



netidee

PROJEKTE

ContractVis

Entwickler Doku | Call 12 | Projekt ID 2116

Lizenz CC-BY-3.0 AT

Inhalt

1	Was ist es?	3
2	Für wen ist es? Wem hilft es wodurch?	3
3	Wie funktioniert es?	4
3.1	HighLighter Web Extension.....	5
3.2	HighLighter Web App	6
3.3	HighLighter Backend.....	7

1 Was ist es?

Im Projekt ContractVis wurde das Tool HighLighter entwickelt, das als visueller Assistent beim Umgang mit AGB-Texten, Datenschutzbestimmungen und anderen Texten eingesetzt werden kann.

Das Tool bereitet Vertragstexte mittels Text Analyse und Informationsvisualisierung auf, sodass ein Überblick direkt im Web Browser möglich ist. Davon ausgehend können Menschen gezielt alle Segmente zu einem Thema wie Bezahlungsmodalitäten oder Rückgabe im Detail lesen oder den Vertragstext einem anderen Text gegenüberstellen, um Unterschiede in diesen Themen zu beurteilen. ContractVis HighLighter ist also kein Black-Box Algorithmus, sondern ein Assistent, der mündige Menschen bei Entscheidungen unterstützt.

2 Für wen ist es? Wem hilft es wodurch?

Zur Zielgruppe gehören Personen, die bewusste Entscheidungen bei Online-Käufen oder derartigen Services treffen möchten und dabei auch einen begrenzten Aufwand haben wollen, um einen guten Überblick über das Kleingedruckte zu bekommen.

Zwei mögliche Nutzungsszenarien sind:

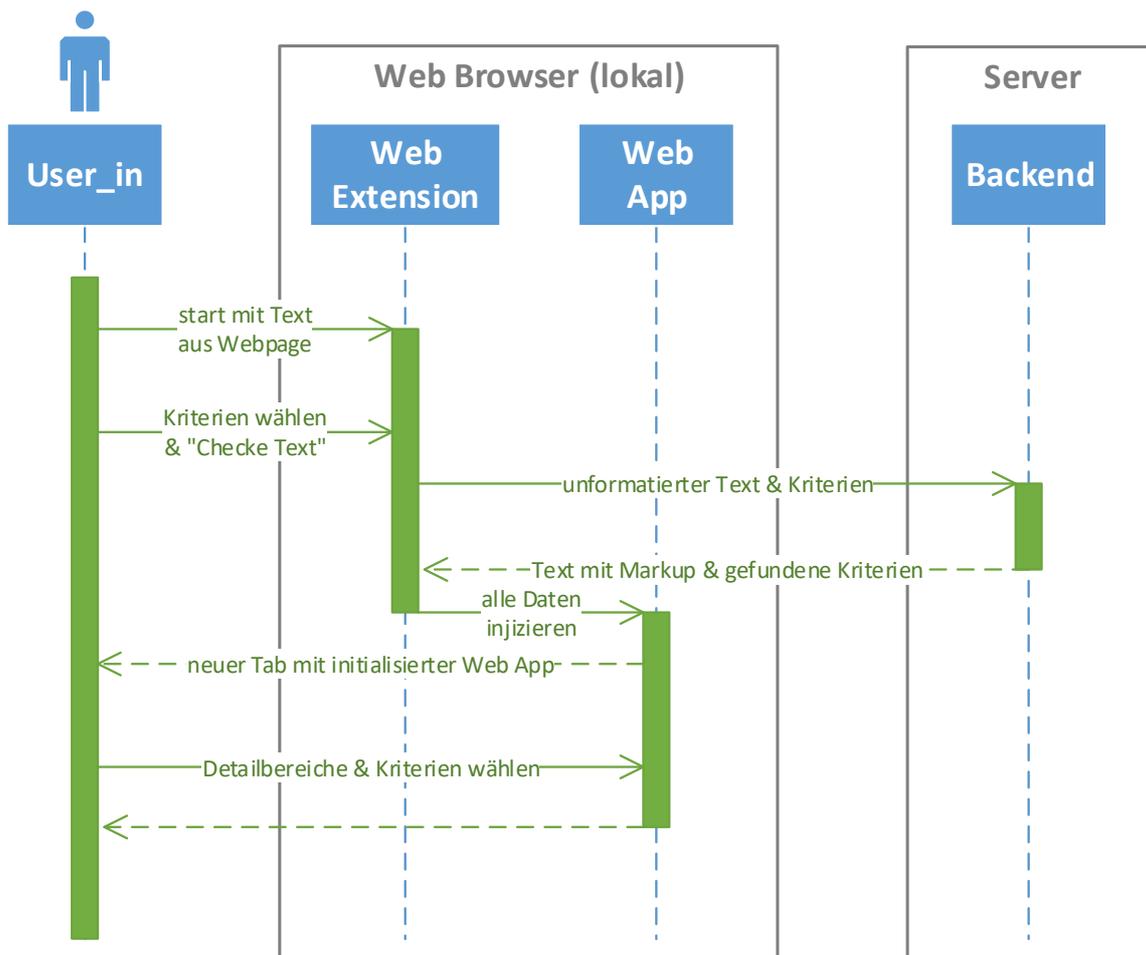
- Du willst dir online einen Rucksack kaufen und findest einen, der dir gefällt bei zwei verschiedenen Anbietern. Du nutzt das Tool, um die AGBs und die Einkaufsbedingungen der Anbieter zu vergleichen und das für dich beste Angebot zu finden, bevor du den Einkauf tatsächlich tätigst.
- Du möchtest bei einem Online-Shop mehrere Artikel kaufen und folglich liegt viel Geld in deinem Warenkorb. Du möchtest schnell und einfach herausfinden, ob du auch auf Rechnung zahlen kannst und wie dein Einkauf versandt wird. Du nutzt das Tool, um die AGBs nach diesen Kriterien zu filtern und zu vergleichen, damit du an die für dich wichtigen Infos problemlos gelangst.

3 Wie funktioniert es?

für Entwickler: Übersicht und detailliertes Systemkonzept, SW-Struktur

Für Umsetzung von ContractVis HighLighter haben wir eine dreiteilige Software Architektur entworfen:

- Die User_innen verwenden primär eine Web Extension, also ein Browser Add-on. Damit können sie Rechtstexte oder andere zu analysierende Texte direkt auf Websites markieren und so in den HighLighter importieren.
- Zur Analyse des Texts wird dieser über ein REST Interface an einen Backend Server geschickt und dort verarbeitet.
- Die Ergebnisse werden in einer Web App, also einer dynamisch im Web Browser laufenden Webseite, dargestellt. Dabei werden die analysierten Texte von der Web Extension direkt in die Web App injiziert, sodass die personen-generierten Daten niemals am Webserver landen.



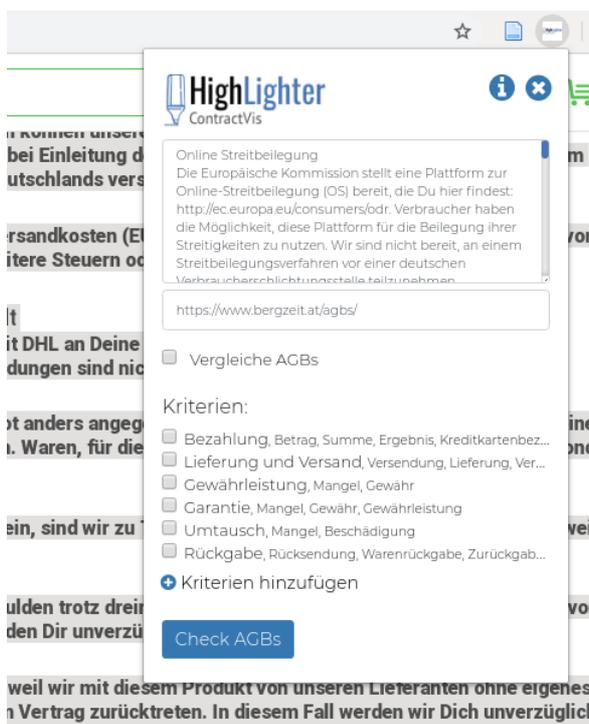
Für diese Architektur sprechen vor allem zwei Designargumente:

- Die Web Extension kann schlank konzipiert werden - mit weniger Lines of Code und weniger eingebundenen Softwarepaketen. Gerade bei einer Web Extension, die in einem Browser Store platziert werden soll, gelten da strengere Auflagen. Währenddessen können sich das Backend und die Web App auf die Textanalyse bzw. die interaktive Visualisierung konzentrieren.
- Daten der User_innen werden, wenn überhaupt, nur lokal in der Browser Extension gespeichert. Zwar werden sie für die Analyse an das Backend geschickt, aber auch dort ist keine Speicherung vorgesehen (was im Source Code auch gut nachvollzogen werden kann).

Wir implementieren alle drei Komponenten mit *TypeScript* basierend auf einer Build Umgebung mit *node.js*, *npm* und *webpack*.

3.1 HighLighter Web Extension

Die Web Extension übernimmt den ausgewählten Text aus dem aktuellen Web Browser Tab, erlaubt Kriterien auszuwählen, startet einen REST Aufruf zur Textanalyse im Backend und injiziert die Ergebnisse in die Web App, die in einem neuen Browser Tab gestartet wird. Außerdem erlaubt sie 2 Texte zu vergleichen und speichert dazu die letzten 5 geladenen Texte.



Eingebundene NPM Packages (Auswahl):

- jquery
- SweetAlert2
- webextension-polyfill

Einrichtung:

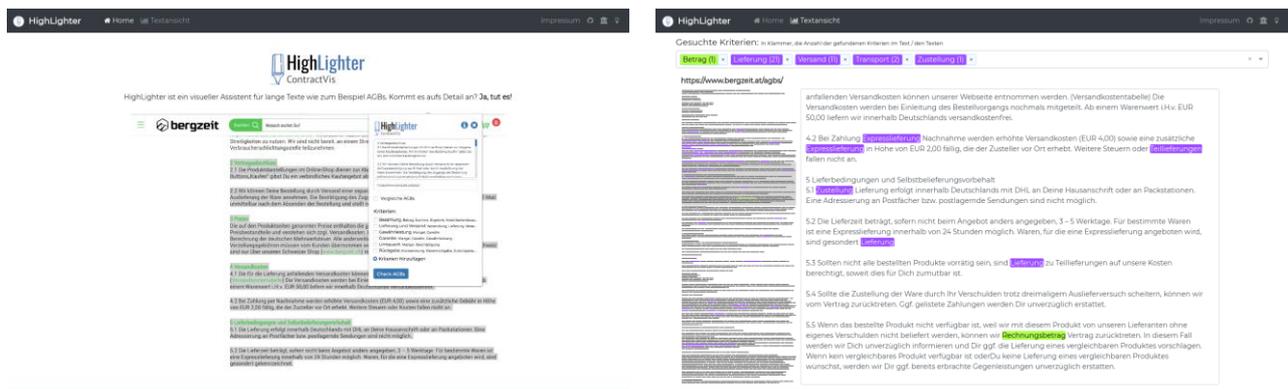
- „npm install“ – zur Installation aller benötigten Packages (lt. package.json)
- npm run dev:chrome – builden mit Hot-Reload
- im Web Browser (z.B. Chrome) den Entwicklermodus aktivieren
- das Verzeichnis dist/chrome in die Extensions Seite von Chrome ziehen

Weiterführende Informationen & Issue Tracking:

<https://github.com/fhstp/highlighter-webextension/>

3.2 HighLighter Web App

Die Web App ist eine Angular-basierte Website mit drei Seiten: Die Home Seite zeigt einführende Informationen und eine Onboarding Materialien für neue User_innen. Die Textansicht zeigt die gehighlighteten Texte in einer Overview/Detail Darstellung. Die dritte Seite ist ein Impressum.



Eingebundene NPM Packages (Auswahl):

- Angular Version 6.0.3
- Bootstrap Version 4.1.3
- Fontawesome Version 4.7.0
- D3.js Version 5.7

Einrichtung:

- `npm install` – zur Installation aller benötigten Packages (lt. `package.json`)
- `ng serve` – starten eines lokalen Testservers (lädt automatisch neu)
- im Browser erreichbar unter <http://localhost:4200/>

Weiterführende Informationen & Issue Tracking:

<https://github.com/fhstp/highlighter-webapp>

3.3 HighLighter Backend

Der Backend Server bietet ein REST Interface das Texte und Markup Kriterien entgegennimmt und den markierten Text sowie die gefundenen Vorkommnisse zurückliefert. Weitergehende linguistische Analysen wären möglich, aber aufgrund der derzeitigen Usage Szenarios nicht erforderlich.

Eingebundene NPM Packages (Auswahl):

- `express` (web framework)
- `body-parser` (parse incoming request bodies)

Einrichtung:

- „`npm install`“ – zur Installation aller benötigten Packages (lt. `package.json`)
- Im Projektverzeichnis die `.env` Datei öffnen oder anlegen und darin den benötigten TCP Port der REST Schnittstelle festlegen:

```
SERVER_PORT = XXXX
```
- HTTPS SSL Zertifikat (z.B. von `lets encrypt` oder `acme.sh` oder aus anderer Quelle). Die Dateien `cert.pem` und `key.pem` müssen im Ordner `ssl` kopieren oder verlinken sein.
- ggf. Port Freigabe in der Firewall
- „`npm start`“ ausführen.

Weiterführende Informationen & Issue Tracking:

<https://github.com/fhstp/highlighter-backend/>