Richtlinie zum „Software-Sprint“

Feminizidmap – Datenbanksoftware zum Dokumentieren von Femi(ni)ziden in Deutschland

---

Schlussbericht

Zuwendungsempfänger:

Lisa Passing

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS21S14 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Femi(ni)zide werden in Deutschland nicht explizit in der Kriminalstatistik erfasst. Erst seit wenigen Jahren wertet die Statistik Morde in intimen Partnerschaften aus, nur eine von mehreren Unterkategorien von Femi(ni)ziden.

In den Sozialwissenschaften wird seit vielen Jahren zu diesem Phänomen geforscht und in der Zivilgesellschaft nehmen sich zunehmend mehr Projekte diesem Thema an. Ihnen fehlt eine solide Datengrundlage, da offizielle Zahlen fehlen. Die Daten müssen von Hand gesammelt werden, oft geschieht das pro Projekt in einem Spreadsheet.

Hier setzt Feminizidmap an: Eine Datenbank mit automatisierter Auswertung soll Projekte dabei unterstützen Fälle zu dokumentieren, Daten strukturiert zu sammeln und anderen zur Verfügung zu stellen.

Beitrag des Projektes zu den Zielen der Förderinitative „Software-Sprint“

Die entstehende, quelloffene Software unterstützt sie beim dokumentieren, verfolgen, verwalten und auswerten von Fällen, entsprechende Rohdaten und automatisierte Auswertungen bleiben online als Informationsquelle für alle bestehen.

Ausführliche Darstellung der Ergebnisse

Das Kerntool, an dem ich im Rahmen der Förderung gearbeitet wurde heißt Mapper. Mapper nutzt eine klassische Server-Client-Architektur. Der Server, der sowohl eine interne, authentifizierte API bereitstellt, als auch später die öffentliche API, ist mit dem Framework Ruby on Rails umgesetzt und nutzt eine Postgres Datenbank. Der Frontendclient, der die interne API konsumiert, basiert auf dem Framework VueJS. Für einen einfachen Einsatz auf Produktionsservern stehen Dockerfiles und ein Docker-Compose-Script bereit.

Mapper soll alle wesentlichen Aufgaben übernehmen, die im Austausch mit dem Team erarbeitet wurden. Davon sind folgende vollständig oder teilweise umgesetzt:

- NutzerInnenverwaltung mit verschiedenen Rollen und Rechten

- Interface um Datenstruktur der gesammelten Daten veränder- bzw erweiterbar zu machen

- Konzept und Beginn der Umsetzung des Dateneingabe-Interface, welches sowohl ein Schritt-für-Schritt-Workflow als auch ein punktuelles Springen zwischen Datenebenen ermöglicht

- Alle Änderungen am Datenstatz werden automatisch gespeichert und Bearbeitungslogs geführt werden. Hieraus soll später "History" pro Eintrag entstehen, aus der zur Not vergangene Zustände wiederhergestellt werden können.

Weitere Aufgaben sind noch in frühen Konzeptionsphasen, sollen aber in vor der Übernahme durch das Team fertig gestellt werden, da sie weitere aufwändige, händische Arbeit ersetzen. Dazu gehören:

- Automatisiertes erstellen von PDFs der Originalquellen zur Dokumentation

- Such- und Filter-Interface um schnell bestimmte Fälle zu finden und weiter bearbeiten zu können

- Review- bzw Nachbereitungworkflow sowie ein Veröffentlichungsmechanismus von jährlichen Berichten und Statistiken

Neben der Mapper entstand auch die auf dem Ruby Framework Sinatra basierte App Briefkasten, die der Zivilgesellschaft das direkte, und auf Wunsch anonyme, Melden von Femi(ni)ziden an das Projekt ermöglicht.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Für die Zielgruppe es Dokumentationsteams ist ein Prototyp entstanden, der in den Grundzügen alle wichtigen Arbeitsschritte im Team abbildet. Der Prototyp ist noch nicht reif genug den bisherigen, Spreadsheet-basierten Workflow zu ersetzen. Das bisherige Feedback aus dem Team stimmt zuversichtlich, dass viel manuelle Arbeit und häufige Fehlerquellen zukünftig wegfallen werden.

Die Pläne für den Prototyp bis zur benutzbaren App sind bereits ausgearbeitet und warten auf ihre Umsetzung, die nach der Förderphase langsamer laufen wird.

Als aktivistisches Open Source Projekt hoffe ich sowohl Unterstützung in der Tech-Community zu finden, als auch das Projekt selbst zu übersetzen und für ähnliche Dokumentationsgruppen nutzbar zu machen.

Mich persönlich hat die Arbeit am Projekt vor allem in den Bereichen Projektplanung, Konzeptionierung von Datenbank bis User-Experience mehr gefordert als ich es gewohnt bin, weil ich normalerweise bestehende Konzepte umsetze. Hier habe ich definitiv viel dazu gelernt.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Die erste Version des Datenbankschemas wurde fast eins zu eins von der Projekt bisher genutzten Tabellenstruktur übernommen. Nachdem ich darauf aufbauend bereits die ersten Endpunkte der Schnittstelle programmierte, stellte sich in mehreren Meetings mit den Freiwilligen aus dem Projekt heraus, dass das Schema an einigen Stellen nicht nur vereinfacht werden kann, sondern an anderen Stellen viel flexibler gedacht werden muss.

Daraufhin plante ich das Datenbankschema komplett neu, mit einem viel höherem Abstraktionslevel, und verwarf alle Teile, die mit dem alten Schema zusammenhingen.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Das Projekt ist für EndnutzerInnen noch nicht live.

Der Code kann auf GitHub eingesehen werden https://github.com/feminizidmap/feminizid-mapper. Technische Dokumentation und Entwicklungsblog befinden sich unter https://tech.feminizidmap.org/.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Der Arbeits- und Meilensteinplan waren von vorn herein sehr optimistisch angelegt. Bei der Planung und Umsetzung der Umstrukturierung des Datenbankschemas verlor ich viel Zeit, die ich nicht mehr aufholen konnte und deswegen früh alle „Nice-to-Haves“ strich.

Dazu kamen, dass ich die Ansprüche an die User-Experience des Dateneingabeinterfaces unterschätzte. Nach mehreren Gesprächen und „Shadowing“ von Freiwilligen bei ihrer Arbeit wurde klar, dass das Interface sowohl Flexibilität für verschiedene Workflows bieten muss, aber gleichzeitig potentielle Fehler bei der Eingabe verhindern und, wo es Sinn macht, Schritte

automatisieren soll. Die Konzeptionierung nahm viel nicht-eingeplante Zeit in Anspruch.

Insgesamt konnte ich große Teile des Arbeitsplans nicht oder nicht vollständig umsetzen, aber realistischer planen.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Keine.