# **Architecture**

## Frontend (ReactJS)

- Users interact via a web-based interface.
- Supports file uploads, transcription retrieval, and search.

HTTP requests

## Backend (FastAPI)

- Provides RESTful endpoints to view, search and upload audio file and transcriptions
- Uses whisper model to process audio and transcribe text

SQL queries

## **SQLite Database**

- transcriptions table for storage
- transcriptions\_v table for full text search on audio filename

# Containerization:

The application is fully containerized using Docker, providing consistent environments for development, testing, and deployment.

Docker Compose is used to orchestrate the containers.

### Monolithic Backend:

The backend is structured as a monolith, with the API gateway and business logic within the same FastAPI application.

For larger applications, a microservices architecture could be considered.

## **SQLite3 Limitations:**

SQLite3 is suitable for small to medium-sized applications.

For applications with high concurrency or large data volumes, a more robust database (e.g., PostgreSQL, MySQL) might be necessary.

# **Testing Strategy:**

Separate Docker containers are used for backend and frontend testing, ensuring isolated test environments.

This setup is suitable for CI/CD pipelines, providing consistent test results.

## **CORS Handling:**

CORS middleware is implemented in the FastAPI backend to allow cross-origin requests from the React frontend.

#### **Environment Variables:**

Environment variables are used to configure the application, such as the API URL for the frontend and database settings for the backend.

## **Deployment:**

The containerized application can be deployed to various environments, including cloud platforms (e.g., AWS, Google Cloud, Azure) and on-premises servers.

## **API URL:**

The frontend uses the REACT\_APP\_API\_URL environment variable to connect to the backend, allowing for easy configuration across environments.

### FFMPEG:

FFMPEG is installed within the backend docker container, in order to handle the audio files.