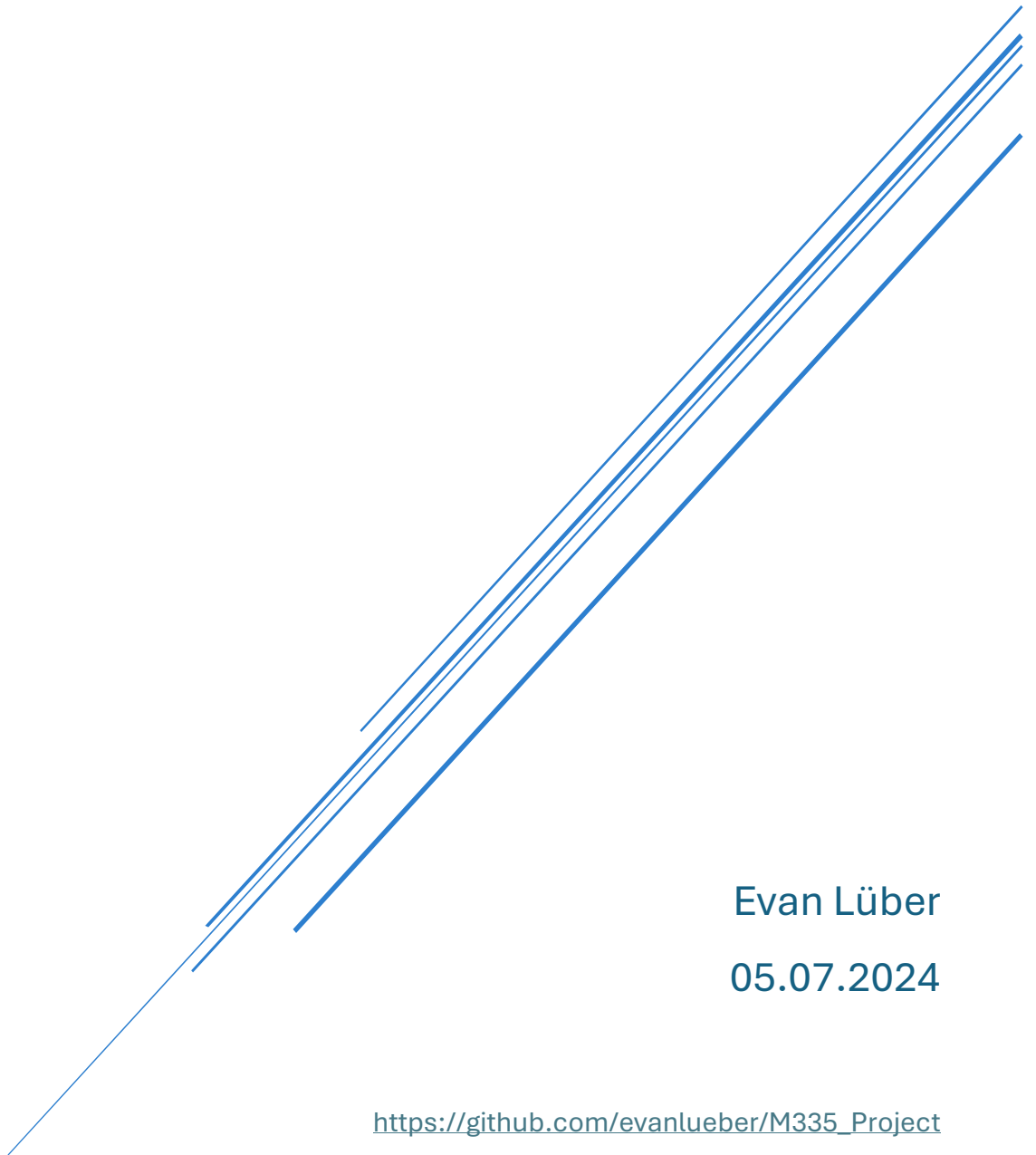


DOKUMENTATION

M335 LB-B



Evan Lüber

05.07.2024

https://github.com/evanlueber/M335_Project

Inhalt

1	Kurzbeschreibung.....	2
2	Motivation.....	2
3	Funktionale Anforderungen.....	2
4	Nicht Funktionale Anforderungen.....	2
5	Anwendungsfalldiagramme	3
5.1	Audio aufnehmen	3
5.2	Audio löschen.....	4
5.3	Audio anhören	4
5.4	Text schreiben und abspielen	5
6	Testkonzept.....	5
6.1	Testumgebung	5
6.2	Testmethode.....	5
6.3	Testfälle.....	6
7	Testprotokoll	6
7.1	Testfälle.....	6
7.1.1	Tonaufnahme erstellen.....	6
7.1.2	Tonaufnahme anhören	6
7.1.3	Tonaufnahme löschen.....	7
7.1.4	Text abspielen	7
7.2	Ausgeführte Tests.....	7
7.2.1	Test 1	7
7.2.2	Test 2.....	8
8	Layer-Diagramm.....	8
9	Komponenten-Diagramm	9
10	Website.....	10
11	Versionskontrolle.....	11
12	Reflexion	11
13	Quellenverzeichnis	12
13.1	Hilfsquellen Code	12
13.2	Abbildungen	12

1 Kurzbeschreibung

yApp ist eine App, die den Zweck der Audioaufnahme und der Textvorlesung erfüllt. Mit yAPP es ist möglich seine Gedanken per Tonaufnahme festzuhalten. Diese Audios werden in der App gespeichert und können angehört werden. Wenn die Aufnahme nicht mehr benötigt wird, kann sie ganz einfach gelöscht werden. Ausserdem enthält yAPP eine Funktion, mit der man sich einen Text vorlesen lassen kann, denn man zuvor in ein Textfeld eingegeben hat.

2 Motivation

Meine Motivation hinter der App war es mehrere Funktionen in ein Programm einzubinden. Oftmals fand ich die Apps für die Aufnahme von Audio und Vorlesen von Text ein wenig unübersichtlich. Deshalb habe ich meine eigene programmiert. Ich habe schon oft nach Vorleseprogrammen gesucht, jedoch habe ich kaum welche gefunden, die funktioniert haben und meistens sahen diese sehr unprofessionell und nicht sicher aus.

3 Funktionale Anforderungen

- Als Aktuator wird der Lautsprecher eingesetzt. Er ist dazu da, die Aufnahme oder einen geschriebenen Text in Sprache wiederzugeben.
- Als externe Schnittstelle wird eine Text-To-Speech Package verwendet, um den geschriebenen Text des Benutzers in Sprache umzuwandeln.
- Damit die Aufnahmen nach dem Schliessen der App nicht verschwinden, sondern beim wieder Öffnen der App noch vorhanden sind, wird ein lokaler Speicher verwendet.
- Die Aufnahme, die Liste der Aufnahmen und die Text-To-Speech Seite wird je in einem eigenen Tab zu finden sein.
- Es besteht eine Interaktion mit der Systemapplikation, beim Verwenden des Mikrofons für das Ausnehmen eines Audios.
- Für das App-Icon wird ein Bild verwendet, welches von GPT-4 generiert wurde.

4 Nicht Funktionale Anforderungen

- App soll keine Bugs haben
- App soll auf IOS und Android funktionieren
- App soll einfach zu bedienen sein

5 Mockup

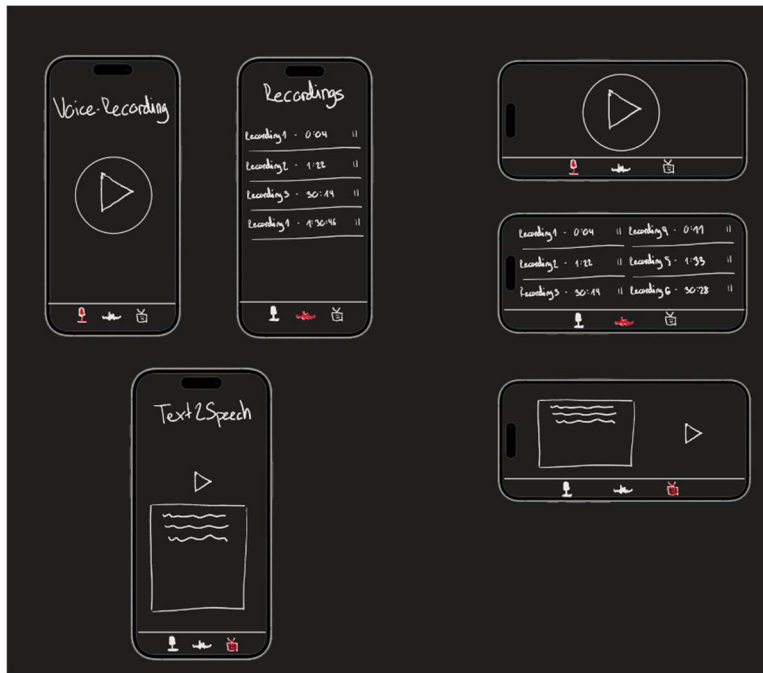


Abbildung 1: Mockup

6 Anwendungsfalldiagramme

6.1 Audio aufnehmen

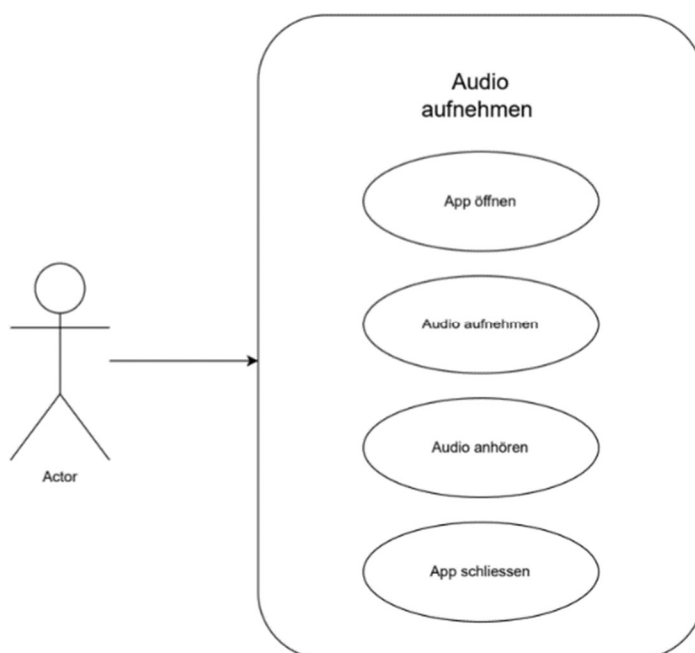


Abbildung 2: Audio aufnehmen

6.2 Audio löschen

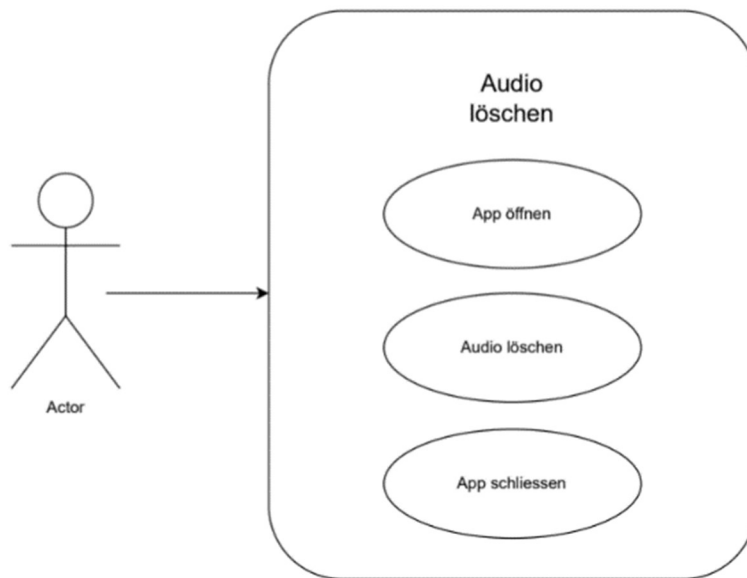


Abbildung 3: Audio löschen

6.3 Audio anhören

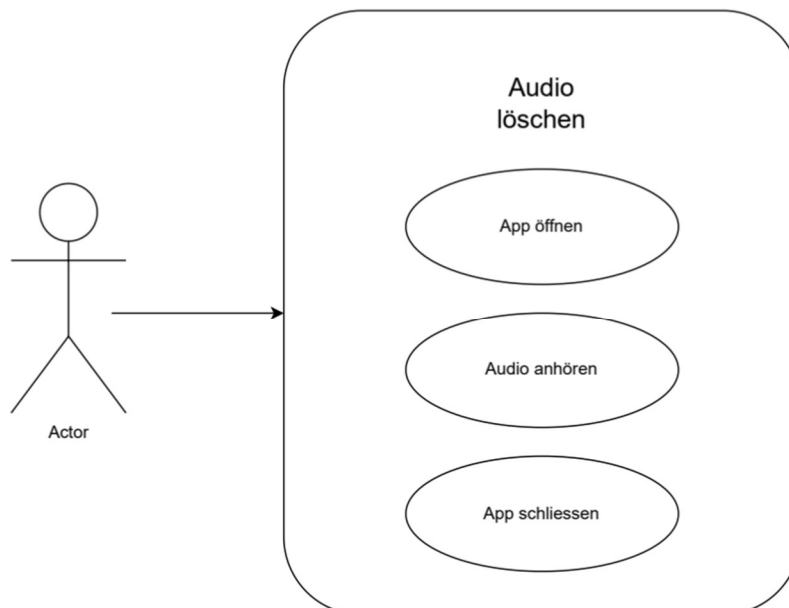


Abbildung 4: Audio anhören

6.4 Text schreiben und abspielen

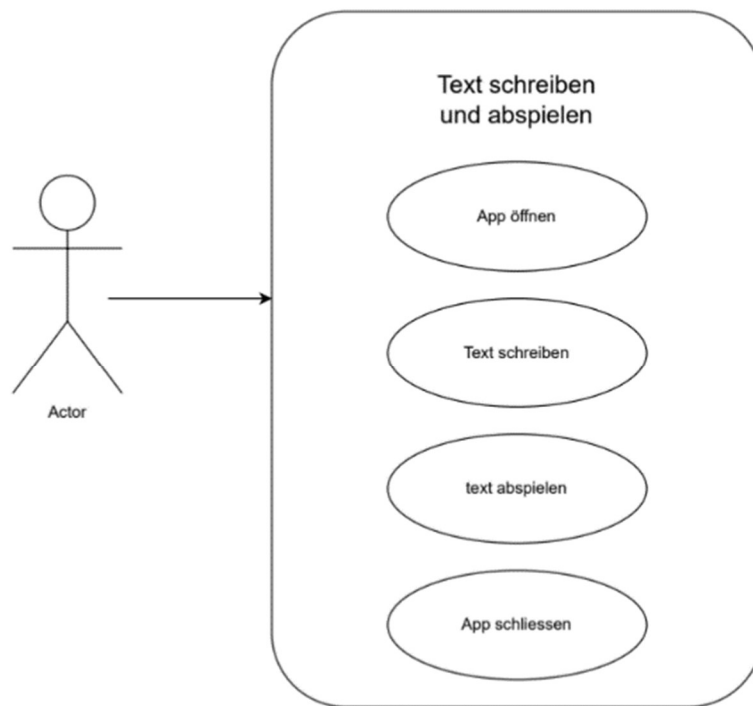


Abbildung 5: Text schreiben und abspielen

7 Testkonzept

7.1 Testumgebung

- iPhone 15, IOS 18.0 BETA
- Samsung A53,
 - One UI version 6.1,
 - Android version 14

7.2 Testmethode

- Manuelle Tests: Testen durch Benutzer, um Benutzerfreundlichkeit zu gewährleisten.
- White-Box: Testen der Logik und Funktion der App
- Black-Box: Funktionstest der Benutzeroberfläche

7.3 Testfälle

- Checken von Sound beim Starten der Aufnahme
- Testen der Wiedergabe-Funktion
- Testen der Lösch-Funktion
- Testen der Text-To-Speech Funktion
- Abrufen des lokalen Speichers
- Benutzeroberfläche auf unterschiedlichen Betriebssystemen

8 Testprotokoll

8.1 Testfälle

8.1.1 Tonaufnahme erstellen

- **Identifikation:** TC-01
- **Vorbedingung:** Die App ist geöffnet.
- **Ausführung:**
 - Benutzer startet Aufnahme
 - Benutzer nimmt etwas auf
 - Benutzer stoppt Aufnahme
- **Erwartetes Resultat:** Aufnahme wird in der App unter Aufnahmen lokal gespeichert.

8.1.2 Tonaufnahme anhören

- **Identifikation:** TC-02
- **Vorbedingung:** Die App ist geöffnet, eine Aufnahme ist vorhanden und Handy ist nicht lautlos.
- **Ausführung:**
 - Benutzer öffnet den Tab Aufnahmen
 - Benutzer drückt auf den Play-Button
 - Benutzer hört sich die Aufnahme an
- **Erwartetes Resultat:** Die Aufnahme wird über den Handylautsprecher abgespielt.

8.1.3 Tonaufnahme löschen

- **Identifikation:** TC-03
- **Vorbedingung:** Die App ist geöffnet und eine Aufnahme ist vorhanden.
- **Ausführung:**
 - Benutzer öffnet den Tab Aufnahmen
 - Benutzer drückt auf den Mülleimer
- **Erwartetes Resultat:** Die Aufnahme wird unwiderruflich gelöscht.

8.1.4 Text abspielen

- **Identifikation:** TC-04
- **Vorbedingung:** Die App ist geöffnet.
- **Ausführung:**
 - Benutzer öffnet den Tab TextToSpeech
 - Benutzer schreibt einen Text
 - Benutzer drückt den Play-Button
- **Erwartetes Resultat:** Der Text wird von einer Roboterstimme vorgelesen.

8.2 Ausgeführte Tests

8.2.1 Test 1

- 1 **Datum:** 05.07.2024
- 2 **Tester:** Ael Banyard
- 3 **Version:** 1.0

Testfall	Ergebnis
TC-01	Erfolgreich
TC-02	Erfolgreich
TC-03	Erfolgreich
TC-04	Erfolgreich

8.2.2 Test 2

- 4 **Datum:** 05.07.2024
5 **Tester:** Aakash Sethi
6 **Version:** 1.0

Testfall	Ergebnis
TC-01	Erfolgreich
TC-02	Erfolgreich
TC-03	Erfolgreich
TC-04	Erfolgreich

9 Layer-Diagramm

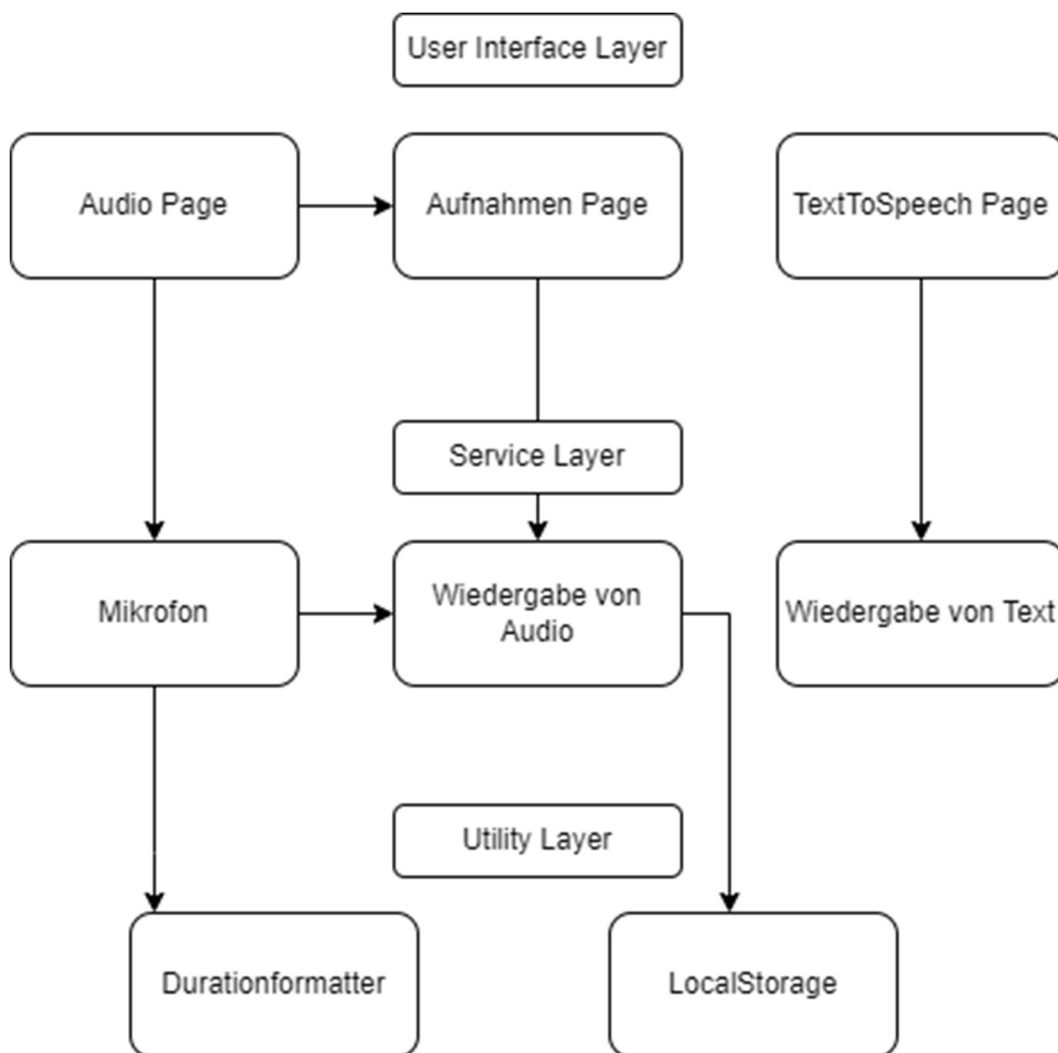


Abbildung 6: Layer-Diagramm

10 Komponenten-Diagramm

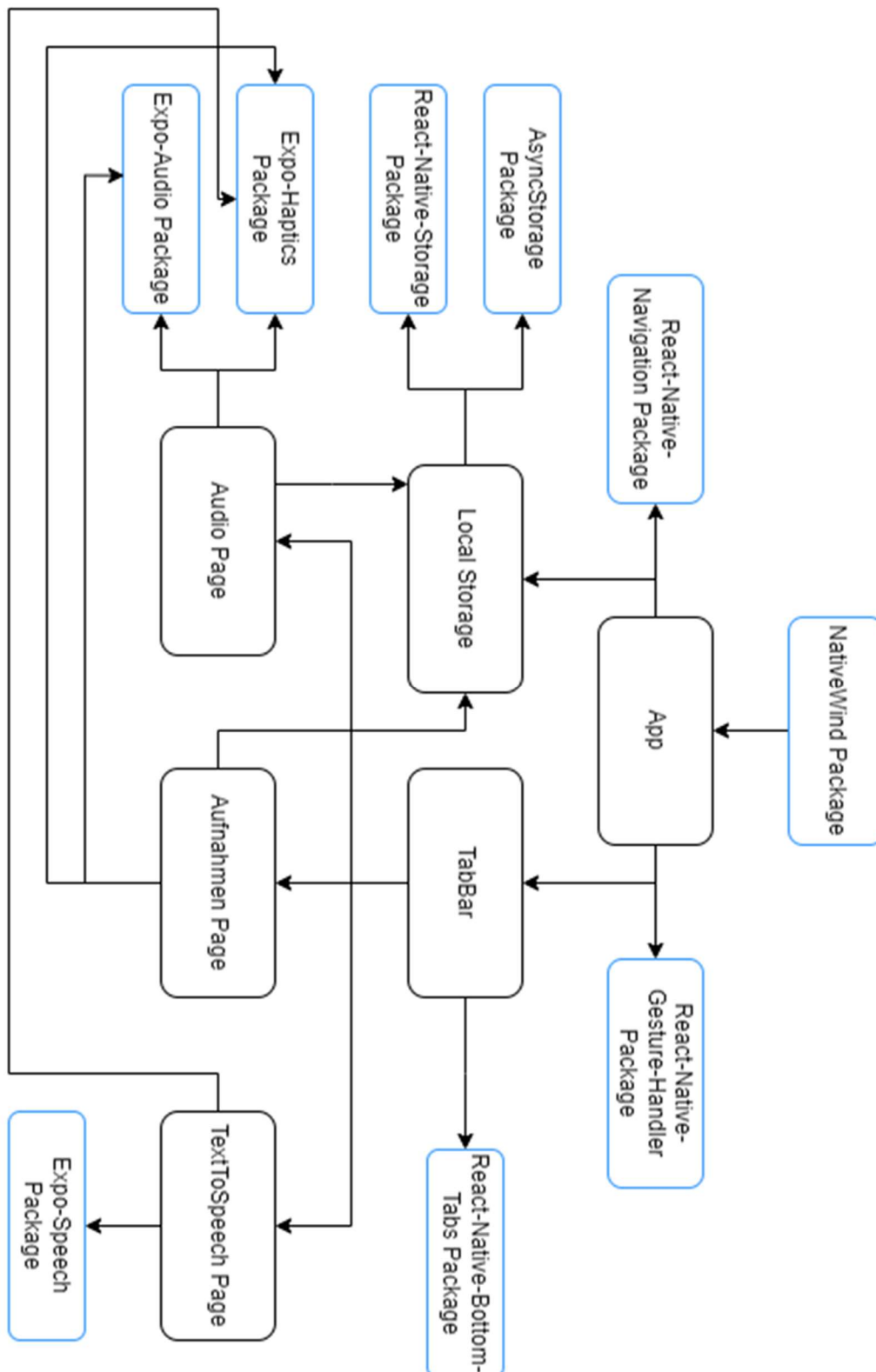


Abbildung 7: Komponenten-Diagramm

11 Website



yAPP

Evan Lüber

Vorschau



Beschreibung

- ✓ Machen Sie Ihre eigene **Tonaufnahme**
- ✓ **Hören** Sie sich Ihre Tonaufnahme an
- ✓ Ihre Tonaufnahme gefällt Ihnen nicht? **Löschen** Sie diese einfach
- ✓ Keine Lust selbst zu lesen? **TextToSpeech** liest für Sie

Abbildung 8: Website

12 Versionskontrolle

Ich habe die Versionskontrolle mit GitHub geführt.

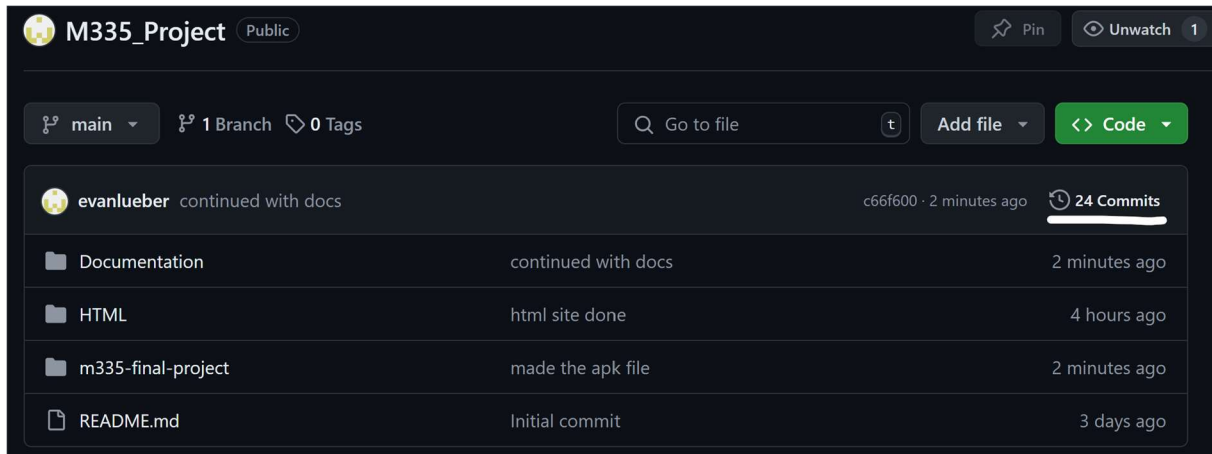


Abbildung 9: Versionskontrolle

13 Reflexion

Mein Ziel bei der App war es, meine Gedanken festhalten zu können, ohne dieses aufschreiben zu müssen. Ausserdem ist es oft so, dass Apps wie meine mit einer Datenbank verbunden sind. YAPP allerdings kann ganz ohne Internetverbindung genutzt werden und alles Aufnahmen werden auf dem lokalen Speicher gespeichert, was mehr Privatsphäre liefert. Die Funktion, sich einen Text vorlesen zu lassen, fand ich sehr gut, da es manchmal anstrengend sein kann, einen langen Text zu lesen. Ich persönlich kann mich besser auf den Text fokussieren, wenn ich ihn höre.

Ich habe alle Ziele, die ich mir vorgenommen haben, erreicht, ausser dem Einbauen des Beschleunigungssensors. Dafür hatte ich leider nicht mehr genügend Zeit. Dies wäre eine Verbesserungsmöglichkeit. Ausserdem könnte man noch einbauen, dass man die Aufnahmen aus der App auf sein Handy herunterladen kann, damit sie nicht nur im lokalen Speicher der App gespeichert sind.

Mit React Native klarzukommen, war leicht, da es meiner Meinung nach gleich wie React ist, jedoch nur andere Bezeichnungen hat. Schwer fand ich das Einbinden der Audioaufnahme und das Speichern der Audios im lokalen Speicher in einem anderen Tab. Dafür habe ich mir Hilfe von YouTube geholt. Ich habe genug früh angefangen und habe nicht geträdelt, was ich sehr gut fand, denn so hatte ich schlussendlich keinen Stress.

14 Quellenverzeichnis

14.1 Hilfsquellen Code

React Native Dokumentation, <https://reactnative.dev/docs/getting-started>

Expo Dokumentation, <https://docs.expo.dev/>

freeDomCamp, <https://www.freecodecamp.org/news/tailwindcss-in-react-native-expo/>

Github, Benutzer: sunnylqm, <https://github.com/sunnylqm/react-native-storage>

Github Copilot, <https://github.com/features/copilot>

ChatGPT, <https://chatgpt.com/>

YouTube, https://www.youtube.com/watch?v=pd_Ez9Kbi2c

14.2 Abbildungen

Abbildung 1: Mockup	3
Abbildung 2: Audio aufnehmen	3
Abbildung 3: Audio löschen	4
Abbildung 4: Audio anhören	4
Abbildung 5: Text schreiben und abspielen	5
Abbildung 6: Layer-Diagramm	8
Abbildung 7: Komponenten-Diagramm.....	9
Abbildung 8: Website.....	10
Abbildung 9: Versionskontrolle.....	11