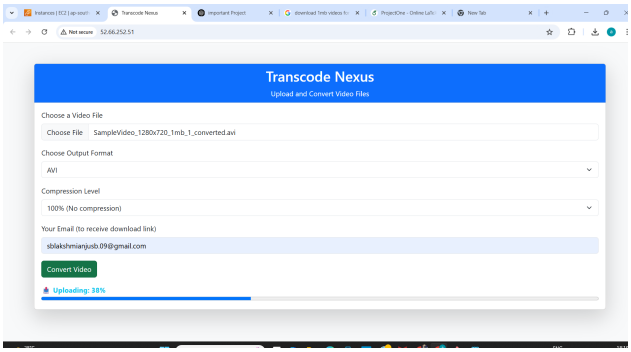


Figure 2: Output http://(public-ip-address) Uploading Progress

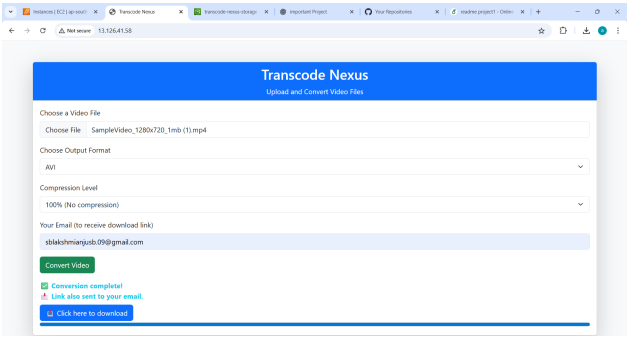


Figure 3: Output http://(public-ip-address)

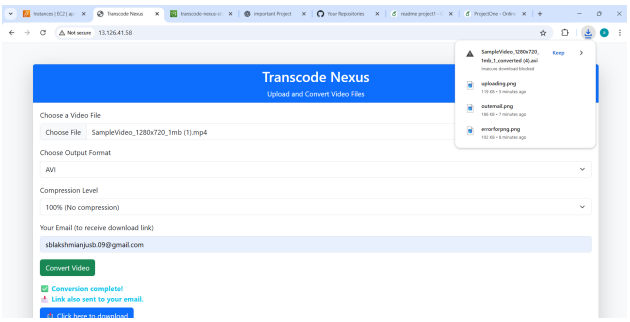


Figure 4: Click Here to Download option helps to downlod video there itself)

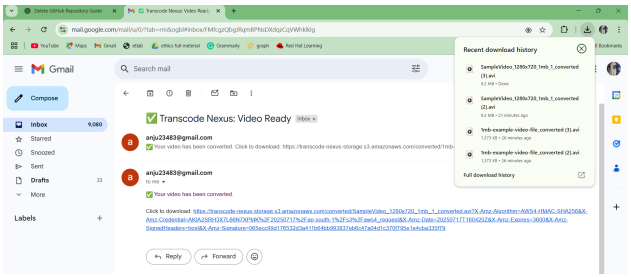


Figure 5: S3 link of the video is received in the mail for download

Future Enhancements

- User login and authentication system
- Video conversion presets and templates
- Subtitles support
- WebSocket-based live progress feedback
- CI/CD pipeline with GitHub Actions

License

Built with ♡by Anju Lakshmi
as a Cloud & DevOps Capstone Project (2025)