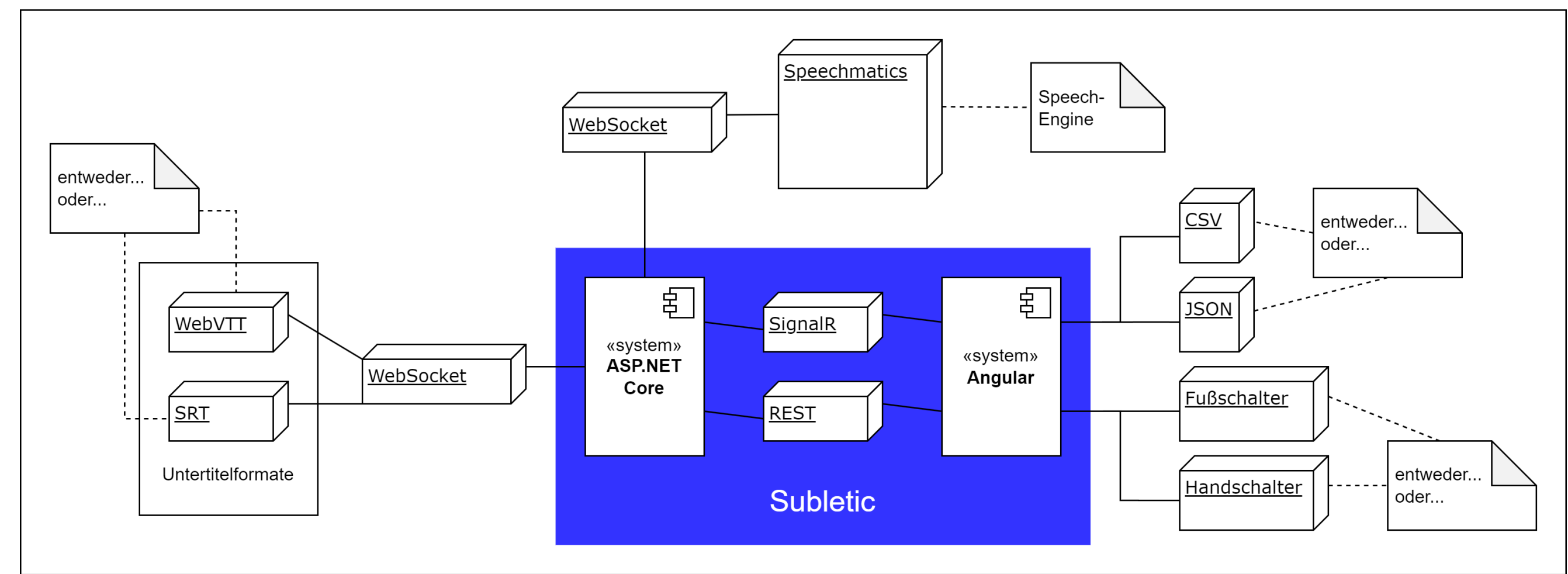


## Subletic: Live Stream Editor zur Korrektur von Untertiteln

Mit der von unseren Entwicklern erdachten Software, möchten wir ein Werkzeug schaffen, welches das Korrigieren von Live-Untertitelung, so einfach wie möglich macht. Dabei liest unsere Software einen beliebigen Videostream ein, erstellt mittels einer Speech-Engine eine Untertitelung und bietet die generierten Untertitel dem Anwender zu Korrektur an. Dieser prüft den Untertitel und korrigiert diesen, falls nötig, sodass der Untertitel an den Endkunden ausgeliefert werden kann. Das Endergebnis? Ein perfekt untertitelter Videostream, intuitiv bedienbar, DSGVO-konform und unter Beachtung der Barrierefreiheit.

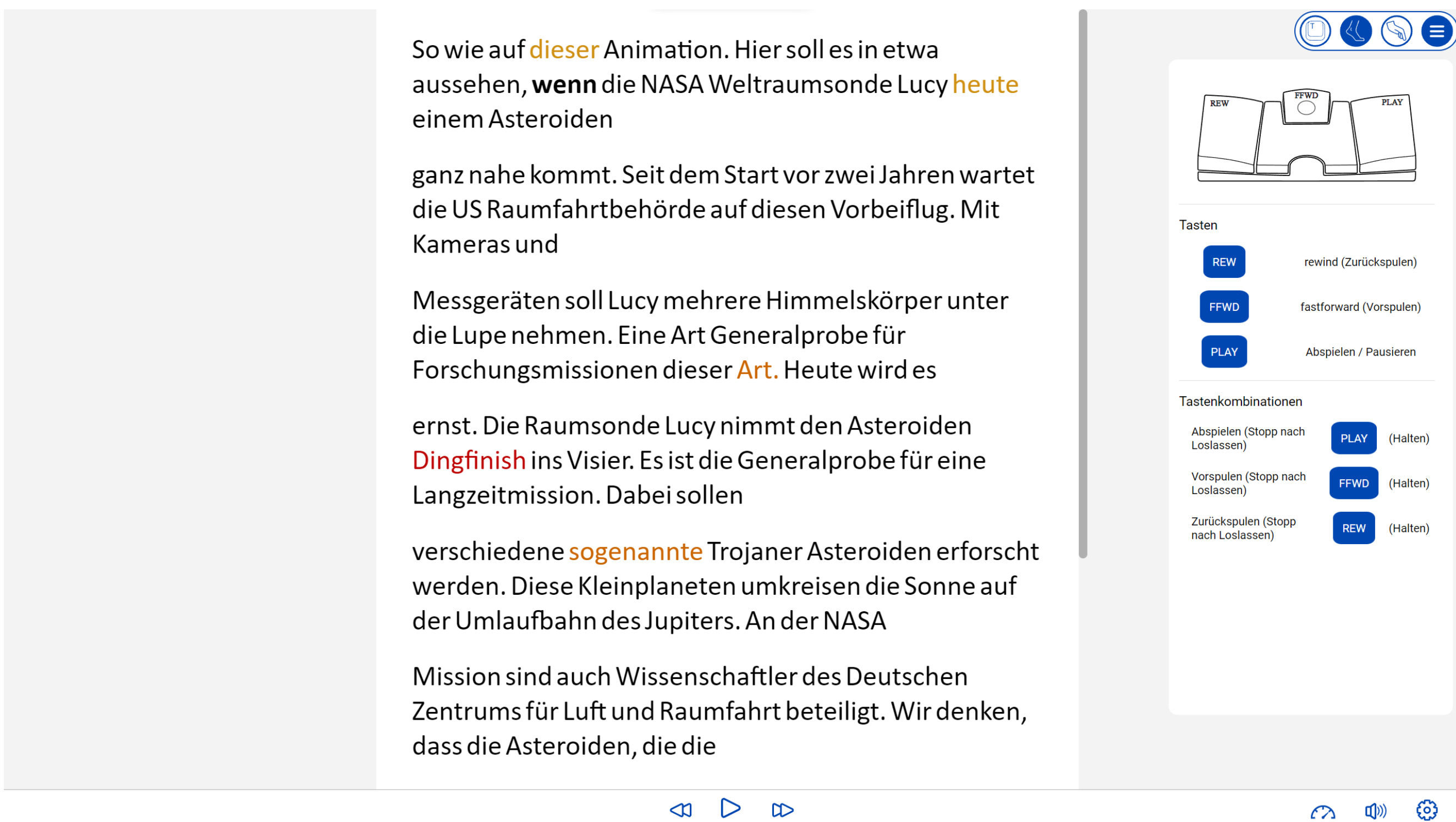
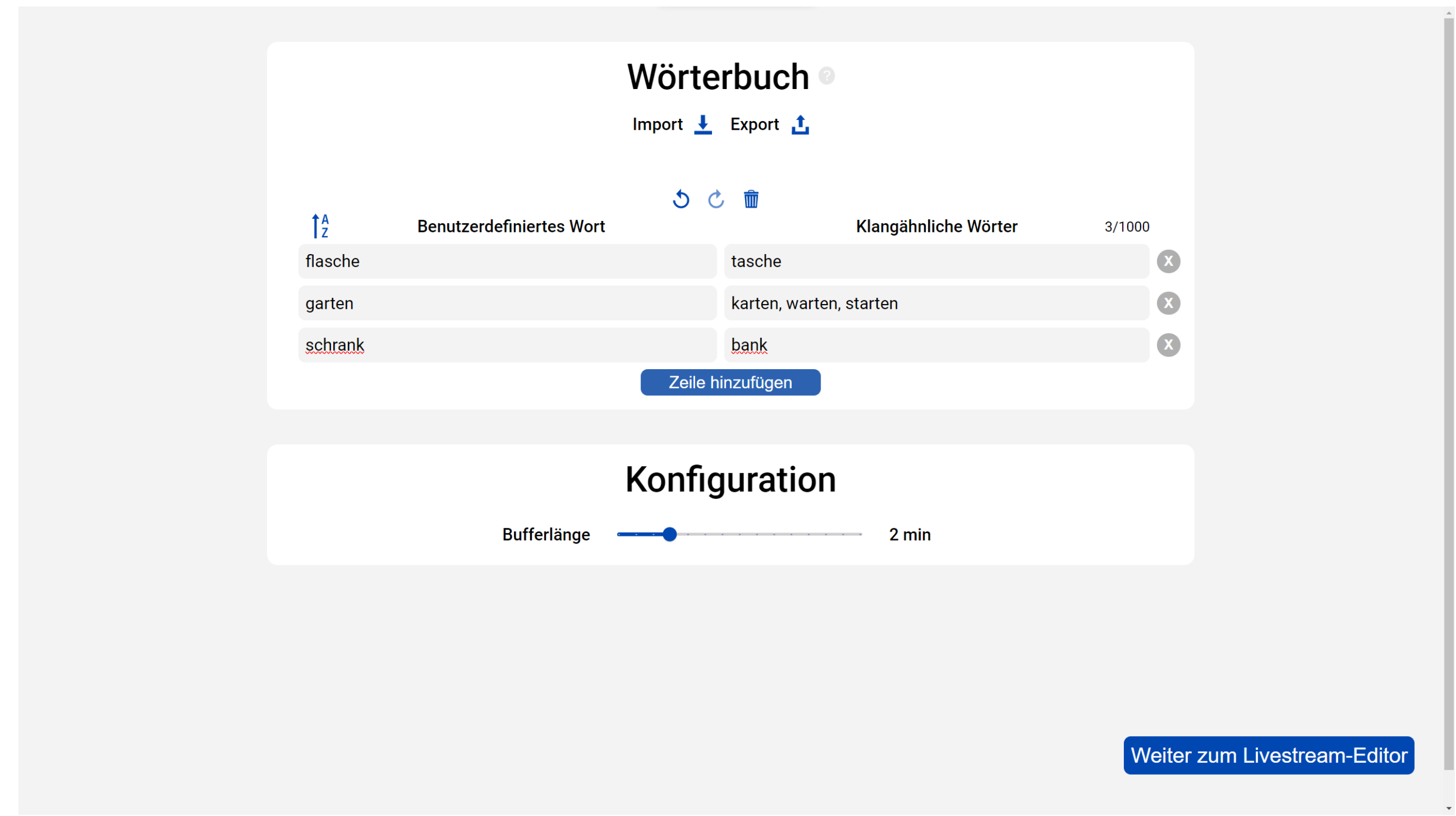


### Zielstellung

Live-Untertitelung existiert schon heute, wie die automatische Untertitelung von YouTube zeigt. Diese ist jedoch noch fehleranfällig und eignet sich daher nicht für Live-Streams, bei denen ein hoher Wert auf Korrektheit gelegt wird, wie es unter anderem bei Landtagsdebatten der Fall ist. Dabei besitzen diese Technologien das Potenzial, Berufsgruppen wie Stenografen zu entlasten, und lässt Menschen, die auf Untertitel angewiesen sind, von besserer Untertitelung profitieren. Unser Ziel ist es, die bereits heute existierende Software für automatische Untertitelung, um Echtzeit Korrektur zu erweitern. Dabei sollen die Bedürfnisse von Stenografen unter Beachtung der DSGVO-Konformität umgesetzt werden. Somit kann unsere Software überall da eingesetzt werden, wo es auf fehlerlose Untertitel und die Einhaltung des Datenschutzes ankommt.

### Funktionalitäten

Unsere Software erlaubt es dem Nutzer, die aus dem Eingabe-Stream generierte Transkriptionen, in sogenannte Sprechblasen, in einer Liste zu sehen und zu editieren. Zudem werden Wörter, bei dem sich die Speech-To-Text Engine unsicher ist, mit sogenannten Konfidenzwerten, farbig markiert. Zusätzlich zur reinen visuellen Repräsentation des Gesagten, ist laufend das Audio des Video-Streams zu hören. Dies bietet die Grundlage, dass sich der Stenograf vergewissern kann, fehlerhafte Wörter zu korrigieren. Außerdem ist möglich, in der Audio vor und zurückzuspringen, um bei Unsicherheiten sich nochmal zu vergewissern. Die Zeitspanne, ab wann Sprechblasen und der Audio-Stream aus dem Zeitfenster zur Bearbeitung verschwinden ist durch den Nutzer individuell Anpassbar. Des Weiteren besitzt die Software die Möglichkeit häufig fehlerhaft erkannte Wörter in einem sogenannten Dictionary zu speichern und diese an die Speech-Engine zu übertragen. Somit ist es von Beginn an möglich potenzielle Fehlerquellen zu eliminieren und dem Stenografen vermehrt Arbeit abzunehmen.



### Qualitätssicherung

Da unsere Software von jungen Entwicklern programmiert wird, haben wir von Anfang an auf Code-Reviews, Unit-Tests und Linting in der CI-Pipeline gesetzt, um die Qualität unseres Codes zu gewährleisten. Erfahrende Teammitglieder unterstützen die jungen Entwickler durch Code-Reviews und geben ihr Wissen weiter. So haben wir eine durchschnittliche Methoden-Komplexität von 2 und eine Klassenkopplung von 16. Die durchschnittlichen Zeilen pro Methode beträgt 6 und die von Klassen 125. Die Line-Coverage liegt bei 66% im Backend und 75% im Frontend. Um die User-Experience zu verbessern, sind unsere Frontend-Entwickler angehalten die 60:30:10-Faustregel anzuwenden. Diese hilft uns ein schlichtes und stimmiges Design durchzusetzen und den Blick des Anwenders durch Akzentfarben zu lenken.



Wir bedanken uns bei allen Beteiligten: