Konzept PSE Demo Video 2

Team

Tobias Brunner, Thea Waldleben, Jan Wolfensberger, Janni Lazar, Sascha Künzler

Thema

Vorstellung realisierte Erweiterung an Medcodesearch und vertiefte Erklärung des Proof of Concept

Zielgruppe

Zielgruppe 1: Andere Entwickler
Zielgruppe 2: eonum (?)

Inhalt

Medcodesearch wurde nun mit der Krankenkassen Leistungsverordnung als durchsuchbare Quelle erweitert. Hierfür werden die jährlich aktualisierten PDF Dokumente mit einem Crawler aus dem Web geholt, geparst und in die Datenbank gespiesen. Via Elasticsearch kann man nun den gesamten Inhalt via Frontend durchsuchen. Die Darstellung der Suchresultate wurde gemäss Kundenwunsch angepasst. Es wurde noch Breadcrumbs hinzugefügt und verwandte KLV-Leistungen werden automatisch verlinkt.

Es ist uns gelungen, im KLV-Anhang eine Struktur zu finden, so dass wir diesen Anhang als strukturierte Quelle integrieren konnten. Die Integration als strukturierte Quelle bedeutete einen grösseren Programmieraufwand, bietet jedoch deutlich mehr Vorteile beim Verknüpfen verschiedener Datenbankeinträgen. Nach dem erfolgreichen Abschluss der KLV Integration, einigten wir uns mit eonum darauf, im zweiten Teil des Praktikums ein Proof of Concept zur Integration von unstrukturierten Quellen zu erarbeiten.

Das neue Ziel war, herauszufinden, ob es möglich ist ganze PDFs in Elasticsearch zu indexieren ohne grosse Verarbeitung, wie wir es nun mit dem KLV gemacht haben. Auf Wunsch von eonum sollen wir es mit dem Dokument ‚Medizinischen Kodierhandbuchs' zeigen, welches ein sehr unstrukturiertes Format hat. Da das Medizinische Kodierhandbuch das Komplexeste der zu integrierenden Dokumente ist, sollte der Proof of Concept auch für weniger komplizierte Dokumente einwandfrei funtionieren.

In den letzten zwei Wochen, in denen wir noch am Poc gearbeitet haben, konnten wir beweisen, dass dies geht. Man braucht hierfür nicht mehr ein Parser, welcher alle Edge-Cases berücksichtigt und eine komplizierte Implementierung voraussetzt. Ein Dokument kann via Rake-Task in einzelne Seiten zerstückelt werden. Die einzige Verarbeitung die es nun noch braucht, um die Dokumente durchsuchbar zu machen sind folgende: Man muss via Gems den Text aus den einzelnen PDF-Seiten herauslesen, sowie die PDF-Seite in Base64 umwandeln. Den herausgelesenen Text und der Base64-Code wird in die Datenbank geschrieben. Bei mehrsprachigen Dokumenten muss man diese Verarbeitung jeweils für die erste Seite aller PDFs machen, dann für die zweite Seite etc. also eine parallele Verarbeitung. Pro Quelldokument braucht es ein Model, welche alle Sprachen beinhaltet. (Man könnte auch ein Model für alle Dokumente machen, doch dies würde die Durchsuchung verlangsamen). Elasticsearch kann nun die Datenbank indexieren und durchsuchbar machen. Wenn man nun via Frontend nach etwas sucht, geht die Suchanfrage via Elasticsearch die extrahierten Texte durchsuchen. ES gibt eine ID zurück, wo diese übereinstimmung in der Datenbank zu finden ist. Um nun das Formiatierungsproblem des unstrukturierten Inhalts zu lösen, kann man als Backend-Response das Base64 zurückgeben. Dies kann man dann im Frontend wieder in das originale PDF zurückkonvertieren mithilfe eines Angular Components. Darin ist sogar die Durchsuchung des PDFs möglich, was das Highlighten im Original-PDFs ermöglicht. Wenn man nun weiterlesen möchte auf der nächsten Seite, kann man einfach ‚weiterblättern' indem als BAckend-Request das Base64 der nächsten Seite angefordert und wieder anzeigt wird.

Bei der Umsetzung haben wir stetig beachtet, dass das ganze skalierbar sein muss. Mithilfe dieses Proof of Concept sollte es nun möglich sein, neue unstrukturierte Quellen mit verhältnissmässig wenig Aufwand bei Medcodesearch hinzuzufügen.

Es war spannend und lehrreich, verschiedene Ansätze der Datenverwertung auszuarbeiten. Wenn es reicht ein Dokument Seitenweise anzuzeigen und keine komplexe Verknüpfbarkeit nötig ist, ermöglicht unser Proof of Concept sehr einfach PDF-Quellen volltext zu durchsuchen und anzuzeigen. Sollen Daten für komplexere Verknüpfungen verfügbar sein, lohnt es sich pro Quelle ein Parser zu erstellen.

Ablauf

Erweiterungen an Medcodesearch

Proof of Concept

Kernbotschaft

Verschiedene Verwendungszwecke von Daten erlauben es uns, mehr oder weniger strukturierte Datenbanken aufzubauen.

Dauer

8-10 Minuten

Musik

Kein

Off-Stimme

Das ganze Video ist hinterlegt mit einer Off-Stimme, welche einem durch das ganze Video begleitet.