**Das unten wurde um 4 Uhr nachts aufs „Papier“ gebracht, dementsprechend können noch einige Fehler in Formulierung sowie Rechtschreibung enthalten sein. Wenn du/Ihr wollt könnt Ihr diese selbstständig Verbessern. Allgemein steht es dir/euch frei alles zu ändern, wie ihr es für richtig haltet. Ich hoffe es ist inhaltlich und mengenmäßig angemessen/ausreichend. Ihr könnt mir ja dann Feedback geben.**Ergonomie und Erprobung von Schlüsselbegriffen

**Ergonomie:**

Unter Ergonomie versteht man die wechselseitige Anpassung zwischen Menschen und ihren Arbeitsbedingungen. Auch in der Softwareentwicklung sollte auf die Ergonomie der Anwendung geachtet werden. Dabei sollte man folgende Kriterien/Eigenschaften im Blick behalten:

|  |  |
| --- | --- |
| Attribut | Beschreibung |
| Konsistenz | Der Nutzer sollte auf Bekannte Muster und Funktionen vertrauen können. |
| Verfügbarkeit | Features und Funktionen sollten nicht den Fluss der Software blockieren. |
| Verständlichkeit | Ähnliche Funktionen/Tools sollten in gleichen Sektionen zu finden sein und lassen sich auch ähnlich bedienen. |
| Automatisierung wiederholter Aufgaben | Repetitive Aufgaben sollten automatisiert werden oder optional automatisierbar sein. |
| Umgehende Rückmeldung | Der Nutzer sollte sowohl über fehlgeschlagene als auch durch erfolgreiche Aktionen informiert werden. (Fehler sollten „aufdringlich“ gezeigt werden z.B. Popup, Erfolge nur „nebenbei“ z.B. Statusziele) |
| Selbsterklärung | Durch Namen oder visuelle Elemente sollte sich der Sinn bzw. die Auswirkung einer Funktion erahnen lassen. |
| Anpassbarkeit | Der Nutzer sollte Optionen nach persönlicher Präferenz einstellen können |
| Fehlertoleranz | Fehler sollten vom Verursacher rückgängig gemacht bzw. korrigiert werden können |
| Erwartungskonformität | Funktionen sollten korrekt betitelt werden und nicht unvorhersehbare Auswirkungen mit sich bringen |
| Höflichkeit | Der Nutzer sollte auf Fehler, bei Aufforderungen oder einfachen Statusmeldungen immer Höflich auf seine Situation aufmerksam gemacht werden. |

Gerade bei Benutzereingaben sollte darauf geachtet werden, dass es zu möglichst wenig Missverständnissen kommen kann und die Eingaben für die vorgesehene Auftrittsumstände angemessen ausgewählt werden. Ferner sollte ebenfalls überlegt werden wie Komplex die Nutzereingaben sein können, sollen und dürfen. Generell sollten häufig auftretende Eingaben möglichst kurzgehalten werden, um Zeit und Aufwand bei der Eingabe niedrig zu halten. Beim Fall der Spracherkennung sollte ein gesunder Kompromiss zwischen für den Arbeiter angenehm aufzusagen als auch von der Spracherkennungssoftware schnell und zuverlässig erkennbar sein.

**Erprobung von Schlüsselbegriffen:**

Bei der Erprobung von Schlüsselbegriffen sollten die oben genannten Punkte der Ergonomie im Hinterkopf behalten werden. Jedoch spielt bei den Schlüsselbegriffen die zuverlässige Erkennung in Speech–to–Text Software eine wichtigere Rolle. Dabei sollten die Begriffe sorgfältig ausgewählt werden. Um die Fehlerquote zu senken hilft es, sich vorab Gedanken zu machen was der S2T-Sw helfen könnte Wörter besser zu differenzieren. Jedoch darf ein Erproben der Begriffe in einem Umfeld, welches dem zukünftigen Einsatzgebiet ähnelt, nicht vernachlässigt werden. Dabei muss sowohl die Fehlerquote als auch die Ergonomie betrachtet werden, wobei eine niedrige Fehlerquote automatisch die Ergonomie verbessert.

Im Folgenden wird am Beispiel Saplexa die Erprobung und das Ersetzen der Schlüsselbegriffe erklärt.

Zum Beginn der Entwicklung wurden die Schlüsselbegriffe „Start Number“ und „End Number“ genutzt, um das zusammensetzen einer neuen Ziffernkette zu Beginnen bzw. zu Beenden. Schnell wurde dabei klar, dass das Wort „Number“ weder zur besseren Erkennung noch zur Ergonomie beiträgt und wurde damit schnell verworfen. Damit waren die neuen ergonomischeren Schlüsselwörter „Start“ und „Stop“ (jeweils englisch ausgesprochen, da ein englisches Spracherkennungsmodell verwendet wurde).

In den im Entwicklungsprozess üblichen Unittest fiel eine häufige Fehlerkennung der beiden Worte auf, was wiederum die Ergonomie der Sprachbedienung durch eine hohe Fehlerquote verschlechterte. Beim genaueren Betrachten der Begriffe über eine Audioverarbeitungssoftware ist zu erkennen, dass die Wörter sich in Verschiedenen Audiovisuellen Formen kaum unterscheiden.

In den untenstehenden Tabellen sind zwei verschiedene Arten zur Visualisierung von Audiodaten.

Die Bilder obige Zeile der Tabellen zeigt ein sogenanntes Spektrogramm oder speziell bei Audiodaten auch Sonagramm genannt, welches die Energiedichte der Daten in bestimmten Frequenzbereichen über einen Zeitraum aufzeigt. Die Darauffolgende Zeile zeigt die eher übliche Waveform(Wellenform).

Die Folgende Tabelle stellt die bei Saplexa sich als problematisch herausgestellten Begriffe gegenüber.

|  |  |
| --- | --- |
| START | STOP |
|  |  |
|  |  |

Die Aufnahmen wurden mit einem für den Privatgebrauch sehr gutem Mikrophon in einer ruhigen Umgebung aufgenommen. Somit sind wenig Verunreinigungen wie Hintergrundrauschen in den Audiosamples enthalten. Es sind zwar Unterschiede zu erkennen, jedoch sind diese eher über den Zeitraum zu erkennen und können bei weniger optimalen Bedingungen sowie anderer Aussprache noch weniger erkennbare Unterschiede aufweisen.

|  |  |
| --- | --- |
| HANA | OKAY |
|  |  |
|  |  |

In der obigen Tabelle stehen sich die „neuen“ ausgetauschten Schlüsselbegriffe „Hana“ und „Okay“ der Anwendung gegenüber. Es sind direkt mehrere Unterschiede zu erkennen, was sich auch in der Fehlerkennungsrate der Speech-to-Text-Anwendung wiederspiegelt. Die Begriffe sind recht kurz geblieben, was sich zusammen mit der geringeren Fehlerquote positiv auf die Ergonomie der Applikation auswirkt. Eventuell könnten Fehlerkennung durch Wörter mit drei bis fünf Silben nochmals gesenkt werden.