**Description**

When the user uploads audio files using the React frontend, the Flask server transcribes the audio files using an AI model which is locally available (i.e. there are no calls to external services from Flask). When transcription is complete, the results are stored into an SQLite database file, which is on the same container as the Flask server.

When the user searches for a filename pattern, Flask will read the SQLite database file and return the relevant transcriptions.

Note: When the backend service starts for the first time, it downloads the files for the AI model from the internet.

**Considerations**

- The frontend and backend services are running on separate docker containers and communicate via a docker network. This separation of the services enables flexibility of implementation (e.g. one could easily swap out the frontend/backend service for another implementation).
- Usage of docker enables ease of distribution as users only need to pull the images from the docker repo and start the containers. System compatibility would not be an issue.