



SOFTWARE SPRINT (PROTOTYPE FUND) AUSWAHLRUNDE MÄRZ 2018

Konsolidierter Schlussbericht

Förderkennzeichen:

01S18S24*
01S18S25
01S18S26
01S18S27*²
01S18S28*
01S18S29
01S18S30*
01S18S31*
01S18S32*
01S18S33*
01S18S34*
01S18S35
01S18S36
01S18S37
01S18S38*
01S18S39
01S18S40
01S18S41*
01S18S42

Vorhabenbezeichnung: Software Sprint – 19 Einzelvorhaben

Laufzeit der Einzelvorhaben: 01.10.2018-31.03.2019 (*01.09.2018-28.02.2019;*² 15.09.2018-14.03.2019)

Die diesem Bericht zugrunde liegenden Vorhaben wurden mit Mitteln des Bundesministeriums für Bildung und Forschung unter den o.g. Förderkennzeichen gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den jeweils genannten Autoren (Zuwendungsempfängern).

Berichtsblatt

1. ISBN oder ISSN	2. Berichtsart (Schlussbericht oder Veröffentlichung) Schlussbericht	
3. Titel Software Sprint – 19 Einzelvorhaben		
4. Autor(en) [Vorname(n), Name(n)] Manuela Blechschmidt, Julia Friesel Benedikt Kaffai Petra Hagemann, Ulrich Fouquet Thieke, Böckhoff, Rudolph, Reissner Princiya Marina Sequeira Julia Anaya Gonzalez Michiel Nuyts Holger Levsen Eric Hartsuyker Dittler, Meurer Svante Schubert Just, Scheunemann Timo Lundelius Jan Bölsche Jan Philip Steimel Alexander Couzens Ostendorff, Blume, Langhabel Nieß, Heinzmann, Brock-Wenzek, Sievers Kochetkova, Zemke	5. Abschlussdatum des Vorhabens 28.02.2019/14.03.2019/31.03.2019	6. Veröffentlichungsdatum 13.06.2019
	7. Form der Publikation	
	9. Ber. Nr. Durchführende Institution	
8. Durchführende Institution(en) (Name, Adresse) 01IS18S24 Blockparty GbR - Manuela Blechschmidt & Julia Friesel 01IS18S25 Benedikt Kaffai 01IS18S26 StalkerBuster - Petra Hagemann und Ulrich Fouquet GbR 01IS18S27 Thieke, Böckhoff, Rudolph und Reissner GbR 01IS18S28 Princiya Marina Sequeira 01IS18S29 Julia Anaya Gonzalez 01IS18S30 Johannes & Nuyts GbR 01IS18S31 Holger Levsen 01IS18S32 Eric Hartsuyker 01IS18S33 Dittler Meurer GbR 01IS18S34 Svante Schubert 01IS18S35 JustScheunemann Sensorimotor GbR 01IS18S36 Meine.Luftdaten.info GbR c/o We-Build.City GmbH 01IS18S37 Jan Bölsche 01IS18S38 Jan Philip Steimel 01IS18S39 Alexander Couzens 01IS18S40 Open Legal Data - Ostendorff Blume Langhabel GbR 01IS18S41 Nieß Heinzmann Brock-Wenzek Sievers GbR 01IS18S42 Kochetkova Zemke GbR	10. Förderkennzeichen 01IS18S24-01IS18S42	
	11. Seitenzahl 82	
	12. Fördernde Institution (Name, Adresse) Bundesministerium für Bildung und Forschung (BMBF) 53170 Bonn	
16. Zusätzliche Angaben	13. Literaturangaben	
	14. Tabellen	
	15. Abbildungen	
17. Vorgelegt bei (Titel, Ort, Datum) Deutsches Zentrum für Luft- und Raumfahrt e. V. (DLR) DLR Projektträger Gesellschaft, Innovation, Technologie Softwaresysteme und Wissenstechnologien Berlin		

18. Kurzfassung

Der Schlussbericht umfasst die Einzelschlussberichte der vierten Auswahlrunde der Fördermaßnahme Software Sprint (Einreichungsdatum Skizze: 31.03.2018). Die Auswahl der Vorhaben erfolgte nach externer Begutachtung unter Abstimmung mit dem BMBF.

19. Schlagwörter

Software Sprint, Prototype Fund

20. Verlag

21. Preis

Document Control Sheet

1. ISBN or ISSN	2. type of document (e.g. report, publication) Final report
3. title Software Sprint – 19 Einzelvorhaben	
4. author(s) (family name, first name(s)) Manuela Blechschmidt, Julia Friesel Benedikt Kaffai Petra Hagemann, Ulrich Fouquet Thieke, Böckhoff, Rudolph, Reissner Princiya Marina Sequeira Julia Anaya Gonzalez Michiel Nuyts Holger Levsen Eric Hartsuyker Dittler, Meurer Svante Schubert Just, Scheunemann Timo Lundelius Jan Bölsche Jan Philip Steimel Alexander Couzens Ostendorff, Blume, Langhabel Nieß, Heinzmann, Brock-Wenzek, Sievers Kochetkova, Zemke	5. end of project 28.02.2019/14.03.2019/31.03.2019
	6. publication date 13.06.2018
	7. form of publication
8. performing organization(s) (name, address) 01IS18S24 Blockparty GbR - Manuela Blechschmidt & Julia Friesel 01IS18S25 Benedikt Kaffai 01IS18S26 StalkerBuster - Petra Hagemann und Ulrich Fouquet GbR 01IS18S27 Thieke, Böckhoff, Rudolph und Reissner GbR 01IS18S28 Princiya Marina Sequeira 01IS18S29 Julia Anaya Gonzalez 01IS18S30 Johannes & Nuyts GbR 01IS18S31 Holger Levsen 01IS18S32 Eric Hartsuyker 01IS18S33 Dittler Meurer GbR 01IS18S34 Svante Schubert 01IS18S35 JustScheunemann Sensorimotor GbR 01IS18S36 Meine.Luftdaten.info GbR c/o We-Build.City GmbH 01IS18S37 Jan Bölsche 01IS18S38 Jan Philip Steimel 01IS18S39 Alexander Couzens 01IS18S40 Open Legal Data - Ostendorff Blume Langhabel GbR 01IS18S41 Nieß Heinzmann Brock-Wenzek Sievers GbR 01IS18S42 Kochetkova Zemke GbR	9. originator's report no.
	10. reference no. 01IS18S24-01IS18S42
	11. no. of pages 82
12. sponsoring agency (name, address) Bundesministerium für Bildung und Forschung (BMBF) 53170 Bonn	13. no. of references
	14. no. of tables
	15. no. of figures
16. supplementary notes	
17. presented at (title, place, date) Deutsches Zentrum für Luft- und Raumfahrt e. V. (DLR) DLR Projektträger Gesellschaft, Innovation, Technologie Softwaresysteme und Wissenstechnologien Berlin	

18. abstract The final report includes the individual final reports for the 4th call of participants within the public funding activity Software Sprint. The selection of the projects (call No. 4 deadline: 31.03.2018) took place after external evaluation under coordination with the BMBF.	
19. keywords Software Sprint, Prototype Fund	
20. publisher	21. price

Richtlinie zum „Software-Sprint“

Blockparty

Schlussbericht

Zuwendungsempfänger: Blockparty GbR

Julia Friesel, Manuela Blechschmidt

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S24 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Soziale Netzwerke fördern gesellschaftliche Teilnahme und erleichtern Wissenstransfer und Gedankenaustausch.

Die meisten bestehenden Angebote sammeln und missbrauchen jedoch die Daten Ihrer User. Daneben gibt es auch Lösungen mit dezidiert nicht-kommerziellen und emanzipatorischen Ansätzen. Diese sind jedoch häufig technisch komplex, schwierig zu installieren und zum Teil mit schlechter Usability. Dadurch werden sie viel zu wenig genutzt, auch wenn sich immer mehr Menschen Alternativen zu Facebook und Co. wünschen. Hier setzen wir an. Wir wollen einen niedrighschwelligen Zugang zu einem nicht-kommerziellen Netzwerk bieten, der sich durch gute Usability und Einfachheit auszeichnet.

Die geplanten Meilensteine waren:

- Technische Einarbeitung und Konzeption mit Wireframes
- Grafisches Design der Anwendung
- Programmierung der Desktop-Anwendung mit Basis-Funktionen: Erstellen eines neuen Netzwerks, Beitreten zu einem Netzwerk, Betrachten der Nachrichten in bestehenden Netzwerken
- Usability-Tests
- Automatisierte Tests und Qualitätssicherung
- Dokumentation zur Software und Sicherstellen einer hohen Testabdeckung
- Website für den Download und die Dokumentation, inklusive Beispielen und Vorschlägen für Anwendungsfälle
- Blog-Post über das Tool
- Talk bei einer geeigneten Konferenz um die Bekanntheit zu erhöhen

Beitrag des Projektes zu den Zielen der Förderinitiative

„Software-Sprint“

Unser Zielgruppe sind alle Menschen, die gern soziale Netzwerke nutzen, aber nicht um jeden Preis: für jene, die die bereits bestehenden nicht-kommerziellen, technisch zu anspruchsvollen Lösungen scheuen, für die, die sich ein Netzwerk nach eigenen Bedürfnissen gestalten wollen. Wir wünschen uns auch für uns selbst ein solches Tool und sehen großes Potenzial für die thematische Nutzung z. B. als Elternnetzwerk.

Unser Tool ist auch für Menschen interessant, die in Gebieten oder Umständen leben, in denen es keine stabile Internetverbindung gibt, da Blockparty offline-fähig ist.

Es wird eine Website für den Download geben und wir wollen das Projekt in Talks und über Blogs verbreiten.

Ausführliche Darstellung der Ergebnisse

Von den geplanten Meilensteinen konnten nicht alle erzielt werden. Wir haben uns viel Zeit genommen für die technische Einarbeitung und die Konzeption mit Wireframes, sowie das grafische Design der Anwendung. Die Programmierung der Desktop-Anwendung mit den Basis-Funktionen offenbarte dann viele unerwartete Herausforderungen; die Funktionalität zum Erstellen eines neuen Netzwerks und das Beitreten zu einem Netzwerk wurden fertig gestellt, nahmen aber wesentlich mehr Zeit als geschätzt in Anspruch. Gründe hierfür waren, dass wir uns die Funktionalität erst erschließen und feststellen mussten, dass die Dokumentation der bestehenden Scuttlebutt-Module oft lückenhaft oder veraltet ist. Hinzu kam, dass viele SSB-Module nur für die Nutzung im Hauptnetz geschrieben und getestet sind. Sobald wir sie mit Blockparty in einem „Altnet“, also mit einer alternativen Netzwerk-ID benutzen wollte, offenbarten sich Bugs und fehlende Funktionalität, die unter anderem zur Folge hatten, dass wir für den Moment einen inoffiziellen, gepatchten Fork von Scuttlebutt benutzen.

Das Betrachten der Nachrichten in bestehenden Netzwerken ist rudimentär umgesetzt.

Während der wöchentlichen Teambesprechung stellten wir schnell fest, dass die Zeit knapp werden würde und beschlossen, uns auf das Erstellen neuer Netzwerke zu konzentrieren, da es sich hier um den grössten Mehrwert von Blockparty handelt.

Wir konnten die Dokumentation zur Software mithilfe von Readme-Dateien auf Github sicherstellen, und haben eine Website zur Information und die Dokumentation, inklusive Beispielen und Vorschlägen für Anwendungsfälle erstellt. Beim Scuttle Camp in Neuseeland im März 2019 wurde das Projekt informell in Workshops vorgestellt und stiess auf grosses Interesse.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Auch wenn Blockparty nicht komplett fertig gestellt wurde, stellt es bereits einen Mehrwert für alle dar, die dezentralisierte private soziale Netzwerke mit Secure Scuttlebutt aufsetzen wollen. Auf SSB selber gab es diesbezüglich bereits Interesse, da mit der steigenden Nutzerzahl von SSB auch die Ideen für alternative Anwendungen abseits des „Mainstream“ sozialen Netzwerks mehr werden.

Was sich auf jeden Fall festhalten lässt, ist dass unsere Arbeit an Blockparty viele Bugs und Stolperfallen für SSB-EntwicklerInnen offenbar wurden. Ein weiterer Effekt unserer Arbeit: Auf dem Scuttle Camp gab es viele Interessierte, die Elemente von Blockparty weiterverwenden wollen, speziell den Wizard zum Aufsetzen eines Netzwerks, der als Stand-Alone-Lösung zum Einbinden in größere Anwendungen zur Verfügung steht. Für das Blockparty-Team hat die Arbeit am Projekt in jedem Fall eine Weiterentwicklung in professioneller und fachlicher Hinsicht dargestellt. Wir haben zu hundert Prozent mit Pairing am Code gearbeitet, also jederzeit mit dem 4-Augen-Prinzip, das die Arbeit zwar manchmal langsamer macht, aber Bugs verhindert und vor allem die Wissensweitergabe unterstützt.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Die Desktop-Anwendung wurde mit Electron erstellt, also mit Browser-Technologien. Für die Erstellung des Frontends wollten wir uns daher für ein Framework entscheiden, das das Management einer solchen Single-Page-Application erleichtert. Dabei wollten wir etwas auswählen, das leichtgewichtig und anwenderinnenfreundlich ist. Wir uns beide bereits einmal das Framework "choo" angeschaut, und es sah vielversprechend aus. Wir haben aber relativ schnell festgestellt, dass es einiges Bugs hat, die uns das Leben erschwert haben. Daher haben wir zu Beginn des Projekts zu "mutant" gewechselt, einem komplett auf Observables basierenden Framework, in dem bereits andere SSB-Clients geschrieben sind. Die Umstellung war nicht zu aufwendig, das Erlernen der neuen Logik allerdings schon.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

- <https://github.com/blockparty-ssb>
- <https://blockparty-ssb.github.io/>
- #blockparty Channel auf Secure Scuttlebutt

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

In der ersten Projekthälfte haben wir wesentlich mehr Arbeit mit dem Projekt verbracht. Das lag zum einen daran, dass uns initiale Schwierigkeiten beim Setup und bei der Einarbeitung viel mehr aufgehalten haben als zunächst gedacht. Zum anderen konnten wir diese Zeit aber auch investieren, weil wir sie zur Verfügung hatten. Uns war es lieber, diese Anfangsprobleme konzentriert zu lösen, als sie aufzuschieben. Hinzu kam, dass ein Teammitglied für den Februar eine Reise unter anderem zum Scuttle Camp geplant hatte, sodass wir wussten, dass zum Ende des Projekts weniger Zeit zur Verfügung stehen würde.

Richtlinie zum „Software-Sprint“

GuideLight – Werkzeuge für mehr digitale Mündigkeit

Schlussbericht

Zuwendungsempfänger: Benedikt Kaffai

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen **01IS18S25** gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Mit der Verabschiedung der Datenschutzgrundverordnung bzw. dem Inkrafttreten im Mai 2018 haben wir Europäer*Innen eine Reihe von verbrieften Rechten im Bezug auf personenbezogene Daten und deren Verarbeitung durch Dritte erhalten. Neben einem Recht auf Auskunft, haben Nutzer*Innen auch die Möglichkeit, der Verarbeitung ihrer Daten zu widersprechen, sie korrigieren, migrieren oder löschen zu lassen. Durch die Datenschutzgrundverordnung in die Pflicht genommen, stellen Unternehmen, Behörden etc. den Nutzer*Innen Einstellungsmöglichkeiten, Opt-Ins, Anfrage- und Kontaktformulare zu Verfügung, um diese Rechte in Anspruch nehmen zu können.

Diese Kommunikations- und Einstellungsmöglichkeiten sind allerdings uneinheitlich gestaltet, in der Regel kaum verständlich oder auf den Websites versteckt. Die Benutzeroberflächen scheinen teilweise so gestaltet zu sein, dass Nutzer*Innen verleitet werden, entgegen ihrer eigenen Interessen zu handeln; bewusst oder unbewusst werden hier sogenannte *Dark Pattern* als Gestaltungsprinzip genutzt, um unerwünschtes Nutzer*Innenverhalten zu vermeiden.

Die primäre Motivation für das Projekt *GuideLight* ist es, das Wissen um die Rechte von Verbraucher*Innen und Nutzer*Innen zu verbreiten und die Inanspruchnahme allen möglichst niederschwellig, kostenfrei und datensparsam zugänglich zu machen. Das Projekt soll ein Bewusstsein dafür schaffen, dass wir mündiger und sicherer im Umgang mit unseren Daten werden können und uns bei diesem Vorhaben unterstützen.

Um Datenschutzeinstellungen vorzunehmen, oder Auskunft über gespeicherte personenbezogene Daten zu erlangen, soll *GuideLight* eine zentrale Anlaufstelle bereitstellen. Von dieser aus sollten über einen verständlichen, und für verschiedene Behörden und Unternehmen einheitlich gestalteten Generator, Anschreiben aus Textbausteinen und Adressdaten sowie Angaben der Nutzer*Innen erstellt werden. Das Ziel von *GuideLight* ist es, den *Dark Pattern* entgegenzuwirken und die Hürden für den souveränen Umgang mit persönlichen Daten möglichst niedrig zu setzen.

Meilenstein war vor allem die juristische, inhaltliche Recherchephase, gefolgt von der Planung der Softwarearchitektur, dem Gestalten der grafischen Oberfläche, Nutzer*Inneninterviews und der Umsetzungs- und Testing-Phase.

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

Wer ist die Zielgruppe für Deine Lösung? Wie profitiert sie von den Ergebnissen Deines Projekts? Welche Bezüge gibt es zu den Themenfeldern und Zielen des Software Sprints?

Im Sinne des Mottos des *Prototypefunds*, „Power to the Users“, versucht *GuideLight* ein Werkzeug für Verbarucher*Innen zu sein, um mündig und souverän mit den eigenen Daten umzugehen. Durch quelloffene Software ist *GuideLight* in seiner Funktion transparent und nachvollziehbar. Der Service wird ohne Werbung und Kosten für Nutzer*Innen betrieben.

Die primäre Zielgruppe von *GuideLight* sind Endverbraucher*Innen im deutschsprachigen Raum, die bewusst mit ihren personenbezogenen Daten umgehen wollen, bzw. sich über einen solchen Umgang informieren wollen. Über die Web-App können sie bei über 2000 Unternehmen und Behörden einfacher ihre Rechte rund um personenbezogene Daten in Anspruch nehmen. Dabei werden keinerlei Daten über sie gespeichert. Die Erstellung der Anschreiben erfolgt lokal.

Als erweiterte Zielgruppe sind Entwickler*Innen. Von diesen kann *GuideLight* und die offenen *Repositories* als Vorlage genutzt werden, eigene Ideen und Projekte umzusetzen. Dabei scheint vor allem die Entwicklung der Software unter dem Aspekt der Datensparsamkeit interessant. Die Software verzichtet im Frontend weitestgehend auf die Verwendung von Bibliotheken und anderen externen und potentiell unsicheren Quellen, und das Backend liefert die Anwendung einmal komplett an die Nutzer*Innen aus und lässt die Anwendung dann autonom und lokal laufen.

Außerdem können die Struktur, Ausgestaltung und die Inhalte der Datenbank mit Unternehmens- und Behördendaten für Dritte von Interesse sein.

Ausführliche Darstellung der Ergebnisse

Welche konkreten Ergebnisse hast Du erzielt? Konnten alle Meilensteine erreicht werden? Welche zusätzlichen Erkenntnisse hast Du aus der Projektarbeit gewonnen, auch im Hinblick auf die Begleitung durch die Open Knowledge Foundation?

Zum Ende der Förderphase hat das Projekt weitgehend alle Ziele erreicht, die im Antrag beschrieben wurden. In einigen Teilbereichen ist *GuideLight* im Funktionsumfang sogar umfangreicher geworden (die Anzahl an möglichen Anschreiben) als ursprünglich geplant, wohingegen ein geplantes Feature (Wiki um Datenbank zu pflegen) nicht realisiert werden konnte.

Mit der Abschlusspräsentation „Demo Day“ am 02. März 2019 wurde die Website *guide-light.de* online gestellt. Auf ihr sind in mehreren Blogbeiträgen Erklärungen zur Datenschutzgrundverordnung und nützliche Links zu Websites, Videos und Podcasts zum Thema vorhanden.

Über eine Suchfunktion können Nutzer*Innen aus einer Datenbank zu circa 2000 Unternehmen und Behörden jene herausuchen, bei denen sie eine Anfrage zu personenbezogene Daten stellen wollen, oder andere Anliegen zu personenbezogenen Daten haben. Die Datenbank fasst Unternehmen und Behörden in mehrere Kategorien zusammen: 20 Adresshändler, 21 Auskunftsteien, 461 Banken, 8 Ministerien, 14 Sicherheitsbehörde, 12 Landeskriminalämter, 6 Energieversorger, 47 Inkassounternehmen, 10 Kundenkarten Unternehmen, 1368 Meldebehörden, 9 Mieterauskunftsteien, 16 Telekommunikationsprovider, 9 Social Media Unternehmen, 8 Technologieunternehmen, 1 Verlag, 58 Versicherer und 2 Verwertungsgesellschaften und 5 weiteren Unternehmen aus sonstigen Branchen.

Durch eine Suchmaske können Nutzer*Innen nach Eigennamen und Kategorie einzelne Entitäten finden. Wählt man eine aus, wird aus den vorhandenen Daten eine Übersichtseite zum Unternehmen bzw. zur Behörde generiert. Diese umfasst einen einleitenden Text, einen Link zu jeweiligen Webpräsenz der Entität und einen zur Datenschutzerklärung. In der unteren Hälfte der Übersichtsseite zum Unternehmen, oder der Behörde werden nun die zur Verfügung stehenden Aktionen angezeigt. Nutzer*Innen können grundsätzlich zwischen zehn Aktionen auswählen. Diese beziehen sich immer auf bestimmte Artikel in der DSGVO und starten jeweils einen eigenen Anschreibengenerator.

1. Artikel 15: Auskunftsrecht der betroffenen Person
2. Artikel 16: Recht auf Berichtigung
3. Artikel 17: Recht auf Löschung („Recht auf Vergessenwerden“)
4. Artikel 18: Recht auf Einschränkung der Verarbeitung
5. Artikel 20: Recht auf Datenübertragbarkeit
6. Artikel 21: Widerspruchsrecht
7. Artikel 7: Einwilligung Widerruf
8. Artikel 21 Absatz 2: Widerspruchsrecht Direktwerbung
9. Artikel 22: Standpunkt zu automatisierte Entscheidungen im Einzelfall einschließlich Profiling
10. Artikel 12 Absatz 3: Modalitäten für die Ausübung der Rechte (Mahnung)

Je nach gewähltem Unternehmen bzw. Behörde sind nur bestimmte Aktionen verfügbar. So ist ein Anschreiben mit der Aufforderung zur Datenübertragung (siehe 5.) von bspw. Spotify zu Apple Music eine vorstellbare Aktion, jedoch von einem Ministerium zur anderen nicht möglich. Auch der Anschreibengenerator zum Widerspruch zur Direktwerbung (siehe 8.) wird bspw. bei Sicherheitsbehörden nicht angeboten. Die ursprüngliche Idee zum Projekt sah nur die erste Aktion, nämlich das Erzeugen von Auskunftersuchen nach Artikel 15 (siehe 1.) vor. Eine Erweiterung des Anschreibenportfolios erschien aber vor allem nach Interviews mit Nutzer*Innen als sehr wünschenswert und auch ohne zu großen Aufwand umsetzbar.

Im Generator selbst beantworten Nutzer*Innen in der Regel zwischen vier und fünf Fragen. Je nach Anliegen und gewählter Aktion werden den Nutzer*Innen dabei verschiedene Fragen gestellt:

*Welche Datensätze sollen korrigiert, geändert, oder gelöscht werden? Von welchem zu welchem Anbieter sollen Daten übertragen werden? Wie kann sich die Nutzer*In der Institution gegenüber eindeutig identifizieren? Soll das erzeugte Anschreiben via E-Mail direkt verschickt werden, oder soll es als PDF zum Download bereitgestellt werden und mit Adressdaten versehen werden, um es als Brief zu verschicken? Soll ein Mensch automatisierten Entscheidungen prüfen? ...*

Je nach ausgewähltem Adressat werden spezifische Fragen übersprungen bzw. Auswahlmöglichkeiten erweitert oder beschränkt. Verlangt eine Behörde eine Ausweiskopie als Identitätsnachweis, so wird Nutzer*Innen im Generator die Möglichkeit gegeben eine solche an das Anschreiben anzuheften. Diese Funktion wird aber beim Generator für Anschreiben an bspw. Social Media Unternehmen, oder Internetprovider bewusst nicht angeboten, da hier in der Regel die Nennung des Nutzernamens bzw. einer Rechnungsnummer völlig ausreichend ist. Auch werden je nach Art der Entität und Anliegen Hinweistexte angepasst. Diese Hinweis- und Hilfetexte führen die Nutzer*Innen durch den Generator. So wird bspw. beim Erstellen eines Auskunftersuchens an Google zusätzlich zum Anschreibengenerator ein direkter Link angeboten, um solche Daten direkt beim Unternehmen herunterzuladen.

Ist das Anschreiben erstellt, kann es in den meisten Fällen direkt an die jeweiligen Datenschutzbeauftragten der Unternehmen bzw. Behörden geschickt werden. Nutzer*Innen bekommen aber auch die Möglichkeit, die erzeugten Briefe als PDF zu speichern und per Post zu verschicken; dies wird vor allem dort angeboten, wo E-Mail Adressen nicht vorhanden sind, oder nicht der bevorzugte Weg der Kommunikation sind.

Im Verlauf der Projektarbeit und durch Interviews mit Nutzer*Innen in der Recherchephase wurde deutlich, dass der eigentliche Wert des hier beschriebenen Systems in der Datenbank mit Unternehmens- und Behördendaten liegt. Primäres Ziel der Arbeit war damit auch, diese strukturiert und nachhaltig aufzubauen und mit aktuellen Informationen und Links zu füllen. Vergleichbare Angebote im Netz scheinen hier oftmals veraltete Datensätze zu nutzen und sind damit unbrauchbar. Für Verbraucher*Innen existieren bereits eine Anzahl ähnlicher Angebote, die über die Datenschutzgrundverordnung informieren und allgemeine Vorlagen für Anschreiben bereitstellen. In Interviews konnte jedoch festgestellt werden, dass die Hürde solche Anschreiben zu nutzen oft relativ hoch ist. Nutzer*Innen sind hierbei verunsichert und haben zahlreiche individuelle Fragen beim Ausfüllen der Vorlagen. So ist es ein weiter Weg von einer allgemeinen Vorlage zum Auskunftersuchen nach Artikel 15 zu einem Anschreiben mit korrekter Anschrift und den nötigen Informationen an bspw. PayPal (Europe) S.à.r.l. et Cie, S.C.A.. Mit zahlreichen Beispielen, erklärenden Texten und der gezielten Einschränkung bzw. Erweiterungen von Optionen und Eingabemöglichkeiten senkt *GuideLight* hier für Verbraucher*Innen die Hürde, ihre Anliegen zu formulieren und einzufordern.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Welcher Nutzen ergibt sich für die Zielgruppe aus den Ergebnissen Deines Projekts? Welche weiter-gehenden Effekte ergeben sich aus der Open-Source-Stellung der Ergebnisse? Gibt es Ideen für die Weiterentwicklung Deiner Lösung und Pläne zu deren Umsetzung? Hat die Arbeit in dem Projekt Dich in Deiner persönlichen, fachlichen Weiterentwicklung unterstützt?

GuideLight richtet sich an zwei Zielgruppen (siehe auch die Antwort zu Frage 2).

Die Website richtet sich an Endverbraucher*Innen, die Anschreiben zu datenschutzrechtlichen Themen an über 2000 Unternehmen und Behörden generieren und mit wenigen Klicks zu verschicken können.

Für Entwickler von Open-Source-Software bietet *GuideLight* eine Infrastruktur mit zahlreichen Daten zu Unternehmen und Behörden.

In Zukunft wird angestrebt, den Anschreibengenerator in bestehende Projekte/Websites zum Thema Datenschutz und DSGVO bzw. BDSG zu integrieren. So können Menschen sich nicht nur über die Themen informieren, sondern auch gleichzeitig aktiv werden.

Im Projekt konnte ich mich persönlich enorm weiterentwickeln. Zum einen als Entwickler im Bereich der Web-Front und -Backend und im Umgang mit Datenbanken, zum anderen im Bereich des Projekt- und Produkt-Managements. Im Bereich Projektmanagement und inhaltliche Ausrichtung des Projekts halfen die Coachings und Betreuung durch den *Prototypefund* sehr. Lenkten sie den Blick doch immer wieder von der reinen technischen Arbeit an Software und Systemarchitektur hin zu den Nutzer*Innen und dem angestrebten Endprodukt.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Gab es Arbeiten bzw. Lösungsansätze, die nicht weiter verfolgt wurden? Was waren die Hintergründe, und wie bist Du alternativ vorgegangen?

In einer frühen Skizze für das Projekt wurde angestrebt, dass Nutzer*Innen, sofern die Möglichkeit bestünde, Einstellungen zum Datenschutz direkt auf den Websites der Unternehmen vornehmen können sollten. *GuideLight* wäre dann als eine Art Browserplugin in der Lage gewesen, durch Overlays die Navigation auf den Websites zu unterstützen, so wie inhaltliche und rechtliche Hinweise zu Einstellungen zu geben. Diese Vorstellung stellte sich nach kurzer Testphase als, wie erwartet, zu ambitioniert heraus. Vor allem die im Antrag für dieses Projekt schon beschriebene Heterogenität der Einstellungsmöglichkeiten bzw. Kontaktformulare auf den Websites von Unternehmen und Behörden erzeugt hier eine hohe Komplexitätshürde. Durch den Fokus auf das Erzeugen von Anschreiben, die via E-Mail oder Post verschickt werden können, ist es möglich diese Komplexität zu abstrahieren und ein einheitliches „Interface“ zu schaffen - den Anschreibengenerator.

Auch war es das Ziel mittels eines Wikis die Datenbank zu öffnen und von einer Community aus Interessierten pflegen zu lassen. Dieses Anliegen besteht weiterhin, ist jedoch aufgrund der zeitlichen Beschränkung des Projekts und der dadurch noch nicht ausreichend erfolgten Öffentlichkeitsarbeit nicht realisiert.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Wo können sich Interessenten detailliert über Deine Projektergebnisse informieren (z.B. Webseite, GitHub, Veröffentlichungen)?

Link zur Webseite

<https://www.guide-light.de>

Link zu den Repositories:

<https://gitlab.com/kaffai/GuideLight-website>

<https://gitlab.com/kaffai/GuideLight-data>

<https://gitlab.com/kaffai/GuideLight-design>

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Gab es im Projektverlauf Ereignisse, die eine Anpassung der Planung erforderlich machten – z.B. Mehr- oder Minderaufwand bei der Bearbeitung von Teilaufgaben?

Durch die Verschiebung des Projektstarts von September auf Oktober musste die Zeitplanung angepasst werden. Dies führte allerdings zu keinerlei projektinternen Problemen.

Innerhalb des Projekts musste die inhaltliche und rechtliche Recherchephase ausführlicher erfolgen, wodurch weniger Zeit für die eigentliche Programmierarbeit blieb. Auch dies führte letztendlich zu keinerlei Abstrichen in der Projektumsetzung im Bezug auf Features und Umfang.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Gab es Entwicklungen anderer Personen oder Institutionen, die Einfluss auf Deine Arbeiten und die Zielsetzung hatten? Wenn ja, worin bestand dieser und wie bist Du damit umgegangen?

Die Website <https://youdata.eu> wurde zu Beginn der Förderphase online gestellt, die ein sehr ähnliches Konzept wie *GuideLight* verfolgt. Sie beschränkt sich auf das Erzeugen von E-Mails (*GuideLight* weicht bei nicht vorhandenen E-Mail Adressen auf den Postweg aus) zu Auskunftersuchen (*GuideLight* erweitert hier die Möglichkeit auf mehrere Arten von Anschreiben). Durch den Kontakt mit dem Entwickler konnte *GuideLight* direkt zum Projektstart mit den Datensätzen von youdata.eu arbeiten.

Richtlinie zum „Software-Sprint“ StalkerBuster - Identifizierung von Stalking-Software

Schlussbericht

Zuwendungsempfänger:

StalkerBuster - Petra Hagemann und Ulrich Fouquet GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S26 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren.

Kurze Darstellung der Aufgabenstellung und Motivation

Digitales Stalking verbreitet sich immer mehr. Dabei werden z.B. Smartphones von Partner.innen oder anderen vertrauten Personen präpariert, ausgelesen oder anderweitig missbraucht, um diese sozial zu kontrollieren. Taucht ein solcher Verdacht auf, ist er technisch meist nur aufwändig zu überprüfen und zu belegen. Dies betrifft nicht nur die Betroffenen selbst, sondern auch Berater.innen in spezialisierten Frauenzentren etc. Der StalkerBuster soll rasche und begründbare Einschätzungen ermöglichen, aufgrund derer Betroffene und Berater.innen besser informierte Entscheidungen in einer sozial heiklen Situation treffen können. So hilft der StalkerBuster dabei, Kontrolle und Vertrauen in die eigenen Kommunikationsgeräte zurückzugewinnen.

Der StalkerBuster ist eine auf einem RaspberryPi laufende Software, mit der sich das auf Stalkingprogramme zu untersuchende Gerät verbinden muss. Anders als viele andere Vorgehensweisen oder Programme, wie z.B. Virenscanner, untersucht der StalkerBuster nicht das Gerät selbst auf Stalkingsoftware, sondern protokolliert den ein- und ausgehenden Internetverkehr. Auf diese Weise können nicht nur spezielle Stalkingapps, sondern auch das Verhalten anderer Programme auf dem Gerät sichtbar gemacht werden.

Das Protokoll des Netzverkehrs wird auf eine Weise aufbereitet, die eine rasche Einordnung der Situation erlaubt und Material für tiefergehende Untersuchungen sichert. Filterung und Analyse des Netzwerktraffics geht dabei auf öffentlich einsehbare Regeln zurück. Auch wird das Verhalten bekannter und verbreiteter Stalkingapps und -technologien analysiert, so dass diese mit großer Wahrscheinlichkeit erkannt werden können. Liegen Hinweise auf Stalking vor, kann gezielt auf dem Gerät nach der entsprechenden Stalking-Technologie gesucht werden.

Als Meilensteine für die Entwicklung des StalkerBusters wurden festgelegt:

- Festlegung Anfangs-Requirement und zu verwendende Libraries
- Einrichtung der Filter- und Alarm-Datenbank - read-only
- Lauffähigkeit Basis-Software
- (optional) Erstellung Reproduzierbarer Build für RaspberryPi3-Image
- Feedback aus konsolidierter Testaktion der Testgruppe einarbeiten
- Webinterface

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

Die Zielgruppe des StalkerBusters sind Berater:innen in Frauenzentren etc., die bei Verdacht von Cyberstalking entweder nicht das technische Know-How oder die Zeit haben, die Geräte ihrer Klient:innen eingehend zu untersuchen. Der StalkerBuster unterstützt sie hierbei und gibt schnelle Hinweise, ob tatsächlich Geräte ausspioniert werden. Falls dies der Fall ist können die vom StalkerBuster gesicherten Informationen weitergehend untersucht werden. Gleichzeitig können die Berater:innen den Betroffenen gezielte psychologische Unterstützung anbieten.

Der StalkerBuster kann ganz im Sinne des Themas der 4. Runde des Software-sprints "Power to the User" mit Hilfe der Berater:innen den Klient:innen der Beratungszentren die Kontrolle über ihre Kommunikationsgeräte zurückgeben. Die Klient:innen können nach der Analyse ihres Netzwerkverkehrs nicht nur ungewollte Stalkerware entfernen, sondern auch selbstbestimmt bei anderen Apps entscheiden, ob eine festgestellte Kommunikation nach Außen gewünscht ist und diese bei Bedarf unterbinden. So kann Vertrauen in die eigenen Kommunikationsgeräte zurückgewonnen werden.

Ausführliche Darstellung der Ergebnisse

Da der StalkerBuster leider nicht auf einem bereits vorhandenen Tool aufsetzen konnte, sondern komplett neu konzipiert und erstellt werden musste, hat die Entwicklung einzelner Module mehr Zeit als erwartet in Anspruch genommen. Obwohl Prototypen der einzelnen Module und entsprechende Tests zu diesen fertig gestellt sind, konnten sie noch nicht zusammengeführt werden. Somit konnte noch keine lauffähige Version des StalkerBusters mit Berater:innen getestet werden. Ein einsatzfähiger Prototyp soll jedoch bis zum Sommer 2019 fertiggestellt und im Anschluss in Frauenberatungszentren getestet werden.

Da während der Förderphase das Thema Cyberstalking auf verschiedenen Veranstaltungen diskutiert und der StalkerBuster vorgestellt wurde, konnte eine Online-Gruppe von Interessierten gegründet werden, die sich zum Thema aus-

tauscht und mit der gemeinsam Lösungen gegen Cyberstalking gefunden werden sollen. Auch haben verschiedene Softwareentwickler Interesse angekündigt, an einer möglichen Weiterentwicklung mitzuarbeiten.

Die von der Open Knowledge Foundation unterstützte Vernetzung und wöchentlichen Onlinetreffen waren sehr hilfreich zur Selbstmotivation, aber auch, um bestimmte Probleme mit den anderen Geförderten zu besprechen und gemeinsam Lösungen zu finden. Insbesondere bei der Erstellung des Webinterfaces des StalkerBusters konnte das UI/UX-Mentoring gezielte Hilfestellungen bei der Verbesserung des Layouts geben.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Für die Zielgruppe des StalkerBusters, den Betreuer:innen von Frauenzentren etc., wird ein leicht bedienbares Gerät bereitgestellt, das sie bei ihrer Arbeit unterstützt. Da meist nicht die Zeit oder Expertise vorhanden ist, um eingehende Geräte-Analysen bei Cyberstalkingverdacht vorzunehmen, kann der StalkerBuster eine erste Übersicht hierzu geben. Die Ergebnisse können mit den Klient:innen besprochen werden und bei Stalkingverdacht genauer untersucht werden.

Da der StalkerBuster als Open Source Software entwickelt wurde, können der komplette Code oder einzelne Teile heruntergeladen, genutzt oder auch weiterentwickelt werden. Insbesondere der Netzwerkscanner hat Interesse bei verschiedenen anderen Open-Source-Projekten geweckt und es wird geprüft, ob dieser so oder in angepasster Form eingesetzt werden kann. Weiterhin ist geplant, die Software auch für breitere Einsatzmöglichkeiten zu erweitern, um Netzwerkverkehr von Mobil- oder IoT-Geräten für die Nutzer:in sichtbar zu machen. Ziel ist es, der Nutzer:in die Möglichkeit zu geben, einen Überblick über die Netzwerkkommunikation ihrer Geräte zu geben und nicht erwünschte Kommunikation zu unterbinden. So kann die Nutzer:in Kontrolle über ihre Geräte zurückgewinnen.

Durch das Projekt konnte das Team sich fachlich im Bereich der Netzwerkanalyse spezialisieren. Die dafür notwendigen Erfahrungen, das teils unvermeidliche Trial-and-Error, wäre ohne die Unterstützung des Software-Sprints schlicht nicht möglich gewesen.

Für Petra Hagemann konnte das Projekt darüber hinaus einen wesentlichen Beitrag zum Schritt in die Selbständigkeit leisten.

Das Team konnte im fachlichen Bereich des Cyberstalkings Gleichgesinnte und Interessierte kennenlernen, die das Projekt auch mit ihrer teils erheblichen Expertise in Zukunft begleiten und helfen werden, die Weiterentwicklung zu verstetigen.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Einen wichtigen Punkt, den wir vorläufig zurückgestellt haben, ist das Mitschneiden und Aufbrechen verschlüsselter Kommunikation. Dies hat sich als technisch immer schwieriger umsetzbar erwiesen, da neuere Versionen von Android (ab Version 7) genauere Prüfungen von Zertifikaten verlangen, respektive die Installation eigener Zertifikate erheblich erschweren. Durch die Einführung von TLS 1.3 wird unser Ansatz noch einmal deutlich erschwert.

Unter dem Gesichtspunkt der Datensicherheit haben wir uns daher entschlossen, bei verschlüsselten Verbindungen auf ein Aufbrechen der Kommunikation zu verzichten. Wir möchten kein Gerät anbieten, das Laien das einfache Aufbrechen verschlüsselter Kommunikation ermöglicht. TLS 1.3 und andere Techniken zum Schutz vertraulicher Kommunikation sollten aus unserer Sicht unterstützt und nicht umgangen werden.

Im Ergebnis bedeutet dies, dass wir uns bei der Analyse der Datenströme mit Metadaten und unverschlüsselten Daten begnügen.

Für ältere Geräte (Android Version < 7) werden wir ggf. noch eine Möglichkeit zum Mitschneiden verschlüsselter Kommunikation anbieten, das wird aber nicht mehr Teil unseres grundsätzlichen Lösungsansatzes sein.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Eine Webseite zum Thema Stalking ist unter <https://stalkerbuster.de> verfügbar. Hier sind umfangreiche Literaturhinweise zum Thema Stalkerware, Tipps zum Absichern der eigenen Kommunikationsgeräte sowie Links zu Beratungs- und Betreuungsstellen zu finden, die Betroffene unterstützen können.

Der komplette Open Source Code ist unter <https://github.com/stalkerbuster> veröffentlicht.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Aufgrund des Mehraufwands für einzelne Komponenten konnte im Förderungszeitraum noch keine integrierte Vollversion mit allen geplanten Komponenten erstellt werden. Die Fertigstellung soll bis Sommer 2019 erfolgen.

Eine Änderung der Planung erfolgte vor allem hinsichtlich zweier Umstände: Erstens erwies sich das Aufbrechen der Verschlüsselung als technisch und sozial schwieriger/fragwürdiger, als wir uns das im Vorfeld vorgestellt hatten.

Zweitens machte die Erstellung eines einfachen Interfaces für den Stalkerbuster mehr Arbeit als geplant, da sich die basalen Netzwerktools (vor allem für WLAN-Management) auf verschiedenen Betriebssystemen viel grundsätzlicher unterscheiden, als wir das vorhergesehen hatten. Selbst im eingeschränkten Kreis der Debian-basierten Systeme (auf den wir uns anfänglich einschränkend kapriziert hatten) gibt es mindestens drei weit verbreitete und gleichzeitig sehr unterschiedliche Toolsets für das Management kabelloser Netzwerk-Verbindungen.

Da unser Projekt aber wesentlich davon lebt, auch für unbedarfte Anwender.innen handhabbar zu sein, war die Unterstützung wenigstens dieser wenigen Toolsets unabdingbar - bedeutete aber gleichzeitig auch dreifache Arbeit. Diese Ausweitung der Arbeit war womöglich ein Produkt schlechter Planung: Wir hatten mit solch einer Komplikation an so einer basalen Stelle nicht gerechnet.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Das Thema Cyberstalking findet erfreulicherweise mehr Beachtung in den Medien. So berichtet das Online-Magazine *Motherboard* von *Vice* in seiner thematischen Serie [When Spies Come Home](#) regelmäßig über Stalking und Spysoftware.

Konkrete Hilfestellungen zum Auffinden von Spyapps wurden im Computermagazin [c't](#), von der vom BMJV geförderten Informationsseite [mobilsicher.de](#) oder auch vom [Bundesverband der Frauenberatungsstellen und Frauennotrufe in Deutschland](#) veröffentlicht. Seit Kurzem hat auch das auf Sicherheitssoftware spezialisierte russische Unternehmen Kaspersky Lab eine deutliche [Kennzeichnung von Stalkerware](#) in seine Antivirenprogramme aufgenommen.

Anders als der StalkerBuster untersuchen all diese Ansätze das Mobilgerät selbst. Dies bedeutet, dass nur bereits bekannte und genauer untersuchte Stalkerware gefunden wird, dies allerdings zuverlässiger als der StalkerBuster dies kann. Der breitere Ansatz des StalkerBusters, in dem sämtlicher Netzwerkverkehr mitgesniffelt wird, verspricht ein umfassenderes Bild darüber, welche Programme welche Daten wohin senden. So können auch weniger bekannte Stalkingsapps aufgespürt werden und darüberhinaus eine Übersicht des Internetverkehrs sämtlicher auf dem untersuchten Gerät installierten Programme gegeben werden.

Richtlinie zum „Software-Sprint“

Freigeist

Schlussbericht

Zuwendungsempfänger:

Thieke, Böckhoff, Rudolph und Reissner GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S27 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Was war Deine Motivation? Welches Problem wolltest Du mit Deinem Projekt lösen? Wie war die geplante Vorgehensweise zur Problemlösung (auch Angabe der wichtigsten Meilensteine)?

Die Inklusion von Menschen mit kognitiven Schwächen in den Arbeitsalltag ist schwierig. Häufig werden sehr triviale Arbeitsaufgaben für diese Menschen angeboten, da der Betreuungsaufwand für komplexere Tätigkeiten groß ist. Mit dem Projekt „Freigeist“ soll eine größere Vielfalt an Arbeitsplätzen für diese Nutzergruppe ermöglicht werden. Es sollen Module und Anregungen entwickelt werden, mit welchen neue und moderne Arbeitsplätze mit Hilfe von IT für Menschen mit kognitiven Schwächen entstehen können. Hierzu sollen kleine, digitale Orthesen entwickelt werden, welche Menschen mit kognitiven Schwächen bei der Erledigung von einfachen Arbeitsaufgaben (z.B. Montagetätigkeiten oder Kommissionierung) unterstützen.

Im Sinne eines interaktiven Tutorials sollen die Nutzer dieses Systems durch einzelne Arbeitsschritte begleitet werden. Verschiedene (IoT-)Sensoren erkennen wesentliche Arbeitstätigkeiten der Nutzer und lösen entsprechend die nächste Arbeitsanweisung aus. Hierdurch können die Nutzer trotz ihrer kognitiven Schwächen mehrstufige Arbeitstätigkeiten selbstständig erledigen.

Für das geförderte Projekt plante der Antragsteller, modular einsetzbare Eingabe- und Ausgabegeräte zu programmieren, welche je nach Bedarf zur Begleitung unterschiedlicher Arbeitsaufgaben eingesetzt werden können. Als Zentralrechner sollte zur Datenverarbeitung ein Raspberry Pi verwendet werden. Das Programm für die Datenverarbeitung sollte dabei hauptsächlich in der Programmiersprache Python geschrieben werden. Das User-Interface zur Konfiguration des Programms soll in einem Web-Browser (HTML5, CSS, JavaScript) dargestellt werden.

Auf der Seite der Dateneingabe sollen zehn bis fünfzehn Sensoren und Schnittstellen für die digitalen Prothesen eingebunden und getestet werden. Auf der Seite der Datenausgabe sollten unter anderem LEDs, Smart-Watches, Video-Projektionen, Motorensteuerungen als modulare Elemente zur Unterstützung der Menschen mit kognitiven Schwächen am Arbeitsplatz getestet werden. Die verwendete Hardware sollte, wie auch die Software, als Open-Source-Komponenten entwickelt und der Allgemeinheit in Form von Bauplänen online zugänglich gemacht werden.

Das Projektteam plante für den Förderzeitraum eine Vernetzung in Deutschland mit den großen Institutionen der Wohlfahrt (z.B. Caritas, Lebenshilfe, Diakonie, AWO, etc.) und wollte so gemeinsam die Arbeitsergebnisse testen und in die Praxis führen.

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

Wer ist die Zielgruppe für Deine Lösung? Wie profitiert sie von den Ergebnissen Deines Projekts? Welche Bezüge gibt es zu den Themenfeldern und Zielen des Software Sprints?

Die Nutzer der Vorhabensergebnisse sind Menschen mit kognitiven Schwächen. Die Zielgruppen des Projektes beinhalten auf B2B-Ebene auch die vielen sozialen Träger und Einrichtungen der Wohlfahrt, welche Menschen mit Behinderungen in Deutschland beschäftigen.

Durch das Projekt und die damit verbundene Öffentlichkeitsarbeit wird der Allgemeinheit gedient. Es haben sich aus dem Projekt bereits heute Impulse für neue, barrierefreie Arbeitsplätze auch bei Dritten (Institutionen und auch Privatpersonen) ergeben. Dies lässt sich aus der positiven Rückmeldung des Fachpublikums und Anfragen zur Zusammenarbeit bei unseren Vorträgen schließen.

Ausführliche Darstellung der Ergebnisse

Welche konkreten Ergebnisse hast Du erzielt? Konnten alle Meilensteine erreicht werden? Welche zusätzlichen Erkenntnisse hast Du aus der Projektarbeit gewonnen, auch im Hinblick auf die Begleitung durch die Open Knowledge Foundation?

Hauptergebnis des geförderten Projekts ist eine Muster-Station, an welcher Menschen mit kognitiven Schwächen bei der Verpackung von Geschenken unterstützt werden. Hierfür wurden 8 modular einsetzbare Sensoren und Ausgabegeräte, sowie eine dafür spezialisierte Software entwickelt. Bei den Sensoren handelt es sich um

- 3 Distanzsensoren zur Vermessung der Geschenkverpackung in den Achsen X, Y und Z,
- um einen Lichtsensor mit welchem die korrekte Position des Geschenks verifiziert werden kann,
- um einen Lichtsensor mit welchem die Position des Messers für das Zuschneiden des Geschenkpapiers verifiziert werden kann,
- um ein Encoder mit welchem die Länge des Geschenkpapiers beim Abwickeln gemessen werden kann,

- um eine Wägezelle auf welcher ein Tesa-Roller positioniert wurde und beim Abriss des Tesastreifens der entsprechende Arbeitsschritt durch den kurzen Gewichtsimpuls bestätigt werden kann
- um eine LED-Einheit mit welcher einzelne Arbeitsschritte durch Lichtsignale bestätigt werden können.

Die entstandene Software verbindet die Ein- und Ausgabegeräte miteinander und unterstützt den Nutzer entsprechend der Programmierung bei der Reihenfolge und Ausführung der Arbeitstätigkeit.

Mit dieser Station können wir nun prototypisch darstellen, wie digitale Assistenz in Werkstätten bei Montage-, Kommissionierungs- oder Verpackungstätigkeiten assistieren können. Damit wurde das Hauptziel des Projekts erreicht.

Neben dem Hauptergebnis wurden folgende Nebenergebnisse erreicht:

- Wir haben im Laufe des Projekts erkannt, dass die Nutzer des zu entwickelnden Systems selbstständig wesentliche Parameter der Software und der Sensoren einstellen können müssen, damit die Nutzung im Alltag erfolgreich ist. Dafür ist eine entsprechende Benutzeroberfläche (Back-End) notwendig. Die Benutzeroberfläche wurde bei der Muster-Station in Form einer Konfigurations-Webseite für die Geschenkverpackung umgesetzt / gezeigt.
- Zur einfachen Übertragung der Software auf unterschiedlichen Computer ist es hilfreich die komplette Library in einem Docker-Container zu bündeln. Dies kann in Einzelfällen jedoch zu Einbußen in der Leistungsfähigkeit der Software führen.
- Wir konnten das Projekt auf mehreren Veranstaltungen und Einrichtungen der Wohlfahrt präsentiert. Unter anderem beim Berliner Johannesstift, den Bonner Werkstätten, der Netzwerkveranstaltung „CareInvest Connect“, dem Caritaskongress und der Werkstättenmesse in Nürnberg. Insbesondere auf dem Caritaskongress und der Werkstättenmesse traf das Projekt auf sehr positive Resonanz, aus welcher eine Fortführung des Projekt und zukünftige Implementierung in der Praxis sehr wahrscheinlich wird. Für weitere Vorträge zu unserem Projekt sind wir bereits eingeladen.
- Die Begleitung durch die Open-Knowledge-Foundation war für die Projektplanung und das Zeitmanagement sehr hilfreich.
- Leider konnte der Meilenstein „Überführung in die Praxis“ bisher nicht erreicht werden, da sich herausstellte, dass die angesprochenen Werkstätten zwar grundsätzlich an dem System interessiert waren, aber keine konkreten Anwendungsmöglichkeiten in diesem kurzen Projektzeitraum zu finden waren. Es zeigte sich, dass das unter anderem daran lag, dass die Planungszeiträume der Werkstätten in der Regel länger sind und die Mitarbeiter sich zunächst mit dem Thema „digitale Assistenzsysteme“ fachlich tiefer befassen müssen.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Welcher Nutzen ergibt sich für die Zielgruppe aus den Ergebnissen Deines Projekts? Welche weitergehenden Effekte ergeben sich aus der Open-Source-Stellung der Ergebnisse? Gibt es Ideen für die Weiterentwicklung Deiner Lösung und Pläne zu deren Umsetzung?

Hat die Arbeit in dem Projekt Dich in Deiner persönlichen, fachlichen Weiterentwicklung unterstützt?

Aus dem Projekt haben sich konkret nutzbare Sensormodule und eine Basissoftware ergeben, welche sich für die Zielgruppe wie erhofft individuell anpassen lassen. Neben der ursprünglich angesprochenen Hauptzielgruppe der Werkstätten für Menschen mit Behinderungen und Arbeitgeber des ersten Arbeitsmarkts haben nun auch Privatpersonen mit uns Kontakt aufgenommen, welche an einzelnen Lösungen unseres Projekts interessiert sind. Durch die Open-Source-Stellung des Projekts können sich nun auch diese Personen selbstständig mit unserer Lösungen befassen. Konkret möchte beispielsweise eine erblindete Schneiderin mit uns besprechen, ob und wie unser Encoder-Sensor ihr dabei helfen könnte wieder selbstständig Näharbeiten zu erledigen. Auch diese Anfrage gibt Impulse zur Weiterentwicklung des Projekts.

Die Arbeit am Projekt hat uns viele Erkenntnisse über die Zusammenhänge in der Sozialwirtschaft gebracht und auch die aktuelle wie zukünftige Stellung digitaler Assistenzsysteme in der Wohlfahrt aufgezeigt. Dies ist für die Fortführung des Projekts hilfreich.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Gab es Arbeiten bzw. Lösungsansätze, die nicht weiter verfolgt wurden? Was waren die Hintergründe, und wie bist Du alternativ vorgegangen?

Die praktische Einsatz der Sensoren und Software in einer Werkstatt für Menschen mit Behinderungen ist bisher noch nicht erfolgt. Dies liegt unter anderem daran, dass die Werkstätten das Thema digitale Assistenten mangels Leuchtturmprojekte in vielen Fällen noch nicht für sich entdeckt haben und auch aufgrund fehlender Fachkompetenz in den meisten Fällen noch nicht entwickeln können. Aus diesem Grund haben wir uns bei der Entwicklung des Projekts darauf konzentriert, eine Muster-Station als Impulsgeber für Werkstätten zu entwickeln.

Aus Zeitgründen konnten wir die Anwendung des Docker-Containers bisher in der Praxis noch nicht testen.

Wir haben für unser Projekt auch einen Stromfluß-Sensor entwickelt und programmiert, welcher in der Muster-Station keine Anwendung fand, da sich hierfür keine Unterstützung für einen sinnvollen Arbeitsprozess ergeben hat.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Wo können sich Interessenten detailliert über Deine Projektergebnisse informieren (z.B. Webseite, GHitHub, Veröffentlichungen)?

www.github.com/freigeistlab

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Gab es im Projektverlauf Ereignisse, die eine Anpassung der Planung erforderlich machten – z.B. Mehr- oder Minderaufwand bei der Bearbeitung von Teilaufgaben?

Der Arbeits- und Kostenplan wurde eingehalten.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Gab es Entwicklungen anderer Personen oder Institutionen, die Einfluss auf Deine Arbeiten und die Zielsetzung hatten? Wenn ja, worin bestand dieser und wie bist Du damit umgegangen?

Das Vorgänger-Projekt „inklusive und interaktives Bistro“ hat in der Praxis neue Impulse zur Entwicklung des geförderten Projekts gegeben. Insbesondere die Nutzerfreundlichkeit der Software für das geförderte Projekt profitiert davon. Durch das Projekt sind Kontakte zu anderen Institutionen (u.a. Fraunhofer-Institut) entstanden, welche an ähnlichen Thematiken arbeiten, wodurch das Projekt in Zukunft profitieren wird.

Lightbeam

Schlussbericht

Zuwendungsempfänger: Princiya Marina Sequeira

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S28 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Was war Deine Motivation? Welches Problem wolltest Du mit Deinem Projekt lösen? Wie war die geplante Vorgehensweise zur Problemlösung (auch Angabe der wichtigsten Meilensteine)?

Ich hatte 2017 an Outreachy teilgenommen (ein dreimonatiges Praktikumsprogramm, um mehr Vielfalt in OpenSource zu bringen) und mein Projekt war Lightbeam. Während dieser Zeit habe ich mich intensiv mit dem Thema Online-Datenschutz beschäftigt. Dies war meine Hauptmotivation, mich für den PrototypeFund-Zuschuss zu bewerben und weiterhin an Lightbeam zu arbeiten. Durch dieses Programm wollte ich der Öffentlichkeit die Grundlagen des Online-Datenschutzes und des Web-Trackings mit Lightbeam vermitteln.

Der geplante Ansatz bestand aus verbesserten Visualisierungen für Lightbeam, die aus drei Visualisierungsmodi bestehen:

1. Gesteuertes Diagrammlayout erzwingen - Zeigt die Netzwerke und Hierarchien von Websites von Erst- und Drittanbietern
2. Uhrmetapher - Zeigt periodische Browserverlaufsdaten und die Beziehung zwischen den besuchten Websites und den verbundenen Dritten
3. Verfolgung von Trends und Mustern - Diese Ansicht bietet eine analytische Perspektive der Online-Verfolgung. Der Benutzer kann sehen, wie sich jede Verbindung eines Drittanbieters im Laufe der Zeit verhält. In diesem Modell wird der Browserverlauf durch die größeren Quadrate in der mittleren Reihe dargestellt, während die kleineren Quadrate darunter die Cookies dieser Site sind. Die Grafik oben visualisiert jeden Dritten und wie sein Tracking-Level im Laufe der Zeit wächst.

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

Wer ist die Zielgruppe für Deine Lösung? Wie profitiert sie von den Ergebnissen Deines Projekts? Welche Bezüge gibt es zu den Themenfeldern und Zielen des Software Sprints?

Lightbeam ist ein Versuch, dem breiten Internetpublikum die Funktionsweise von Web-Tracking zu erläutern. Dadurch erhalten die Benutzer die Möglichkeit, die Kontrolle über ihre Online-Privatsphäre zu übernehmen. Web-Tracking ist zwar nicht zu 100% böse, seine Funktionsweise wird jedoch kaum verstanden. Ihre persönlichen Daten werden ohne Ihr Wissen an Dritte weitergegeben, um neue Wege zu finden, um Sie zu überzeugen, Geld auszugeben.

Einzelne Benutzer können in Echtzeit sehen, mit wem sie sich verbinden, und Entscheidungen über ihren Online-Datenschutz treffen. Experten, Power-User und nachfragende Bürger können Lightbeam nutzen, um die Beziehungen der Tracker zu erforschen und in ihren eigenen Daten nach Mustern zu suchen.

Open Community Research-Plattform: Öffentlich verfügbare Lightbeam-Daten, die durch die willige Teilnahme von Lightbeam-Benutzern erhoben werden und Web-Tracking-Muster basierend auf der Geografie aufdecken können.

Ausführliche Darstellung der Ergebnisse

Welche konkreten Ergebnisse hast Du erzielt? Konnten alle Meilensteine erreicht werden? Welche zusätzlichen Erkenntnisse hast Du aus der Projektarbeit gewonnen, auch im Hinblick auf die Begleitung durch die Open Knowledge Foundation?

Die wichtigsten Meilensteine wurden erreicht, die im ersten Antragsverfahren angegeben wurden.

1. Recherchieren, um Trends und Muster zu verfolgen und zu verstehen
2. Visualisierung zur Verfolgung von Trends und Mustern
3. Studieren Sie Forschungsmaterial in Bezug auf Online-Datenschutz und -Überwachung, sprechen Sie auf Konferenzen und bei lokalen Treffen
4. Benutzertests, Feedback-Erfassung

Darüber hinaus konnte ich die folgenden Ergebnisse erzielen:

1. Sprechen Sie mit politischen Entscheidungsträgern, der französischen und niederländischen DPA (Datenschutzbehörde), um die GDPR besser zu verstehen und den Anwendungsbereich von Lightbeam zu erweitern
2. Arbeitete an ähnlichen Open-Source-Projekten, um herauszufinden, wie PII-Daten (persönliche identifizierbare Informationen) während einer Browsersitzung durchgesickert werden

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Welcher Nutzen ergibt sich für die Zielgruppe aus den Ergebnissen Deines Projekts? Welche weiter gehenden Effekte ergeben sich aus der Open-Source-Stellung der Ergebnisse? Gibt es Ideen für die Weiterentwicklung Deiner Lösung und Pläne zu deren Umsetzung? Hat die Arbeit in dem Projekt Dich in Deiner persönlichen, fachlichen Weiterentwicklung unterstützt?

Einzelne Benutzer können mehrere Lightbeam-Datensätze importieren, um die Beziehungen der Tracker zu untersuchen und nach Mustern in ihren eigenen Daten zu suchen. Experten, Power-User, anfragende Bürger und Forscher können den öffentlich verfügbaren Tracking-Datensatz nutzen, um auf geografischer Basis Web-Tracking-Muster zu erlernen.

Nächste Schritte:

- Gemeinsame Zusammenarbeit mit Forschern der Universität Tilburg, Niederlande, um die geografischen Verfolgungsmuster von Web-Trackern zu verstehen
- Gemeinsame Zusammenarbeit mit Mozilla Web Fellow zum Erstellen des GDPR-Konformitätswerkzeugs, mit dem Benutzer Website-Verstöße melden können

All diese Arbeit hat mir sehr geholfen, mein Wissen über Online-Datenschutz und Web-Tracking weiter zu verbessern.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Gab es Arbeiten bzw. Lösungsansätze, die nicht weiter verfolgt wurden? Was waren die Hintergründe, und wie bist Du alternativ vorgegangen?

Die ursprüngliche Idee bestand darin, die Leistung der vorhandenen Lightbeam-Visualisierung zu verbessern. Mir wurde jedoch bewusst, dass es sinnvoll wäre, die Finanzierungszeit für die Erarbeitung eines neuen Konzeptentwurfs zu nutzen, und beschloss, mich darauf zu konzentrieren, neuere Arten der Darstellung von Lightbeam-Daten zu prototypisieren.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Wo können sich Interessenten detailliert über Deine Projektergebnisse informieren (z.B. Webseite, GHitHub, Veröffentlichungen)?

- <https://github.com/mozilla/lightbeam-we>
- <https://github.com/princiya/lightbeam-import>
- <https://princiya777.wordpress.com/category/lightbeam/>
- <https://addons.mozilla.org/en-GB/firefox/addon/lightbeam/>
- Der öffentliche Datensatz zur Verfolgungsstudie basierend auf der Geografie wird Ende Mai 2019 verfügbar sein.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Gab es im Projektverlauf Ereignisse, die eine Anpassung der Planung erforderlich machten – z. B. Mehr- oder Minderaufwand bei der Bearbeitung von Teilaufgaben?

Die Konzeptionsphase ging etwas länger, weil ich zu viele Ideen hatte. Ich habe jedoch versucht, den Spielraum einzugrenzen, um mich auf das Ziel des Sprints zu konzentrieren.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Gab es Entwicklungen anderer Personen oder Institutionen, die Einfluss auf Deine Arbeiten und die Zielsetzung hatten? Wenn ja, worin bestand dieser und wie bist Du damit umgegangen?

Das Forschungsteam des [GlobalDataJustice-Projekts](#) verwendet Lightbeam, um das Verhalten des Web-Trackings an verschiedenen geografischen Standorten zu verstehen. Ich arbeite mit ihnen in enger Zusammenarbeit zusammen und wir beabsichtigen, das Web-Tracking-Datenset gegen Ende Mai 2019 zu veröffentlichen.

Richtlinie zum „Software-Sprint“

OnBaSca

Schlussbericht

Zuwendungsempfänger:

Julia Gonzalez

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen **01IS18S29** gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Die Tor Directory Authorities führen Bandbreiten-Scanner aus, um die Kapazität der Relais zu messen, die verwendet werden, um die Last im Tor-Netzwerk auszugleichen.

Dieses Projekt zielt darauf ab, einen Bandbreiten-Scanner zu implementieren, der benutzerfreundlicher und flexibler ist und es kann genauere Messungen auf eine sicherere Weise erhalten, um die Leistung des Netzwerks zu verbessern.

Durch eine bessere Messung des Netzwerks und durch schnellere Erkennung von Angriffen wird die Last besser ausgeglichen und die Benutzererfahrung verbessert. Es wird auch die Sicherheit und Privatsphäre der Benutzer verbessern, indem Sie sicherere Methoden zum Scannen des Netzwerks haben.

Es wird in Python unter Verwendung von gut gepflegten Tor-Bibliotheken, txorcon und Stem, implementiert werden, die auch von zwei neuen Bandbreiten-Scanner-Implementierungen verwendet werden, die noch nicht für die Produktion bereit sind, wie BwScanner und andere, die noch nicht öffentlich sind.

Sie wird entweder verbessert werden oder Teile werden wiederverwendet, um einen flexibleren zu haben, der verschiedene Scan-Algorithmen verwenden kann, wie zufällige Ausgänge oder Hilfsexits oder verschiedene Dateiserver oder Dateien, die zum Messen verwendet werden.

Es wird Ergebnisse im selben Format wie Torflow produzieren, so dass sie verglichen werden können, und es kann progressiv ohne den Bedarf eines Flaggentages eingesetzt werden.

Die Benutzerschnittstelle wird eine Befehlszeile sein, da sie nur von den Bandbreitenbehörden ausgeführt werden soll. Es wird mit Chutney getestet werden und mit Schatten simuliert.

Es wird möglicherweise für die Betriebssysteme gepackt, auf denen die Bandbreitenautoritäten ausgeführt werden.

Um die erhaltenen Messungen zu analysieren, werden metrics.torproject.org, tomrittervg.github.io/bwauth-tools/ verwendet sowie andere Skripte implementiert werden.

Die Entwicklung wird mithilfe agiler Methoden und Iterationszyklen erfolgt, die hauptsächlich aus folgenden Schritten bestehen: Implementieren eines neuen Features, Ausführen und Analysieren der Ergebnisse.

- Schreiben / aktualisieren Sie die Spezifikation der Bandbreitenbehörde
- Implementieren Sie eine einfache, modulare, nutzbare, die in der Bandwidth Authorities und Produkt-Ergebnisse ähnlich wie Torflow laufen kann
- Analysieren Sie, wie sich die Ergebnisse von Torflow unterscheiden, wie sie sich auf das Netzwerk auswirken und welche weiteren Funktionen zu implementieren sind
- Implementieren Sie die neuen Funktionen
- Simulationen ausführen und Ergebnisse analysieren
- Dokumentenumsetzung und Ergebnisse
- Implementieren Sie Testanzüge
- technischer Bericht
- Kommunikation mit dem Tor-Netzwerkteam
- Verbreitung in anderen Kommunikationskanälen und der "State of the Onion" Vortrag von Tor in Konferenzen

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Tor-Verzeichnisbehörden können Bandbreitenscanner einfacher einsetzen, was die Dezentralisierung, Fehlertoleranz und Genauigkeit der Bandbreitenmessungen erhöht.

Dank der Diagnose des neuen Bandbreitenscanners können Tor-Relaisbetreiber die Bandbreite ihrer Relais oder das Netzwerk besser verstehen, was sie zur Erhöhung ihrer Relaiskapazität speichern können.

Durch genauere Messungen würden Tor-Endbenutzer letztendlich von einer höheren Benutzersicherheit und Erlebnisqualität profitieren.

Da es sich bei der Software um Open-Source-Software handelt und wir eine Spezifikation[0] geschrieben haben, können Entwickler und Forscher den Bandbreitenscanner auf kompatible Weise weiter verbessern oder Parser implementieren, um Probleme im Netzwerk oder bei Relais zu erkennen.

Entwickler und Forscher können den Scanner zur Lösung der gemeldeten Probleme[1] und die im Blogbeitrag[2] erwähnte zukünftige Arbeit weiter verbessern.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Der Algorithmus zur Bandbreitenskalierung wurde nicht geändert, da es mehr Zeit in Anspruch nehmen würde, die Veränderungen im Netzwerk zu beobachten, mehr zu experimentieren und zu forschen.

Es wurde gemeinsam mit dem Tor-Netzwerkteam entschieden, dass der Scanner stattdessen Diagnosewerte meldet, die helfen, Probleme im Netzwerk, bei Relais oder Bandbreite zu verstehen.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Benutzer finden Informationen im Blog-Post[2], Relay-Betreiber, Entwickler und Forscher finden detaillierte Informationen im Abschnitt "Get Involved" des Blog-Posts, der Scanner-Website[3], dieser Projekt-Website[4], die mehrere Präsentationsfolien und die Bandbreiten-Datei-Spezifikation[0] enthält.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Die neuen Diagnoseberichtsaufgaben benötigten etwas mehr Stunden als geplant.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Die wichtigsten Ergebnisse sind im Blogbeitrag[2] zu sehen.

Die gesamte Arbeit wurde in Zusammenarbeit mit dem Tor-Netzwerkteam durchgeführt, das während des gesamten Projekts Feedback, Vorschläge und alternative Ziele lieferte.

Das ermöglichte es mir, vom Tor-Netzwerkteam zu lernen.

Referenzen

[0] <https://gitweb.torproject.org/torspec.git/tree/bandwidth-file-spec.txt>

[1] https://trac.torproject.org/projects/tor/query?status=accepted&status=assigned&status=merge_ready&status=needs_information&status=needs_review&status=needs_revision&status=new&status=reopened&component=%5ECore+Tor%2Fsbws&col=id&col=summary&col=status&col=type&col=priority&col=milestone&col=component&order=priority

[2] <https://blog.torproject.org/how-bandwidth-scanners-monitor-tor-network>

[3] <https://sbws.rtfid.io>

[4] <https://onbasca.rtfid.io>

Richtlinie zum „Software-Sprint“

PT-for-LEAP – Verbesserte anonymisierte Kommunikation

Schlussbericht

Zuwendungsempfänger:

Johannes & Nuyts GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S30 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Was war Deine Motivation? Welches Problem wolltest Du mit Deinem Projekt lösen? Wie war die geplante Vorgehensweise zur Problemlösung (auch Angabe der wichtigsten Meilensteine)?

LEAP Encryption Access Project- <https://leap.se/> & <https://bitmask.net/> - ist gegründet worden, um online frei und sicher kommunizieren zu können. Seit seiner Gründung im Jahr 2012 haben wir dafür eine Open-Source Client-Server-Architektur für die Bereitstellung von VPN-Diensten und verschlüsselter E-Mail aufgebaut.

In den vergangenen Jahren wurde die Umgehung von Internetzensur durch den Einsatz neuer Detektionsverfahren seitens von Unternehmen oder staatlichen Institutionen eingeschränkt.

Um weiterhin die Software für einen leicht aufzusetzenden und global erreichbaren VPN Dienst bereitstellen zu können, haben wir im Rahmen der Förderung Pluggable Transports (PT) client- und serverseitig implementiert. Pluggable Transports ist eine zusätzliche Transportschicht, in der Datenpakete gekapselt werden, um deren Eigenschaften zu obfusizieren. Damit ist es möglich, viele Deep Package Inspection (DPI) Verfahren auszuhebeln, mithilfe derer oftmals Internetzensur durchgesetzt wird. Gleichzeitig schützt Pluggable Transports sowohl Nutzer*innen als auch Dienstleister*innen von LEAP davor, von Zensoren als solche identifiziert zu werden.

Für die Umsetzung von Pluggable Transports sollten zunächst die verfügbaren Protokolle, die PT unterstützen, vor dem Hintergrund stetig weiterentwickelter DPI-Verfahren auf ihre Aktualität und

Effektivität evaluiert werden. Daraufhin sollten bestehende Implementierungen der in Frage kommenden Obfuskierungsprotokolle recherchiert, auf ihre Einsetzbarkeit in die bestehende Softwarearchitektur der Clients (Bitmask) und des LEAP-Servers bewertet und nötige Anpassungen an PT-Bibliotheken vorgenommen werden. Durch die Entwicklung von Puppetmodulen sollte die serverseitige Einrichtung der Pluggable Transports Software automatisiert werden. Die vorhandene Server-API zur Abfrage der Client-Konfiguration sollte um die nötigen PT-Parameter erweitert werden. Die transparent ablaufende Selbstkonfiguration von Bitmask sollte unter Zuhilfenahme der Server-API um den Einrichtungsschritt von Pluggable Transports erweitert werden. Pluggable Transports sollte für Bitmask auf den Plattformen Android, Linux und Mac unterstützt werden. Ein Client-Userinterface zur Aktivierung von Pluggable Transports sollte ebenso wie eine Server-Schnittstelle für Administrator*innen zur manuellen Konfiguration der unterstützten Protokolle, IP und Ports implementiert werden. Mithilfe von User*innentests sollten die entwickelten grafischen Oberflächen bewertet werden.

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

Wer ist die Zielgruppe für Deine Lösung? Wie profitiert sie von den Ergebnissen Deines Projekts? Welche Bezüge gibt es zu den Themenfeldern und Zielen des Software Sprints?

Zielgruppen von LEAP sind einerseits Journalist*innen und zivilgesellschaftlich und politisch aktive Gruppen. Die Software wird in der Praxis aber auch von anderen Menschen, die einen freien Zugang zu Informationen im Internet erhalten wollen, genutzt. Die Playstore-Statistiken unserer Android-Clients zeigen, dass besonders viele Menschen aus Ländern mit restriktivem Internetzugang den LEAP-Dienst nutzen. Wir erwarten, dass die Nutzer*innenbasis in China, Kasachstan und anderen Staaten, die erfolgreich VPN durch Deep Package Inspection blockieren, mit der Veröffentlichung der im Rahmen der Förderung erfolgten Implementierung wachsen wird.

Eine weitere wichtige Zielgruppe sind Community-Service-Provider und NGOs, die ihren Mitgliedern eine verschlüsselte Verbindung zu ihren Servern anbieten wollen. Diese erweitern mithilfe von PT ihre Reichweite. Außerdem ist diese Software natürlich auch nützlich für Menschen, die z.B. ihr Internetverhalten für andere Menschen aus ihrer Umgebung abschirmen möchten.

Ausführliche Darstellung der Ergebnisse

Welche konkreten Ergebnisse hast Du erzielt? Konnten alle Meilensteine erreicht werden? Welche zusätzlichen Erkenntnisse hast Du aus der Projektarbeit gewonnen, auch im Hinblick auf die Begleitung durch die Open Knowledge Foundation?

Wir haben zunächst die PT-Protokolle Obfs2, Meek, Scramblesuit, Domain Fronting und Obfs4 bewertet und basierend auf unserer Onlinerecherche Obfs4 als primär einzusetzendes Obfuskierungsprotokoll identifiziert.

Obfs4Proxy und shapeshifter-dispatcher wurden im Anschluss als mögliche Implementierungen prototypisch client- und serverseitig integriert. Beide Softwares basieren auf der Programmiersprache Go und sind dadurch theoretisch auf allen von LEAP unterstützten Plattformen einsetzbar. Um plattformspezifischen Hardwarearchitekturen zu unterstützen, mussten Skripte zum Cross-Compilieren des Sourcecodes entwickelt werden.

Die Skripte ermöglichten in einem späteren Schritt auch die Erweiterung unseres Continuous Integration (CI) Prozesses zum automatisierten Bauen und Testen auf unseren Build-Servern.

Um das Puppet-Modul zur automatisierten Einrichtung von shapeshifter-dispatcher auf den LEAP-Servern entwickeln zu können, mussten wir zunächst ein System-Update der vorhandenen Puppet-Module von Debian 8 („Jessie“) auf Debian 9 („Stretch“) vornehmen. Der Umfang dieser Arbeit wurde deutlich unterschätzt, da nicht nur das Betriebssystem aktualisiert wurde, sondern auch sehr viele Komponenten sehr eng mit einander verknüpft waren und geupdatet werden mussten. So musste zum Beispiel die Puppetmodule der Nutzer*innendatenbank, Monitoring und E-mail zum Teil ganz ersetzt werden.

Die Provider-API wurde so angepasst, dass auf Anfrage des Clients neben dem Port und der IP des unobfuskierten VPN-Gateways auch die Verbindungsdaten der unterstützten Pluggable-Transport-Technologie mitgeliefert werden. Hier wurde ein Augenmerk auf Erweiterbarkeit gelegt, sodass zukünftig auch andere Obfuskierte Protokolle ohne erneute Änderungen der Schnittstelle unterstützt werden können.

Wir haben in der Entwicklungsphase der Clienttechnologie zunächst unseren Fokus auf Android gelegt. Dort wurde der Selbstkonfigurationsprozess um den Einrichtungsschritt des shapeshifter-dispatchers ergänzt. Dieser schließt die Installation des Binaries, das Starten selbigen in einem eigenen Prozess und die Identifikation des nach dem Start zur Verfügung gestellten Proxy-Ports ein. Um den Batterieverbrauch durch Bitmask zu vermindern, wurde der Lebenszyklus des shapeshifter-dispatcher an jenen des OpenVPN-Prozesses gebunden. Bei der Android-Implementierung wurde ebenfalls Wert auf eine mögliche Ergänzung weiterer Protokolle gelegt.

Wir haben in einem späteren Schritt Shapeshifter-Dispatcher in unseren ebenfalls auf Go basierenden Desktop-Client Bitmask-lite prototypisch integriert, sodass die Obfuskierte Technologie auf Linux, Mac und Windows rudimentär unterstützt wird.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Welcher Nutzen ergibt sich für die Zielgruppe aus den Ergebnissen Deines Projekts? Welche weitergehenden Effekte ergeben sich aus der Open-Source-Stellung der Ergebnisse? Gibt es Ideen für die Weiterentwicklung Deiner Lösung und Pläne zu deren Umsetzung?

Hat die Arbeit in dem Projekt Dich in Deiner persönlichen, fachlichen Weiterentwicklung unterstützt?

Hat die Arbeit in dem Projekt Dich in Deiner persönlichen, fachlichen Weiterentwicklung unterstützt?

Der bisherige Stand der Entwicklung wurde noch nicht in den produktiv eingesetzten Source Code gemergt, da eine bessere Absicherung des Codes und einige Bugfixes ausstehen. Wir planen in den kommenden Wochen diese Schritte zu vollziehen. Wir gehen davon aus, dass Endnutzer*innen aus Weißrussland, China, Türkei, Russland, Turkmenistan, Irak, Iran, Oman und den Vereinigten Arabischen Emiraten jene LEAP-Provider erreichen werden können, die ihr System aktualisieren und Pluggable Transports aktivieren.

Der Einsatz von Pluggable Transports zur Umgehung von Internetzensur steht in einem permanenten Spannungsverhältnis zur Weiterentwicklung von Gegentechnologien zur Erkennung selbiger sowie zu politischen Entwicklungen weltweit. Es wird Nutzer*innen temporär die Möglichkeit geben, sich sicher über das Internet auszutauschen, bis Internet Service Provider oder staatliche Institutionen das von LEAP unterstützte Obfuskierungsprotokoll beispielsweise mithilfe von Machine-Learning-Algorithmen erkennen und blockieren können. Folgerichtig wird in der Zukunft das von LEAP entwickelte VPN neue Protokolle implementieren, um im Technologiewettbewerb der Internetzensur einen Schritt voraus zu bleiben. Gleichzeitig benötigt es neben technischen Lösungen weiterhin enorme zivilgesellschaftliche Anstrengungen, um die politischen Verhältnisse zugunsten freier Gesellschaften zu verändern. Wir wollen mit unserer Technologie gerade jene Gruppen unterstützen, die dafür einstehen.

Durch die Open-Source-Stellung unserer Software ermöglichen wir einen community-basierten Ansatz der Weiterentwicklung der Software. Sie ermöglicht außerdem eine öffentliche Kontrolle des Source Codes, was bei sicherheitskritischer Software ein besonders wichtigen Faktor darstellt. Insbesondere die Clients können als Referenzimplementierungen für andere Entwickler*innen dienen, da unseres Wissens nach noch keine anderen Open-Source-Projekte mit dem gleichen Technologie-Stack existieren. Während unserer Arbeit standen wir in Verbindung mit anderen Menschen aus der Network Traffic Obfuscation Gemeinschaft und haben auch Pluggable Transports Dokumentationen auf den neuen Stand gebracht, die andere Entwickler*innen nützen können.

Ein weiteres von Providern gewünschtes Feature ist ein zuverlässiges Monitoring von Gateways und anonymisierten Nutzungsdaten. Wir möchten nach der Förderung verschiedene Möglichkeiten hierzu untersuchen und implementieren.

Insbesondere in Bezug auf die aktuellen Entwicklungen von Obfuskierungstechnologien haben wir uns als Team weiterentwickelt.

Die Begleitung des Prototypenfunds durch die angebotenen Workshops hat uns auch geholfen, besser einzuschätzen in wie weit sich das Betreiben eines VPN-Providers zur Unterstützung der Softwareentwicklung ökonomisch rentiert. Der Extra-Input von außen ist wichtig, um sich nicht in technischen Details zu verlieren und um sich vor Augen zu führen, dass ein regelmäßiger Kontakt mit den Endnutzer*innen und den Service-Providern wichtig sind.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Infolge der prototypischen Integration mussten wir die fehlende Interoperabilität von Obfs4Proxy und shapeshifter-dispatcher feststellen, obwohl sie über das gleiche Protokoll kommunizieren. Wir haben uns schließlich dafür entschieden, shapeshifter-dispatcher auf allen Plattformen gleichermaßen einzusetzen.

Wir hatten darüber hinaus ursprünglich geplant, eine grafische Konfigurationsoberfläche für die Clients zur Aktivierung und Deaktivierung von Pluggable Transports zu implementieren. Wir haben uns jedoch dagegen entschieden, da wir die Komplexität der Software für den User möglichst gering halten wollen. Stattdessen implementierten wir eine Autodetektion der unterstützten Obfuskierungsprotokolle, einen automatischen Verbindungsaufbau unter Nutzung selbiger und einen optimierten Fallbackmechanismus für fehlgeschlagene Verbindungsversuche.

Während der Anfangsphase stellte sich heraus, dass Domain Fronting eine Technologie ist, die nicht mehr in der Praxis umsetzbar ist, weil alle große Plattformen dieses unterbinden.

Ein erheblicher Anteil der vorhandenen Dokumentation verweist noch auf veraltete Protokolle, die wir testweise eingesetzt haben, aber relativ schnell ersetzen mussten.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Wir haben zwar offen an unserem Quellcode gearbeitet, aber wir befürchten, dass es im Moment noch einige Probleme auftreten können, wenn Menschen diesen einfach kopieren. Wir haben vor, weitere Berichte über die Integration von Pluggable Transports in unsere Software auf den Zeitpunkt des ersten Releases, der jene Änderungen beinhaltet, zu verschieben. Der Quellcode existiert unter <https://Oxacab.org/leap/> und werden unseren Bericht auf <https://leap.se> veröffentlichen.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Die Aktualisierung der Puppet-Module als Voraussetzung zur serverseitigen Implementierung von Pluggable Transports hat sich als deutlich komplexer und langwieriger erwiesen, als wir in der Planungsphase vermutet hatten. Außerdem hatte die Behebung von Serverausfällen in unserem Build-System zu Verzögerungen geführt. Infolgedessen konnten wir einige geplante Ziele nicht erreichen. So werden im Moment noch zu wenig Optionen für Administrator*innen geboten, um Pluggable Transports zu konfigurieren. Eine Fallbackdiscovery der angebotenen Transports existiert ebenfalls nicht.

Bei der Implementierung für das Serverplattform fehlt im Moment auch noch die Integration mit dem Command Line Interface.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Gab es Entwicklungen anderer Personen oder Institutionen, die Einfluss auf Deine Arbeiten und die Zielsetzung hatten? Wenn ja, worin bestand dieser und wie bist Du damit umgegangen?

Wie schon erwähnt, war die Deaktivierung von Domain Fronting bei Google, Amazon und anderen Anbietern ein kleiner Rückschlag. Zum Glück gibt es weitere Obfuskierungstechnologien, sodass wir Obfs4 einsetzen konnten. Der Ausfall unserer Buildsysteme und Infrastruktur hat uns bestimmt eine Woche gekostet, aber weil es um einen nicht an uns kommunizierte Umzug ging, konnten wir nach Umbau und Neukonfiguration nachher wieder weiter. Nachdem wir unsere erste Verbesserungen für die Webseite und Anleitungen auf <http://pluggabletransports.info> vorgeschlagen hatten, wurden wir von Internews kontaktiert. Dieser Austausch war bereichernd und vor allen sehr motivierend: zu hören das die Arbeit sinnvoll ist und geschätzt wird. Wir haben in die letzte 6 Monaten auch intensiver zusammen gearbeitet mit Übersetzer*innen wodurch die Software auch durch mehr Menschen benutzt werden kann. Unsere Android Applikation ist inzwischen in 14 Sprachen übersetzt, von Deutsch und Englisch bis zum Baskisch, Hebräisch und Farsi.

Richtlinie zum „Software-Sprint“

ReproBuilds – Reproducible Builds in der Wirklichkeit

Schlussbericht

Zuwendungsempfänger:

Holger Levsen

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen **01IS18S31** gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Seit Ende 2014 beschäftige ich mich mit Reproducible Builds, wobei wir uns bis zum Projektbeginn im wesentlichen damit beschäftigt haben, Reproducible Builds theoretisch möglich zu machen, d.h. wir haben Fehler(klassen) in Quellcode(s) analysiert und Patches bereitgestellt, und dies (verschiedenen intensiv) in verschiedenen (Linux und anderen) Distributionen. Ein Schwerpunkt meiner Arbeit innerhalb des Reproducible Builds Projektes liegt bei der Entwicklung und Betrieb des Services <https://tests.reproducible-builds.org> welches folgende Projekte kontinuierlich testet: Debian, Archlinux, OpenWrt, coreboot, NetBSD, FreeBSD und F-Droid.

Die bisherigen Tests und der Betrieb von <https://tests.reproducible-builds.org> richteten sich an die Entwickler von den Projekten, nicht aber an Userinnen und User.

Ziel des Prototypefund Projekt war es, die Entwicklung dahin zu lenken, Userinnen und User zu ermöglichen, selber zu entscheiden, ob sie nicht-reproduzierbare Software installieren.

Anders gesagt, Ziel war es, die theoretische Ebene für eines oder mehrere der getesteten Projekte zu verlassen und praktisch zu werden.

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

Freie Software ist die Basis des Software Sprints, da Freie Software reviewed, verbessert, geteilt und genutzt werden kann. (Die 4 Freiheiten von Freier Software.) Allerdings gibt es – außer Reproducible Builds – keine Möglichkeit, sicher zu sein, dass die verwendete Software wirklich Freie Software ist, weil sich die 4 Freiheiten auf Sourcecode beziehen, während Binärcode verwendet wird und es – außer durch Reproducible Builds – nicht sichergestellt werden kann, dass ein bestimmter Binärcode von einem bestimmten Sourcecode abstammt.

Ausführliche Darstellung der Ergebnisse

Mein erstes Ziel war die Erstellung von <https://tests.reproducible-builds.org/archlinux/> wo die (theoretischen) Testergebnisse von Reproduzierbaren Archlinux untersucht und dargestellt werden sollten, um darauf aufbauend dann Archlinux praktisch reproduzierbar zu machen. Ersteres ist gelungen, zweiteres leider noch nicht, aufgrund der Probleme die ich weiter unten in „kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben“ (für Debian) beschrieben habe. Diese Probleme sind dieselben für Archlinux und Debian.

Ich betrachte diesen Teilschritt trotzdem als (teilweise) erfolgreich, weil er gezeigt hat/zeigt, dass

- a.) Archlinux sich auch reproduzierbar machen lässt (es also keine prinzipiellen Probleme gibt)
- b.) dazu geführt hat, diverse Probleme in Archlinux zu beheben und es
- c.) dadurch eine Einbindung der Archlinux Entwickler Community in das Reproducible Builds Projekt gegeben hat. Vor diesen Tests war ein Archlinux Entwickler Teil des Reproducible Projekts, heute sind 4-5 Leute aktiv.

Anfang Dezember 2018 war ich dann damit beschäftigt, den Reproducible Builds Summit in Paris auszurichten (siehe <https://reproducible-builds.org/events/paris2018/> inkl. einer ausführlichen Dokumentation der Ergebnisse des Summits). Ein technisch herausragendes Ergebnis dieses Summits war die Integration der (woanders erstellten) Ergebnisse der Tests von OpenSUSE in unsere SQL Projektdatenbank, mit dem Ziel diese Ergebnisse im JSON Format bereitzustellen, um dann Usertools zu entwickeln. Derzeit bieten wir JSON für Debian, Archlinux und OpenSUSE an (und seit dem grade vergangenen Wochenende auch für Alpinelinux). Die Entwicklung besagter Usertools befindet sich noch im Alphastadium aufgrund der weiter unten beschriebenen Probleme.

Anfang 2019 habe ich dann begonnen zu untersuchen, wie sich die 93% theoretische Reproduzierbarkeit von Debian in der Realität widerspiegelt und bin zu der etwas deprimierenden Zahl von 54% real reproduzierbaren Paketen gekommen, was im wesentlichen an diesen drei Bugs liegt:

- #894441: binNMUs should be replaced by easy "no-change-except-debian/changelog-uploads" (as of 20190302, 12% of all binaries in Buster were binNMUs.)

- #900837: Mass-rebuild of packages for reproducible builds (blocked by #894441)
- #862538: security.debian.org: Please POST .buildinfo files to buildinfo.debian.net

Diese drei Bugs sind allerdings Probleme in der Debian Infrastruktur und Prozessen, die zum Einem eine Kooperation verschiedener Teams in Debian nötig machen und zum Anderen nicht während des Release-Freezes adressiert werden können. Debian produziert ca. alle zwei Jahre ein neues Release, dazu wird vorher die Distribution „eingefroren“, so daß nur minimal invasive Änderungen gemacht werden. Dieses Mal begann der Freeze am 20. Januar 2019, so daß wir diese Änderungen erst jetzt im Sommer 2019 nach dem Release vorantreiben können.

Die Erkenntnisse daraus führten nach Projektende dazu, daß ich <https://buildinfos.debian.net> aufgesetzt habe, weil der bestehende (und von einem anderem Teammitglied betriebene) <https://buildinfo.debian.net> Service für unsere Zwecke nicht ausreicht. Eine Zusammenführung dieser zwei ähnlichen Services steht noch aus, ist aber geplant.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Das Prototypfund Projekt hat meine Sicht auf das Reproducible Builds Projekt massiv verändert: dadurch das ich schon 5 Jahre an diesem Problem gearbeitet habe, bin ich quasi ein wenig betriebsblind geworden: theoretisch sind von 28000 Source Paketen zB in Debian „nur“ 93% reproduzierbar, d.h. wir haben noch ca. 1940 Source Pakete, die wir nicht reproduzieren können.

Diese 1940 Source Pakete sind ein (großes) Problem, bzw. 1940 Probleme, aber sie verstellten den Blick auf ein größeres Problem, daß wir – bis Ende 2018 - nämlich nur die Reproduzierbarkeit in der Theorie getestet haben, nicht aber in der Praxis.

Diesen Fokuswechsel, erst meinerseits, dann sukzessive weitergegeben an andere Projektteilnehmer, betrachte ich persönlich als den größten Effekt dieses Prototypfund Projekts, da dies die weitere Arbeit entscheidend verändert hat.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Leider gibt es derzeit noch ein theoretisches und praktisches Grundproblem, d.h. leider kein Basissystem reproduzierbar ist, da bestimmte Pakete (perl, gcc-8, apt, sudo, groff, nss, xorg-server und python2.7 um nur 8 unreproduzierbare Pakete von 71 unreproduzierbaren Paketen der 1337 meist installierten Debian-Paketen zu nennen) nicht reproduzierbar sind. Dadurch ist es derzeit noch nicht möglich, ein (kleines) System zu bauen, das 100% reproduzierbar ist.

Aus diesem Grund habe ich an der praktischen Reproduzierbarkeit von diversen Systemen (Debian, Archlinux, OpenWrt) gearbeitet, bis ich an kurz- und mittelfristig unlösbare Probleme stieß und habe mich dann einem anderen System zugewandt, bis ich wieder irgendwo hängenblieb, so daß ich mir dann wieder ein anderes System vorgenommen habe.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Bis Ende März 2019 haben wir in 206 wöchentlichen Blogposts über die Arbeit unseres Projektes informiert (seit April 2019, also nach Projektende, schreiben wir monatliche Blogposts), diese finden sich unter <https://reproducible-builds.org/reports/> wobei für das Prototypefund Projekt die Wochen 176-201 relevant sind.

<https://reproducible-builds.org> ist die generelle Webseite des Projekts, unsere Testergebnisse finden sich auf <https://tests.reproducible-builds.org>

Sourcecode findet sich unter <https://salsa.debian.org/reproducible-builds/> und unter <https://salsa.debian.org/qa/jenkins.debian.net/>

<https://www.youtube.com/watch?v=PSxm2DbDHG8> ist ein 90 Sekunden langes Video über meine Arbeit, produziert vom Prototypefund.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Die Arbeitsplanung war sehr dynamisch, da kontinuierlich an der Reproduzierbarkeit von verschiedenen Projekten gearbeitet wurde und dies immer in Kooperation mit anderen Stakeholdern geschah. Dies hatte den Vorteil, daß es quasi keine Leerläufe gab, hatte allerdings auch den Nachteil, daß dadurch manche Arbeiten unterbrochen wurden (z.B. Archlinux) und erst nach Wochen oder Monaten wieder aufgenommen wurden.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Neben der Entwicklung und des Betriebs von <https://tests.reproducible-builds.org> besteht eine weitere Hauptrolle von mir innerhalb des Reproducible Builds Projekts darin, Anfragen und Arbeiten von beteiligten oder sich beteiligenden wollenden Projekten zu koordinieren und fördern, sowohl im IRC (Chat) als auch auf unseren Mailinglisten oder bei Konferenzen. Diese Arbeiten nehmen wöchentlich zwischen 1-10 Stunden (von mir) in Anspruch, die sich dann durch die Mitarbeit anderer teilweise extrem auszahlen. Zum Beispiel wurde Herr Alexander Couzens durch mich intensiv motiviert, diverse Reproduzierbarkeitsprobleme in OpenWrt zu lösen.

Mitte Dezember 2018 erhielt ich eine Email von OSUOSL (Oregon State University Open Source Labs), daß wir Zugriff auf acht Server mit *jeweils* 144 GiB RAM und 32 Cores bekommen würden (zusätzlich zu den bestehenden 44 Testsystemen). Die Einbindung dieser Server in unser existierende Test-Setups war dann eine unerwartete Aufgabe.

Im September 2018 wurde außerdem unser Logo fertiggestellt, welches dann an diversen Stellen eingebunden werden mußte.

Der Reproducible Builds Summit in Paris vom 11.-13. Dezember, an dem 50 Leute aus 25 Projekten teilgenommen haben, hat sowohl in der Vor- als auch Nachbereitung einige Zeit gekostet, wobei die Synergieeffekte dieses Treffens beachtlich sind und den Zeitaufwand mehr als wettmachen.

Richtlinie zum „Software-Sprint“

SecureDrop – Schutz für Journalisten und Quellen

Schlussbericht

Zuwendungsempfänger:

Eric Hartsuyker

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S32 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Dieses Projekt wurde motiviert, SecureDrop für Quellen, Journalisten und Administratoren benutzerfreundlicher zu machen. Die Sicherheit ist kompliziert und oft sind die Unsicherheiten auf einen Benutzerfehler zurückzuführen. Wenn Software einfacher gemacht wird, gibt es weniger Fehler. SecureDrop zielt darauf ab, die Quellen angesichts des von Massenüberwachung ausgehenden Risikos so gut wie möglich zu schützen. Ein neuer Journalisten-Workflow und die Aktualisierung von altem Code, Software-Packages und Betriebssystemen.

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

Die Hauptzielgruppe für dieses Projekt sind Whistleblower und die Journalisten, die mit ihnen kommunizieren. Journalisten profitieren von zusätzlichen Quellen, die vertrauliche Informationen senden, und sie fühlen sich durch ihre Anonymität dazu befugt. Quellen profitieren von einem geringeren Risiko der staatlichen Verfolgung, da sie bei der Kommunikation mit Quellen identifiziert werden. Infolgedessen profitiert die gesamte Gesellschaft davon, Machtmissbrauch ausgesetzt zu sein.

Ausführliche Darstellung der Ergebnisse

Wir konnten das Betriebssystem, auf dem SecureDrop ausgeführt wird, von Ubuntu 14.04 auf Ubuntu 16.04 aktualisieren. Wir haben einen Prototyp von einem Qubes basierenden Client mit Qt erstellt. Die Client-Funktionen erforderten Ergänzungen und Änderungen an der API. Der Client verfügt über die meisten wichtigen Funktionen, um von Journalisten verwendet zu werden. Einige der vorgeschlagenen Funktionen konnten nicht implementiert werden, da Daten aus

Benutzerforschungsergebnissen mit unseren ursprünglichen Annahmen in Konflikt standen. Durch die Zusammenarbeit mit einem UX-Forscher konnten wir jedoch erhebliche Änderungen an der UX vornehmen, anstatt die von uns vorgeschlagenen großen Änderungen.

Das Größte, was ich gelernt habe, waren die Grundlagen für die Erstellung eines nativen GUI-Clients mit Qt. Dies wird in Zukunft sehr nützlich sein, um Anwendungen zu erstellen, die benutzerfreundlicher sind als Befehlszeilentools.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Quellen profitieren am meisten von Verbesserungen an SecureDrop, die bei Verwendung der Anwendung dem größten Risiko ausgesetzt sind. Ein Teil dieser Sicherheit ist für den Benutzer sichtbar, aber ein Großteil davon befindet sich hinter den Kulissen.

SecureDrop wurde in einer Reihe von Vorträgen und Forschungsarbeiten als hervorragendes Beispiel für sichere Open-Source-Software zitiert, die ein sehr spezifisches Ziel für die Benutzer hat. Andere aus der Community profitieren davon, SecureDrop als Vorlage für die vollständige Bereitstellung einer sicheren Anwendung verwenden zu können. Die Entwicklung von SecureDrop ist für die gesamte Open Source-Community geeignet.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Ein Meilenstein, den wir ursprünglich vorgeschlagen hatten, war die Möglichkeit, einem Journalisten eine Quelle zuzuweisen. Dies war eine Idee, die wir uns ausgedacht und prototypisiert haben. Wir testeten und recherchierten mit Benutzern und stellten fest, dass diese Lösung bei den meisten Arbeitsabläufen nicht funktionierte und nur von einer Minderheit von Journalisten verwendet werden konnte. Dies zeigt, wie wichtig das Testen von Benutzern ist.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

SecureDrop kann auf verschiedene Arten kontaktiert werden:

Website – <https://securedrop.org>

Chat – <https://gitter.im/freedomofpress/securedrop>

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Es wurden einige erste Meilensteine festgelegt, die für 6 Monate Arbeit für 1 Person sinnvoll waren. Wir wussten jedoch, dass diese möglicherweise nicht durch zeitliche Einschränkungen oder Planänderungen erreicht werden. SecureDrop hat viele ungelöste Tickets, und wenn mehr Zeit übrig war, wurden andere Tickets zugewiesen.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

SecureDrop existiert als Teil eines großen Ökosystems verwandter Technologien zum Schutz der Privatsphäre. Ein Beispiel, bei dem ein anderes Projekt unser Projekt durchschnittlich hat, war ein Fehler in Firefox. Dieser Fehler verursachte Probleme mit dem Verhalten des Addons NoScript. Der

Tor Browser basiert auf Firefox und wird mit NoScript geliefert. Wir mussten mit den Upstream-Projekten zusammenarbeiten, um dieses Problem zu identifizieren und zu umgehen.

Richtlinie zum „Software-Sprint“

EMASS – E-Mail-Accountverwaltung mit Self-Service

Schlussbericht

Zuwendungsempfänger:

Dittler Meurer GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S33 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Bei den meisten Webdiensten haben die technische Administrator*in auch die komplette Macht über den Inhalt des Systems. Datenschutzfreundliche Webdienste sollten den Nutzer*innen aber möglichst viel Autonomie bieten und den Administrator*innen möglichst wenig Macht.

Mit dem Projekt sollte eine bestehende E-Mail Accountverwaltung als Open Source Software weiterentwickelt und insbesondere um die neuen Funktionen E-Mail-Postfachverschlüsselung und Account-Wiederherstellungs-Prozess erweitert werden. Verschlüsselte Postfächer sind für Administrator*innen nicht mehr einsehbar und die Verschlüsselung kann nicht umgekehrt werden. Dazu waren folgende Meilensteine vorgesehen:

1. Heben der Codebase auf Symfony 4.0
2. Datenbankabstraktion
3. Interner Audit
4. Erweiterung der Tests
5. Integration des Alias-Self-Service
6. Einmal-/SPAM-Aliase
7. Integration von Trees
8. Integration Domain-spezifischer Admins
9. 2FA implementieren
10. Dokumentation erstellen
11. Bekanntmachung
12. Erfahrungen und Probleme aus Praxiseinsatz berücksichtigen

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

Wir wollten mit dem Projekt kleinere E-Mail-Provider und deren Nutzer*innen unterstützen. Dazu haben wir uns mit anderen unkommerziellen Mailbetreibern und ausgetauscht und sie über die Veröffentlichung informiert. E-Mail-Nutzer*innen profitieren konkret von dem Projekt, wenn sie einen E-Mail-Account bei einem Provider haben. Damit trägt das Projekt insbesondere zum Themenfeld „Datenschutz“ bei.

Ausführliche Darstellung der Ergebnisse

Die E-Mail Accountverwaltung wurde deutlich in ihren Funktionen erweitert, der Quellcode an vielen Stellen grundsätzlich überarbeitet und mit umfangreichen Unit-Tests, sowie funktionalen Tests, eine gute Grundlage für anhaltende Codequalität gelegt. Mit Ausnahme der Zweifaktorauthentifizierung konnten wir alle zu Beginn definierten Meilensteine erfolgreich erreichen. Die bedeutendsten neu eingeführten nutzer*innenseitigen Funktionalitäten sind:

- E-Mail Postfachverschlüsselung mittels Dovecot MailCrypt
- Wiederherstellungsprozess für Accounts bei verloren gegangenem Passwort
- Domainadmin-Selfservice: Optional Admin-Funktionen für ganze Domains
- Alias-Selfservice: Nutzer*innen können sich selbst gewählte sowie Zufalls-Aliases anlegen
- Grundlegende Arbeiten am User Interface für bessere Benutzer*innenfreundlichkeit

Die Projektbegleitung durch die Open Knowledge Foundation hat uns insbesondere bei der organisatorischen und administrativen Arbeit deutlich unterstützt. Auch die zwei in der Projektförderung enthaltenen Coachings (in unserem Fall UI/UX Coachings durch Simply Secure) haben uns stark weitergeholfen.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Die erreichten Nutzer*innen profitieren unmittelbar in der Sicherheit ihrer E-Mail-Daten. Die Inhalte auf dem Server liegender E-Mails sind nicht länger den Administrator*innen der E-Mail-Server zugänglich.

Die Veröffentlichung der Software als Open Source bringt viele Vorteile mit sich. Einmal können auch andere (kleine) E-Mail-Provider die Software für sich verwenden, oder zumindest als Referenz für eigene Lösungen verwenden. Zum anderen aber ist es den Nutzer*innen und weiteren Interessierten möglich, die Funktionalität der Software zu überprüfen und damit zu verifizieren, dass die vom E-Mail-Provider versprochene Sicherheit auch erfüllt wird. Wissensverteilung, Vermeidung von doppelter Arbeit und Transparenz sind also starke Gründe für Open Source.

Wir haben bereits Ideen, das Projekt weiterzuentwickeln. So wollen wir die Accountverwaltung zu einem generischen Authentication- und Identity-Provider umbauen, welcher die Verwaltung von Nutzer*innen und ihrer Rechten service-unabhängig ermöglicht.

Als Beteiligte konnten wir unsere fachlichen Kompetenzen im Rahmen des Projekts ausbauen.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Wir hatten zu Beginn vor eine andere Implementierung von E-Mail-Postfachverschlüsselung zu verwenden. Nach ausführlicher Untersuchung der verfügbaren Lösungen wurde uns klar, dass Dovecot MailCrypt die zukunftssträchtigere und technisch saubere Implementierung ist. Wir haben uns daraufhin für diese Lösung entschieden.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Unser Projekt findet sich unter <https://github.com/systemli/userli>. Die zugehörige Dokumentation ist unter <https://systemli.github.io/userli/> zu finden.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Wir sind selbst überrascht, wie realistisch sich unsere anfängliche Projektplanung retrospektiv darstellt. Die Entwicklung von Unit-Tests zu Beginn war deutlich arbeitsaufwändiger als gedacht, da sie auch andere Probleme offenbarte. Die meisten anderen Teilaufgaben entsprachen aber in etwa dem geschätzten Arbeitsaufwand.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Es gab keine nennenswerten Einflüsse durch andere Institutionen, die den Verlauf des Projekts maßgeblich beeinflusst haben.

Richtlinie zum „Software-Sprint“

"Papierloser Alltag"

Schlussbericht

Zuwendungsempfänger:

Svante Schubert

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen **01IS18S34** gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Was war Deine Motivation? Welches Problem wolltest Du mit Deinem Projekt lösen? Wie war die geplante Vorgehensweise zur Problemlösung (auch Angabe der wichtigsten Meilensteine)?

Der Idee des papierlosen, umweltfreundlichen Alltags für jeden ist nicht neu, doch nicht einmal die digitale Rechnung hat sich durchgesetzt.

Vieles spricht dafür; allein 220 Fußballfelder deckt das Papier des Zahlungsverkehrs der deutschen Autoindustrie mit Zulieferern ab. Naturressourcen könnten geschont und lästige Arbeit gespart werden. Doch es fehlt die Akzeptanz, sowie unterstützende Software. Ein Henne Ei Problem.

Trotz eines Rahmen gebenden EU Standards erschuf jedes EU Land seine eigene eRechnung Syntax: allein in Deutschland existieren zwei eRechnungsformate und in Europa fast ein Dutzend, die alle EU-weit akzeptiert werden müssen.

Ein Wildwuchs, der wie geschaffen ist für eine offene, zentrale Lösung zur Harmonisierung der eRechnungsformate.

Aktuelle Meilensteine

1. Erhalt der Editorversionen der Spezifikation im Officeformat (anstatt Papier oder strukturloses PDF) durch Teilnahme im DIN Gremium „Elektronisches Rechnungswesen“
2. Einarbeitung in die Spezifikation und Extraktion der Daten aus den SyntaxBinding/Mapping Tabellen, welche die Semantik mit einem der beiden XML Formate verbindet
3. Datenvalidierung / Sanitytests, um die Qualität der Daten und der später darauf aufbauenden Software zu gewährleisten

4. Laden der Spezifikations Daten in eine Graph Datenbank, sowie weitere Datenvalidierungen der Spezifikation mit Hilfe der Graphdatenbank
5. Generierung eines möglichst starken Rahmen der späteren Software
6. Manuelle Implementierung der Lücken, damit die Test Suite der EU für Implementierungen der Spezifikation (en16931) passiert wird.

Das Projekt befindet sich im Meilstein 4 von 6. Es hat sich aufgrund des unerwart, hohen Zeitaufwandes für die Sicherung der Datenqualität der Spezifikation verzögert, doch wird das Projekt über den Sommer weitergeführt, um es zu vollenden.

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

Welche Bezüge gibt es zu den Themenfeldern „Civic Tech“ und „Data Literacy“ des Software Sprints oder zu weiteren gesellschaftlich relevanten Zielen bzw. Lösungsansätzen?

Wir möchten Bürokratie abbauen, indem wir eine offene, EU-taugliche eRechnungssoftware aufbauen. Andere offene Projekte können hierauf aufsetzen, ohne sich mit der Komplexität aller Formate befassen zu müssen. Die Annahme von eRechnungen, deren Prüfung und Erstellung ist heutzutage eine technische Basisfunktionalität. Erst die Weiterverarbeitung, wie das automatische Ausfüllen einer Überweisung, eine Erinnerung oder Anbindung an Steuersoftware würden den Nutzer*innen Vorteile bringen. Wir möchten erreichen, dass nicht nur wenige große Firmen diese Optionen besitzen, sondern sie allen Softwareentwickler*innen zur Verfügung steht, um innovative papierlose Anwendungen für alle zu entwickeln.

Ausführliche Darstellung der Ergebnisse

Welche konkreten Ergebnisse hast Du erzielt? Konnten alle Meilensteine erreicht werden? Welche zusätzlichen Erkenntnisse hast Du aus der Projektarbeit gewonnen, auch im Hinblick auf die Begleitung durch die Open Knowledge Foundation?

Wir haben die Daten für die Software nach einigen Rückschlägen doch noch aus der EU Spezifikation extrahieren können und aufgrund der Problematik mit der Datenqualität die Extraktionswerkzeuge als OpenSource den EU Editoren unter GitHub ^[1] bereitgestellt. Auch werden wir Anfang April zum CEN Plenary nach Reykjavik reisen, um über unsere Arbeit zu berichten und die versuchen die Weiterentwicklung der e-Rechnung zu beeinflussen.

Vor allem besser maschinenlesbare Spezifikationen mit Tests beim Standard aber auch kostenfrei erhältliche Spezifikationen erhältlich sind, die sich besser warten lassen (als derzeit mühsam über CEN Corrigendas).

¹ <https://github.com/svanteschubert/en16931-data-extractor>

Wir sind im Dialog mit den Entwicklern aus Australien, die im Auftrag der dortigen Regierung den Software Standard für das Web auf UN/CEFACT Ebene erstellen und glauben auch diesen positiv zu beeinflussen (dass dieser ebenso auf die EU Semantik aufsetzt und wir eine weltweite bessere Interoperabilität erreichen).

Folgende Arbeiten möchten wir noch über den Sommer zu Ende bringen, denn keine „halbfertige Brücke“ ist benutzbar:

1. Laden der Spezifikations Daten in eine Graph Datenbank, sowie weitere Datenvalidierungen der Spezifikation mit Hilfe der Graphdatenbank
2. Generierung eines möglichst starken Rahmen der späteren Software
3. Manuelle Implementierung der Lücken, damit die Test Suite der EU für Implementierungen der Spezifikation (en16931) passiert wird.

Die OpenSource e-Rechnungs Software wird über GitHub veröffentlicht werden, auch arbeiten wir an einer Partnerschaft aus der Industrie, damit wir nicht an späteren Kunden vorbeiarbeiten und unser Produkt auch benutzt wird.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Welcher Nutzen ergibt sich für die Zielgruppe aus den Ergebnissen Deines Projekts? Welche weitergehenden Effekte ergeben sich aus der Open-Source-Stellung der Ergebnisse? Gibt es Ideen für die Weiterentwicklung Deiner Lösung und Pläne zu deren Umsetzung?

Als Zielgruppe sehen wir jede*n Empfänger*in und Sender*in von Rechnungen, der*die nicht länger auf Papier und manuelle Bearbeitung angewiesen sein möchte. Dabei soll unser eRechnungs-Infrastrukturprojekt zunächst anderen Entwicklungsteams dienen, welche sich mit eRechnungen beschäftigen und basierend auf diesem Projekt innovative Lösungen aufbauen möchten. Vor allem andere Open-Source-Systeme, die unser Modul nutzen wollen, ohne sich mit der Materie der vielen EU-Formate im Detail beschäftigen müssen und sich dadurch auf die eigentlichen Funktionen fokussieren können.

Hat die Arbeit in dem Projekt Dich in Deiner persönlichen, fachlichen Weiterentwicklung unterstützt?

Ich konnte mich tief in die Thematik der elektronischen Rechnung einarbeiten und dabei nationale, sowie internationale Experten über die verschiedensten Fachgremien kontaktieren (DIN, CEN und UN/CEFACT seien genannt).

Ich denke ohne die Förderung, hätte ich diesen riesigen Kraftaufwand für das OpenSource Projekt nicht ohne weiteres auf mich genommen. Danke! 😊

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Gab es Arbeiten bzw. Lösungsansätze, die nicht weiter verfolgt wurden? Was waren die Hintergründe, und wie bist Du alternativ vorgegangen?

Eine Extraktion der Daten aus den PDF war äusserst fehlerhaft, da im PDF Dateiformat „Struktur“ nicht von vornherein im Design vorgesehen war. Doch gelang eine Teilnahme beim Verein der Deutschen Standardisierungsnorm (DIN) im Gremium „Elektronisches Geschäftswesen“.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Wo können sich Interessenten detailliert über Deine Projektergebnisse informieren (z.B. Webseite, GitHub, Veröffentlichungen)?

Neben des bereits erwähnten GitHub repos für die Extraktion der Daten <https://github.com/svanteschubert/en16931-data-extractor> (gedacht für EU Editoren der Spezifikation), werden noch ein weiteres Repo mit der Software über den Sommer folgen.

Außerdem ist ein Vortrag auf dem CEN Plenum in Reykjavik geplant, sowie technische Vorträge auf der OpenSource Konferenz FOSDEM in Belgien, sowie XML Konferenz im kommenden Jahr in Prag.

Eine eigene Webseite mit Dokumentation mit Hilfe von Partnern aus der Industrie wird angestrebt.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Gab es im Projektverlauf Ereignisse, die eine Anpassung der Planung erforderlich machten – z.B. Mehr- oder Minderaufwand bei der Bearbeitung von Teilaufgaben?

Es gab einen unerwarteten Mehraufwand bei der Extraktion der Daten aus den Office Dokumenten der en16931 Spezifikation. Vor allem war aber die Datenqualität der Spezifikation weitaus schlechter als angenommen und alle möglichen Datenkontrollen & Validierungen wurden durchgeführt und als ein eigenes Vorprodukt für die EU Editoren veröffentlicht: <https://github.com/svanteschubert/en16931-data-extractor>

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Gab es Entwicklungen anderer Personen oder Institutionen, die Einfluss auf Deine Arbeiten und die Zielsetzung hatten? Wenn ja, worin bestand dieser und wie bist Du damit umgegangen?

Auf Konferenzen und auch mit Interessierten wurden lange Diskussionen über das Thema geführt, die in die Arbeit des Projektes mit einfließen.

Sensorimotor

Schlussbericht

Zuwendungsempfänger:

JustScheunemann Sensorimotor GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S35 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Wir entwickeln Open-Source-free-Design Hardware und Software für Robotik-Antriebe und wollen so die vielen kleinen Robotik-Projekte weltweit fördern und durch kostengünstige und modifikationsfreundliche Hardware die Kreativität der Entwickler*innen, Forscher*innen und Bastler*innen anregen und sie ein Stück weit unabhängiger von bestehenden industriellen Lösungen machen. Wir haben unsere Wurzeln in der universitären Robotik-Forschung und in der Maker-Szene und möchten unsere Entwicklungen den Robotikenthusiasten weltweit zugänglich machen.

Roboter und KI halten Einzug in unseren Alltag. Wir glauben, dass die Entwicklung für die Menschheit vorteilhaft und sicher sein kann - wenn die zugrundeliegende Technologie frei von Interessen der Wenigen ist und in die Hände von Forscher*innen und Entwickler*innen weltweit gelegt wird. Hierfür sind freie Hardware-Designs entscheidend, denn sie stärken die Freiheit der Nutzer*innen und geben ihnen wieder Kontrolle über ihre Hardware. Mit freien Hardware-Designs können Maschinen auch endlich wieder repariert und modifiziert werden. Wir möchten zur Demokratisierung der Robotik beitragen, indem wir ein völlig freies Hardware-Design und die dazugehörige Software für Robotik-Antriebe entwickeln.

Ziel unseres Projektes war es, die Nutzer*innen in die Lage zu versetzen, einfache Motoren in SmartServos zu verwandeln und somit für beliebige Robotikaufgaben einsetzbar zu machen. Hierfür haben wir eine Platine entwickelt, die mit zahlreichen DC-Motoren kompatibel ist und mit minimalem Aufwand angeschlossen werden kann. Mit ihr erhalten Nutzer*innen ein reichhaltiges sensorisches Feedback vom Motor (Winkel, Temperatur, Stromstärke, Spannung) und es gibt eine Vielzahl an Control-Modi, die eine exakte Regelung von Position, Geschwindigkeit und Drehmoment ermöglichen.

Die Kommunikation zu den Motoren erfolgt über ein standardisiertes Bus-System. Dadurch können bis zu 256 Motoren über eine Strecke von bis zu einem Kilometer miteinander gekoppelt werden und von einem einzigen Host-Controller gesteuert werden, was insbesondere für Automatisierungs- und

Robotik-Projekte von großem Vorteil ist. Die Firmware ist in C/C++ geschrieben und wir verwenden ein open-source build/test system.

Ziel des Sensorimotor während der Förderungsphase war es, die fertige Hardware um notwendige Software zu ergänzen und somit für alle nutzbar zu machen. Mit einer Arduino-kompatiblen C/C++ Bibliothek und einer Raspberry Pi-kompatiblen Python-Bibliothek können nun alle Motoren im Bus für viele unterschiedliche Projekte komfortabel angesteuert werden.

Meilensteine des Projekts waren:

1. C/C++ Bibliothek inklusive Unittests, Beispielcode und Tutorials für Arduino
2. Python-Bibliothek inklusive Unittests, Beispielcode und Tutorials für Raspberry Pi
3. Erstellung einer umfassenden Dokumentation inkl. weiteren Anwendungsbeispielen.

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

Unsere Zielgruppe sind Wissenschaftler*innen, Ausbilder*innen, Maker und Entwickler*innen kleinerer Unternehmen, insbesondere in strukturell schwächeren Regionen. Durch free-design Hardware besteht die Möglichkeit zur Modifikation und Eigenproduktion und damit einhergehend die Förderung unabhängiger Entwicklungen. Durch ausführliche Dokumentation, Tutorials und Veröffentlichung aller Schaltpläne auf einer Plattform wie GitHub, wird ein weltweiter Wissensaustausch ermöglicht.

Ausführliche Darstellung der Ergebnisse

Im geförderten Projektzeitraum wurden die folgenden Coding-Projekte umgesetzt:

Protokoll, C++ Library, Python-Library, Arduino-Library, Microcontroller-Firmware, Protokoll-Plugin für Logic Analyzer, Servo Platine.

#Protokoll

Ein offenes Protokoll ist die Voraussetzung dafür, dass das Projekt als offene Plattform verwendet werden kann. Die initiale Entscheidung fiel daher auf ein möglichst einfaches, binäres Protokoll um die Verarbeitung durch leistungsschwache Plattformen wie den ATmega328 zu ermöglichen. Die zweite Priorität war die Sicherung der übertragenen Daten vor Übertragungsfehlern, was die Nutzung von einfacher und kosteneffizienter Hardware erlaubt.

Dieses Ziele führten zu der Entwicklung eines simplen und robusten Protokolls mit Prüfsumme zur Fehlererkennung, somit wurden unsere Initiellen Anforderungen an diese Komponente erfüllt.

C++ library

Bei der C++ Library handelt es sich um die Referenzimplementation des Protokolls für PC. Sie wurde als library konzipiert um sie für viele Programmiersprachen und Entwicklungsumgebungen verwendbar zu machen.

Python library

Bei der Python-library handelt es sich um einen sog. "Wrapper" für die C++ library, das heißt bereits implementierte Funktionen werden für die Python-Programmiersprache bereitgestellt. Dies erlaubt das Verwenden in einsteigerfreundlichen Python-Projekten.

Arduino-Library

Die Arduino library ist eine Implementation des Protokolls für die Arduino Plattform. Sie stellt keinen Wrapper der C++ Library dar, sondern ist eine komplett unabhängige Entwicklung mit Fokus auf Kompatibilität für die Arduino-Plattform. Um die Bibliothek auch für Beginner nutzbar zu machen, wurde Wert auf eine möglichst einfache API gelegt.

Microcontroller-Firmware

Die Firmware stellt das Gegenstück zu den Bibliotheken dar, und hat die Aufgabe zwischen dem Protokoll und Steuerbefehlen auf Hardwareebene zu übersetzen.

Protokoll-Plugin für Logic Analyzer

Das Protokoll-Plugin bietet die Möglichkeit für Entwickler die Kommunikation zwischen Bibliothek und Servoplatine auf Hardwareebene zu betrachten. Dieses Projekt soll es einfacher gestalten eigene Protokollerweiterungen zu entwickeln und zu testen.

Die gesetzten Meilensteine wurden erreicht: Die genannten Komponenten sind fertig gestellt. Das Implementieren von Projekten auf Basis unserer Plattform ist möglich, und wurde bereits parallel während der Entwicklungsphase getestet.

Das Arbeiten mit einer Kombination von sowohl Software, Hardware und Firmware ist eine äußerst interessante und erfüllende Erfahrung, von der wir hoffen sie mit vielen Nutzern der Plattform teilen zu können.

Gerade aus dem Zusammenspiel der verschiedenen Komponenten ergaben sich natürlich viele potentielle Fehlerquellen die für Probleme sorgten; Im Speziellen die Hardware verhielt sich zeitweise anders als erwartet, und führte somit zu Schwierigkeiten die bei z.B. beim Entwickeln einer Webanwendung oder Handy-App sicher nicht aufgetreten wären.

Diese Probleme wurden jedoch bereits in der Planungsphase vorhergesehen, und für ihre Lösung Zeit und Ressourcen eingeplant. Wir sind überzeugt davon das sich mittels der Integration von Soft- und Hardware viele neue Problembereiche lösen und revolutionieren lassen, und wünschen uns daher vom PrototypeFund und der OKF auch in Zukunft die Bereitschaft, auch Hardware-nahe Softwareprojekte zu unterstützen.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Wir wollen mit dem Sensorimotor eine Alternative zu bestehenden hochpreisigen closed-source SmartServos, wie z.B. Robotis/Dynamixel, anbieten. Durch unsere freie Hardware wird es ermöglicht, diese Technologie zu studieren, zu modifizieren und eigene Ideen darauf aufzusetzen. Dies gibt insbesondere kleineren, nicht finanzstarken Projekten mehr Unabhängigkeit und Möglichkeiten, denn: free hardware enables everybody.

Da unsere Zielgruppe sind Wissenschaftler*innen, Ausbilder*innen, Maker und Entwickler*innen kleinerer Unternehmen sind, ergeben sich für diese, durch die Veröffentlichung unserer free-design Hardware und der dazugehörigen Software, die Möglichkeit zur Modifikation und Eigenproduktion und damit einhergehend die Förderung unabhängiger Entwicklungen. Durch ausführliche Dokumentation, Tutorials und Veröffentlichung aller Schaltpläne auf einer Plattform wie GitHub, versuchen wir alle Interessierten einen möglichst einfachen Einstieg in eigene Projekte zu ermöglichen.

Mittlerweile gibt es eine ganze Reihe an Projekten, zum Teil von uns selbst angestoßen, aber wir haben auch komplett von uns unabhängigen Robotikprojekten mit der Verfügungstellung von Sensorimotor-Platinen und Software helfen können.

Die Arbeit am Sensorimotor hat uns nicht nur technisch weiter gebracht – mit den erzielten Ergebnissen ist der Grundstein für unser eigenes open-source Robotikprojekt gelegt - sondern uns auch als Team zusammen gebracht und gelehrt trotz aller Widrigkeiten zusammen zu arbeiten.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Das Projekt wurde mit dem Ziel begonnen eine vollständige neue Plattform für die Entwicklung von Mechatronikprojekten zu erschaffen und die bestehenden Plattformen zu ergänzen. Als Teil dieses Plans hatten wir uns vorgenommen mit evtl. vorhandener contingency, also Pufferzeit, nützliche Tools für Entwickler und Nutzer zu implementieren, z.B.:

- * eine Mobile App für die Fernsteuerung von Servomotoren
- * eine Grafische Anwendung zur Visualisierung von gemessenen Werten durch den Smartservo
- * einen Bootloader, um das Updaten von Firmware zu vereinfachen

Jedes dieser Unterprojekte wurde begonnen, jedoch haben wir uns entschieden die Bibliotheken, Firmware und Dokumentation zu priorisieren. Daher waren die genannten optionalen Milestones zum Zeitpunkt der Abgabe noch nicht fertig.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Alle unsere Entwicklungen sind unter <https://github.com/suprememachines> zu finden.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Wie bereits beschrieben ergaben sich aus dem Zusammenspiel der verschiedenen Komponenten zahlreiche Probleme - im Speziellen war das Verhalten der Hardware anders als erwartet und führte somit zu Schwierigkeiten.

Auch wenn diese Probleme zu erwarten waren und wir in der Planungsphase für ihre Lösung Zeit und Ressourcen eingeplant hatten, war unsere Schätzung etwas knapp bemessen. Um die Fertigstellung der wichtigsten Komponenten rechtzeitig zu schaffen, mussten wir die Arbeit an einigen optionalen Unterprojekten reduzieren, so dass diese zum Zeitpunkt der Abgabe noch nicht fertig sind.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Nein gab es nicht.

Richtlinie zum „Software-Sprint“

MeineLuftdaten – Meine.Luftdaten.info

Schlussbericht

Zuwendungsempfänger:

Meine.Luftdaten.info GbR
Timo Lundelius / Geschäftsführer
c/o We-Build.City GmbH
Herlingsburg 16, 22529 Hamburg

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen **01IS18S36** gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei Timo Lundelius.

Kurze Darstellung der Aufgabenstellung und Motivation

Was war Deine Motivation? Welches Problem wolltest Du mit Deinem Projekt lösen? Wie war die geplante Vorgehensweise zur Problemlösung (auch Angabe der wichtigsten Meilensteine)?

Das Feinstaub-Sensornetzwerk Luftdaten.info ist als zivilgesellschaftliches, ehrenamtliches Projekt gestartet. Unsere Motivation in diesem Antrag war es, die seit Jahren aufgebauten Datenschätze und die alle 3 Minuten eintreffenden Echtzeitwerte alle Bürgern weltweit und persönlich zugänglich zu mache.

Das Problem war vorher, dass nur Datenspezialisten die gesammelten Messwerte für sich speziell zugeschnitten nutzen konnten. Also z.B. wie ist die Feinstaubbelastung bei mir, um meinen Arbeitsplatz herum.

Um das Problem zu lösen war die geplante Vorgehensweise, nach einer intensiven Konzept- und Designphase diese dann auf User-Experience und User Interface Kriterien mit Nutzern „vor“-zu testen um dann die guten Ergebnisse auch zu programmieren.

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

Wer ist die Zielgruppe für Deine Lösung? Wie profitiert sie von den Ergebnissen Deines Projekts? Welche Bezüge gibt es zu den Themenfeldern und Zielen des Software Sprints?

Die Zielgruppe für unsere Lösung ist der/die „Ottonormal“-VerbraucherIn. Der Informations-Abo-Service von Luftdaten.info richtet sich an alle Menschen, die wissen wollen, wann

in ihrer Nähe die Grenzwerte für Feinstaub überschritten werden. Die Nutzer dieses Services profitieren davon, dass sie personalisierte "Warnungen" oder Hinweise über Feinstaub-Grenzwert-Überschreitungen direkt auf ihr Smartphone / Computer bekommen.

Der größte Bezug zum Software Sprint ist der "Open"-Ansatz. Das Luftdaten.info Projekt basiert fast vollständig auf Open Source Software, veröffentlicht ausschließlich Open Data und nutzt sehr viel Open Hardware.

Ausführliche Darstellung der Ergebnisse

Welche konkreten Ergebnisse hast Du erzielt? Konnten alle Meilensteine erreicht werden? Welche zusätzlichen Erkenntnisse hast Du aus der Projektarbeit gewonnen, auch im Hinblick auf die Begleitung durch die Open Knowledge Foundation?

Wir haben einen voll funktionsfähigen Prototypen unter app.luftdaten.info veröffentlicht und testen nun mit ca. 50 Testnutzern die Funktionalitäten.

Ja, es konnten alle Meilensteine erreicht werden. Allerdings konnten wir nicht den Qualitätsanspruch, den wir selbst an uns hatten, erfüllen. Dies betraf insbesondere die erste Phase "Konzeption" und User-Tests. Durch die Abwesenheit von klaren, disziplinären Entscheidungsstrukturen konnte gegen die oben genannte Entwicklung nicht ausreichend gegengesteuert werden.

Größte Erkenntnis war, wie aufwendig der Kommunikationsbedarf zwischen überregionalen Teams ist und wo die Grenzen zwischen ehrenamtlichem Engagement und teil-hauptberuflichem Betrieben von Open-Projekten liegt.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Welcher Nutzen ergibt sich für die Zielgruppe aus den Ergebnissen Deines Projekts? Welche weitergehenden Effekte ergeben sich aus der Open-Source-Stellung der Ergebnisse? Gibt es Ideen für die Weiterentwicklung Deiner Lösung und Pläne zu deren Umsetzung?

Ja, nach dem Projekt gibt es deutlich klarere Vorstellungen darüber welche Konzepte eine erfolgreiche Weiterarbeit erlauben und wie diese umgesetzt werden können. Darüber hinaus wurde auch deutlicher welche Ziele und Ideen ohne hauptamtliche Strukturen nicht umsetzbar sind.

Hat die Arbeit in dem Projekt Dich in Deiner persönlichen, fachlichen Weiterentwicklung unterstützt?

Teilen des Teams hat die Arbeit an dem Projekt große fachliche und persönliche Weiterentwicklungen gebracht. Insbesondere die Erfahrung über das Spannungsfeld zwischen Ehrenarbeit und hauptamtlicher Kollaborations-Arbeit.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Gab es Arbeiten bzw. Lösungsansätze, die nicht weiter verfolgt wurden? Was waren die Hintergründe, und wie bist Du alternativ vorgegangen?

Keine.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Wo können sich Interessenten detailliert über Deine Projektergebnisse informieren (z.B. Webseite, GitHub, Veröffentlichungen)?

Bürger-App mit Feinstaub-Abo-Service: <https://app.luftdaten.info/>

Quellcode: <https://github.com/luftdaten>

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Gab es im Projektverlauf Ereignisse, die eine Anpassung der Planung erforderlich machten – z.B. Mehr- oder Minderaufwand bei der Bearbeitung von Teilaufgaben?

Da unser Team aus 4 Personen bestand, war die Verteilung des Budgets auf 6 Monate knapp und erlaubte keine Vollzeit-Arbeit an dem Projekt, was zu Lasten der Fokussierung und Konzentration auf das Projekt führte. Eine höhere Fördersumme, ggf. angepasst an die Aufwände und Teammitglieder wäre hier wünschenswert.

Durch das Prinzip der quartalsweisen, nachträglichen Kostenerstattung war es für einzelne Teammitglieder immer wieder eine starke Herausforderung die notwendige Liquidität aufzubringen.

Wünschenswert wäre hier eine (zumindest) teilweise Vorfinanzierung von Projektmitteln da unser Team aus Privatpersonen besteht und nicht z.B. durch öffentliche Institutionen die nötige Mittelversorgung hatte.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen]

Gab es Entwicklungen anderer Personen oder Institutionen, die Einfluss auf Deine Arbeiten und die Zielsetzung hatten? Wenn ja, worin bestand dieser und wie bist Du damit umgegangen?

Nein.

Richtlinie zum „Software-Sprint“

ssb-photos

Schlussbericht

Zuwendungsempfänger:

Bölsche, Jan

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S37 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Internetplattformen spielen heute eine zentrale Rolle im Leben vieler Menschen. Die Nutzung von Facebook, twitter, WhatsApp und Instagram ist zum Alltag geworden und stellt in manchen Regionen der Welt das wichtigste Kommunikationsmedium zwischen Freunden und Familienmitgliedern dar. Ein Problem besteht darin, dass Anwender häufig nicht wissen, was mit ihren Daten geschieht; ob sie ausreichend vor dem Zugriff dritter geschützt sind, oder an wen der Plattformanbieter die persönlichen Daten verkauft.

Darüber hinaus vertritt die Benutzeroberfläche dieser Plattform häufig nicht die Interessen der Nutzer, sondern die der Anbieter. So kann man sich auf Facebook beispielsweise nicht sicher sein, ob ein Posting wirklich

alle Freunde erreicht, denn ein Algorithmus entscheidet, wer was zu lesen bekommt.

Fotos spielen eine zentrale Rolle bei dieser Kommunikation. Mit ssb-photos versuche ich eine Alternative zu den etablierten Plattformen anzubieten, die einen direkten Austausch der Daten zwischen den Nutzern ermöglicht, deren Algorithmen transparent sind und die im Interesse ihrer Nutzer agiert.

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

Es ist ein menschliches Grundbedürfnis, Momente in Form von Fotos festzuhalten und Menschen teilhaben zu lassen, die nicht vor Ort sein können. Ssb-photos will Nutzern, die derzeit eine zentralisierte Internetplattform für den Austausch dieser Fotos nutzen, eine Alternative anbieten, die ihre Privatsphäre schützt und von den Benutzern selber betrieben werden kann.

In einer späteren Ausbaustufe sollen Bildbearbeitungsfunktionen hinzu kommen, die auch professionelle Fotografen ansprechen, denn diese Berufsgruppe profitiert besonders von der Echtzeitfähigkeit von ssb-photos.(Änderungen an Entwicklungsparametern können live von Zuschauern verfolgt werden)

Ausführliche Darstellung der Ergebnisse

Im Rahmen des Software Sprints ist eine Applikation für Mac OS entstanden, die es erlaubt, Fotos per Drag and Drop von einer Kamera oder aus dem Dateisystem zum importieren. Es existiert ein Tool, mithilfe

dessen ein Nutzer einen Einladungscode für die eigene Fotobibliothek erstellen kann. Dieser Einladungscode kann z.B. per email an Freunde oder Familienmitglieder versandt werden. Wendet man eine solchen Invite-Code an, erhält man dieselbe Ansicht wie der einladende Benutzer. Alle Beteiligten können nun die Bilder in eine Baumstruktur anordnen. Es können neue “Ordner” (ähnlich wie Fotoalben, aber flexibler) angelegt werden. Nutzer können die Metadaten der Bilder einsehen und ändern. Diese Metadaten umfassen sowohl von der Kamera erfasste Daten (Aufnahmedatum und ggf. -ort, Belichtungsdauer, ISO-Wert, verwendetes Objekt etc), als auch einen frei wählbaren Titel. Die dezentrale Datenbank ist nicht destruktiv, d.h. Daten werden nur angehängt, nicht geändert. Jedes Bild hat somit eine Historie. Es ist nachvollziehbar, wer wann welche Änderung vorgenommen hat, und jede Änderung kann zurück genommen werden. Änderungen, die nicht vom Inhaber der Bildbibliothek stammen, werden als “change request” vermerkt, per default aber angezeigt. In einer späteren Ausbaustufe können Nutzer entscheiden, ob sie nur die Fassung des ursprünglichen Autoren sehen möchten, oder auch die der anderen Nutzer.

Jede Änderung wird sofort sichtbar und ist innerhalb von wenigen Sekunden bei allen Beteiligten des sozialen Netzes zu sehen. Diese Echtzeitfähigkeit erleichtert das kollaborative Einordnen grosser Fotomengen, etwa nach einem gemeinsamen Urlaub.

Professionelle Fotografen können ihre Kunden in die Bildbibliothek einladen, die während eines Foto-Shootings entstanden ist. Fotograf und Kunde können dann gemeinsam eine Sortierung der Bilder vornehmen während sie beispielsweise am Telefon sprechen. Diese Echtzeitfunktion ist auf der Strecke Berlin – Wellington (über 18 Tausend Kilometer) getestet und die Latenz ist gering genug für diesen Anwendungsfall.

Ein Modul ist entstanden, das die Historie eines Bildes wie einen U-Bahn Plan (“Perlenschnur”) darstellt. Wird ein Bild verändert, entsteht

automatisch eine neue Version (ein Punkt auf dem Diagram). Jede Version kann nun wiederum verändert werden. Basiert mehr als eine Version auf derselben Vorgängerversion, entsteht eine Abzweigung (engl. Fork). Dann konkurrieren zwei Version darum, die aktuelle Variante sein. Dieser Baum an Varianten ist sehr hilfreich im kreativen Prozess. Dieses Modul ist unter Einfluss einer durch die Open Knowledge Foundation vermittelten UI-Beratung entstanden, die sehr inspirierend war. Es muss allerdings nicht in die Applikation integriert werden.

Weitere Features, die im gegebenen Zeitrahmen noch nicht umgesetzt werden konnten sind:

- eine Portierung auf Microsoft Windows und Linux
- Bildbearbeitungsmöglichkeiten wie Belichtung, Sättigung, Ausschnitt

Die Software ist sehr modular aufgebaut und besteht bereits aus über 50 Modulen (Repositories auf GitHub und NodeJS Module auf npmjs.com). Mein Hoffnung ist, so eine Basis geschaffen zu haben, die es anderen Entwicklern sehr erleichtert, andere Medientypen auf ähnliche Weise zu dezentralisieren. Denkbar ist eine Videoschnittsoftware, ein Tools zur Untertitelung von Videos, eine Umgebung, in der Autoren und Lektorat gemeinsam an einem Buchtext arbeiten können. All diese Anwendungen können von vielen der 50 Module profitieren.

Ich selber plane viele dieser Module zu Nutzen, um einen geteilten Kalender für Eltern von Kleinkindern zu entwickeln. In diesem Bereich verdient die Privatsphäre besonderen Schutz und der Austausch der Daten in Echtzeit erleichtert die Organisation von Arztbesuchen, Kitaaufenthalten etc.

Hat die Arbeit in dem Projekt Dich in Deiner persönlichen, fachlichen Weiterentwicklung unterstützt?

Es war besonders hilfreich, genug finanziellen und damit zeitlichen Spielraum zu haben, die Dinge so gut umzusetzen, wie ich es derzeit in der Lage bin. Der Lerneffekt bei der Arbeit an Projekten, die durch ihr enges Budgetvorgaben und strikten Deadlines kein qualitativ hochwertiges Arbeiten möglich machen, ist eher gering. Ein Lerneffekt stellt sich erst ein, wenn ich die Freiheit habe, jedes Detail auszufeuern, und für meine Begriffe perfekt umzusetzen. Dann nämlich ergeben sich Überraschungen; erst dann werde ich mit Aspekten konfrontiert, die mir neu sind und die ich nicht ausreichend durchdacht hatte. Nur so kann ich als Softwareentwickler an einer Aufgabe wachsen. Und ich meine, dass das in diesem Projekt sehr gut funktioniert.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben. Gab es Arbeiten bzw. Lösungsansätze, die nicht weiter verfolgt wurden? Was waren die Hintergründe, und wie bist Du alternativ vorgegangen?

Ganz zu Anfang der Förderphase musste ich die Entscheidung fällen, welche Version der zugrundeliegenden Software “secure scuttlebutt” (ssh) ich verwenden würde. Eine Besonderheit von ssb-photos ist, dass es –

ähnlich wie das ebenfalls in dieser Runde geförderte Projekt “Blockparty” – beliebig viele, voneinander isolierte soziale Netzwerke erlaubt. Diese Funktion ist zwar grundsätzlich in ssb vorhanden, wird aber von keiner anderen Applikation derzeit genutzt, weshalb sie kaum getestet ist. Ich wusste aus Erfahrung, dass eine ältere Version von ssb mit mehreren Netzen funktioniert, und ein schneller Test ergab, dass die aktuelle Version dies nicht mehr tut. Ich hatte also zu entscheiden, ob ich die alte Version als Basis verwenden, oder mich daran mache, die neue Version soweit zu reparieren, dass ich sie für ssb-photos nutzen kann. Da auch andere davon profitieren würden, und weil ich schon in der Vergangenheit mit den ssb-Entwicklern zusammen gearbeitet hatte, entschied ich mich dafür, die aktuelle Version zu reparieren.

Weil ssb ebenfalls sehr stark modularisiert ist, erstreckten sich meine Arbeiten über fünf oder sechs GitHub-Repositories und die diverse Pull-Requests haben nur im Bezug aufeinander Sinn gemacht. Ich musste feststellen, dass es durch diese Fragmentierung sehr schwer fällt, eine Änderung erfolgreich zu kommunizieren. Dazu kam, dass andere Entwicklungen parallel dazu stattfanden, sodass meine Änderungen nicht mehr auf den nun geänderten ssb Code passten. Ich habe in dieser Zeit sehr viel kommuniziert, auf Probleme aufmerksam gemacht Korrekturen an Code und Dokumentation eingebracht und auch neue ssb Module geschrieben. Viele dieser Dinge sind heute in den ssb Repositories übernommen, aber ungefähr die Hälfte wurde von anderen ssb-Entwicklern auf andere Art und Weise umgesetzt, sodass es mir nie gelang, eine Multi-Netzwerk-kompatible, aktuelle Version von ssb zu erhalten. Ich habe dieses Vorhaben nach fünf Wochen aufgegeben und stattdessen ein paar kleinere Bugs an einer älteren Version korrigiert und konnte dann endlich damit beginnen, am Applikationscode zu arbeiten.

Ich hatte später die Gelegenheit mit einigen der beteiligten Entwickler in Neuseeland zu sprechen, und mir wurde versichert, dass es einigen so ergangen war wie mir.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Ein Screenshot und eine Installationsanleitung, sowie eine Anleitung zum Einladen neuer Nutzer, ist hier zu finden:

<https://github.com/regular/ssb-photos>

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Gab es im Projektverlauf Ereignisse, die eine Anpassung der Planung erforderlich machten – z.B. Mehr- oder Minderaufwand bei der Bearbeitung von Teilaufgaben?

Nein.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Um zu vermeiden, die Bemühungen des Projektes “Blockparty” (ebenfalls in dieser Prototypefund-Runde) zu duplizieren, habe ich Funktionen ausgeklammert, die durch dieses Projekt abgedeckt werden. Insbesondere eine benutzerfreundliche UI zum Erstellen eines neuen ssb Netzwerkes und Aufsetzen eines sogenannten “pub servers”, sowie dem Erzeugen von Invite codes. All diese Funktionen sind in ssb-photos zwar vorhanden,

erfordern aber Kenntnisse der Unix Kommandozeile. Zu einem späteren Zeitpunkt hoffe ich, ein freundliches, grafisches User-Interface, das im Rahmen von Blockparty entwickelt wird, in ssb-photos zu integrieren.

Richtlinie zum „Software-Sprint“

Dingsda – ein echtes Internet der Dinge

Schlussbericht

Zuwendungsempfänger:

Jan Philip Steimel

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen **01IS18S38** gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Viele Freiberufler*innen, Institutionen, Privatmenschen und kleine Firmen (z.B. Theaterkompanien wie "machina eX") sammeln über die Jahre eine Unmenge von Equipment, Werkzeugen, Kostümteilen und anderweitig nützlicher Produktionsmittel an. Sie verstauben in Lagerräumen und werden ein, zwei mal im Jahr genutzt - Während jüngere, unterprivilegierte Menschen, Gründer*innen, Studierende versuchen ihre Ziele zu verfolgen dafür aber ihre Hände an die Produktionsmittel bekommen müssen. Oft leihen wir untereinander, aber dafür wissen wir selten wer was eigentlich hat. Dingsda soll dem Abhilfe schaffen und ein Set an digitalen Tools zur Verfügung stellen mit dem sich kleine Firmen unkompliziert miteinander vernetzen können und/oder ihr Material/Lagerbestände inventarisieren. Dazu wollte ich dingsda als nodejs Server aufbauen, ein Datenbank-Design entwerfen, eine API und ein Framework aus Helferskripten konstruieren und schließlich alles in einer Benutzerfreundlichen WebApp zusammenführen um schließlich mit ersten Tester*innen aus der Theaterszene auf einer Testinstanz zu beginnen die genauen Benutzungsweisen und Abläufe zu testen.

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

Dingsda richtet sich an kleine Firmen, Künstler*innen, Kollektive, die keine Möglichkeit haben mit großen komplexen und teuren Inventarisierungslösungen zu arbeiten, aber in ihrem Alltagsgeschäft regelmäßig mit physischen Produktionsmitteln operieren. Dabei leistet Dingsda einen Beitrag zur Nachhaltigkeit (da Produktionsmittel so effizienter und Ressourcen schonender benutzt werden und überflüssige Anschaffungen vermieden werden), und zur Wirtschaftlichkeit kleinerer Unternehmen (durch die Vernetzung von Produktionsmitteln untereinander ermächtigen sich kleine Startups, Firmen und Freiberufler*innen gegenseitig)

Ausführliche Darstellung der Ergebnisse

Welche konkreten Ergebnisse hast Du erzielt? Konnten alle Meilensteine erreicht werden? Welche zusätzlichen Erkenntnisse hast Du aus der Projektarbeit gewonnen, auch im Hinblick auf die Begleitung durch die Open Knowledge Foundation?

Durch die Förderung im Rahmen der Initiative Software-Sprint konnte ich Dingsda weit genug entwickeln um es noch während der letzten Wochen der Förderphase mit ersten Test-Benutzer*innen (zwei Theaterkompanies und verschiedene Einzelpersonen) auf einem Testserver in Gebrauch zu nehmen.

Die meisten Meilensteine wurden erreicht. Auf dem Weg musste ich diverse Modifikationen an den Projektzielen vornehmen um Dingsda als offene, erweitertere Codebasis zu entwickeln:

Es wurde eine API entwickelt, die verschiedenste Transaktionen von Gegenständen und Inventarisierungen ermöglicht (Hinzufügen, Modifizieren, Ausleihanfragen, Verleih, Rückgabe, Suche, Geosuche, Userverwaltung etc.).

Desweiteren wurde ein Datenbankdesign für Non-SQL Datenbanken auf Basis des Datenbank Managementsystems CouchDB entwickelt das auch auf andere Datenbanksysteme übertragen werden könnte.

Schließlich wurden API und Datenbank zusammengeführt und in einem beispielhaften funktionierenden WebClient für die direkte Nutzung bereit gemacht.

Auf Basis des veröffentlichten Codes ist es möglich eigene Server Instanzen mit Datenbank, API und WebClient aufzusetzen und nach Belieben zu modifizieren.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Durch die Testrunden verbreitete sich die Idee von dingsda in den Kreisen in denen ich gehofft hatte Anschluss zu finden: Eine größere Theaterkompanie beauftragte mich im Anschluss an den Förderzeitraum spezielle Plug-Ins für ihre Nutzungsbeispiele in das von mir entwickelte System hinzuzufügen und plant es in der nahen Zukunft für alle ihre Projekte einzubinden. Zwei weitere Theaterkompanies benutzen dingsda bereits seit der ersten Testphasen und führten verschiedenste Equipemt-Verleihvorgänge aus.

Dabei konnte ich bereits erste Kooperationspartner*innen gewinnen, die im git erste Tickets hinterließen mit Verbesserungsvorschlägen und Wünschen für die weitere Entwicklung von Dingsda. Mein Plan sieht vor Dingsda als Open Source Projekt weiter zu entwickeln und überdies erste Monetarisierungsmöglichkeiten für Services auf Basis des Dingsda Projekts zu entwickeln. Ersteinmal sehe ich aber ein langsames organisches Wachstum der Codebase und weitere Test- und Optimierungsrunden mit der bereits laufenden Plattform vor.

Ich konnte mich durch die Arbeit an dem Projekt als Designer und Developer enorm fachlich und persönlich weiterentwickeln: Die Lerneffekte die sich aus der Arbeit an Dingsda ergaben, führten dazu, dass ich persönlich und professionell wachsen, mir neue Fähigkeiten und Wissen aneignete und auf deren Basis das Projekt weiter vorantreiben konnte. Das wird sich sehr wahrscheinlich auch auf den weiteren Verlauf meiner Karriere auswirken.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Das Ziel ein vollständiges Framework zu bauen, mit dem man flexibel verschiedene Leih- und Inventarisierungssysteme erstellen kann, musste ich soweit modifizieren, dass die konkrete Arbeit an den funktionalen Teilen (API, Datenbank, Client) relevanter wurde als die abstraktere Arbeit an Bibliotheken und Konventionen für das Framework: Die dabei gewonnen Erkenntnisse mündeten allerdings durchaus in etwas, das nach ein paar Rekursionen tatsächlich zu einem Framework aus Client-Bibliothek(en), Notations-, API- und Transaktions- Konventionen wachsen könnte. Das war nur durch intensives Testen mit Benutzerinnen in einem vollständigen Testsystem mit UIs und konkreten Tools möglich und daher anders zu gewichten als ich das zu Beginn vorsah.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Die Website www.dingsda.org erklärt die Projektziele und kostet unter <https://dingsda.org/two> den ersten Dingsda Testserver auf dem bereits erste Companies und Menschen Dinge tauschen und verwalten.

Wer sich lieber selbst einen Server aufsetzten will, kann auch unter <https://bitbucket.org/machinaeXphilip/dingsda/> im git alles dafür finden. Dort findet sich auch bereits die erste Version der Dokumentation, die auch nach Projektende noch weiter wachsen sollte.

Bei Interesse an der Plattform können sich interessierte Menschen auch gerne direkt an info@philipsteimel.de wenden.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Keine wesentliche Änderung gegenüber der ursprünglichen Planung des Projekts

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Ich hatte Kontakt mit zwei weiteren Software Projekten die sich mit sharing auseinander setzen aber andere Use-Cases abdecken als Dingsda. Mit beiden bin ich weiterhin in Kontakt und mit mindestens einer Initiative (Common Bookings) plane ich vonseiten des API-Designs in der Zukunft zu kooperieren..

Richtlinie zum „Software-Sprint“

OpenWrt Test

Schlussbericht

Zuwendungsempfänger:

Alexander Couzens

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen **01IS18S39** gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Was war Deine Motivation? Welches Problem wolltest Du mit Deinem Projekt lösen? Wie war die geplante Vorgehensweise zur Problemlösung?

Auf Millionen von Wlanroutern, Accesspoints sowie IoT Geräten wird eine Version des Betriebssystem OpenWrt benutzt. Neben kommerziellen Benutzern gibt es viele freie Software Projekte die OpenWrt als Basis benutzen, wie z.B. die freifunk Community oder der onion Router. OpenWrt ist eines der wenige Betriebssysteme, welche für Geräte mit 4MiB Flash (vgl. Festplattenspeicher) sowie 32 MiB Arbeitsspeicher zu laufen. Dabei unterstützt OpenWrt ca. 1000 verschiedene Geräte. Für die Softwareentwicklung ist es wichtig, neue Softwareversionen zu testen, sodass sichergestellt wird, dass diese korrekt funktionieren.

In dem Förderungsprojekt wurde ein automatisches Testsystem für OpenWrt entwickelt, welches die speziellen Anforderung der hardwarenahen Entwicklung entspricht.

Ziele

- 10 Testcases
- 10 unterschiedliche Geräte integriert
- Feedback von Entwicklern und Maintainern
- "How to setup LAVA" und "How to create a Testsetup?"

- Vortrag/Workshop zu den OpenWrt Tests
- Webseite zur Übersicht der Tests
- Grafische Auswertung von Tests

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

Wer ist die Zielgruppe für Deine Lösung? Wie profitiert sie von den Ergebnissen Deines Projekts? Welche Bezüge gibt es zu den Themenfeldern und Zielen des Software Sprints?

Das Testsystem ermöglicht eine schnellere Entwicklung, dadurch können mehr Geräte mit weniger Aufwand unterstützt werden. Durch die zusätzlichen unterstützten Geräte kann auch alte Hardware länger betrieben werden, weil diese durch OpenWrt wieder aktuelle und sichere Software mit Sicherheitsupdates erhält. Dies trägt zur Vermeidung des Elektroschrottes bei.

Zusätzlich können auch andere Projekte, wie z.B. das freifunk Projekt, das Testsystem nutzen um ihre Software und Anpassung an OpenWrt zu testen.

Ausführliche Darstellung der Ergebnisse

Welche konkreten Ergebnisse hast Du erzielt? Konnten alle Meilensteine erreicht werden? Welche zusätzlichen Erkenntnisse hast Du aus der Projektarbeit gewonnen, auch im Hinblick auf die Begleitung durch die Open Knowledge Foundation?

Es wurden mehr als 4000 Testcases ausgeführt auf zur Zeit 4 aktiven Geräten. Bei 4 weiteren Geräten muss der Bootloader modifiziert werden, um diese in das System einzupflegen. Durch den verschobenen Start des Projektes konnte das Projekt nicht wie geplant auf der Entwicklerkonferenz OpenWrt-Summit in Lissabon (<http://openwrtsummit.org/>) vorgestellt werden, sondern nur ein Konzept mit den Entwicklern und Maintainern besprochen werden und so Feedback eingeholt werden. Ein Vortrag auf der nächsten Konferenz ist geplant. Eine Auswertung der automatischen Tests steht noch bevor. Eine feste Integration in den Entwicklungszyklus in das Projekt soll auch beim nächsten Entwicklertreffen besprochen werden.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Welcher Nutzen ergibt sich für die Zielgruppe aus den Ergebnissen Deines Projekts? Welche weitergehenden Effekte ergeben sich aus der Open-Source-Stellung der Ergebnisse? Gibt es Ideen für die Weiterentwicklung Deiner Lösung und Pläne zu deren Umsetzung?

Das automatische Testsystem kann die Entwicklungsgeschwindigkeit fördern sowie die Qualität, da frühzeitig und automatisch verschieden Geräte getestet werden. Zusätzlich können Entwickler Zugang zu den Systemen erhalten, um neue Treiber zu entwickeln. Es ist außerdem geplant, alle verschiedenen

Testsysteme (wie z.B. unittest, build tests) in einer gemeinsamen Webseite übersichtlich darzustellen.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Wo können sich Interessenten detailliert über Deine Projektergebnisse informieren (z.B. Webseite, GitHub, Veröffentlichungen)?

Die komplette Entwicklung befindet sich im Gitlab Projekt unter <https://code.fe80.eu/openwrt-lava> sowie unter <https://lava.fe80.eu/>

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Gab es im Projektverlauf Ereignisse, die eine Anpassung der Planung erforderlich machten - z.B. Mehr- oder Minderaufwand bei der Bearbeitung von Teilaufgaben?

Die Planung hätte kleinere Meilensteine vorgeben sollen und mehr zeitliche Reserve einplanen sollen. Aufgrund dessen wurden viele Ziele nicht erfüllt, sondern nur zum Teil angegangen, um die weiteren Ziele nicht aus den Augen zu verlieren. Es wird jedoch weiter an dem Projekt gearbeitet, sodass geplant ist die fehlenden Ziele 2019 noch zu erfüllen.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Gab es Entwicklungen anderer Personen oder Institutionen, die Einfluss auf Deine Arbeiten und die Zielsetzung hatten? Wenn ja, worin bestand dieser und wie bist Du damit umgegangen?

Es gab Austausch mit dem Projekt boardfarm sowie mit anderen Firmen, die Ihre eigene (Closed-Source) Testumgebungen benutzen. Dabei wurden die unterschiedlichen Anforderungen und Umsetzungen besprochen. Sowie die Vor- und Nachteile der unterschiedlichen Ansätze.

Des weiteren gab es auch einen Austausch zwischen anderen OpenSource Hardware Testsystemen, welche nicht OpenWrt als Ziel haben. Sowie der Entwicklergemeinschaft rund um das Softwareprojekt LAVA, auf dessen Basis dieses Projekt realisiert wurde.

Open Legal Data

Schlussbericht

Zuwendungsempfänger:

Open Legal Data - Ostendorff Blume Langhabel GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S40 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Was war Deine Motivation? Welches Problem wolltest Du mit Deinem Projekt lösen? Wie war die geplante Vorgehensweise zur Problemlösung (auch Angabe der wichtigsten Meilensteine)?

Open Legal Data ist eine freie Plattform, die juristische Inhalte der Öffentlichkeit nachhaltig so zur Verfügung stellt, dass sie leicht sowohl von Personen als auch von Maschinen genutzt werden können. Dazu bedienen wir uns neuesten Verfahren der automatischen Aufbereitung und Erschließung von öffentlichen Urteilen und Rechtsnormen. Somit werden nicht nur die Dokumente zugänglich gemacht, sondern auch die darin enthaltenen Information. Dadurch soll der Rechtsstaat transparenter werden und der Umgang mit Rechtsthemen für Nicht-Juristen erleichtert werden, wobei wir zunächst technisch versierte Bürger ansprechen. Das Projekt verpflichtet sich dabei den Prinzipien der Open Data Bewegung.

Ein Urteil wird im Namen des Volkes gesprochen, ist damit Ausdruck des Demokratieprinzips (Art. 20 GG) und nicht urheberrechtlich geschützt. Dennoch sind viele Urteile nur unzureichend für die Allgemeinheit zugänglich. Dies führt dazu, dass die Akzeptanz der Justiz abnimmt und Unverständnis für verschiedene Urteile herrscht. Neben der Unzugänglichkeit fehlt eine überzeugende und transparente amtliche Statistik der Rechtsprechung. Dies wird u.a. auch beim Deutschen Juristentag thematisiert und kritisiert. Mit unserem Projekt machen wir die Rechtsprechung für alle zugänglich und mit Hilfe von unabhängigen und auf Open Data basierten Statistiken die Rechtsprechung transparenter.

Der Kern des Projekts ist eine Website inkl. API. Über die Website werden alle Inhalte optisch ansprechend und übersichtlich dargestellt. Zusätzlich ermöglicht eine Suchfunktion das Finden von relevanten Inhalten. Neben der Website wird der Austausch von juristischen Daten vor allem durch eine API ermöglicht. Über die API sind nicht nur alle Inhalte in maschinenlesbarer Form abrufbar, sodass sie von anderen Diensten genutzt werden können, sondern wir bieten auch eine Möglichkeit, Inhalte mit Zusatzinformationen, wie Erklärungen und Streitwerte, anzureichern (Open Annotations). Das Projekt ist in Python geschrieben mit dem Django Framework einschließlich REST-API Addon als Grundgerüst. Elasticsearch wird für die Suchfunktion benutzt. Als Datenbank ist MySQL im Einsatz. Des Weiteren stellen wir API Clients und Jupyter Notebooks mit direktem Zugriff auf alle Daten zur Verfügung, um weiterführende Analysen zu erleichtern.

Für die Umsetzung des Projektes haben wir uns fünf Meilensteine gesetzt:

- 1) Open Source Veröffentlichung mit angemessener Codequalität sowie integrierten Test und grundlegender Dokumentation
- 2) Vollständige Verfügbarkeit von deutschen und europäischen Rechtsnormen und Verordnungen
- 3) Über 100.000 Urteilstexte inklusive Metadaten von deutschen und europäischen Gerichtshöfen (darunter EuGH, BGH, BVerfG, BVerwG) sind maschinenlesbar über die API abrufbar
- 4) Eine öffentlich konfigurierbare Data Processing Pipeline, die es externen Entwicklern ermöglicht gezielt Informationen aus den Daten zu extrahieren, z.B. Streitwert oder verwendete Rechtsnormen
- 5) Veröffentlichung eigener Analysen

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

Wer ist die Zielgruppe für Deine Lösung? Wie profitiert sie von den Ergebnissen Deines Projekts? Welche Bezüge gibt es zu den Themenfeldern und Zielen des Software Sprints?

Die primäre Zielgruppe von Open Legal Data sind interessierte Aktivisten, Journalisten und Entwickler, die mit unserer Hilfe Analysen, Artikel, Apps etc. erstellen können. Um unsere Zielgruppen zu erreichen, haben wir u.a. Open Legal Data beim 35. Chaos Communication Congress¹ vorgestellt. Mit der Veröffentlichung eines Datensatz von Gerichtsurteilen haben wir den Global Legal Tech Hackathon² unterstützt. Mit der Hilfe unserer Daten gewann das Team VizLaw den Legal Tech Hackathon in Berlin³. Auf diese Weise konnten wir den Nutzen von Open Source und Open Data für den juristischen Bereich demonstrieren und nachhaltig eine Diskussion auch über die Grenzen der „Open Data-Szene“ hinaus anstoßen.

Ausführliche Darstellung der Ergebnisse

Welche konkreten Ergebnisse hast Du erzielt? Konnten alle Meilensteine erreicht werden? Welche zusätzlichen Erkenntnisse hast Du aus der Projektarbeit gewonnen, auch im Hinblick auf die Begleitung durch die Open Knowledge Foundation?

Durch die Förderung konnten wir aus einer Idee im Laufe des Projektzeitraums einen nutzbaren Prototypen entwickeln. Nahe zu alle selbstgesteckten Meilensteine konnten erreicht werden. Der gesamte Quellcode der Open Legal Data Plattform ist auf GitHub unter der freien MIT Lizenz veröffentlicht⁴ (MS1). Alle dazugehörigen Module, wie Zitaterkennung oder Named Entity Recognition, werden ebenfalls zur Verfügung gestellt. Über die Plattform können nicht nur Inhalte abgerufen werden. Unser Annotation API (Data Processing Pipeline) ermöglicht es Dritten, die auf der Plattform enthalten Dokumente mit zusätzlichen Informationen anzureichern (MS4). Wie diese

¹ #35C3 Open Legal Data - Lighting Talk, <https://www.youtube.com/watch?v=5KHcXCmqfiE>

² Global Legal Hackathon 2019, <https://globallegalhackathon.com/>

³ VizLaw uses our API and wins Berlin Legal Tech Hackathon 2019, <http://openlegaldata.io/events/2019/02/27/vizlaw-wins-berlin-legal-tech-hackathon.html>

⁴ GitHub – Open Legal Data Plattform, <https://github.com/openlegaldata/oldp>

Funktion genutzt werden kann, demonstrieren wir beispielhaft mit Streitwerten⁵. Auch das im Rahmen des Software Sprints geschaffene Named Entity Recognition Modul spielt als Schritt der Data Processing Pipeline Daten über die API in die Datenbank ein. Das inhaltliche Angebot konnten wir auf mehr als 250.000 juristische Dokumente ausbauen. Während des Projektzeitraums haben wir uns inhaltlich auf Urteilstexte fokussiert, sodass MS3 mit mehr als 150.000 Urteilen übertroffen werden konnte. Bezüglich Gesetzestexte, konnte MS2 nicht vollständig erreicht werden, da die Gesetze der Länder bisher noch nicht über die Plattform abrufbar sind. Auf Basis der gesammelten Inhalte konnten wir bereits eigene Analysen, wie z.B. ein Zitatnetzwerk deutscher Gerichtsentscheidungen⁶, erstellen (MS5).

Insgesamt hat die Förderung uns ermöglicht große Fortschritte mit unserem Projekt zu machen. Insbesondere auch konnte die Kernwebsite von einem Ecosystem an Komponenten, Beispielen und Anwendungen umgeben werden. Durch die gemeinsamen Workshops und Coaching konnten wir uns professionalisieren und ein Netzwerk aufbauen, sodass wir auch langfristig das Projekt vorantreiben können.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Welcher Nutzen ergibt sich für die Zielgruppe aus den Ergebnissen Deines Projekts? Welche weitergehenden Effekte ergeben sich aus der Open-Source-Stellung der Ergebnisse? Gibt es Ideen für die Weiterentwicklung Deiner Lösung und Pläne zu deren Umsetzung? Hat die Arbeit in dem Projekt Dich in Deiner persönlichen, fachlichen Weiterentwicklung unterstützt?

Als eines der ersten Open Source Projekte im juristischen Bereich konnten wir nachhaltig für die Vorteile der Open Source Idee werben. Das Beispiel VizLaw zeigt wie Innovation durch Open Data gefördert werden kann. Um Open Legal Data auch langfristig betreiben zu können, sind wir bereits in Gesprächen, um das Projekt auch auf andere Länder zu erweitern. Über ein universitäres Forschungsprojekt soll zusätzlichen der wissenschaftliche Nutzen nachhaltig erhalten bleiben.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Gab es Arbeiten bzw. Lösungsansätze, die nicht weiter verfolgt wurden? Was waren die Hintergründe, und wie bist Du alternativ vorgegangen?

Es wurden keine geplanten Arbeiten gestoppt. Während des Förderzeitraums wurden lediglich einige Teilaufgaben anders priorisiert. Zum Beispiel haben wir uns auf ausschließlich Urteile als Inhalte fokussiert. Andere Dokumentarten, wie Gesetze, werden wir jedoch in Zukunft auch hinzufügen.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Wo können sich Interessenten detailliert über Deine Projektergebnisse informieren (z.B. Webseite, GHitHub, Veröffentlichungen)?

Die erste Anlaufstelle für mögliche Nutzer ist unsere Website, auf welcher alle Inhalte abgerufen werden können und wir über unsere Arbeit in einem Blog berichten. Neben der Website sind wir u.a.

⁵ Litigation Value Extraction, https://github.com/openlegaldata/oldp-notebooks/blob/master/notebooks/litigation_value.ipynb

⁶ Analysis: Citation Network of German Court Decisions, <http://openlegaldata.io/de/citation-network.html>

auf Twitter⁷ präsent, dort konnten wir unsere Followerzahl innerhalb des Förderzeitraums um 620% steigern. Für die Entwickler-Community sind alle unsere Arbeiten auf GitHub⁸ verfügbar.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Gab es im Projektverlauf Ereignisse, die eine Anpassung der Planung erforderlich machten – z.B. Mehr- oder Minderaufwand bei der Bearbeitung von Teilaufgaben?

Alle Teilaufgaben wurden wie geplant bearbeitet.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Gab es Entwicklungen anderer Personen oder Institutionen, die Einfluss auf Deine Arbeiten und die Zielsetzung hatten? Wenn ja, worin bestand dieser und wie bist Du damit umgegangen?

Die Veröffentlichung des Projektes OffeneGesetze.de im Dezember 2018 durch die Open Knowledge Foundation veranlasste uns dazu, den Schwerpunkt von Open Legal Data auf insbesondere Rechtsprechung zu verschieben.

⁷ <https://twitter.com/openlegaldata/>

⁸ <https://github.com/openlegaldata/>

Richtlinie zum „Software-Sprint“

arso – Archipel

Schlussbericht

Zuwendungsempfänger:

Nieß, Heinzmann, Brock-Wenzek u. Sievers GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S41 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Das Internet bietet viele Wege, Material zu veröffentlichen, doch ist dessen Erhalt und dauerhafte Zugänglichkeit keineswegs sicher. Oft landen Medien auf proprietären Plattformen, wo alles an deren Fortbestehen, Geschäftsinteresse und Zugangsoffenheit hängt. Ein mögliche Alternative ist die eigene Website, die aber Kosten hat, nur mit Aufwand gut kopierbar und bei temporären Unterfangen wie Konferenzen oder sozialen Bewegungen oft mit den Inhalten irgendwann wieder offline ist. Auch große Mediensammlungen sind davon z.B. bei auslaufender Finanzierung betroffen. Digitale Selbstbestimmung, die über den Moment hinaus reicht, braucht also Werkzeuge, um Dinge auf Dauer teilen und zugänglich machen zu können.

Grundsätzlich ist das System für unterschiedlichste Arten und Formen von Archiven einsetzbar. Unser Fokus wird auf bereits bestehenden Sammlungen und Archiven gemeinnütziger TrägerInnen liegen. Zu Beginn wird es dabei um bereits vorhandenes, öffentlich zugängliches Material gehen. Wird der NutzerInnen-Kreis über diese erste Schiene hinaus erweitert, werden wir auch auf datenschutzrechtliche Einschränkungen sowie auf das "Recht auf Vergessen"

aufmerksam machen. Technische Wege zur Depublizierung werden ebenfalls implementiert.

Wie:

Im Zentrum steht die archipel-app: Eine Desktop-App, um Archive zu erstellen, anzusehen und zu teilen. Archipel basiert auf Dat, einem progressiven Open-Source-Projekt. Dieses versionierte, dezentrale und sichere P2P-Dateisystem mit Datenbank ist unser Backend. Die Archipel-App wird in Javascript/Node "universal" geschrieben und läuft damit sowohl als offline-first P2P-App, als installierte Desktop-App (via Electron) sowie als Webapp an einem simplen Backend. Die App wird über verschiedene Plugins Inhalt hinzufügen (Medien importieren, ganze Websites archivieren, Twitter-Feeds inkl. verlinkter Medien auslesen) sowie grundlegende Metadaten erfassen können. Sie kann erstellte Archive über das Dat-Netzwerk veröffentlichen, außerdem andere Archive lokal spiegeln, teilen und betrachten. Vorhandene Open-Source-Tools (Hashbase, Database) können die Archive auf Servern spiegeln und für alle Browser zugänglich machen, diese werden wir geringfügig für unseren Usecase erweitern. Durch eine Vielzahl vorhandener Libraries aus dem Dat-Umfeld und diverse Vorarbeiten können wir die App als MVP im Prototype-Fund zuverlässig umsetzen. Wir legen dabei mehr Wert auf saubere Implementierung, Tests, Dokumentation und Reusability als auf möglichst viele Features gleich zu Beginn. Bei der Entwicklung werden wir auch die Skalierbarkeit hin zu vielen und/oder großen Archiven untersuchen. Das Dat-Projekt, auf dem wir aufbauen, wurde schon für sehr große Datensätze (mehrere hundert GB) erfolgreich eingesetzt. Das Dat-Protokoll ist für eine weitgehende Skalierbarkeit ausgelegt. Im produktiven Einsatz wurden hier schon mehrere Zehntausende Archive über einzelne Rechner erfolgreich geteilt. Wir werden entsprechende Weiterentwicklungen verfolgen und auch selbst im Rahmen des Förderprogramms Tests mit großen, vielen und weit verteilten Archiven vornehmen.

Meilensteinen

1. Backend für die Verwaltung mehrerer Dat-Strukturen (hyperdb, hyperdrive, hypergraph).
2. Im Backend UI-kompatible Wrapper für die grundlegenden Funktionen der Dat-Strukturen erstellen. Also öffentliches announcing der Strukturen, Anlegen von Dateneinträgen, etc. Hierbei zunächst Fokus auf Dateien, Metadatenmanagement in einer späteren Iteration.
3. Entwicklung eines modularen Setups für das User Interface mit React.js und verschiedenen Bundling-Konzepten.
4. Entwicklung des konkreten User Interfaces: Archivverwaltung, Datei-Management.
5. Entwicklung eines Metadaten-Store zum Anlegen und Verwalten von Meta-Informationen über die Dateien (zunächst Titel, Beschreibung, Tags). Dies beinhaltet entsprechende Backend- und Frontend-Komponenten. Das Backend ist ein Triple-Store aufbauend auf den Dat-Datenstrukturen.
6. Exemplarische Importer: Ein algorithmisches Hinzufügen großer Datenbestände, z.B. ganzer Websites. Importer sind immer stark vom jeweilig importierten Inhalt abhängig und müssen dementsprechend immer wieder neu angepasst werden.
7. Metadaten-Hub. Die Darstellung/das Browsing des Archivs anhand der bestehenden Metadaten zu dessen Inhalten.

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

Mit arso/Archipel tragen wir folgendermaßen zu den Zielen bei: Archipel soll den Erhalt von kleinen Blogs und Websites sowie der Datenbestände gemeinnütziger Initiativen, Vereine und Grassroots-Bewegungen vereinfachen. Gerade diese oft mit begrenzten finanziellen Mitteln ausgestatteten gesellschaftlichen Akteure sind von Data-Deterioration, also dem Verfall der von ihnen erstellten digitalen Inhalte, betroffen. Archipel leistet einen Beitrag dies aufzuhalten und die von den

genannten Akteuren erstellten Inhalte zu archivieren. Dies ist insbesondere wichtig, da diese gemeinnützigen Initiativen oft Fördergelempfänger*innen sind und der Verfall ihrer Arbeit auch den Verfall von Steuergeldern bedeutet.

Ausführliche Darstellung der Ergebnisse

Archipel ist eine Node.js-Anwendung die sowohl lokal als auch remote ausgeführt und in beiden Fällen mittels Webrowsers, im ersten auch als Desktop-App (via Electron) bedient werden kann. Im Backend ist es möglich unterschiedliche Datenstrukturen des Dat-Stacks (hyperdb, hyperdrive, hypergraph) zu verwalten und die in Dat bereits vorhandene peer-to-peer-Technologie zu nutzen. Die Dat-Strukturen werden zu Archiven zusammengefasst und können als gesamtes Archiv oder einzeln im peer-to-peer-Netz geteilt werden. Mittels des User Interfaces ist es möglich mehrerer dieser Archive zu verwalten. Dies umfasst das Erstellen eines neuen Archivs, das Hinzufügen des Archivs von anderen Peers, sowie die Änderungen an einer remote-Kopie eines Archivs zurück-zu-synchronisieren. Innerhalb eines Archivs ist es möglich Dateien und Ordner hochzuladen und zu erstellen. Gängige Dateitypen (Rohtext, PDF, JPEG, PNG, MP3, OGG, ...) können dargestellt bzw. wiedergegeben werden. Die Möglichkeit diese Dateien mit Metadaten zu versehen und im Archiv anhand dieser Metadaten zu browsen ist in Grundzügen vorhanden. Ebenso ist die Möglichkeit bestehende Webseiten und Blogs in ein Archiv zu importieren in Grundzügen vorhanden.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Zur Zielgruppe zählen alle, die an dem langfristigen Erhalt ihrer digitalen Inhalte interessiert sind. Insbesondere möchten wir Vereine, Initiativen und Grassrootsprojekte ansprechen. Diese leisten oft einen immensen Beitrag zur politischen und kulturellen Bildung und erstellen zu diesem Zweck liebevoll Webseiten und Blogs mit äußerst interessanten Beiträgen. Leider haben eben diese Akteure oft eine begrenzte Lebensdauer und damit auch die von ihnen

veröffentlichten Inhalte. Zudem sind die Inhalte oft nur dann zu finden, wenn man genau weiß wonach man sucht.

Wir konnten Grundlagen schaffen für die Weiterarbeit an diesem Usecase. Wir haben sehr viel gelernt in den letzten 6 Monaten, sowohl im Feld der Softwareentwicklung als auch beim Projektmanagement und dem Aufbau eines zukunftsfähigen Projekts.

Wir haben nicht nur offenen Quelltext produziert, sondern sind nun auch Teil der aktiven Entwickler-Community rund um das Dat-Projekt, dessen Fortentwicklung unserem Projekt unmittelbar zunutzen kommt. Die enge Verzahnung dieser Projekte ist nur dank Open-Source möglich.

Konkreter Nutzen für die genannten Zielgruppen ergibt sich, wenn unser Prototyp die notwendige Weiterentwicklung erfährt, um vom Alpha-Stadium zum getesteten und zuverlässigen Release weiterentwickelt werden kann. Wir sind optimistisch, dass wir mit arso diese nächsten Schritte gehen können - dank der Grundlagenarbeit, die im Rahmen des Software-Sprints geleistet werden konnte. Konkrete, anstehende Weiterentwicklungen sind:

- Tests, Error-Reporting, Stabilität
- Ausbau der Metadaten-Verwaltung
- Entwicklung von Werkzeugen um Websites und Blogs direkt aus einem Archipel-Archiv zu betreiben um so die Trennung von Veröffentlichung, Sicherung und Archivierung aufzuheben

Für die Umsetzung dieser und weiterer Schritte sehen wir nicht nur technisch, sondern auch organisatorisch mit den im Prototype Fund gewonnen Erfahrungen optimistisch entgegen.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Die Kopplungspunkte zwischen Front- und Backend in unserem isomorphen Stack mussten wir mehrfach überdenken und teilweise neu ansetzen, da die erste Implementierung die Umsetzung wichtiger Features erschwert hat. Im Rahmen eines iterativen Programmierkonzepts ist dies jedoch ein normaler Vorgang.

Aktuelle Entwicklungen von JavaScript haben zu teils unerwarteten Komplexitäten geführt aufgrund von Inkompatibilitäten zwischen Browsern (Frontend) und NodeJS (Backend). Diese konnten wir jedoch soweit möglich in den Griff bekommen.

Die Architektur der Metadatenverwaltung erforderte und erfordert ausgiebige Abwägung zwischen der Komplexität von RDF und semantischer Korrektheit einerseits und dem Anspruch, unnötige Komplexität zu vermeiden andererseits.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

<https://arso.xyz/archipel>

<https://github.com/arso-project/archipel>

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Die Entwicklung mancher von uns verwendeten Open-Source-Modulen aus dem Dat-Stack verlief langsamer, als von uns (und dem Dat-Projekt selbst) zu Beginn der Förderperiode erwartet. Dadurch musste von unserer Seite teils mehr Aufwand für Grundlagenarbeit investiert werden. Aufgrund dessen mussten wir den Metadatenaspekt in Teilen weiter nach hinten und damit aus der Förderperiode hinaus verschieben.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Unser Produkt basiert in weiten Teilen auf dem Dat-Projekt. Dort steht derzeit eine größere Erneuerung an ("multidrive hyperdrive"), die in der Fertigstellung mehr Zeit in Anspruch nimmt als anfangs erwartet. Das hat uns teilweise vor Probleme gestellt. Zugleich haben wir enorm profitiert von der Entwickler-Community rund um das Dat-Projekt.

Schlussbericht

Zuwendungsempfänger:

Kochetkova Zemke Gbr

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01|S18S42 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Es sollte ein Webdesktop entwickelt werden, der das Teilen, Speichern und Bearbeiten von Dateien und Projekten ermöglicht. Webdesktop bedeutet, dass die Funktionalität eines Betriebssystems auf einer Webseite nachempfunden wird, inklusive Filesystem, der Möglichkeit Applikationen auszuführen und mit diesen Dateien aus dem Filesystem zu bearbeiten. Dateien können privat gespeichert oder mit der Öffentlichkeit, bzw. mit bestimmten Gruppen geteilt werden.

Das "universeOS" soll anonyme und pseudonymisierte Kollaboration, Recherche, Veröffentlichung und Durchsuchen von komplexen Informationen auf Basis von Open Source Technologie ermöglichen. Das öffentliche Dateisystem kann zu einem Archiv digitaler Informationen werden, dass Themen erfassbar abbildet.

Gerade in Ländern, wie etwa Russland, China oder die Türkei, wo die freie Meinungsäußerung stark eingeschränkt ist und Journalist*innen oder Menschenrechtsaktivist*innen bedroht sind, fehlt es an offenen Kollaborations-Lösungen die es ermöglicht gemeinsam an Recherchen zu arbeiten, oder Dokumente auszuwerten und der Öffentlichkeit bereitzustellen, ohne sich selbst zu gefährden. Bei proprietären Softwarelösungen ist oft unklar ob und wie Daten weitergegeben werden, zudem sind sie in bestimmten Ländern mitunter nicht erreichbar. Durch den offenen Ansatz, kann jeder sein eigenes UniverseOS aufsetzen und somit Nachteile zentralisierter Systeme, insbesondere in Bezug auf Zensurmaßnahmen umgehen.

Das System soll das Verstehen aktueller politischer oder gesellschaftlicher Ereignisse und Entwicklungen vereinfachen, indem Themen zusammengefasst, verlinkt und kollaborativ bearbeitet werden können.

Zielgruppe sind hierbei Recherchegruppen, Journalist*innen so wie z.B Menschenrechtsaktivist*innen in Ländern wie bsp. Russland oder China. Welche pseudonymisiert mit dieser Plattform recherchieren, kollaborieren und publizieren können

Zusätzlich soll ein Gegenangebot für Tools angeboten werden, die von Organisationen und Unternehmen genutzt werden, die primär von kommerziellen (Cloud-)Anbietern angeboten werden. Nutzer*innen sollen die Macht und die Kontrolle über ihre eigenen Daten behalten können.

Die Software besteht im wesentlichen aus zwei Komponenten: Einem Server und der GUI.

Der Server beinhaltet eine Datenbank, einen Storage und eine API, um mit der GUI zu interagieren. Storage und Datenbank sollen nicht öffentliche Daten verschlüsselt speichern. Für die Implementierung wurde das NodeJS Framework Express genutzt.

Im Frontend wurde das Nutzererlebnis eines Betriebssystems auf Basis von vueJS nachempfunden, inklusive Dateisystem und Applikationen.

Die umfassendste Änderung bestand in einem Wechsel der Programmiersprache von Go zu NodeJS für die Serverapplikation. Wir hatten zunächst geplant, das Backend in Go umzusetzen, haben jedoch für den Prototypen zu NodeJS gewechselt, da dies einerseits eine einheitliche Umsetzung des Gesamtprojektes (Frontend & Backend) in Javascript ermöglicht hat und zum anderen ein deutlich schnelleres Prototyping ermöglicht hat.

Zusätzlich hatten wir zunächst geplant, das Signal Protokoll zur Verschlüsselung von Dateien und Nachrichten zu nutzen. Da das auf Instant-Messaging ausgelegte Protokoll nicht auf die Verschlüsselung von Dateien ausgelegt ist, sind wir auf die Crypto Bibliothek "NaCL" ausgewichen.

Aufgrund der Einarbeitung in das Signal Protokoll und den Wechsel zur NaCL Bibliothek ist ein deutlicher Mehraufwand entstanden, da wir für die Nutzerregistrierung, während welcher auch die Generierung der Schlüssel stattfindet, statt wie geplant einer Arbeitswoche insgesamt beinahe drei Wochen benötigt wurden. Dies hat zu einer deutlichen Verschiebung des Zeitplans geführt.

Die Repos sind unter Gitlab verfügbar:

Universe Node Backend

http://gitlab.com/creators-of-the-universe/universe_node

Universe Frontend

https://gitlab.com/creators-of-the-universe/universe_frontend