

Multimediasysteme - Projektantrag

Michael Blaimschein, Patrick Grubauer, Elias Engelbert Plank,
Fabian Sengseis, Julian Hermann Weingartner, Benedikt Alexander Zöchmann,

29. März 2025

Gruppenmitglieder

Mitglieder der Gruppe:

- Michael Blaimschein - k12416067
- Patrick Grubauer - k12410380
- Elias Engelbert Plank - k12408679
- Fabian Sengseis - k12411306
- Julian Hermann Weingartner - k12428284
- Benedikt Alexander Zöchmann - k12410383

Thema: Smart Session Summary

In unserem Projekt wollen wir eine *Smart Session Summary* Applikation umsetzen, mit der man beliebige Mediendateien (z.B. `.mp4`-Video) automatisch transkribieren kann. Aus solchen Transkripten soll unsere Applikation dann auch eine Zusammenfassung generieren können, die die wichtigsten Ereignisse im Transkript anführt. Die Anwendung soll dabei eine Benutzeroberfläche anbieten, in der der Benutzer eine Mediendatei hochladen kann – diese soll dann transkribiert/zusammengefasst werden.

Der primäre Anwendungsfall unserer Applikation ist das automatische Zusammenfassen von aufgezeichneten Besprechungen. Das nimmt den Teilnehmern einiges an Arbeit ab, da man sich vollständig auf die Besprechung selbst konzentrieren kann, und sich nicht damit beschäftigen muss, die wichtigsten Punkte selbst mitzuprotokollieren.

Wir sind der Meinung, dass unser Projekt ideal zu Multimediasysteme passt, weil wir viele Themen aus dem Fach praktisch umsetzen. Zum Beispiel müssen wir mit Audio- und Videodateien arbeiten, diese umwandeln und so verarbeiten, dass man sie transkribieren kann. Dabei geht es auch um Kodierungen und verschiedene Formate. Zusätzlich brauchen wir eine gute Verarbeitungskette: vom Hochladen der Datei, über das Transkribieren mit `whisper.cpp`, bis hin zur automatischen Zusammenfassung. Auch das Anzeigen der Ergebnisse in einer einfachen Benutzeroberfläche gehört dazu. Unser Projekt verbindet also viele Inhalte aus dem Fach – von der Medienverarbeitung bis zur Umsetzung in einer funktionierenden Anwendung.

Zielformulierung

Die folgenden Funktionalitäten sollen umgesetzt werden:

- Benutzerverwaltung mit OAuth-Login
- Möglichkeit zum Hochladen einer beliebigen Mediendatei, die eine Tonspur enthält (z.B. `mp4`, `wav`, `mp3`)
- Automatische Generierung eines Transkripts
- Automatische Generierung einer Zusammenfassung (künstliche Intelligenz notwendig)
- Effizientes Persistieren von bereits erstellten Transkripten/Zusammenfassungen
- Analytiken bezüglich der bereits generierten Zusammenfassungen
- Diverse Konfigurationsmöglichkeiten

Realisierung

Derzeit planen wir, die *Smart Session Summary* über verschiedene Microservices zu implementieren. Somit können wir eine bessere Aufgabenteilung erreichen, und jeder kann mit Technologien arbeiten, die ihm vertraut sind.

Verwendete Technologien

C++, gRPC und whisper.cpp

Für den Microservice, der das Transkribieren und Zusammenfassen durchführt, wollen wir C++ in Kombination mit gRPC und `whisper.cpp` verwenden. Mit `whisper.cpp` wird das tatsächliche transkribieren durchgeführt, über gRPC soll man den Service dann erreichen können.



Abbildung 1: Banner von `whisper.cpp` – Quelle: <https://github.com/ggerganov/whisper.cpp>

Java/Kotlin und Spring Boot

Die *Settings*-, *Analytics*- und *Persistence*-Services wollen wir in Java bzw. Kotlin mit dem Spring Boot Framework implementieren.



Abbildung 2: Logo von Spring Boot – Quelle: https://en.m.wikipedia.org/wiki/File:Spring_Boot.svg

PostgreSQL

Zum persistieren der Daten wollen wir eine PostgreSQL-Datenbank verwenden.

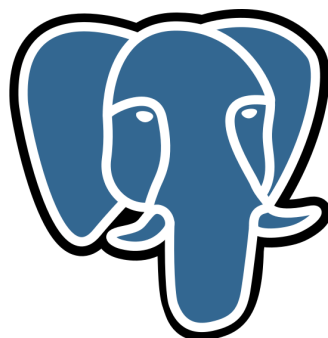


Abbildung 3: Logo von PostgreSQL – Quelle: https://en.m.wikipedia.org/wiki/File:Postgresql_elephant.svg

Grobe Skizzierung der angestrebten Lösung

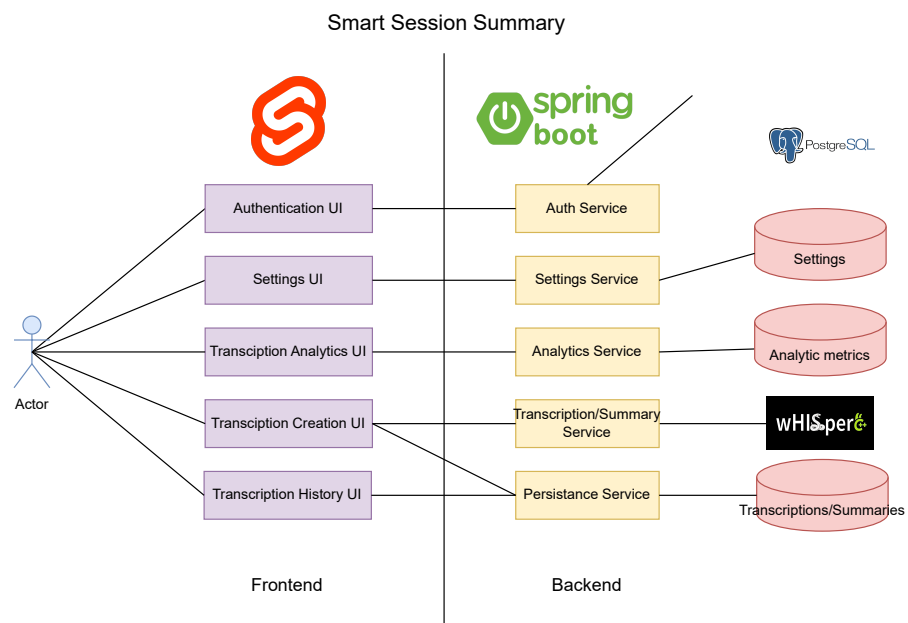


Abbildung 4: Übersicht über die Architektur – Quelle: eigens erstellt mit *drawio*