

OSTFALIA HOCHSCHULE

EKDI-PROJEKT

Lastenheft

*Martin Krause 70478294, Tom Strunz 70476813, Patricia
Weber 70477439, Kristin Altmann 70476503, Marcel Pulst
70477125*

supervised by
Prof. JENSEN

20. Januar 2022

Inhaltsverzeichnis

1	Zweck und Ziel	3
2	Abgrenzung	3
3	Begriffe	3
4	Soll-Stand	3
4.1	Funktionen	4
4.2	Daten	4
4.3	Regeln	4
4.4	Nichtfunktionale Anforderungen	5
5	Dokumentenhistorie	5

1 Zweck und Ziel

Es soll ein Programm entwickelt werden, welches mit dem Aufrufen des „Lies.java“-Java-Programms den Inhalt einer beliebigen URL vorlesen soll. Dabei muss beachtet werden, dass die nicht dargestellten Bestandteile, zum Beispiel Tags und Metadaten, nicht vorgelesen werden sollen. Dieses Programm soll entwickelt werden, um zum Beispiel Personen mit Sehbehinderung Webseiten hinter URLs inhaltlich zeigen zu können.

2 Abgrenzung

Das Programm soll kein Graphical User Interface (GUI) enthalten und dem Programm wird es ebenfalls fast nur möglich sein, deutsche Webseiten vorzulesen, aufgrund von den auf Deutsch eingelesenen Silben.

3 Begriffe

- 'URL': Adressierung der vorzulesenden Website
- 'vorlesen': Abspielen von wav-Dateien, die den Worten des Texts entsprechen
- 'Text': der Haupttext auf der Website
- 'Start': Beginn/ Fortsetzen des Vorlesens
- 'Stopp': Unterbrechen des Vorlesens
- 'parsen': Zerlegen und umwandeln für eine Weiterverarbeitung
- 'Silbe': eine Buchstabenkombination einer Silbe im grammatikalischen Sinne zuweisbar ist, gilt dies als Silbe. Da nicht alle Silben der deutschen Sprache ausgesprochen werden können, werden Buchstaben, die nicht einer Silbe zugeordnet werden können, als Silbe aus nur einem Buchstaben betrachtet.

4 Soll-Stand

Was soll erreicht werden? Das Programm soll einen Text von einer Website, deren URL dem Programm übergeben wurde, einlesen und dann vorlesen. Wie soll das erreicht werden? Das Programm besteht im Wesentlichen aus zwei Teilen: dem Einlesen und dem Vorlesen. Beim Einlesen werden zuerst Daten von der Website gelesen und der HTML-Code geparkt. Für das Vorlesen müssen Start, Stopp, das Trennen des Textes in Silben und das Wandeln der Silben in gesprochene Sprache. Die Silben müssen dementsprechend noch ausgesprochen werden. Was kann noch erreicht werden? Es könnten eine Warnungen ausgegeben werden, wenn der Text wahrscheinlich nicht auf Deutsch ist, zu lang ist oder die wav-Dateien, die zum Vorlesen nötig sind, nicht gefunden werden. Außerdem könnte der vorgelesene Text ausgegeben werden.

4.1 Funktionen

- LF1: Beim Programmstart, wird eine URL übergeben.
- LF2: Das Vorlesen kann gestartet werden und beliebig pausiert und wieder weitergespielt werden. Außerdem kann das Programm beliebig beendet werden.
- LF3: Der Text aus einer Webseite wird herauskopiert.
- LF4: Der Text wird geparkt und die Steuerzeichen werden herausgefiltert.
- LF5: Der Text wird in Silben bzw. einzelne Buchstaben aufgeteilt.
- LF6: Die Silben werden nacheinander abgespielt.
- LF7: Der Thread kümmert sich in einer Endlosschleife um das Vorlesen nach einer bestimmten kurzen Pause.

4.2 Daten

- LD1: Die ersten Daten mit dem das Programm konfrontiert wird ist die übergebene URL.
- LD2: Mit Tastatureingabe „l“ beginnt das Programm mit dem Vorlesen.
- LD3: Mit Tastatureingabe „s“ kann das Vorlesen gestoppt und wieder weitergespielt werden.
- LD4: Mit Tastatureingabe „e“ kann das Programm beendet werden.

4.3 Regeln

- LR1: Das Programm soll mit dem Aufruf `java Lies „URL“` den Inhalt einer beliebigen URL vorlesen.
- LR2: Andere Libraries als die der Oracle JDK oder OpenJDK dürfen nicht verwendet werden.
- LR3: Der Aufruf externer Services oder Programme zum Vorlesen ist nicht erlaubt.
- LR4: Nicht dargestellte Bestandteile einer Website, d.h. Tags oder Metadaten, dürfen nicht vorgelesen werden.
- LR5: Falls eine nicht bekannte Webseite aufgerufen werden soll, wird eine Fehlermeldung ausgegeben.
- LR6: Falls eine nicht vorhandene Ton-Datei abgespielt werden soll, wird eine Fehlermeldung ausgegeben.

4.4 Nichtfunktionale Anforderungen

- NA1: Das Programm soll von Personen mit Sehbehinderung bedient werden können.

5 Dokumentenhistorie

1. — Martin Krause — Ersterstellung — 01.11.21
2. — Kristin Altmann — Bearbeitung (1. & 2. Punkt) — 06.11.21
3. — Martin Krause — Funktionen und Daten überarbeitet — 06.12.21
4. — Tom Strunz — Begriffe, Soll-Stand überarbeitet — 13.12.21
5. — Patricia Weber — Regeln, Nichtfunktionale Anforderungen — 13.12.21
6. — Martin Krause — Deckblatt, Inhaltsverzeichnis, Funktionen bearbeitet — 13.12.21
7. — Martin Krause — Deckblatt bearbeitet — 19.01.22
8. — Marcel Pulst — Dokument korrigiert — 20.01.22