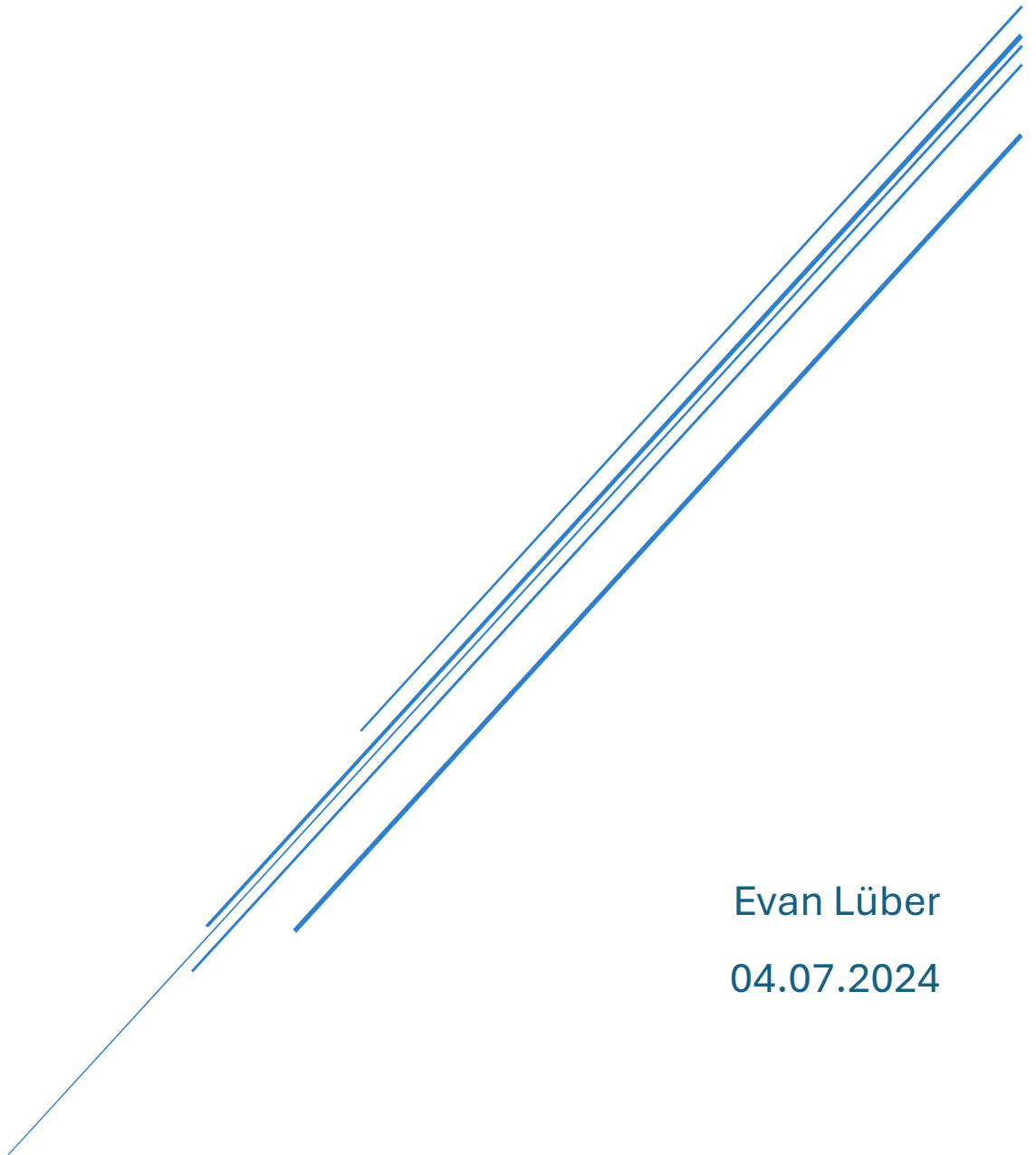


# DOKUMENTATION

M335 LB-B



Evan Lüber

04.07.2024

## Inhalt

|     |                                       |   |
|-----|---------------------------------------|---|
| 1   | Funktionale Anforderungen .....       | 2 |
| 2   | Nicht Funktionale Anforderungen ..... | 2 |
| 3   | Anwendungsfalldiagramme.....          | 3 |
| 3.1 | Audio aufnehmen .....                 | 3 |
| 3.2 | Audio löschen.....                    | 3 |
| 3.3 | Audio anhören .....                   | 4 |
| 3.4 | Text schreiben und abspielen .....    | 4 |
| 4   | Testkonzept .....                     | 5 |
| 4.1 | Testumgebung .....                    | 5 |
| 4.2 | Testmethode.....                      | 5 |
| 4.3 | Testfälle .....                       | 5 |
| 5   | Layer-Diagramm .....                  | 6 |
| 6   | Komponenten-Diagramm.....             | 7 |
| 7   | Quellenverzeichnis .....              | 8 |
| 7.1 | Hilfsquellen Code .....               | 8 |
| 7.2 | Abbildungen .....                     | 8 |

# 1 Funktionale Anforderungen

- Als Sensor wird der Beschleunigungssensor verwendet. Er soll erkennen, wann das Handy horizontal oder vertikal ist, um die App auf dem Handy mitzudrehen.
- 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 API 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.

# 2 Nicht Funktionale Anforderungen

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

## 3 Anwendungsfalldiagramme

### 3.1 Audio aufnehmen

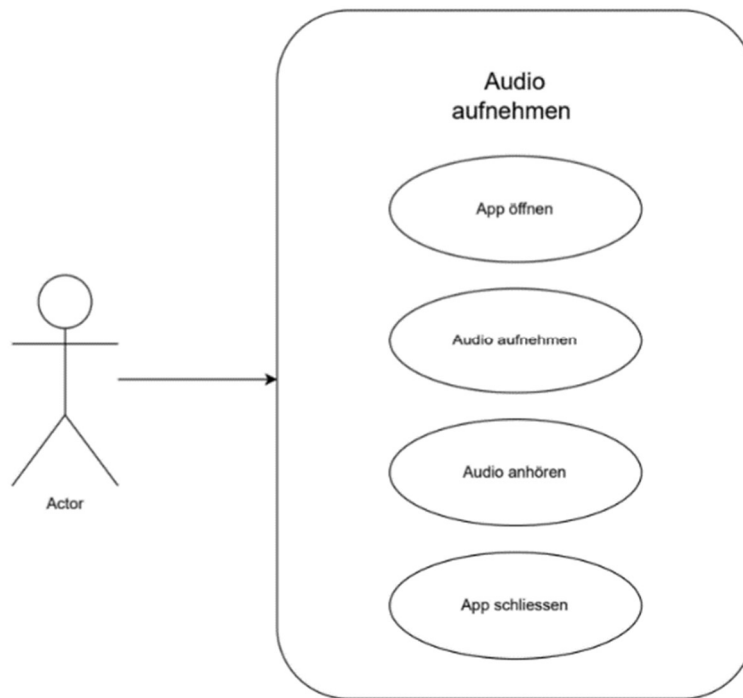


Abbildung 1: Audio aufnehmen

### 3.2 Audio löschen

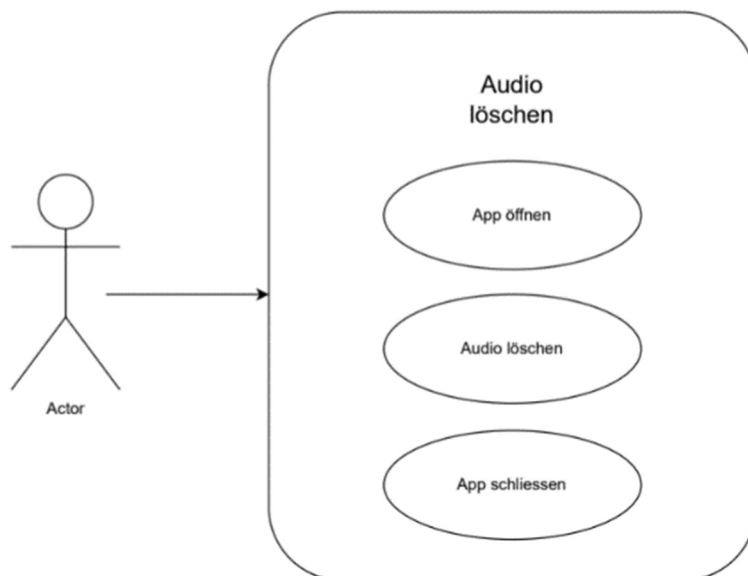


Abbildung 2: Audio löschen

### 3.3 Audio anhören

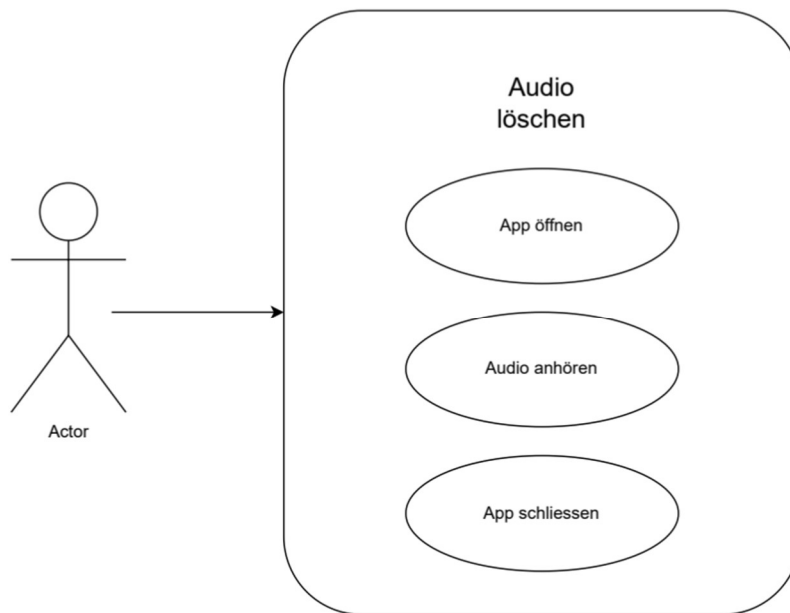


Abbildung 3: Audio anhören

### 3.4 Text schreiben und abspielen

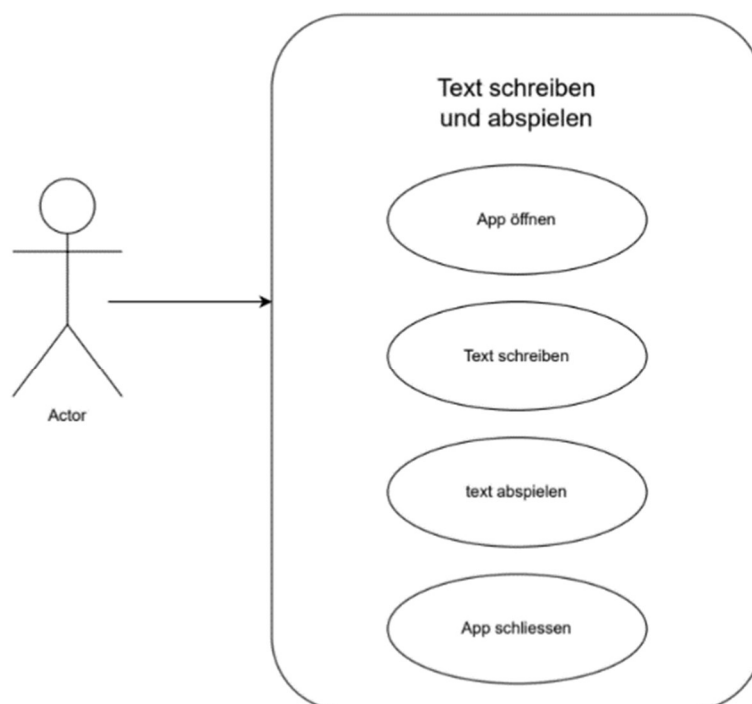


Abbildung 4: Text schreiben und abspielen

## 4 Testkonzept

### 4.1 Testumgebung

- iPhone 11, IOS 16.6.1
- iPhone 15, IOS 18.0 BETA
- Samsung A53,
  - One UI version 6.1,
  - Android version 14

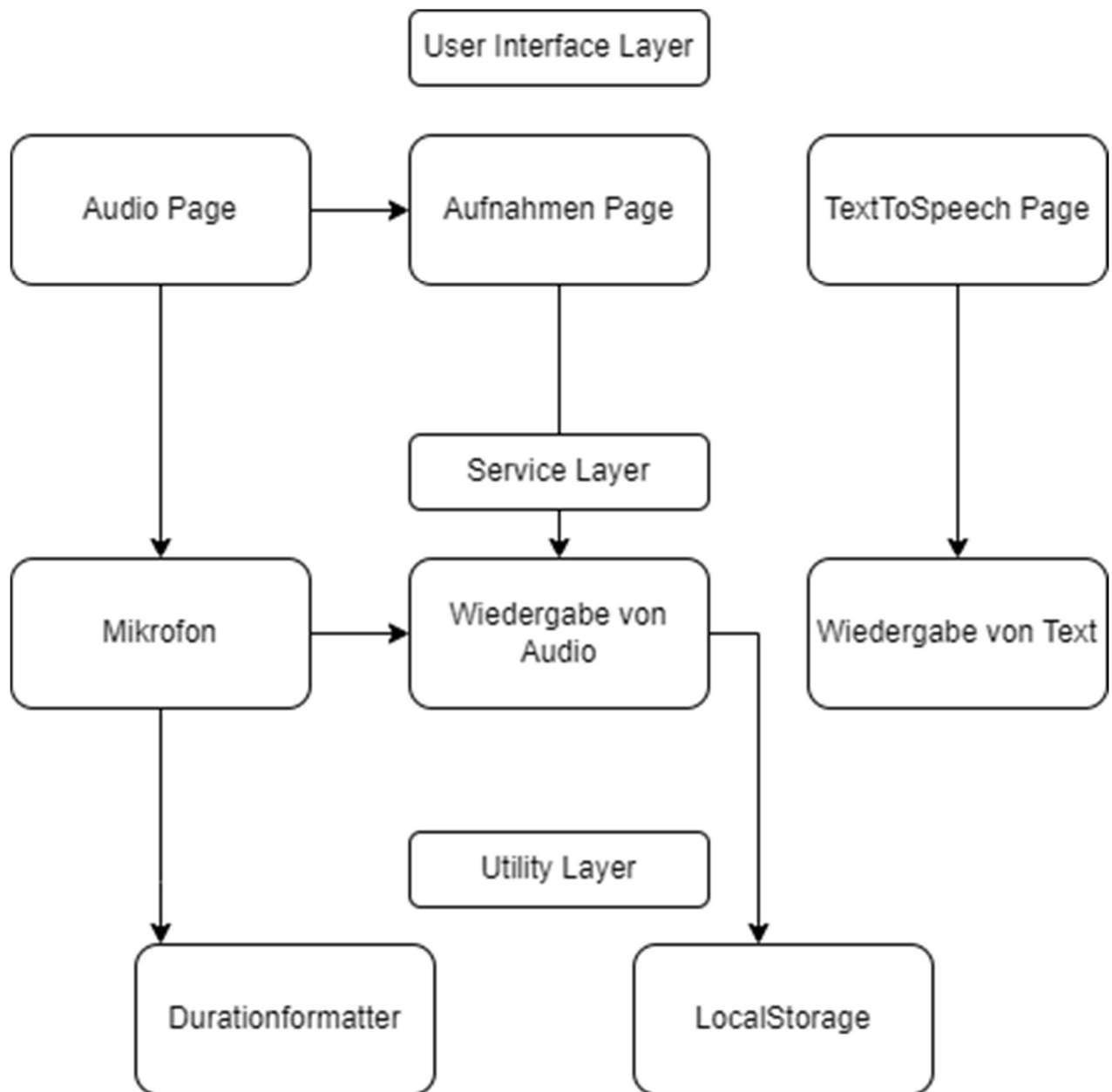
### 4.2 Testmethode

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

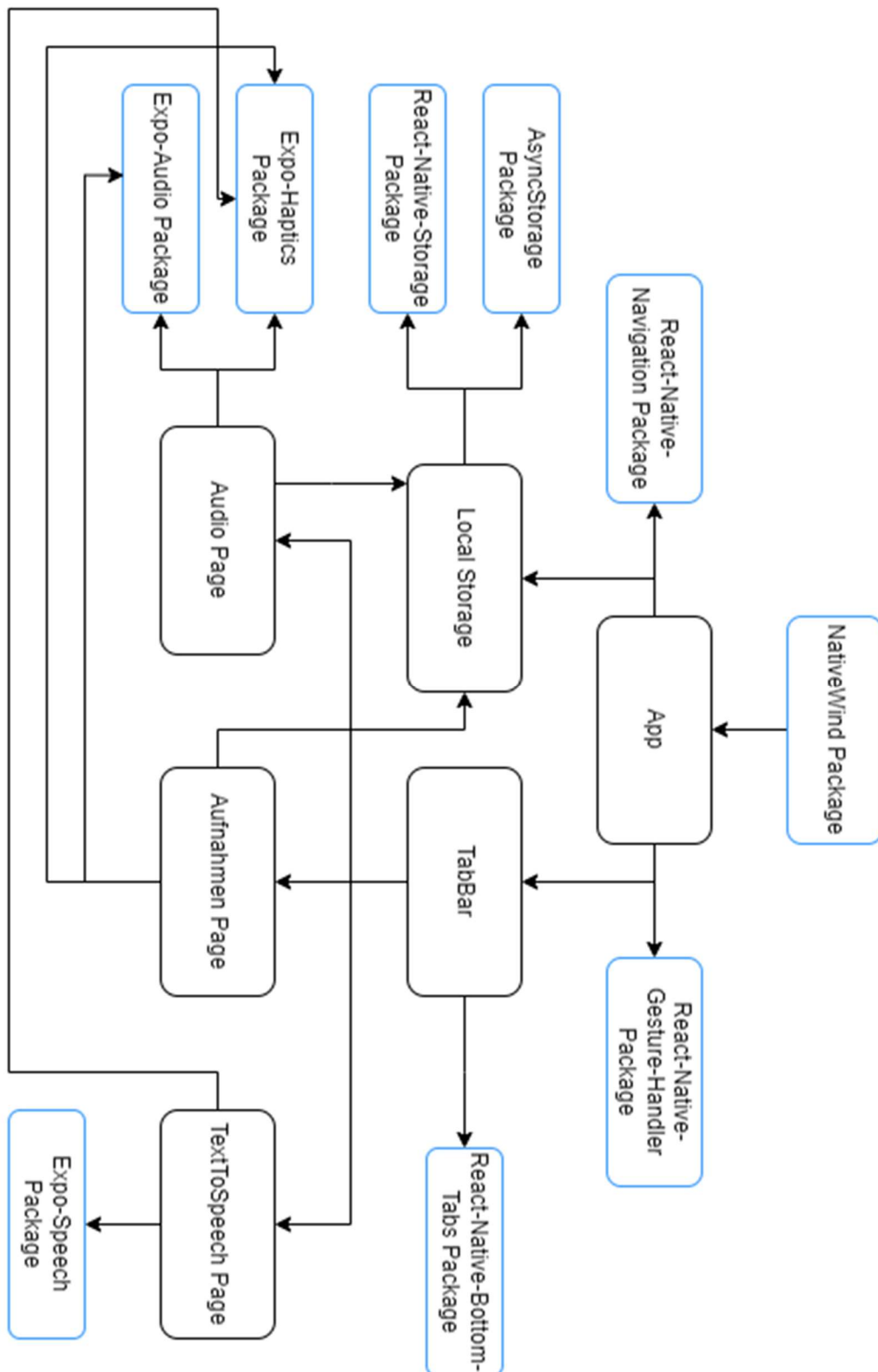
### 4.3 Testfälle

- 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

## 5 Layer-Diagramm



## 6 Komponenten-Diagramm





## 7 Quellenverzeichnis

### 7.1 Hilfsquellen Code

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

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

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

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

### 7.2 Abbildungen

|   |   |
|---|---|
| Abbildung 1: Audio aufnehmen .....              | 3 |
| Abbildung 2: Audio löschen .....                | 3 |
| Abbildung 3: Audio anhören .....                | 4 |
| Abbildung 4: Text schreiben und abspielen ..... | 4 |