

MultimodalRouting - Multimodale Routing Engine für Rollstuhlfahrer und Fahrradlogistiker

Schlussbericht

Zuwendungsempfänger:

Lötzsch, Tilsch & Sultova GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S01 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Die Community um OpenStreetMap und Wheelmap pflegt eine umfangreiche Sammlung freier Daten über Barrierefreiheit, die für Rollstuhlfahrer und Lastenräder sehr nützlich sind. Wir wollen das Potential dieser Daten nutzen, um eine für die Bedürfnisse dieser Nutzer optimierte Navigationsanwendungen zu entwickeln.

Primäre Motivation war, körperlich eingeschränkten Menschen bei selbstständiger Verkehrsteilnahme an unbekannten Orten zu unterstützen. Von Anfang an wurde erkannt, dass die im Projekt entwickelten Routinglösungen auch auf viele andere Einsatzzwecke übertragbar sind. In einer bestehenden Kooperation wurde seit längerem nach ähnliche Werkzeugen gesucht, um alternative Innenstadtlogistikkonzepte durch geeignete Software zu unterstützen.

Projektziel war die Entwicklung einer Routingengine, die Navigation für kombinierte Nutzung von nichtmotorisiertem Individualverkehr und öffentlichen Verkehrsmitteln unterstützt und dabei vorhandene Informationen zu akuten Hindernissen (z.B. nicht funktionierende Fahrstühle, Ausfälle/Umleitungen im ÖPNV) berücksichtigt. Besonderen Fokus sollte auf die Verwendbarkeit individualisierbarer Routingprofile gelegt werden.

Wichtigste Meilensteine waren die Untersuchung frei verfügbarer Datenquellen mit einem Fokus auf barrierearmes Routing, die Implementierung einer Routingengine für aggregierte dynamische Daten, sowie der Entwurf eines Frontends, dass verfügbare Daten verständlich visualisiert.

Technische Herausforderung

Suchalgorithmen sind wissenschaftlich ausführlich untersucht. Zur informierten Routensuche in euklidischen Räumen werden meist Varianten des vollständigen und optimalen A*-Algorithmus verwendet. Er hat im Verhältnis zur Anzahl der Knoten eine quadratische worst-case Laufzeit.

Es existiert eine Vielfalt freier Routingengines, mittels derer auf OpenStreetMap-Daten navigiert werden kann. Diese sind üblicherweise auf Performance optimiert, damit auch für lange Strecken effizient optimale Lösungen gefunden werden können. Um das zu erreichen, wird sich auf eine beschränkte Anzahl verfügbarer Routingprofile festgelegt, für die jeweils eine Zuordnung von Streckenabschnitt zu Routingkosten (z.B. Zeitaufwand) statisch vorberechnet wird.

Um während des Routings auf dynamische Größen (z.B. Wahrscheinlichkeit dass ein Fahrstuhl nicht funktioniert, sich ändernde individuelle Nutzerpräferenzen oder aus überwundenen Strecken erlernte Heuristiken) reagieren zu können, muss ein flexiblerer Ansatz verwendet werden. Zu unseren Aufgaben gehörte es daher, entweder eine bestehende Routingengine für die Nutzung mit dynamischen Routingkosten anzupassen oder eine neue Routingengine zu entwickeln.

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

Wir wollen körperlich eingeschränkten Personen mit Hilfe von Public Interest Tech in Form von durch Crowdsourcing gesammelten offenen Daten und einer ehrenamtlich gepflegten Free/Libre Open Source App helfen. Nach unserer Auffassung sollten solche Werkzeuge zur Daseinsvorsorge zählen und halten die Prototypfinanzierung nach dem Paradigma „Public Money, Public Code“ daher für angemessen.

Neben der konkreten App, bietet auch das entwickelte Routingframework ein großes Potential als Plattform weiterer Anwendungen, die Menschen mit besonderen Anforderungen unterstützen können.

Wir wurden darauf aufmerksam gemacht, dass Eltern mit Kinderwagen und Senioren mit Rollatoren ähnliche Anforderungen haben. Es gibt bereits

Überlegungen anderer Entwickler, auf unserer Routingengine aufbauend eine Navigationsanwendung für Blinde zu entwickeln.

Im Kontext von verkehrsmittelübergreifenden Verkehrskonzepten und alternativen Antriebstechnologien, kann multimodales Routing einen Beitrag zum Klimaschutz leisten.

Spannende Anwendungen könnten sich ergeben aus Kooperation mit Nahverkehrsanbietern (Park and Ride), Anbietern von Leihfahrrädern und E-Scootern, sowie im Bereich E-Mobilität (Einbeziehung von Ladezeiten in die Routenplanung). Es existieren zahlreiche Möglichkeiten, wie das Framework für stadtplanerische Fragestellungen ein hilfreiches Werkzeug darstellen kann.

Ausführliche Darstellung der Ergebnisse

Zusammenfassung

Es wurde ein Framework implementiert, mittels dessen Routinganwendungen für Spezialanforderungen unter Verwendung von aggregierten Daten aus unterschiedlichen Quellen gebaut werden können.

Im einfachsten Fall müssen Entwickler nur die anwendungsspezifischen Routingprofile mittels Kostenfunktion selbst implementieren. Dafür kann wahlweise Java oder Javascript verwendet werden.

Das Framework stellt alle anderen notwendigen Komponenten bereit. Dazu zählen Routingengine, Anbindung von Datenquellen, UI sowohl für die Darstellung von Routen in der App, als auch zusätzliche Kartenoverlays zur Darstellung von Debuginformationen. Auf diese Weise können Datenqualität für den benötigten Zweck und Eigenschaften der entwickelten Kostenfunktion visualisiert werden.

Das Backend ist vollständig in Clojure(script) implementiert und kann dadurch sowohl für die JRE als auch nach Javascript compiliert werden. Mit dem Framework gebaute Routinganwendungen können wahlweise als Client-/Server-Architektur oder vollständig im Client laufen. Unsere Build-Pipeline kann die Anwendungen mittels Cordova als native Android-, iOS- oder Electron-Apps bauen.

Auf Basis des Frameworks wurde eine Navigationsanwendung für Menschen mit körperlichen Einschränkungen entwickelt, die sich derzeit in einer frühen Testphase befindet. Eine Integration in eine Logistikanwendung für Fahrradkuriere ist geplant.

Einzelne Funktionen (z.B. Darstellung von Isochronen) können auch als separate Karte in Webseiten eingebunden werden.

Anforderungsanalyse, allgemeine Recherche

~2 Mannwochen

Eine ausführliche Anforderungsanalyse, die gemeinsam mit Vertretern der Zielgruppe durchgeführt wurde, half uns die Bedürfnisse unserer Nutzer besser zu verstehen und zu priorisieren. Besonders wertvoll war es, von Personen mit sehr unterschiedlichen Einschränkungen zu erleben, welche Hindernisse für sie relevant sind, zu sehen bei welchen Alltagssituationen bestehende Lösungen für diese Anwender nicht funktionieren und erklärt zu bekommen, welche Informationen als besonders hilfreich erachtet werden.

Es setzte sich die Erkenntnis durch, wie vielfältig die benötigten Routingprofile sind und dass auch für einen einzelnen Nutzer je nach Situation unterschiedliche Routingprofile benötigt werden (z.B. abhängig davon, ob er mit einer Begleitperson, einem Zusatzmotor oder einer anderen eingeschränkten Person unterwegs ist).

Das Bedürfnis relevante Informationen detailliert und dennoch übersichtlich visualisiert zu bekommen bestätigte sich. Alternative Darstellungsoptionen wurden gemeinsam erörtert.

Untersuchung vorhandener OSM Routingengines

~1,5 Mannwochen

Als vielversprechendste Kandidaten für eine Anpassung ergaben sich Graphhopper, Openrouteservice und BRouter.

Graphhopper zeichnet sich insbesondere durch gut lesbaren Code, umfangreiche Dokumentation, reichhaltige Funktionalität, effiziente Datenstrukturen und eine stetige Weiterentwicklung aus.

Die Einarbeitung in bestehende Routingengines gab hilfreiche Anregungen für die spätere eigene Implementierung.

Untersuchung der Qualität existierender Daten

~1 Mannwoche

Es wurde in verschiedenen Regionen untersucht, ob die verfügbaren Daten für die von uns geplanten Lösungsansätze ausreichen. Dafür wurden sowohl Datensätze konkreter relevanter Orte im Detail betrachtet, als auch qualitative Vergleiche verschiedener Regionen berechnet. Es stellte sich heraus, dass Dresden aufgrund der guten Datenlage ein vielversprechender Ort für Tests ist.

Frontendentwicklung

~5 Mannwochen

Besondere Relevanz für unsere Anwendung hat die übersichtliche Informationsdarstellung auf Karten. Das UI-Coaching half uns eine geeignete Darstellung zu finden, die neben der Beurteilung von Barrierefreiheit auch Auskunft über die Konfidenz der Daten liefert.

Ein entwickeltes Plugin für React-Leaflet, dass beliebige Visualisierungen auf Leaflet- und Mapbox-Karten mittels D3 ermöglicht, kann auch für viele andere Projekte im Bereich Data Science hilfreich sein.

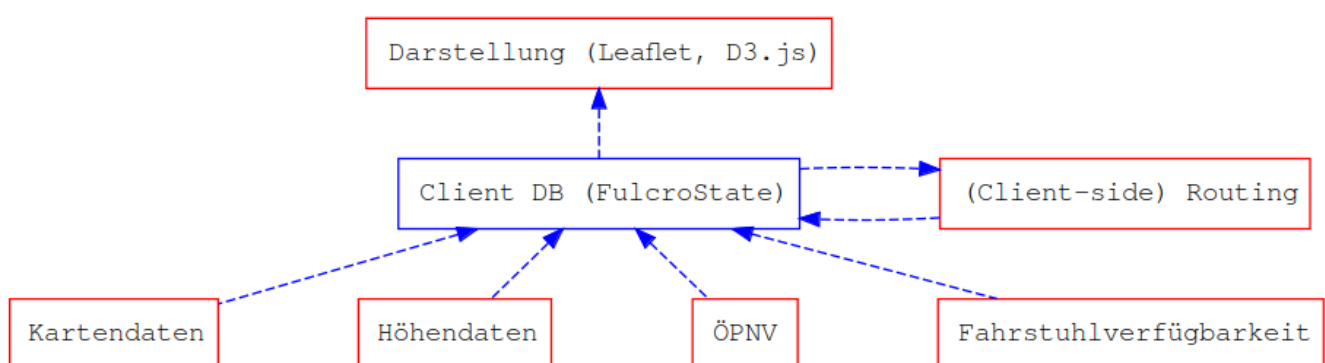
Durch Verwendung von Framework7 kann unsere Webanwendung auf den gängigen Plattformen mit nativem Look and Feel genutzt werden.

Framework

~7 Mannwochen

Wir haben uns von vornherein bemüht, möglichst große Teile der Anwendung für ähnliche Projekte wiederverwendbar zu designen.

Der Kern unserer Architektur besteht aus einer Datenbank, die Geofeatures strukturell ähnlich zum JOSM-Dateiformat speichert. Diese Datenbank kann mit Geodaten verschiedenster Quellen befüllt werden. Die von uns implementierte Routingengine kann verwendet werden, um auf den vereinheitlichten Geofeatures Routen zu berechnen. Routen können selbst wiederum als Pfad oder Relation in der Datenbank abgelegt werden. Das Frontend stellt Methoden bereit, um beliebige Geofeatures darzustellen.

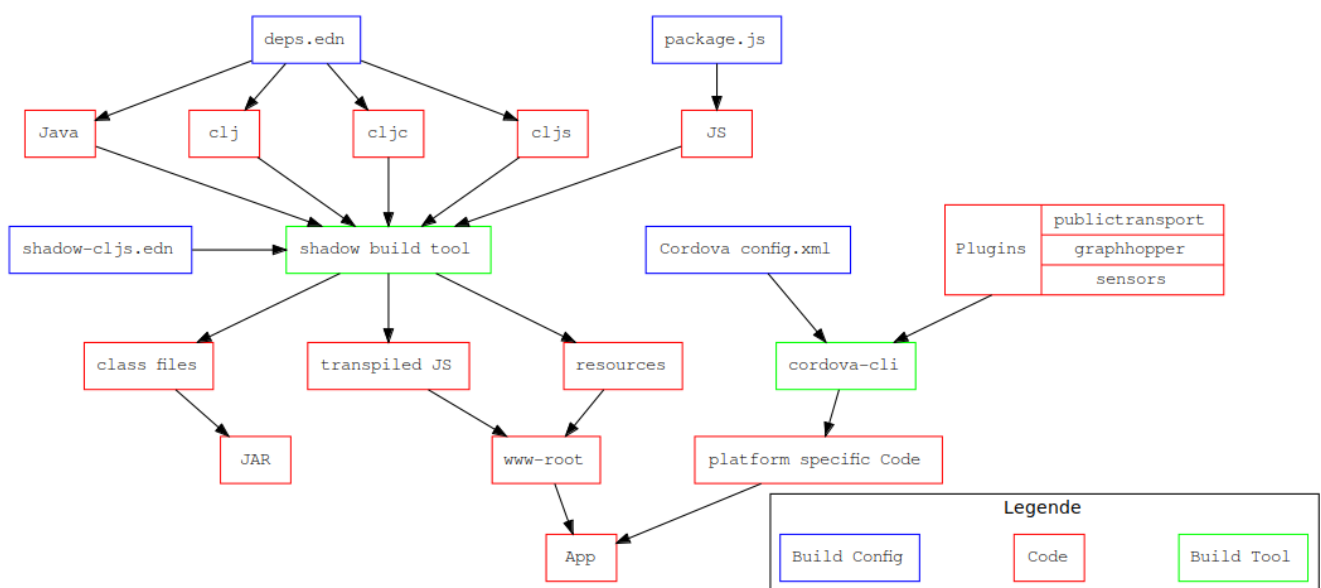


Auf Basis dieser Architektur ist eine effiziente Entwicklung möglich, da für neue Routingprofile geschriebene Kostenfunktionen und hinzugefügte Datensätze beliebig miteinander kombiniert und für Debugging-Zwecke ohne zusätzlichen Aufwand visualisiert werden können.

Das Frontend läuft im Browser. Der Rest des Frameworks ist aufgrund der Implementierung in Clojure(Script) sowohl nach Javascript als auch für die JVM compilierbar.

Anwendungsentwicklung unter Nutzung des Frameworks besteht primär daraus, Kostenfunktionen und optional zusätzliche Anbindungen für Datenquellen wahlweise in Javascript oder Java zu implementieren. Wenn auch dafür Clojure(Script) verwendet wird, behalten Prototyp-Entwickler die volle Flexibilität, welche Teile der Anwendung später im Front- oder Backend laufen.

Das Framework stellt eine Build- und CI-Pipeline bereit, um neben reinen Webanwendungen mittels Cordova auch native Apps für Android und iOS zu bauen.



Um plattformunabhängig, energiesparend und zeitlich hochauflösend den aktuellen Gerätestandort zu ermitteln, nutzen wir das Cordova Background Geolocation Plugin.

Anbindung von Datenquellen

~4 Mannwochen

Für die Nutzung mit unserem Routingframework wurden bereits eine Vielzahl an Datenquellen angebunden. Zum Zweck der Rollstuhlnavigation sind das Kartendaten, Höhendaten, ÖPNV-Fahrplandaten, und Informationen über die aktuelle Funktionalität von Fahrstühlen.

Eine Anbindung an Graphhopper-Routenabfragen kann genutzt werden, wenn für ein in Graphhopper existierendes Routingprofil (z.B. PKW) performante (überregionale) Abfragen gestellt werden sollen.

Während der Entwicklung ist die Nutzung von Overpass-Abfragen als Kartenquelle sehr hilfreich.

Routingengine

~3 Mannwochen

Die entwickelte Routingengine verwendet das einheitliche Geofeature-Format des Frameworks und funktioniert unabhängig von der Herkunft der Daten.

Die Engine expandiert Geofeature (Punkte, Linien, Flächen, Relationen) in einen Routinggraph im Format der Clojure-Bibliothek Loom. Eine vom Anwendungsentwickler übergebene Kostenfunktion, die das Routingprofil definiert, wird verwendet um jeder Kante im Routinggraph ein Gewicht zuzuweisen.

Zur Berechnung der kürzesten Pfade können sowohl die Graphenbibliothek Loom, als auch die zu ihr kompatible Bibliothek Ubergraph verwendet werden. Dafür existieren jeweils Implementierungen von Dijkstra und A*.

Mehrere Loom-Graphen lassen sich problemlos miteinander mergen. Dass erlaubt es die Expansion und Kostenberechnung statischer Daten einmalig, die von dynamischen Daten jedoch zusätzlich so oft wie nötig auszuführen.

Routingergebnisse können wieder in das einheitliche Geofeature-Format gerechnet und die Eingangsdaten für jeden Knoten referenziert werden, indem eine zur Expansion inverse Funktion verwendet wird.

Beta-Tests

~3,5 Mannwochen

Um erste Ergebnisse zu evaluieren und zu verbessern, wurde implementiert, dass von Testnutzern im Alltag zurückgelegte Strecken aufgezeichnet und mit den von uns berechneten Routenvorschlägen verglichen werden können.

Da uns während dieser Projektphase vor allen iOS-Nutzer zur Verfügung standen, wurde der zusätzlich notwendige Aufwand getrieben, um die App auch für iOS bereitzustellen.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Framework

Das entwickelte Routingframework ist vielseitig einsetzbar. Es ermöglicht mit verhältnismäßig geringem Aufwand Routinganwendungen für Spezialanforderungen zu bauen.

Es stellt Anbindungen verschiedener Datenquellen und deren Transformation in ein einheitliches Geofeature-Format, Methoden um daraus Routinggraphen zu generieren, sowie darauf arbeitende Routingalgorithmen und Visualisierungen der Ergebnisse bereit. Die Buildpipeline kann genutzt werden, um Webanwendungen, native Android-, iOS- oder Electron-Apps zu bauen. Die Anwendung kann wahlweise vollständig Clientseitig oder als Client-/Serverlösung gebaut werden.

Zum Bau einer Routinganwendung mit nutzerspezifischen Profilen, muss unter Verwendung unseres Frameworks bestenfalls nur die Kostenfunktion implementiert werden.

Bei Bedarf können neben den bereits angebundenen Datenquellen weitere Quellen implementiert werden.

Derzeit können OpenStreetMap-Karten im JOSM-Format (*.osm), deren Json-Entsprechungen (OSM JSON), sowie im Protocolbuffer-Binärformat (*.pbf) genutzt werden. Ebenso existiert die Möglichkeit Daten per Overpass API und aus PostGIS-Datenbanken im Imposm3-Format abzufragen oder MapboxVectorTiles (*.mvt) zu nutzen.

Höhendaten können als TerrainTiles und Fahrpläne aus GTFS-Quellen eingebunden werden. Echtzeitdaten, ob Fahrstühle funktionieren, können über die APIs von accessibility.cloud und der Deutschen Bahn verwendet werden.

Angedacht, jedoch noch nicht implementiert ist die Nutzung von Daten aus dem Open-Data-Portal der Stadt Dresden zu Bordsteinabsenkungen und Behindertenparkplätzen.

Neben unseren Anwendungen für Rollstuhlfahrer und Fahrradlogistiker gibt es weitere Anfragen zur Nutzung des Frameworks, die wir gerne unterstützen. Es gibt konkrete Pläne für eine Navigationsanwendung für Blinde, die auf unserem Framework basierend eine textuelle und auditive Nutzerschnittstelle bereitstellen will. Ebenfalls in Überlegung ist die Nutzung für einen barrierefreien Campusnavigator.

Noch keine konkreten Pläne, jedoch erste Ideen gibt es in Kooperation mit einer Logistikfirma, die Navigation auf Vorfeldern zur Abfertigung von Luftfracht, sowie zur Optimierung der Anlieferung auf Werksgeländen benötigt.

Bisher hat die Arbeit in dem Projekt vor allem die beteiligten Entwickler in ihrer persönlichen, und fachlichen Weiterentwicklung vorangebracht und mit dem Framework ein großes Potential an Folgeprojekten ermöglicht.

Wir planen das Framework gründlich aufzuräumen, einzelne Funktionalitäten in separate Bibliotheken zu unterteilen und für andere Entwickler verständlich zu dokumentieren.

Routinganwendung für körperlich eingeschränkte Menschen

Unsere Evaluation hat ergeben, dass bereits für sämtliche aus unserer Sicht relevanten Weg- und Hindernissattribute Tags zur Kartographierung in der OpenStreetMap definiert wurden. Leider gibt es teilweise verschiedene Varianten, die berücksichtigt werden müssen.

Die Datenlage ist in verschiedenen Städten, abhängig von der lokalen OSM-Community, sehr verschieden.

Unsere Routinganwendung zeigt den Mehrwert einiger Kartendaten, die von anderen Navigationsanwendungen bisher noch nicht berücksichtigt wurden. Wir untersuchen aktiv, wie durch ehrenamtliche Mapper und durch OpenData-Initiativen ein effizienter Beitrag für körperlich eingeschränkte Menschen erzielt werden kann. Unsere App und die mit ihr verbundene Öffentlichkeitsarbeit soll am Beispiel von Städten mit guter Datenlage motivieren, auch in anderen Regionen schrittweise die Erfassung relevanter Daten zu verbessern.

In Städten, bei denen pro Bus-/Straßenbahnlinie jeweils Fahrzeuge mit einheitlicher Einstiegshöhe verwendet werden, kann ein sehr großer Mehrwert für körperlich eingeschränkte Menschen erzielt werden, indem Haltepunkte danach klassifiziert werden, wie diese nutzbar sind. Dafür sind die 3 Varianten: barrierefrei (wheelchair=yes), bei Verwendung einer Rampe (wheelchair=limited) oder nicht barrierefrei (wheelchair=no) vorgesehen. Aus unserer Sicht ist das der einfachste Weg, wie lokale OSM-Communities einen ersten effizienten Beitrag leisten können.

Für am zweitwichtigsten halten wir das Mappen von Fahrstühlen, Rampen, Stufen/Treppen, Pollern, Schranken und ähnlichen Hindernissen.

Ebenfalls wertvoll ist das Verzeichnen von Straßenübergänge über stark befahrene Straßen mit dem Tag wheelchair, sowie gegebenen Falls die Kennzeichnung der Existenz einer Ampel.

Mit größerem Aufwand verbunden ist das konsequente Mappen von Bürgersteigen (jeweils beide Fahrbahnseiten separat) mit den Attributen für Belag, Glattheit, Breite und Steigung (oder vereinfacht mit dem wheelchair-Attribut).

Nach unseren ersten Erfahrungen, kann eine hilfreiche Rollstuhl-Navigationsanwendung auf Basis von Heuristiken auch dann implementiert werden, wenn nicht jede einzelne Bordsteinabsenkung bekannt ist. Ein nützlicher

Mehrwert wird bereits durch das Mappen unerwartet unzureichender Absenkungen erzielt. Selbstverständlich ist jede verzeichnete Bordsteinabsenkung ein Mehrwert, insbesondere in Kreuzungsbereichen und an anderen wichtigen Übergängen.

Viele OSM-Routingengines können nur auf OSM-Elementen vom Type Linie, jedoch nicht auf solchen vom Typ Punkt routen. So sind sie beispielsweise unfähig, als Punkt gespeicherten Ampeln, Haltestellen oder Fahrstühlen Wegkosten zuzuweisen. Unsere Engine erlaubt bei der Expansion des Routinggraphs aus OSM-Elementen, egal ob Punkt, Linien oder Relation, jeweils einen oder mehrere Graphkanten zu erzeugen. So können bei Ampeln und Haltestellen separat Kosten für erwartete Wartezeit und Überwindung des räumlichen Hindernisses berechnet werden. Bei Fahrstühlen wird eine zusätzliche Graphkante für die erwarteten Kosten bei Defekt hinzugefügt. Diese kann aus Ausfallwahrscheinlichkeit und in diesem Fall nötigem Alternativweg abgeschätzt werden.

Aufgrund der regional sehr unterschiedlichen Datenlage, liefern traditionelle Routingprofile, die Wegen entsprechend ihrer Attribute konstante Streckenkosten zuweisen, nur mäßige Ergebnisse. Insbesondere ist es bei großer Varianz schwierig, die Kosten für verschiedene Wegtypen und der Kombination ihrer Attribute richtig zu gewichten. Wir begegnen der Herausforderung, indem wir eine zweidimensionale Metrik, bestehend aus geschätztem (Zeit-)Aufwand und geschätzter Konfidenz, verwenden. Die Berechnung der finalen Wegkosten im Routinggraph nutzt eine Normalisierung in Abhängigkeit der regionalen Datenlage sowie von Nutzerpräferenzen.

Künftig sollen auch vom Gerät des Nutzers lokal gesammelte Daten genutzt werden, die Aufschluss darüber geben, wieviel Zeit für das Zurücklegen ähnlicher Wege benötigt wurde.

Ein wichtiges Alleinstellungsmerkmal unserer Engine ist der Umgang mit Hindernissen, bei denen wir uns nicht sicher sind, ob der Nutzer sie überwinden kann und möchte. Unser Ziel ist es, dem Nutzer eine selbstbestimmte, informierte Entscheidung zu ermöglichen. Deshalb wollen wir dem Nutzer sowohl die Möglichkeit geben, die als optimal berechnete vollständig barrierefreie Route zu wählen, als auch Abkürzungen die wahrscheinlich mit Aufwand überwindbare Hindernisse beinhalten. Unsere Lösung zeigt daher die als sinnvoll berechneten Alternativen an, visualisiert Hindernisse als solche klar erkenntlich und ermöglicht alle bekannten Informationen über das Hindernis einzusehen.

Künftig wollen wir aus dem vom Nutzer getroffenen Entscheidungen lernen, um die Vorschläge individuell verbessern zu können. Ein neues Nutzerprofil soll zu

Beginn einer sehr konservativen Konfiguration entsprechen, also nach Möglichkeit jeder Form von Hindernissen ausweichen. Wenn der Nutzer sich für Alternativrouten entscheidet und deren Hindernisse überwindet, sollen ähnliche Hindernisse schrittweise im Nutzerprofil freigeschaltet werden. Während der Überwindung von Hindernissen wird die benötigte Zeit gemessen und künftig für die Abschätzung der benötigten Streckenkosten verwendet.

Wir erhoffen uns damit, individuellen Bedürfnissen und Vorlieben entgegen zu kommen, ohne dass jeder neue Nutzer umfangreiche Einstellungen vornehmen muss.

Eine spannende Fragestellung für künftige Experimente ist, wie zuverlässig aus GPX-Tracks, die um Daten von Beschleunigungssensoren angereichert wurden, auf Attribute von Wegen geschlossen werden kann. Dies könnte eine Lösung zur Datenerhebung ermöglichen, bei denen unsere App nutzende Rollstuhlfahrer gebeten werden, die automatisch ermittelten Streckeneigenschaften (z.B. Vorhandensein einer Bordsteinabsenkung) zu bestätigen.

Während der Entwicklung wurde uns von künftigen Nutzern ein großes Interesse gespiegelt. Es besteht ein deutlicher Wunsch nach Autonomie im Alltag und das Bedürfnis insbesondere an unbekannten Orten die Effizienz und Planungssicherheit zum pünktlichen Erreichen von Zielen zu erhöhen.

Wir stehen mit Rollstuhl-Communities in drei verschiedenen Städten in Kontakt, die uns beim Testen unterstützen.

Die im Coaching als Grundlage von nutzerzentrierter Entwicklung angeregten gemeinsamen Tests konnten aufgrund des COVID-19-Lockdowns seit Mitte März nicht wie ursprünglich geplant durchgeführt werden. Der Test- und Entwicklungszyklus soll aber zeitnah wieder aufgenommen werden.

Nach erfolgreichen Tests soll die App bei F-Droid und Google Play veröffentlicht werden. Ob auch eine Verbreitung über den Apple App Store stattfinden wird, ist unsicher. Bisher sind die umfangreichen Bemühungen ein Apple Developer Certificate zu erwerben gescheitert.

Es ist beabsichtigt, die App ehrenamtlich weiter zu entwickeln. Hilfreich wäre es, die App schlank zu halten und möglichst viel der Funktionalität mit anderen Anwendungen im Framework zu teilen.

Anwendung für Fahrradkuriere

Das entwickelte Routingframework soll genutzt werden, um eine auf die Bedürfnisse von Fahrradkurieren abgestimmte Routenberechnung in eine bestehende Logistikanwendung zu integrieren.

Geeignete Routingprofile werden sowohl für Rennradfahrer, als auch Lastenradfahrer benötigt.

Die Routenberechnung kann vom Fahrer zur Navigation, vom Dispatcher für die Einsatz-Planung und vom System zur Abschätzung Entfernungs- und Zeitabhängiger Auftragspreise verwendet werden.

Eine Firmenspezifische Routinganwendung kann es ermöglichen, Verkehrsinformationen, Informationen über Streckensperrungen, sowie Hinweise zu Abhol- und Zustellorten zwischen verschiedenen Fahrern und Dispatchern auszutauschen.

Eine einheitliche Fahrerapp kann sowohl für die Navigation, zum melden der aktuellen Position an den Dispatcher und wartende Kunden, zur Übermittlung neuer Aufträge an den Fahrer, als auch zur allgemeinen Kommunikation zwischen Fahrer und Dispatcher verwendet werden.

Einbettung des Plugins auf Webseiten

Manche Funktionen des Frameworks sollen auch als mit minimalem Aufwand in Webseiten einbindbare Karten zur Verfügung gestellt werden.

Beispielsweise kann die Berechnung von Isochronen genutzt werden, um auf einer Veranstaltungswebsite den Nutzern Orte von Interesse (z.B. Hotels, Restaurants, Haltestellen) anzuzeigen, die dem Veranstaltungsort nicht nur räumlich in der Nähe sind, sondern von diesem auch bequem erreicht werden können.

Vernetzung und Weiterbildung

Neben der fachlichen Arbeit profitierten wir besonders durch den fachlichen Austausch innerhalb des Teams und mit der Community.

Sehr wertvoll für uns waren die im Rahmen der Förderung angebotenen Coachings.

Auf der Konferenz "State of the Map" haben wir uns mit Vertretern der Open-Street-Map Community und Wissenschaftlern aus dem Bereich der Geo-Informationen-Systeme vernetzt. Beim Accessibility Club Summit 2019 bildeten wir uns zu barrierefreier Softwareentwicklung weiter.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

In der Beta-Test-Phase am Ende des Förderungszeitraumes standen uns vor allem Nutzer von Geräten mit iOS zur Verfügung. Daher bemühten wir uns iOS-Builds unserer App zur Verfügung zu stellen. Leider funktioniert dies nur unter Verwendung von Xcode, welches wiederum nur für MacOS existiert. Wir schafften uns daher einen Mac zum bauen und ein iPad für eigene Tests auf der Plattform an. Wir können unsere App nun auch für iOS compilieren, per USB-Kabel installieren und identisch zur Android-App nutzen.

Um Apps auf entfernten iOS-Geräten (insbesondere für Testnutzer in anderen Städten wichtig) installieren zu können, muss die App mit einem Apple Developer Certificate signiert werden. Wir haben daher auch ein solches Zertifikat bei Apple käuflich erworben. Leider wurde das Zertifikat nach kurzer Zeit ohne Begründung von Apple widerrufen. Nach sehr vielen Telefonaten mit dem Apple-Support, bei denen wir immer wieder weitergeleitet wurden, uns jedoch von keiner Abteilung geholfen werden konnte, entschieden wir uns für einen komplett neuen Versuch. Obwohl wir unter dem Namen eines anderen Teammitgliedes, mit anderer Telefonnummer und Mailadresse auf anderer Hardware eine neue Apple ID angelegt, als Apple Developer registriert und über eine andere Kreditkarte das Developer Certificate bezahlt haben, wurde das Zertifikat ebenfalls gesperrt. In Folge dessen trafen wir die Entscheidung, uns zunächst auf Android zu beschränken..

Da iOS eine weit verbreitete Plattform ist und von vielen Rollstuhlfahrern genutzt wird, sollten wir mittelfristig mit Unterstützung erfahrener iOS-Entwickler einen weiteren Versuch unternehmen, um die Anwendung auch im Apple App Store zu veröffentlichen.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Der vollständige Quellcode steht auf der Online-Kollaborations- und Codehosting-Plattform Github zur Verfügung.

<https://github.com/multimodalrouting>

Die Rollstuhl Navigationsapp wurde bei Testnutzern installiert.

Im nächsten Schritt ist eine Veröffentlichung bei F-Droid und Google Play geplant. Die Verbreitung über den Apple App Store kann nicht garantiert werden.

Eine technische Dokumentation des Routingframeworks ist auf der Projektwebsite geplant.

<https://multimodalrouting.github.io>

Gelante Vorträge mussten aufgrund von COVID-19 abgesagt werden. Es ist jedoch angedacht, diese bei künftigen Konferenzen nachzuholen.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Die im Projektantrag beschriebene Anwendung wurde unter Einhaltung des Kosten- und Zeitplans als Prototyp umgesetzt.

Änderungen der geplanten Routingarchitektur

Ursprünglich war ein Backend geplant, das aus zwei separaten Komponenten besteht:

1. Zur Berechnung von Routen auf statischen OSM-Daten sollte eine bestehende Engine ausgewählt und angepasst werden (hinzufügen von parametrisierbaren Rollstuhl-Profilen).
2. Die Berechnung von Routen auf der Kombination von Ergebnissen der ersten Komponente (statische OSM-Daten) und zusätzlichen dynamischen Daten (z.B. Informationen ob Fahrstühle aktuell funktionieren) sollte per Eigenentwicklung gelöst werden.

Grund für diese Architekturüberlegung war es, dass die unterschiedlichen Herausforderungen, bei den beiden zu lösenden Teilaufgaben tendenziell verschiedenen hätten gelöst werden müssen.

Die Laufzeitkomplexität von Routing auf statischen Daten kann um einen spürbaren linearen Faktor gesenkt werden, wenn die Wegkosten aller Streckenabschnitte einmalig vorberechnet wird. Auf Performance optimierte Engines verwenden dafür effiziente low-level Datenstrukturen.

Auf dynamischen Daten ist diese Optimierung nicht möglich.

Die Erkenntnis, dass für Routing innerhalb einer Stadt die Anzahl der Kanten (Wege) klein genug sind, um auch ohne die Optimierung hinreichend schnell zu sein, hat uns umdenken lassen.

Wir führen das Routing auf den statischen OSM-Daten und der zusätzlichen dynamischen Daten gemeinsam in der von uns entwickelten Engine durch. Dies spart nicht nur eine große Abhängigkeit, sondern macht die Lösung auch deutlich

Plattformunabhängiger. Da unsere Engine auch nach Javascript compiliert, ist die Anwendung vollständig im Browser lauffähig.

In allen innerstädtischen Tests wurden die Routen in unter einer Sekunde berechnet.

Wenn nötig (Routen über größere Entfernungen), kann auch bei der jetzigen Architektur hierarchisches Routing verwendet werden.

Entwicklung nativer Apps

Ursprünglich war eine Webanwendung vorgesehen, die später auch als Progressive Web App verwendet werden kann.

Dank der Expertise eines Teammitgliedes, wurde dann aber nicht nur eine Mobile First Strategy angewendet, sondern gleich Hybride Web Apps per Cordova entwickelt.

Nutzung von Machine Learning Methoden

Da unsere AI-Expertin aufgrund eines kurz vor Beginn des Förderungszeitraums begonnenen neuen Jobs keine Kapazitäten für uns zu entwickeln hatte, wurde sie ausschließlich beratend tätig. Die experimentelle Nutzung von Machine Learning Methoden zur Ermittlung von Parametern des Routingprofils wurde daher verschoben und zunächst durch einfachere Heuristiken ersetzt.

Statt dessen wurden mehr Ressourcen in das Tooling des Frameworks, insbesondere die Entwicklung einer Build- und CI-Pipeline investiert.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Die Graphhopper Routing Engine wurde in der Zwischenzeit um Funktionen erweitert, die uns neue Möglichkeiten bieten.

So werden seit September 2019 (v0.13) Vektor-Tiles ausgeliefert. Diese können von uns als eine Datenquelle verwendet werden.

<https://www.graphhopper.com/blog/2019/09/18/graphhopper-routing-engine-0-13-released/>

Im Austausch mit der OSM-Community wurden wir auf das Projekt StreetComplete aufmerksam gemacht. Über von uns definierte Regeln kann damit eine bestehende Community gebeten werden, Datensätze zu vervollständigen, die für besseres Rollstuhlrouting hilfreich sind.

<https://github.com/westnordost/StreetComplete/>

Richtlinie zum „Software-Sprint“

LibEuFin – Freie API und Sandbox für das europäische Finanzsystem

Schlussbericht

Zuwendungsempfänger:

Marcello Stanisci & Florian Dold, Fintech-Bibliotheksentwicklung GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S02 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Die europäischen Zahlungsdienstrichtlinie PSD2 soll unter anderem innovative Finanz-Anwendungen ermöglichen, indem Banken dazu verpflichtet werden, Programmierschnittstellen (APIs) für Kontoinformationen und Zahlungsauslösung anzubieten. Allerdings sind die APIs und Nachrichtenformate die von Banken bereitgestellt werden so vielfältig und komplex, dass kleinere Projekte und Privatpersonen diese kaum nutzen können.

Das Testen war bisher nur mit hohem Aufwand, echten Bankkonten bei diversen Anbietern und proprietärer Software möglich. Im Ergebnis werden Projekte für persönliche Finanzplanung und Banking für Bürger*innen praktisch nur von großen, profitorientierten Unternehmen entwickelt.

LibEuFin soll Entwickler*innen das Entwickeln und Testen von freien Anwendungen im Bereich Online-Banking erleichtern, beispielsweise für die persönliche Finanzplanung.

Zu diesem Zweck sollte im Rahmen dieses Projekts eine Bibliothek implementiert werden, die die verbreitetsten deutschen und europäischen Banking-Standardprotokolle (FinTS/EBICS/ ISO 20022) unter einer vereinfachten Schnittstelle zur Verfügung stellt. Zudem wurde eine Sandbox entwickelt, mit der Entwickler*innen ihre Banking-Anwendungen ausreichend automatisiert testen können, ohne Konten mit Testzugang bei mehreren Banken zu benötigen.

Das Projekt wurde ursprünglich in folgende Meilensteine aufgeteilt:

Meilenstein 1 (1. Sept. - 30. Sept. 2019): Herausarbeiten der wichtigsten Nachrichtenformate und Auftragsarten der Standards (FinTS/EBICS/ISO 20022)

Meilenstein 2 (1. Okt. - 31. Okt. 2019): Entwicklung einer Bibliothek für die Kryptographie und Nachrichtenformate

Meilenstein 3 (1. Nov. - 15. Nov. 2019): Entwicklung der Bibliothek für Kontoinformationen und Zahlungsauslösung

Meilenstein 4 (16. Nov. 2019 - 15. Jan 2020): Entwicklung der Banking-Sandbox

Meilenstein 5 (16. Jan - 31. Jan. 2020): Dokumentation und Lernmaterialien

Meilenstein 6 (1. Feb. - 29. Feb. 2020): Umsetzung einer Beispielanwendung

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

Die Zielgruppe des Projekts sind Entwickler*innen von Software, die Nutzern automatisierten Zugriff auf europäischen Bankkonten ermöglicht. Speziell kleine Unternehmen, Organisationen und Open-Source Entwickler*innen können von einer freien Implementation und Testumgebung profitieren.

Hierbei besteht Zusammenhang zu mehreren Themenfeldern und Zielen des Software Sprints:

1. *Softwareinfrastruktur*. Die Verfügbarkeit von einer Infrastruktursoftware wie LibEuFin, die vereinfachten programmatischen Zugriff auf Bankkonten bietet, ermöglicht die Automatisierung von Prozessen in Projekten und Organisationen. Existierende proprietäre Software für den elektronischen Zugriff auf Bankkonten bietet keine programmierbare Schnittstelle an, sondern nur vom Hersteller definierte feste Abläufe. Zudem muss Softwareinfrastruktur robust sein. Dies ist bei modernen Softwareprojekten grundsätzlich nur mit ausführlichen automatisiertem Testen möglich. Deshalb gibt LibEuFin der *Bank-Sandbox* einen besonders hohen Stellenwert.
2. *Data Literacy*. Der automatisierte Lesezugriff auf Bankkonten mit einer Open Source Softwarebibliothek wie LibEuFin ermöglicht die persönliche Finanzplanung und Auswertung der eigenen Ausgabegewohnheiten, ohne dabei diese Daten an Drittanbieter weiterzuleiten zu müssen.

Ausführliche Darstellung der Ergebnisse

Die folgenden Softwarekomponenten sind in diesem Projekt entwickelt worden:

- Die *LibEuFin Sandbox*, einer Serveranwendung welche einen Bankenserver simuliert der das EBICS-Protokoll (sowie zukünftig andere Protokolle wie FinTS) als Server anbietet. Dabei wird in sehr rudimentärer Weise die Benutzer- und Transaktionsverwaltung einer echten Bank nachgestellt.
- Der *LibEuFin Nexus*, einem Dienst der Anfragen in einer sehr leicht für Entwickler*innen zu benutzenden, auf HTTP+JSON basierten Schnittstelle entgegennimmt, und diese in verschiedene Transport- und Nachrichtenformate (derzeit EBICS und ISO 20022) übersetzt und an den Bankserver versendet. Dabei werden komplexe Prozesse wie Schlüsselverwaltung, Kryptographie und Protokolle zur signierten und segmentierten

Datenübertragung abstrahiert. Nutzer*innen der *LibEuFin Nexus* Programmierschnittstelle müssen sich darüber keine Gedanken machen.

Wie bereits erwähnt gibt es keine frei zugänglichen Testumgebungen für das EBICS-Protokoll oder ähnliche Bankprotokolle. Jedoch gibt es leicht für Privatkunden zugängliche, proprietäre Homebanking-Anwendungen, die gängig von deutschen Banken (GLS, Sparkasse, ...) angeboten werden.

Unsere innovative Strategie war, zuerst die Bank-Seite des Protokolls in der *LibEuFin Sandbox* nachzubauen. Mit mehreren proprietären Homebanking-Anwendungen konnten wir dann leicht verifizieren, ob die Sandbox das jeweilige Protokoll und die Nachrichtenformate korrekt implementiert und wo Anpassungen unseres Programms nötig sind. Somit konnten wir uns indirekt eine validierte Testumgebung verschaffen, welche normalerweise von Banken nicht angeboten wird.

In der nächsten Phase haben wir dann unsere *LibEuFin Sandbox* verwendet, um unsere client-seitige Implementation einfach validieren und testen zu können.

Abschließend konnten wir mit einem echten Firmenkonto bei der GLS Bank verifizieren, dass die Implementation des *LibEuFin Nexus* der Protokollspezifikation entspricht.

Wir haben folgende Funktionalität entwickelt, deren Sandbox-Implementation mit mehreren Homebanking-Anwendungen verifiziert und mit einem echten Bank-Server getestet:

1. Teilnehmerverwaltung und Initialisierung auf Basis von kryptographischen Schlüsseln. Bei diesem (in EBICS und FinTS verwendeten) Verfahren erzeugt die Homebanking-Anwendung bzw. der *LibEuFin Nexus* kryptographische Schlüsselpaare und sendet die öffentlichen Schlüssel digital an die Bank. Zudem muss der/die Kund*in die öffentlichen Schlüssel auf Papier ausdrucken, unterzeichnen und an die Bank senden, welche nach Überprüfung der Unterschrift den Teilnehmer freischaltet.
2. Abrufen von maschinenlesbaren Kontoauszügen und Vormerkposten
3. Zahlungsauslösung auf Basis der SEPA-Überweisung

Andere Vorgänge (z.B. Verwaltung von Lastschriften) können in Zukunft mit relativ geringem Aufwand implementiert werden, da LibEuFin eine umfassende Hilfsbibliothek als Grundlage anbietet.

Alle Komponenten wurden in der Programmiersprache Kotlin entwickelt, welche (dank der darunterliegenden JVM Plattform) die sehr komplexen XML-Standards unterstützt, welche von dem EBICS-Protokoll benutzt werden. Zudem kann eine in Kotlin entwickelte Anwendung sehr einfach in mobile Anwendungen für Android-Smartphones eingebunden werden.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Wir wurden bereits von mehreren Community-Projekten mit Interesse an LibEuFin angesprochen. Als Beispiel darf "MyExpenses" (<http://www.myexpenses.mobi/en/>) genannt werden, eine Open-Source-Anwendung für persönliches Finanzmanagement auf Mobilgeräten. Ein weiteres Projekt ist daran interessiert, LibEuFin für das Ticket-Kaufsystem einer Eventmanagement-Plattform zu benutzen.

Zudem haben sich bereits mehrere Studenten bei uns gemeldet, die im Rahmen des "Google Summer of Code 2020" Projekts (<https://summerofcode.withgoogle.com/>) LibEuFin weiterentwickeln wollen.

Die Dachorganisation "GNU" wird auch im Jahr 2020 von Google als Mentor-Organisation gefördert und bietet in diesem Rahmen auch Projektideen für LibEuFin an. Eine Liste von weiterführenden Projektideen zu LibEuFin ist hier zu finden:

<https://www.gnu.org/software/soc-projects/ideas-2020.html#taler>

Das Taler Projekt (gefördert in Runde 5 des Prototype Funds) wird zudem LibEuFin einbinden, um den praktischen Einsatz der Software-Infrastruktur für ein datenschutzfreundliches Bezahlsystems für "digitales Bargeld" zu ermöglichen.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Wegen der immens hohen Komplexität des EBICS-Protokolls und des ISO 20022 Standards haben wir uns vorerst auf die Implementation dieses einen Übertragungsprotokolls beschränkt. Jedoch wäre die Integration von anderen Übertragungsstandards wie z.B. dem deutschen FinTS Standard mit erheblich weniger Aufwand verbunden, da dieser ebenso wie EBICS Nachrichten in dem auf XML basierenden ISO 20022 Standardnachrichtenformat für Finanzdienste austauscht.

Zudem wurde vorerst auf die Entwicklung einer Beispielanwendung mit graphischer Oberfläche verzichtet, und stattdessen eine an Entwickler*innen gerichtetes Kommandozeilenprogramm als Beispielanwendung entwickelt. Eine umfassende und an Endbenutzer*innen gerichtete grafische Oberfläche soll in dem bereits erwähnten Google Summer of Code Projekt von einem Studenten entwickelt werden.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Quellcode: <https://git.libeufin.tech/>

Dokumentation mit API-Dokumentation und ausführlichen Erläuterungen zu dem EBICS-Protokoll: <https://docs.libeufin.tech/>

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Die Kostenplanung wurde weitgehend eingehalten. Jedoch musste wegen Mehraufwand bei den Meilensteinen 1-4 die für die beiden letzten Meilensteine zur Dokumentation und Entwicklung der Beispielanwendung gekürzt werden.

Zudem wurde die Entwicklung der Sandbox (Meilenstein 4) vorgezogen und parallel mit Meilensteinen 2-4 bearbeitet. Dies ermöglichte schnelles und zuverlässiges Testen mit der vorausgehend beschriebenen Methode der indirekt validierten Protokoll-Implementation.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Bei der Entwicklung der Anforderungen an die Architektur von LibEuFin wurde Rücksprache mit den Entwickler*innen des GNU Taler Projekts gehalten.

Richtlinie zum „Software-Sprint“

ESP - Unabhängiges Solar- Energie Meshnetz

Schlussbericht

Zuwendungsempfänger:

Corinna Aichele

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S03 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Vier Milliarden Menschen auf der Welt haben keinen Zugang zum Internet und mehr als eine Milliarde Menschen haben keinen Zugang zu Elektrizität. Mangel an Kommunikation – der sogenannte *Digital Divide* – erhöht die sozialen Ungleichheiten. Do-it-yourself-Lösungen wie das ESP-Unabhängiges Solar-Energie Meshnetz (im Folgenden als ESP-ISEMS abgekürzt) helfen, energieautarke Internet und Kommunikationstechnologie auch dort zur Verfügung zu stellen, wo gleichwertige Lebensverhältnisse nicht gegeben sind.

Arme Menschen am langen Ende des ökonomischen Long-Tail sind für kommerzielle Provider uninteressant, weil alles – auch Energieversorgung – mitgebracht werden muss und dem keine oder nur geringe Profiterwartungen oder gar Verluste gegenüberstehen. Insbesondere zuverlässige Elektrizität muss in Entwicklungs- und Schwellenländern oder Katastrophenfällen erst bereitgestellt werden. Energieeffiziente und preiswerte energieautonome Systeme sind dafür Voraussetzung.

Zweckentfremdung von IOT-Hardware dank ESP-ISEMS-Software (IOT = Internet of Things)

Ziel des ESP-Projekts für Unabhängige Solar-Energie Meshnetze (kurz ISEMS auf Englisch für Independent Solar Energy Mesh System) ist es, Menschen zu befähigen, ressourcenschonende, unabhängige und energieautarke Kommunikationsinfrastruktur zu errichten, wo sie am notwendigsten gebraucht wird, um z.B. die digitale Spaltung zu überwinden oder zumindest zu verkleinern. Dazu wurde eine Firmware für den ESP32-IOT-Chip von Espressif entwickelt, die den energiesparenden und kostensparenden Betrieb von Kommunikationsrelais zu minimalen Kosten auf der Basis einer IOT-Hardware entwickelt, die sehr wenig kostet und extrem wenig Energie

verbraucht. Immer mit dem Augenmerk, dass die verwendete IOT-Hardware nur minimale Ressourcen bietet (RAM, CPU) und daher clevere und effiziente Lösungen gefragt sind.

Die Firmware für diesen Zweck steht nun dank der im Projekt entwickelten ESP-ISEMS-Softwarelösung (eines LUA-basierten Softwaresystems auf der Basis der Betriebssysteme Free-RTOS) zur Verfügung. Eventuelle spätere Erweiterungen von ESP-ISEMS sind von interessierten Entwickler*innen in LUA leicht zu entwickeln. Parallel zur geförderten Entwicklung der Firmware von Seiten des BMBF wurde ausserhalb und unabhängig von der Förderung in den letzten drei Monaten eine neue angepasste Hardware entwickelt, die den ESP32-Chip mit einem Solarladeregler mit MPPT-Funktion integriert, nach dem die Entwicklung von Prototypen auf der Basis kommerziell verfügbarer universeller Entwicklerplatinen abgeschlossen war.

Die MPPT-Funktion erhöht den Energieertrag eines energie-autonomen Solarsystems um bis zu 40% gegenüber einem System mit einem einfachen Laderegler, der die Ladung nur abregelt, wenn der Akku voll ist. https://de.wikipedia.org/wiki/Maximum_Power_Point_Tracking Der Gesamtpreis des Systems, das nun aus der vom BMBF geförderten Software und der in der Endphase hinzugekommenen (ungeförderten) Hardwareentwicklung besteht, liegt jetzt bei unter 100 € in Kleinserie. In Großserie dürfte der Preis um etwa 40% sinken. 6 Prototypen wurden bereits getestet. Davon befinden sich vier Geräte mit ESP-ISEMS-Software bei interessierten Entwickler*innen und Anwender*innen, die bei einem ESP-ISEMS-Workshop auf dem 36C3-Kongress Ende Dezember teilgenommen haben. So ist bereits seit zwei Monaten Feedback in das Projekt eingeflossen.

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

ESP-ISEMS unterstützt ökonomisch benachteiligte Menschen, die in abgeschiedenen Regionen oder nach einer Katastrophe preiswert drahtlose Netzwerk-Infrastruktur mit WLAN aufbauen wollen oder weiße Flecken in der Netzwerkversorgung schließen wollen, z.B. durch ein solarbetriebenes Kommunikationsrelais auf dem Hügel oder Berg, der ansonsten eine breitbandige Richtfunkstrecke blockieren würde. Durch die Portierung des Systems auf eine IOT-Plattform sind die Kosten und der Materialaufwand stark gesunken – eben unter 100€ für ein Kommunikationsrelais in Kleinserie mit allen dafür erforderlichen Komponenten im Vergleich zu 220-250€ für ein kleines System mit Linux auf OpenWRT-Basis. Ebenso kann das System zur Energieversorgung eines externen Kommunikationssystems mit größerer Leistung dienen und dieses aus der Ferne schalten.

Hilfe zur Selbsthilfe zu ermöglichen ist für die von klassischen ökonomischen Systemüberlegungen abgehängten Bevölkerungsgruppen ein naheliegender Schritt, wenn wir die Lösung für systemische Probleme neu denken und ändern wollen. Hier kann ESP-ISEMS helfen. Die Software ist Open-Source und kann als Grundlage für weitere Entwicklungen dienen. Auch die fertiggestellte Hardware-Entwicklung ist unter Open-Hardware-Lizenz. Auf Wunsch und dank Feedback der Community wird die Hardware gerade zusätzlich um Hardware-Schnittstellen für Umweltsensoren und weitere Kommunikationssysteme (z.B. LoRaWAN) erweitert.

Ausführliche Darstellung der Ergebnisse

Das Projekt wurde erfolgreich abgeschlossen und alle geplanten Funktionen und Meilensteine wurden implementiert und getestet. Die Steuerung und Auswertung eines MPPT-Ladereglers (MPPT = Maximum Power Point Tracking) wurde in LUA implementiert, Daten werden in LUA analysiert, aufbereitet, im CSV-Format per HTTP-Protokoll bereitgestellt und über einen in den Code integrierten Webserver in maschinen- sowie menschenlesbarer Form zur Verfügung gestellt. Das System benötigt nur minimale Ressourcen und ist voll kompatibel mit der ISEMS-Web-App und ISEMS-Data-Collector aus Förderrunde 3. Die Lösung aus Runde 3 war noch auf Linux und wesentlich größere und energieaufwendigere Hardware für die Solarknoten angewiesen, mit minimal etwa vierfachem Energieverbrauch gegenüber der jetzt bestehenden Lösung. Darüber hinaus wurden hinzugefügt: Lua-Shell über Telnet als Debug-Schnittstelle, eine Unix-artige minimale Shell (Kommandozeilen-Interface), Telemetrie-Daten-Upload über MQTT-Protokoll im JSON-Format, Fernsteuerung über HTTP-Kommandos und kryptographische Authentifizierung nach einem vereinfachten Digest-Access-Authentication-Protokoll, angelehnt an das IETF Protokoll RFC 7616, um RAM gegenüber einer SSL/TLS-Lösung zu sparen. Der ESP32 besitzt nur 520 kByte RAM – etwa ein Achttausendstel eines Laptops oder Desktop-Computers mit 4 GB RAM. Trotzdem ist das System nach nur 8 Sekunden betriebsbereit. Ausserdem wurde ein Linux-Kommandozeilen-Tool entwickelt, um das System von einem Linux-basierten Router per skriptbarem Fernzugriff zu steuern. So kann das System aus der Ferne beispielsweise automatisiert aktiviert oder deaktiviert werden. Auch für Tüftler*innen und Bastler*innen mit allgemeinen Interessen an drahtloser Sensorik und unabhängiger Energieversorgung dürfte die ESP-ISEMS-Plattform wegen der effizienten und optimierten Lösung interessant sein. Vorbereitungen für eine kleine Serienproduktion der für ESP-ISEMS angepassten Hardware laufen.

Das Projekt hat bereits viel positives Feedback und Support von anderen Open-Source-Entwickler*innen bekommen. Persönlich konnte ich meine Kenntnisse über die ESP-Hardware-Plattform erweitern. Es werden aus diesem Projekt noch weitere Projekte hervorgehen.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Die Kosten für einen Solarknoten als energieautonome WLAN-Infrastruktur wurden durch das Projekt auf unter 40% gegenüber der Lösung aus Förderungsrunde 3 gesenkt. Auch aus anderen Open-Source- und Maker-Communities gibt es Interesse und Anfragen für das Projekt. Funktionierende Hardware-Prototypen werden wie erwähnt bereits von der Community betrieben und getestet. Es ist zu erwarten, dass auch andere Projekte, die z.B. nach einer energieautonomen Lösung für relativ energiehungrige Umweltsensoren (Feinstaub, Schadstoffe) suchen, von ESP-ISEMS profitieren werden.

Die Arbeit hat meine persönlichen und fachlichen Kenntnisse für die Hardwareplattform und Telemetrie generell ausgebaut. Im Moment arbeite ich bereits an einem Umweltsensor-System, dass auf der ESP-ISEMS Firmware und den gewonnenen Erfahrungen aufbaut. Ich habe gerade Sensoren für Feinstaub, CO², Luftfeuchte, Luftdruck gekauft. Support für diese Sensorik wird jetzt nach und nach als Erweiterungen für ESP-ISEMS hinzugefügt.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Arbeiten, die zu keiner Lösung geführt haben, gab es in diesem konkreten Projekt nicht. Am Anfang des Projektes standen zwei ESP-Chips zur Auswahl, ESP8266 und ESP32. Der ESP8266 ist schon länger auf dem Markt, veraltet und kostet etwa einen Dollar weniger, leistet aber auch viel weniger. Vor allem hat er mit 96 kByte RAM nur sehr wenig Arbeitsspeicher. Daher wurde schon zu Anfang nach kurzen Untersuchungen der Entwicklungsfokus auf den ESP32 beschränkt, da die alte Plattform sich nicht für den Entwicklungsaufwand rentiert. Für die erforderlichen Recherchen wurde nur wenig Zeit aufgewendet.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Der Sourcecode, Dokumentation und Firmware-Images sowie Begleitwerkzeuge sind über Github erhältlich: <https://github.com/ISEMS/ISEMS-ESP32>

Weitere Informationen für die breite Öffentlichkeit sind über die Projekt-Webseite www.isems.de zugänglich.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Das Projekt verlief in seiner Entwicklung erstaunlich reibungslos, bis auf die Tatsache, dass die Analogmessungen des ESP32-Chips nachträglich kalibriert werden mussten, um die Präzision der Messwerte zu erhöhen. Dafür wurde zusätzlich eine Kalibrierungs-Software geschrieben. Allerdings habe ich am Ende aus purer Begeisterung für die ESP-ISEMS-Plattform, die sich immer mehr wie ein minimales Linux-System anfühlt noch weitere Funktionen hinzugefügt und Zeit investiert, die nicht abgerechnet werden konnte. Auch motiviert durch die Tatsache, dass aus der Community Softwarepatches und Erweiterungswünsche angetragen wurden.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Von der Machart und vor allem der Grundidee her, ist das ESP-ISEMS Projekt meines Wissens nach bislang einzigartig. Auf die unkonventionelle Idee, einen WLAN-Chip aus dem Bereich IOT (Internet of Things – Internet der Dinge) von seinem angedachten Zweck zu entfremden und stattdessen als extrem kosten- und stromsparende Lösung für energieautonome WLAN-Infrastruktur mit Solarenergie einzusetzen, ist offenbar noch niemand bislang gekommen und es gibt – zumindest bislang – auch keine mir bekannten ähnlichen Projekte. Von der Hive-Eyes-Community, die Bienenstöcke mit telemetrischen Systemen überwacht, gibt es aktives Interesse an ESP-ISEMS. Von dieser Seite kam auch der Vorschlag das MQTT-Telemetrieprotokoll zusätzlich zu ESP-ISEMS hinzuzufügen, was auch bereits von mir in Zusammenarbeit mit der Hive-Eyes-Community umgesetzt wurde.

Berlin, den 5.3.2020



GreenerWP - Nachhaltige Websites mit WordPress (ehemals LowTechWP)

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S04 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

1. Kurze Darstellung der Aufgabenstellung und Motivation

Um ökologische Probleme wie den Klimawandel effektiv zu bekämpfen, müssen in vielen Bereichen unserer Gesellschaft Optimierungen vorgenommen werden. Das Internet wird ein immer größerer Energie-Verbraucher. Mit neuen Technologien wie der 5G-Mobilfunkkommunikation und erheblichem Wachstum der Cloud wird sich das Problem die nächsten Jahre verschärfen. Ein effizienteres Web wird den Klimawandel nicht stoppen, ist aber ein notwendiger Beitrag dazu. WordPress ist Grundlage eines großen Teils der Website-Projekte (je nach Quelle bis zu 32 % Anteil aller Webseiten); eine Effizienz-Steigerung und die Erprobung von Alternativen wird also ein wichtiger Beitrag sein.

Um das Problem zu lösen, waren technische und inhaltliche Unterstützung der WordPress-Nutzer*innen angedacht. Konkret war geplant:

- Aufbau einer WordPress-Umgebung auf einem Einplatinencomputer mit autarker Energieversorgung über ein Solar-Panel
- Umsetzung eines Referenz-Themes auf Basis des aktuellsten offiziellen WordPress-Themes
- Umsetzung eines WordPress-Plugins
- Umsetzung von Optimierungs-Plugins für reduzierte Bildgrößen, Cache-Einrichtung, Content-Analyse, weitere Optimierungen
- Untersuchung der verschiedenen Optimierungen auf Auswirkungen auf den Energieverbrauch des Servers und Clients
- Mess-Umgebung, die über ein Web-Interface erreichbar ist
- Anpassung von offiziellen WordPress-Themes

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

Zielgruppe des Projektes sind WordPress-Website-Betreiber*innen ohne tiefergehende technische Kompetenzen, denen ökologische Nachhaltigkeit wichtig ist und die in der Lage sind, ein Wordpress-Plugin oder ein Theme zu installieren.

Die Plugins und Themes des Projektes sind in den WordPress-Datenbanken vertreten. Da auch andere Probleme mit den Plugins und Themes gelöst werden können, beispielsweise ein besseres Page-Ranking dank schnellerer Ladezeiten, werden verschiedenste Betreiber*innen erreicht und überzeugt, die Plugins und Themes einzusetzen.

Durch das Projekt werden Nutzer*innen aus verschiedenen Bereichen der Gesellschaft dabei unterstützt, nachhaltige Websites zu erstellen und sich Kompetenzen in dem Bereich zu erarbeiten, welche auch in anderen Projekten angewendet werden können.

3. Ausführliche Darstellung der Ergebnisse

Die wesentlichen Komponenten wurden umgesetzt: Website mit theoretischem Hintergrund und Anleitungen, WordPress-Plugin, Referenz-Theme, Solar-betriebener Einplatinen-Computer, Website-Scanner.

Der zentrale Punkt des Projektes ist die **Projekt-Website** (<https://greenerwp.net>). Hier erhalten Nutzer*innen einen ersten Blick über das Projekt. Eine Schritt-für-Schritt-Anleitung führt Nutzer*innen durch die Anwendung des Projektes.

Neben dem Überblick bietet die Website auch Anleitungen und Vertiefungen. Nutzer*innen sollen nicht nur die Werkzeuge anwenden, sondern auch ein grundlegendes Verständnis der Thematik entwickeln. Dafür gibt es auch innerhalb der Werkzeuge immer wieder Verweise auf die Website, um konkrete Schritte besser nachvollziehen zu können. Dadurch wird ebenfalls ermöglicht, die gewonnen Kompetenzen in anderen (auch WordPress-fernen) Projekten anzuwenden. In einem integrierten Blog wird über die neuesten Entwicklungen berichtet.

Für Menschen, die zum Projekt beitragen wollen, gibt es einen eigenen Bereich, der versucht, die Schwelle zum Beitragen niedrig zu halten und für eine konstruktive und freundliche Atmosphäre zu sorgen.

Das **WordPress-Plugin** ist das wichtigste Werkzeug für Nutzer*innen des Projektes. Es ist in der offiziellen WordPress-Datenbank und kann somit von Nutzer*innen direkt über die WordPress-Admin-Oberfläche installiert und genutzt werden.

Das Plugin besteht aus Analyse-Werkzeugen, Optimierungen und Schritt-für-Schritt-Anleitungen genannt Rezepte.

Mit den Analyse-Werkzeugen lässt sich herausfinden, welche konkrete Seiten der Website am meisten zum Datenverkehr der Website und damit zum Energieverbrauch (insbesondere im Netzwerk) beitragen. Der Datenverbrauch wird dafür direkt im Webbrowser der Besucher*innen gemessen und ausgewertet. Betreiber*innen können dann einzelne Websites optimieren.

Rezepte erleichtern es, Optimierungen durchzuführen, die beispielsweise dritte Plugins benötigen. Im Stil einer Todo-Liste können Schritt für Schritt Verbesserungen durchgeführt werden, beispielsweise Caching durch Installation und Konfiguration eines Caching-Plugins aktiviert werden.

Weiter bietet das Plugin konkrete Optimierung, beispielsweise die Einstellung der Bilder-Qualität. Zudem existiert ein Feature, mit dem für Besucher*innen der Datenverbrauch visualisiert werden kann, insbesondere im Vergleich zum durchschnittlichen Datenverbrauch von Websites.

Für das Projekt wurde ein **Theme** entwickelt, das auch von der Projekt-Website genutzt wird. Ziel war Effizienz und Nutzungsfreundlichkeit.

Als Anschauungsbeispiel wurde WordPress auf einem **Einplatinen-Computer** installiert und auf minimalen Stromverbrauch optimiert. Dadurch war es möglich, eine darauf betriebene Website mit einer kleinen Solar-Anlage zu betreiben.

Entwickelt wurden zudem Werkzeuge, um den Batterie-Stand und Ladestatus der Solaranlage zu messen und auf der Website zu visualisieren. Weiter ist es möglich, den Einplatinen-Computer durch einen vorgeschalteten Microcontroller bei Bedarf (niedriger Ladestand z.B.) herunterzufahren und nach einer bestimmten Zeit wieder hochzufahren.

Das ganze lässt sich auch als Eye-Catcher beispielsweise auf Konferenzen nutzen, um auf das Projekt und das Thema aufmerksam zu machen.

Zudem sind weitere Anwendungsfälle denkbar, beispielsweise der Betrieb von Websites in autarken

6. Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Die Website (<https://greenerwp.net/>) liefert alle wichtigen Informationen zum Projekt. Nutzer*innen können dort einen Überblick bekommen und haben Zugang zu den entwickelten Werkzeugen.

Der Quellcode des Projektes befindet sich in öffentlich zugänglichen Repositories (<https://gitlab.com/greenerwp>).

Nutzer*innen von WordPress können das WordPress-Plugin in der WordPress-Datenbank wiederfinden (<https://wordpress.org/plugins/greenerwp/>).

Der Mitschnitt eines Konferenz-Beitrages gibt einen kurzen Überblick über das Projekt (<https://media.ccc.de/v/36c3-91-helping-wordpress-users-build-climate-friendly-websites>).

7. Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Während des Projektverlauf gab es keine besonderen Ereignisse, die eine Anpassung der Planung erforderlich machten. Im Verlaufe des Projektes verschieben sich etwas die Schwerpunkte, das wirkte sich aber auch nicht Wesentlich auf die Zeitplanung aus.

8. Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Kurz vor Halbzeit des Projektes hatte ich ein Treffen mit einem Entwicklungsteam eines anderen Projektes mit teilweise überschneidender Zielsetzung, aber anderen Schwerpunkten. Der Austausch mit diesem Team half mir dabei, Schwerpunkte innerhalb meines Projektes etwas anzupassen.

Konkret fokussierte ich mich weniger auf den Solar-Teil des Projektes und mehr auf die Software-Lösungen. Zudem führte das zu einer Umbenennung des Projektes von "LowTechWP" hin zu "GreenerWP".

Richtlinie zum „Software-Sprint“

LiViR - Linux- Festplattenverschlüsselung im Ruhezustand

Schlussbericht

Zuwendungsempfänger:

Dittler Meurer GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S05 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Notebookfestplatten werden häufig verschlüsselt, um ihre Daten gegen physischen Diebstahl zu schützen. Im Gegensatz zu proprietären Lösungen, funktioniert dies unter Linux bisher nur während das Gerät vollständig heruntergefahren ist. Tatsächlich ist dies bei Notebooks heute selten der Fall. Im Rahmen des Projekts sollte die größte unkommerzielle Linux-Distribution Debian derart erweitert werden, dass auch das einfache Schließen des Notebookdisplays zum Schutz vor physischem Zugriff führt. Nutzer*innen werden dann beim Öffnen des Gerätes erneut nach ihrem Festplattenpasswort gefragt. Dazu waren folgende Meilensteine vorgesehen:

1. Vergleichen der bisherigen (unvollständigen) Lösungen und Kommunikation mit deren Entwickler*innen.
2. Notwendigen Kernel Patch mit den Linux Kernel Entwickler*innen diskutieren und zur Aufnahme in den Debian Linux Kernel vorschlagen.
3. Proof of Concept weiter verfeinern.
4. Ausführliches Testing Dokumentation des Verfahrens und damit einhergehender Risiken
5. Integration der Suspend-Funktion und zugehöriger Konfiguration in Betriebssystem
6. Entwicklung von Plugins, um weitere sensible Daten aus RAM zu löschen (z.B. private SSH/GnuPG Schlüssel)
7. Integration der Konfiguration des suspend-Features in Gnome Power Management
8. Vorstellung der Lösung bei Betreuern anderer Linux-Distributionen

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

Mit dem Projekt werden verschiedenen Ziele des „Software-Sprint“ verbessert. Grundsätzlich wird vor allem die IT-Sicherheit der Nutzer*innen von Debian-basierten Linuxdistributionen verbessert. Ganz konkret, wird die Sicherheit von Nutzer*innen von Linux-Festplattenverschlüsselung verbessert, indem ihre Laptops auch im Ruhezustand geschützt sind. Freie Software wird aus guten Gründen, viel im Bereich der Zivilgesellschaft und des Journalismus eingesetzt. Die Sicherheit dieser Menschen zu erhöhen, halten wir für einen wichtigen Beitrag zu Civic Tech. Des weiteren wurde auch die Softwareinfrastruktur von Debian und Cryptsetup durch unser Projekt verbessert. Projekte, auf die zehntausende von Menschen täglich vertrauen.

Ausführliche Darstellung der Ergebnisse

Wir haben erfolgreich einen funktionierenden Prototypen entwickelt, der auf vielen Systemen zuverlässig funktioniert. Auch haben wir eine Verbesserung im Linux Kernel entwickelt, die nun offiziell aufgenommen und mit dem Release 5.6 des Linux Kernel veröffentlicht wurde. Unser Code läuft somit auf allen aktuellen Linuxcomputern. Die entwickelte Software ist in das offizielle Debian-Pakete für Linux Festplattenverschlüsselung aufgenommen worden, was eines unserer Hauptziele war. Es kann über die offiziellen Debian-Repositories heruntergeladen werden. Aufgrund von Problemen auf mancher Hardware ist das Paket derzeit noch als „experimentell“ bezeichnet. Wir führen unsere Bemühung Hardwareinkompatibilitäten zu beheben seit dem Ende des Förderzeitraums fort.

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 weiter geholfen. Wichtig war hier vor allem die Planung, welche Funktionalität eine Nutzer*in erwartet, wenn sie unser Paket installiert. Sinnvolle Standardwerte und eine verständliche Anzahl an Einstellungen zu geben war der Schwerpunkt in dieser Beratung. Weiterhin gaben sie uns gute Tipps zur Kommunikation mit der Linux Kernel Community.

Einer breiten Öffentlichkeit wurde unser Projekt am 02.02.2020 durch einen Vortrag auf der FOSDEM 20 vorgestellt. FOSDEM ist die größte Open Source-Messe Europas mit mehr als 8000 Teilnehmer*innen. Es ergaben sich spannende Diskussion über die Integration mit anderen Projekten, welche die physische Sicherheit von Linux-Computern erhöhen.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Zielgruppe waren insbesondere Menschen, die Ihre Daten vor gezielten physikalischen Angriffen schützen müssen - also Journalist*innen, Aktivist*innen, oppositionelle Politiker*innen in aller Welt. Des weiteren profitieren Nutzer*innen von Festplattenverschlüsselung auf Linux-Laptops allgemein, da ihre Daten im Ruhezustand besser geschützt sind. Auch größte Konzerne vertrauen auf die Verschlüsselung mittels Cryptsetup und profitieren letztlich von unserer Entwicklung

Da die Lösung in Debian implementiert wurde, wird sie mittelfristig alle Nutzer*innen von Debian-Derivaten (etwa Ubuntu, Tails, Linux Mint) erreichen, sobald diese auf eine aktuelle Version ihres Betriebssystems aktualisieren. Unsere Software wird bereits zuverlässig auf Ubuntu-Systeme

eingesetzt. Eine Adaption für Qubes OS ist in Diskussion. Dies ist eine völlig neuartige Linuxdistribution mit dem Schwerpunkt auf Sicherheit durch Isolation. Unsere Erwartungen wurden übertroffen, da unsere Arbeit bereits für Fedora adaptiert wurde. Fedora ist die zweite große Linux-Distribution neben Debian, von der sich weitere Distribution ableiten.

Da das Projekt noch nicht ganz abgeschlossen ist, arbeiten wir weiter um den experimentellen Status der Pakete zu entfernen und es in einen breit getesteten, zuverlässigen Zustand zu bekommen. Wie oben geschrieben, handelt es sich hier um Probleme der Hardwareinkompatibilität, welche immer recht komplex sind.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Im kleinen gab es immer mal wieder Ansätze, die sich später als nicht zielführend raus gestellt haben. Im Großen aber hatten wir von Beginn an eine gute Vorstellung von den nötigen Teilschritten. Während der Entwicklung hatten wir kurzzeitig versucht, nicht nur Debian, sondern alle Systemd-basierten Linuxdistributionen anzuvisieren. Da die Entwickler von Systemd unserer Idee allerdings nicht besonders offen gegenüber waren, mussten wir diesen Plan wieder aufgeben. Systemd hat einen anderen Ansatz gewählt, welcher nur die Home-Laufwerke der Nutzer*innen verschlüsselt, da sie perspektivisch den Rest des Systems unveränderbar machen wollen. Diese Lösung liegt allerdings für Desktopsystem noch in weiter Ferne, während unsere Lösung heute funktioniert.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Unser Projekt inklusive Sourcecode, Problemtracking und Anleitungen findet sich unter <https://salsa.debian.org/mejo/cryptsetup-suspend>. Anleitungen zur Installation finden sich unter <https://salsa.debian.org/mejo/cryptsetup-suspend/-/wikis/home>. Weitere Dokumentation wird mit dem Paket „cryptsetup-suspend“ installiert und ist über das Werkzeug „man“ abrufbar. Die Folien und das Video der Präsentation des Projekts auf FOSDEM 20 ist unter https://fosdem.org/2020/schedule/event/dip_close_lid_encrypt/ abrufbar.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Wir waren wesentlich länger mit der Kommunikation mit der Linux Kernel Community beschäftigt, als wir eingeplant hatten. Daher war auch unsere Beratung mit Simply Secure stark auf diesen Bereich fokussiert. Dies ist hauptsächlich darauf zurück zu führen, dass wir als Einzelpersonen an diese Community heran getreten sind und nicht durch eine große Firma unterstützt worden, welche die Kernel-Entwicklung mitfinanziert. Letztlich waren wir aber in unserem Projektzeitraum erfolgreich und konnten die zusätzlich notwendig Zeit durch gradlinige Entwicklung wieder gut machen. Hier erwies sich unsere Vertrautheit mit der Codebasis, sowie das gute Durchdenken des prinzipiellen Problem als besonders hilfreich. Die von uns geplanten Arbeitsziele konnten größtenteils erfüllt werden. Nur „7. Integration der Konfiguration des suspend-Features in Gnome Power Management“ musste aus Zeitgründen entfallen. Wir haben uns hier für einen generischen Weg der Konfiguration über eine Konfigurationsdatei entschieden. Die grafische Integration mit Gnome hätte letztlich zu wenig Nutzen für sehr großen Aufwand gebracht. Probleme mit spezieller Hardware waren in unserem Projektplan nicht vorgesehen und müssen daher auch im Nachhinein gelöst werden.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

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

24. März 2020

Richtlinie zum „Software-Sprint“¹²⁷⁷

Dark Crystal

Schlussbericht

Zuwendungsempfänger:

Pegasus GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen **01IS19S06** 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 zunehmende Internetverfügbarkeit hat eine Verschiebung hin zu leichten Geräten mit geringer Leistung bewirkt, macht Nutzende jedoch immer mehr von ressourcenintensiven Zentralinfrastrukturen abhängig.

Peer-to-Peer-Systeme mit niedriger Bandbreite bieten eine vielversprechende ressourcenschonende Alternative. Das Problem des Schlüsselmanagements ist jedoch ein Hindernis für ihre Akzeptanz. Nutzende heute sind es gewöhnt, diese Verantwortung an zentrale Anbieter abzugeben.

Damit sich dezentrale Architekturen durchsetzen können, braucht es Lösungen für Probleme wie Geräteverlust oder -fehler und vergessene Kennwörter.

Mit unserem Projekt planen wir:

- die Erarbeitung einer transportagnostischen Protokollspezifikation für ein System zur Wiederherstellung kryptographischer Schlüssel;
- die Erstellung eines Entwickler-Toolkit zur Integration von Dark Crystal in vorhandene Projekte – mit Anleitungen, Methodenteil und Code-Modulen;
- die Erweiterung von Dark Crystal um die Funktion des Widerrufs und der Neuausgabe von Schlüsseln.

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?

Unsere Zielgruppe sind Entwickler_innen, die ein Schlüsselverwaltungssystem in ihre Software implementieren wollen.

Im Speziellen wendet sich Dark Crystal an Projekte, in denen die Schlüsselintegrität eine wichtige Rolle spielt. Dies ist in der Regel bei „distributed Systems“ der Fall, da sie ohne einen authentifizierten Server auskommen.

Diese Systeme sind vorteilhaft hinsichtlich der Datensicherheit, da Anwendende die volle Kontrolle über ihre Daten behalten. Sie schneiden auch aus einer ökologischen Sicht sehr gut ab, weil sie kollaborative Anwendungen ohne energieintensive Datenzentren ermöglichen.

Dark Crystal macht diese Systeme sicherer und leichter nutzbar und unterstützt damit deren Verbreitung.

Entwickelnde profitieren von unserer Arbeit durch die Veröffentlichung unseres Werkzeugkastens für Entwickler_innen, den wir frei zugänglich veröffentlichen. Neben unseren Open Source- Modulen enthält dieser Werkzeugkasten auch eine umfangreiche Dokumentation mit Anleitungen, Empfehlungen und Anwendungsbeispielen.

Wir haben darauf geachtet, diesen Werkzeugkasten so flexibel wie möglich zu gestalten. Daher haben wir an Stelle von festen Codes vor allem 'Design Muster' entwickelt, die in vielfältigen Situationen anwendbar sind.

Entwickelnde, die Dark Crystal benutzen, können daher unsere Ideen so umbauen und implementieren, wie es für die Besonderheiten ihres Projektes passt.

Das macht unser Projekt zukunftssicher im Hinblick auf neue Programmiersprachen und Technologien.

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:

- unser Prinzip zur Generierung und Verschlüsselung von Shares in einer allgemeingültigen Form formuliert und vom Transportprotokoll unabhängige Module in NodeJS entwickelt;
- einen allgemeingültigen Standard für unser Protokoll erstellt;
- Berichte über die sozialen und technischen Herausforderungen des Modells zur Verfügung gestellt;
- das Protokoll um ein System zur Sperrung und Neuausgabe von Schlüsseln mit BLS-Threshold-Signaturen erweitert.
- eine allgemeine Einführung zu unserem Konzept und seinen Hintergründen geschrieben.
- Anwendungsvorlagen für die verschiedenen Nachrichtentypen, die wir im Protokoll benutzen, veröffentlicht;
- einfache Erklärungen zu den wichtigsten von uns benutzten Verschlüsselungstechniken erstellt.
- ein Anwendungsbeispiel zur Erstellung einer Benutzungsoberfläche für einen Email- Klienten zur Verfügung gestellt, in welchem das Dark Crystal-Protokoll zur Sicherung der Verschlüsselung und zum Zeichnen von Schlüsseln implementiert ist.

Mit Hilfe von Design-Coaching Sitzungen mit Simply Secure, die von der Open Knowledge Foundation organisiert wurden, haben wir ein Dokument mit Empfehlungen für das Designen von Benutzungsoberflächen erstellt. Simply Secure haben selbst erst kürzlich zu Herausforderungen im Design bei dezentralisierten Systemen geforscht. Dies ist genau das Thema, welchem sich unser Projekt widmet. Das Coaching durch Simple Secure war daher für uns äußerst hilfreich.

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?

Mehrere Softwareprojekte haben bereits ihr Interesse daran bekundet, unser System der Schlüsselwiederherstellung in ihrem Projekt zu verwenden.

In unserem nächsten Projekt, welches in Kürze beginnt, werden wir mit „Briar“ zusammen arbeiten, einer mobilen Nachrichten-App mit Schwerpunkt auf Datenschutz und Schutz der Privatssphäre, um ein System zur

Schlüsselwiederherstellung und „remote Wipe“ in deren Anwendung einzubringen. „Briar“ wurden in der Vergangenheit ebenfalls durch Software-Sprint unterstützt.

Wir haben durch das Projekt viel dazugelernt, vor allem über Aspekte des Designs durch die von der Open Knowledge Foundation organisierten Coaching-Sessions.

Wir haben auch durch das Schreiben des Berichtes und der Module zu den Threshold-Signaturen viel gelernt. Dies war der stärker experimentelle Teil des Projektes, welcher aus einem Anfang 2019 veröffentlichten Forschungspapier entstanden ist, an dem wir mitgeschrieben haben. Dieser Teil beinhaltete die Arbeit an einem für uns neuen Verschlüsselungssystem und die Zusammenarbeit mit Entwickelnden, mit denen wir bisher noch keinen direkten Kontakt hatten. Dieser Teil des Projektes war für unsere persönliche Entwicklung besonders spannend und förderlich.

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?

Das Schlüsselneuausgabesystem auf der Basis von Threshold-Signaturen war der eher experimentelle Teil dieses Projektes und wir sind sehr dankbar für die Möglichkeit, daran zu arbeiten. Wir haben im geförderten Zeitraum dazu ein funktionierendes Modell entwickelt und eine Beschreibung geschrieben. Dennoch ist dieser Teil des Projektes weiterhin weniger stabil und befindet sich in der Testphase. Hier ist noch Arbeit nötig, bevor es einem breiteren Publikum empfohlen werden kann.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

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

Die schriftliche Komponente des Werkzeugkastens für Entwickelnde ist über eine einfache Webseite unter folgendem Link verfügbar:

<https://dark-crystal.gitlab.io/>

Der Quellcode aller unserer Code- Module und unsere verschriftlichtigten Forschungsergebnisse, alle veröffentlicht unter der Affero General Public License version 3, finden sich unter folgendem Link:

<https://gitlab.com/dark-crystal>

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?

Natürlich ist es immer der Fall, dass bestimmte Dinge länger dauern als geplant etwa aufgrund von Fehlern im Code oder anderen Schwierigkeiten.

Aufgrund unserer personellen Verfügbarkeit und aus sonstigen persönlichen Gründen haben wir die Abfolge der Teile unseres Projektes gegenüber dem ursprünglichen Ablauf verändert. Wir haben die Arbeit an den Threshold-Signaturen zuerst abgeschlossen und anschließend an der Schlüsselwiederherstellungskomponente gearbeitet.

Die regelmäßigen Anrufe und die Unterstützung durch die Open Knowledge Foundation waren sehr hilfreich, um ein gutes Projekt Management zu etablieren und haben uns die Möglichkeit zum Dialog und Erfahrungsaustausch mit den Angehörigen anderer Projekte gegeben.

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?

Ein Teil der Grundlagenarbeit, auf der Dark Crystal aufbaut, wurde mit Unterstützung der Ethereum Foundation erarbeitet. Dieses Projekt endete jedoch bereits in 2018.

Richtlinie zum „Software-Sprint“

EmissionsAPI – Emissions Daten API basierend auf Satellitendaten

Schlussbericht

Zuwendungsempfänger: Brockmeyer Haardiek Kerssen Kerssen Kiesow Rosemann GbR

Timo Nogueira Brockmeyer

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S07 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Am Anfang unserer Idee stand die Erkenntnis, dass politische Instrumente zur Umsetzung von Klimazielen in ihrer Effektivität nur für einen sehr kleinen Personenkreis überprüfbar sind. Dementsprechend bleibt die öffentliche Diskussion für viele Menschen abstrakt und schwer greifbar. Um einem breiteren Publikum die Möglichkeit zu geben, sich eine durch Fakten gestützte Meinung zu bilden, sehen wir es als notwendig an, einen einfachen Zugang zu den Datenquellen zu schaffen. Dieser Zugang kann konkret durch interaktive Visualisierungen oder einfach abrufbare Statistiken geschaffen werden. Im Laufe der Recherchen zu unserem Projekt haben wir dabei schnell erkannt, dass die Daten, die wir von der ESA nutzen wollten, zwar öffentlich zugänglich sind, aber nicht auf einfache Art durch standard Webtechnologien zu verarbeiten sind. Daher haben wir uns entschieden, diese fehlende digitale Infrastruktur zu erschaffen, die die wissenschaftlichen Daten aufbereitet und durch ein einfaches Application Programming Interface (API) in einer automatisierbaren, dynamischen Weise ausliefern kann. Diese Schnittstelle kann dann wiederum von einem verbreiteten Personenkreis aus Web-Entwicklern und Data Scientists genutzt werden, um eigene Projekte umzusetzen. Beispielhaft könnte man die Effektivität von Umweltvorschriften - wie Dieselfahrverbote in Innenstädten oder neue Schwefelgrenzwerte in der Schifffahrt - durch einen Vorher/Nacher-Vergleich auf einer Landkarte bewerten und dadurch Argumente mit Daten zu überprüfen.

In der anfänglichen Planung haben wir die im Folgenden aufgeführten Meilensteine definiert. In der Arbeitsweise haben wir dabei ein vertikales Prototyping angestrebt, so dass immer an allen Meilensteinen gleichzeitig gearbeitet wird.

Meilenstein 1: Datenaufbereitung

- Entwicklung der Algorithmen zur automatischen Aufbereitung der Daten
- Entwicklung der Rechenmodelle zur effizienten Datenreduzierung

Meilenstein 2: Datenmodell

- Einbindung der Sentinel-API
- Caching der Daten
- Datenstrukturen innerhalb des Dienstes

Meilenstein 3: Infrastruktur

- Berechnung und Datenfluss
- automatisches und nachhaltiges Deployment

Meilenstein 4: API

- Definition der API Endpunkte
- Anbindung an Module des Backends
- Beispiel-App einer Visualisierung in OpenStreetMap

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

Die primäre Zielgruppe für unser Projekt sind Entwickler und Data Scientists, die Visualisierungen oder Analysen basierend auf den Emissionsdaten des Sentinel-5P Programmes entwickeln möchten. Die Einstiegshürde bei diesen Daten ist jedoch aus zwei Gründen sehr hoch: erstens, benötigen die wissenschaftlichen Daten eine ausführliche Auseinandersetzung mit der Dokumentation durch die ESA um sie korrekt verarbeiten zu können und zweitens, ist eine aufwendige Infrastruktur notwendig um sie dynamisch darzustellen. Es fehlt also eine technische Lösung, die diese Daten auf eine an die in unserer Zielgruppe vorherrschenden Technologien angepasste Art und Weise ausliefert. Unser Projekt schafft diese Lösung und ermöglicht es unserer Zielgruppe schneller und ohne tiefgehendes Fachwissen zu arbeiten. Zusätzlich hoffen wir auf einen Multiplikatoreffekt, indem die Daten einer breiteren Gruppe als bisher zugänglich werden. In diesem Sinne fällt unser Projekt klar unter den Public Interest Tech Begriff, da es eine Lücke in der öffentlichen, digitalen Infrastruktur füllt.

Ausführliche Darstellung der Ergebnisse

Am Ende unserer Förderung steht ein prototypischer Dienst, der ein Datenprodukt (Kohlenmonoxid) der Sentinel-5P Mission in einem einfach in andere Dienste integrierbarem Format (JSON) ausliefert: <https://api.emissions-api.org/ui/>. In diesem Sinne konnten auch alle Meilensteine erfolgreich abgeschlossen werden. Eine Einschränkung von den ursprünglichen Zielen ist lediglich, dass wir uns auf ein Datenprodukt der Sentinel-5P Mission beschränken, statt schon während des Förderzeitraumes alle Datenprodukte anzubieten. Dieser Umstand soll allerdings zeitnah behoben werden (siehe Punkt Weiterentwicklungen). In den folgenden Absätzen werden wir die Ergebnisse nach Themengebieten geordnet darstellen.

Technische Entwicklung

Am Anfang der Projektlaufzeit haben wir einen minimalen Prototypen entwickelt, der bereits alle relevanten technischen Eigenschaften des späteren Dienstes abgebildet hat (vertikales Prototyping). Dieser Prototyp wurde dann als Ausgangsbasis genommen um im weiteren Projektverlauf die verschiedenen technischen Aspekte im Detail zu testen und weiterentwickeln zu können. Als erstes stand dabei das Beschaffen der Daten von der ESA im Mittelpunkt. Dazu musste ein Programm geschaffen werden, das automatisiert die Daten bei der ESA herunterladen kann. Dazu wurde das Pythonpaket `sentinel5dl` entwickelt, das auch als eigenständige Software benutzt werden kann. Es wurde über die Plattform PyPI zur einfachen Installation veröffentlicht. Diese Software findet bereits auch außerhalb unseres Projektes Anwendung durch Dritte.

Der nächste Fokus war das Design der für den Nutzer relevanten API-Endpunkte. Dabei hat uns das begleitende Coaching durch die Open Knowledge Foundation geholfen unsere Nutzer besser zu verstehen. Der dritte Schritt war die Entwicklung eines Paketes zur Datenvorverarbeitung und -filterung (`s5a`), das auch über die Plattform PyPI der Öffentlichkeit zur Verfügung gestellt wurde.

Der größte Teil der technischen Entwicklung wurde durch das Testen und die Optimierung der Datenhaltung in Anspruch genommen. Die großen Datenmengen die durch das Zwischenspeichern und Konvertieren des binären Datenformats der Originaldaten, in ein besser abzufragendes Datenbankformat entstehen, waren wie erwartet eine große Herausforderung. Bei der Bearbeitung dieser Problemstellung kam uns der transparente Umgang mit Wissen in der Open-Source Gemeinschaft zu Gute. Dadurch war es uns möglich, mit Datenbankentwicklern der von uns genutzten Softwareprodukte Kontakt aufzunehmen und mit ihnen über unsere technischen Fragen zu diskutieren. Das hat uns geholfen einige wichtige Optimierungen an den Schnittstellen zu unserer Datenbanklösung vorzunehmen. Letztenendes mussten wir uns dennoch entscheiden, dass die Gesamtdatenmenge aller ESA-Datenprodukte, für uns nicht mit realistisch verfügbaren Hardwarekapazitäten handhabbar ist. Daher haben wir uns schlussendlich für die Projektlaufzeit auf ein einzelnes Datenprodukt (Kohlenmonoxid) konzentriert (siehe auch Ausführungen weiter unten).

Der letzte Punkt der technischen Entwicklung betraf das Aufsetzen des Dienstes auf den Servern. Wir haben dazu zwei automatisierte Set-Ups entwickelt, eines für Rechenzentren und eines für den Amazon Web Service. Als Host konnten wir sowohl Amazon als auch zwei deutsche Universitäten gewinnen. Dadurch konnten wir die Verfügbarkeit unseres Dienstes eine Laufzeit von mindestens einem Jahr nach Förderende garantieren.

Öffentlichkeitsarbeit

Da ein Open-Source Projekt nur von einer lebendigen Gemeinschaft getragen werden kann, haben wir während des gesamten Förderzeitraums Wert auf eine aktive Außendarstellung gelegt. Gleich zu Beginn des Projektes haben wir daher eine Homepage und einen Twitter-Account angelegt, um mögliche Nutzer für unser Projekt zu interessieren. Im Oktober haben wir erfolgreich die virtuelle Veranstaltung Hacktoberfest auf GitHub genutzt, um durch gezielte Veröffentlichung unserer technischen Fragestellungen externe Entwickler zur Mitarbeit zu animieren. Im November haben wir das Projekt in dem Video-Podcast FLOSS Weekly einem breiten internationalen Publikum vorgestellt. Im Dezember folgte dann ein Lightning Talk auf dem 36C3 in Leipzig und dann im Februar ein Lightning Talk auf der FOSDEM in Brüssel.

Neben diesen Events, haben wir uns bemüht die Hürden für Externe die an unserem Projekt teilnehmen oder die Ergebnisse verwerten möchten, so niedrig wie möglich zu halten. Daher haben wir neben der technischen Entwicklungsarbeit einen Schwerpunkt unserer Arbeit auf eine gute Dokumentation mit Contribution Guidelines und auf das Einhalten von Open-Source Best Practices gelegt. Bei der Verbesserung unserer Außendarstellung war uns auch das durch die Open Knowledge Foundation zur Verfügung gestellte Coaching eine Hilfe.

Data Literacy

Ein weiteres Ergebnis des Projektes, ist die ausführliche Begleitung des Nutzers durch Beispiele und Texte die die Data Literacy im Umgang mit den Sentinel-5P Daten erhöhen sollen. Wir haben im Projektverlauf selber viel über die Daten gelernt. Dieses Wissen stellen wir über Blogartikel und begleitende Texte auf unserer Homepage zur Verfügung, um die Nutzer über die Vorteile aber auch die Limitierungen von satellitengestützten Emissionsmessungen aufzuklären. Wir möchten dadurch unsere Nutzer befähigen selber gute Anwendungsfälle für die Daten zu erkennen.

Abschließend möchten wir bemerken, dass es uns ohne die Hilfestellung der Open Knowledge Foundation schwer gefallen wäre in sechs Monaten den gleichen Impact mit unserem Open-Source Projekt zu erreichen. Die Betreuung, Hilfestellung, Unterstützung in der Öffentlichkeitsarbeit und die Vernetzung mit anderen Public Interest Tech Projekten hat uns motiviert, uns eine äußere Struktur gegeben und uns kreativ bereichert.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Die Zielgruppe und der Nutzen der sich für sie aus unserem Projekt ergibt, besteht wie oben erwähnt in dem vereinfachten Zugriffs auf satellitengestützte Emissionsdaten. Neben neu entwickelten Infrastruktur- und Softwarelösungen, gehören zu dieser Vereinfachung auch die Dokumentation und die Begleittexte zum Umgang mit den Daten. Dadurch dass unsere Ergebnisse alle frei lizenziert sind, sind der Weiterverwendung durch Dritte kaum Grenzen gesetzt. Wir erhoffen uns dadurch einen Multiplikatoreffekt der andere Entwickler befähigt, so wie wir auch bei unseren Entwicklungen von der Vorarbeit anderer Entwickler profitiert haben. Für uns persönlich hat dieses Projekt die Möglichkeit bedeutet, den kompletten Prozess, mit allen Aspekten der Organisation und Durchführung eines Open-Source Projekts, einmal konzentriert zu durchlaufen. Das und die intensive, bewusste Auseinandersetzung mit der Schnittstelle von Technik und Gesellschaft, hat unsere Fähigkeiten und unser Verantwortungsbewusstsein geschärft, um in der Zukunft noch aktiver unsere digitale Gesellschaft mitgestalten zu können.

Für die Zukunft der emissions API planen wir eine Reduzierung der Rohdaten auf unserer Serverseite, so dass alle Datenprodukte der Sentinel-5P Mission berücksichtigt werden können. Details haben wir in einem Text auf unserer Homepage veröffentlicht (<https://emissions-api.org/roadmap.html>).

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Wie bereits erwähnt, haben wir im Laufe des Projekts beschließen müssen nicht alle Datenprodukte als Rohdaten anzubieten. Sobald wir im Projektverlauf absehen konnten, dass dieses Problem auftreten kann, haben wir einen zweigleisigen Ansatz verfolgt. Wir haben zum Einen versucht mit externen Experten auszutesten, wo wirklich unsere technischen Grenzen liegen. Zum Anderen haben wir die Idee erkundet Daten zu reduzieren. Dieses Vorgehen hat es uns zum Ende des Projekts ermöglicht unsere ursprüngliche Idee als Prototyp zu präsentieren, gleichzeitig aber schon einen konkreten Plan mit technischen Lösungsansätzen für die Zukunft zu haben.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Die erste Anlaufstelle für mögliche Nutzer ist unsere Homepage <https://emissions-api.org/>. Dort finden sich neben Links zu allen weiterführenden Ressourcen um das Projekt auch erklärende Texte zu den Datensätzen, Anwendungsbeispiele und Formulierungen von Motivation und Zukunftszielen. Ebenso ist dort der Link zu der technischen Dokumentation der API zu finden (<https://api.emissions-api.org/ui/>).

Unsere GitHub-Präsenz (<https://github.com/emissions-api>) stellt den Quellcode für alle Programmteile öffentlich unter der MIT-Lizenz zur Verfügung. Außerdem finden sich hier die technische Dokumentation (<https://docs.emissions-api.org/> und <https://sentinel5dl.emissions-api.org/>), öffentlich dokumentierte Diskussionen zur Entscheidungsfindung im Projektverlauf und Hinweise wie man selber in unserem Projekt aktiv werden kann. Nutzer die an Neuigkeiten zu unserem Projekt interessiert sind, können unserem Twitter-Account @emissions_api folgen und Aufzeichnungen von öffentlichen Vorträgen über unser Projekt sind unter folgenden Links zu finden:

- https://media.ccc.de/v/36c3-10525-lightning_talks_day_3#t=1146
- https://fosdem.org/2020/schedule/event/emissions_api/
- <https://twit.tv/shows/floss-weekly/episodes/555>

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Zum Ende des Projektes standen wir vor der Frage ob wir mehr Zeit in technische Perfektionierung investieren oder in breitere Dokumentation und Begleitung der Nutzer. Wir haben uns letztendlich dazu entschieden einen stärkeren Fokus auf die Dokumentation und die Begleittexte zu legen, da wir darin die nachhaltigere Strategie für das Fortbestehen eines Open-Source Projektes sehen.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Da die momentanen satellitengestützten Emissionsdaten sich gut eignen um große räumliche und zeitliche Muster abzubilden, verweisen wir gerne auf die Projekte OpenSenseMap und lufdaten.info, die durch Bodensensoren erfasste Emissionsdaten lokal und zeitlich fein aufgelöst ausliefern können (dafür mit einer geringeren Abdeckung als Satellitendaten). Diese Projekte sind erfreulich komplementär zu unserem und wir stehen in Austausch über zukünftige Entwicklungen.

StayAlive Insektencounter

Schlussbericht

Zuwendungsempfänger: Nicola Wettmarshausen

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S08 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?

2018 habe ich die Initiative Ecocurious gegründet – eine Citizen Science Community, die sich mit Projekten an der Schnittstelle Natur, Umwelt und Technik beschäftigt. In diesem Teil-Projekt möchten wir etwas gegen den Biodiversitätsverlust vor der eigenen Haustür, also konkret gegen das Insektensterben tun, denn dieses Thema wird aktuell immer brisanter.

Während meiner Recherchen lernte ich, dass das Monitoring von Insekten in der Praxis sehr analog betrieben wird. Fliegende Insekten werden in zeltartigen Malaise-Fallen gefangen und in Alkohol konserviert, um anschließend ihre Biomasse zu bestimmen. Einzelne Familien (Schwebfliegen, Wildbienen) werden dabei manuell aussortiert und ihre Art bestimmt. So arbeiten beispielsweise die Entomologen im Naturkundemuseum Stuttgart im Auftrag der Landesanstalt für Umwelt BW.

Das brachte uns auf die Idee, die Insekten digital zu zählen und zu klassifizieren, sie dabei aber nicht zu töten. Im Juni 2019 überlegten wir gemeinsam, wie wir eine digitale Echtzeitmessung ermöglichen und dadurch zeigen können, welche Klassen von Insekten (Fliegen, Hummeln, Falter usw.) es noch gibt, wie sich der Bestand über die Zeit verändert und welche Schutzmaßnahmen einen direkten Einfluss auf die Populationszahlen haben können. Wir formulierten das Ziel, ein Citizen-Science-Monitoringtool für Umweltschützer, NGOs, Landwirte, Hobbygärtner, Wissenschaftler und weitere Zielgruppen zu entwickeln – ein kostengünstiges Do-it-yourself (DIY)-Gerät auf Raspberry-Pi-Basis, das sich jeder in gemeinsamen Workshops zusammenbauen können soll.

Die Vision: Das Device soll Insekten unterscheiden und lernt dank Künstlicher Intelligenz (KI) ständig hinzu. Die Daten werden später über eine Map im Browser visualisiert. Standorte können verglichen und Tag- und Nachaktivitäten beobachtet werden. So kann jeder selbst zum Bürger-Forscher werden und Nachforschungen anstellen, gezielte Maßnahmen ergreifen und direkt nachvollziehen, ob seine Maßnahmen wirken. Etwa, wenn er bestimmte nektar- und pollenhaltige Pflanzen in seinem Garten säht statt purem englischen Rasen.

Ausführliche Darstellung der Ergebnisse

Wie war die geplante Vorgehensweise zur Problemlösung (auch Angabe der wichtigsten Meilensteine)? Welche konkreten Ergebnisse hast Du erzielt? Konnten alle Meilensteine erreicht werden?

Im Rahmen des Software Sprints konnten wir einen ersten funktionsfähigen Prototyp (Hardware und Software) für beide Messmethoden (optoakustisch und akustisch) entwickeln, einschließlich dem dazugehörigen Datenhandling für beide Messmethoden (*Insects Monitoring Lab* als Prototyp für die Bilderkennung und eine prototypische Plattform für die Wingbeats-Daten).

Da die konkrete Umsetzung der Idee eines Insektencounters eine parallele, teilweise voneinander abhängige Hardware- und Softwareentwicklung erfordert und deshalb nicht innerhalb eines halben Jahres komplett zu realisieren ist, mussten wir unseren experimentellen Ansatz immer wieder anpassen und iterativ umsetzen. Unser Ziel war es, zwei Messmethoden auszuprobieren und einen Proof-of-Concept zu entwickeln. Wir starteten mit diesen zwei Ansätzen parallel in zwei Teams.

Team Eins in Stuttgart/Tübingen (Jürgen, Matthias, Nicola) widmete sich der Anwendung der *Wingbeats*-Messmethode, zu der es bereits Forschung im Bereich Schädlingsbekämpfung gibt. Wir wollten diese Methode für unsere Fragestellung anpassen: Schaffen wir es, möglichst viele fliegende Insekten (und nicht nur „Schädlinge“ wie Moskitos oder Olivenfliegen) damit zu detektieren? Und welche Hardware brauchen wir dazu? Da das Device später von Usern aller Altersstufen benutzt werden soll, testeten wir keine Laser, sondern Infrarot-LEDs. Diese LEDs wurden mit Photodioden auf der gegenüberliegenden Seite eines 3D-gedruckten Rahmens kombiniert (siehe Skizze). Fliegt ein Insekt durch den Rahmen, schattet sein Flügelschlag das IR-Licht im Frequenzbereich von 15-1000 Hz ab. Diese Abschattung kann spezifisch für eine bestimmte Art oder zumindest Familie von Insekten sein. Das Bestechende an dieser Messmethode ist, dass man später ein Audiosignal verarbeitet (Wave-Format), das man zuerst optisch misst und anschließend wandelt. Auf diese Weise stören zum Einen akustische Ereignisse wie Hundegebell, Autos, Flugzeuge etc. nicht und zum Anderen können optische Störungen leichter herausgefiltert werden.

Wir arbeiteten hier mit der Software GNU Radio (<https://www.gnuradio.org/about/>) und übertrugen die Filtereinstellungen später in Python (siehe Github). Mit GNU Radio entstörten Matthias und Jürgen das ankommende Licht-Signal, modulierten und demodulierten es und testeten verschiedene Filtermöglichkeiten. Als wir dann im Spätherbst so weit waren, Wespen von Stubenfliegen unterscheiden zu können, begann allerdings die Winterperiode. Gekaufte Insekten waren nicht in dem Umfang vorhanden, wie wir es uns erhofft hatten. Hinzu kam, dass Insekten wie Florfliegen, Wachsmotten, Heimchen oder Wüstenheuschrecken partout nicht fliegen wollten und dadurch mit dieser Methode nicht messbar waren. Dadurch reichte die Datenmenge nicht aus, um ein KI-Modell trainieren zu können. Wir fingen dennoch an, ein vortrainiertes Modell mit Trainingsdaten von 6 Mückenarten (Daten von iNaturalist aus <https://www.kaggle.com>), bestehend aus einem 1D-CNN (eindimensionales Neuronales Netz) mit Daten von bewegungsfreudigeren Fleischfliegen (Sarcophagidae) mithilfe von TensorFlow (TF) weiter zu trainieren. Dabei nutzten wir TF, um später mit der abgespeckten Version TF Lite (8 Bit) auf dem Raspberry weiterarbeiten zu können.

Auf der Scicar-Konferenz 2019 (<https://www.scicar.de/>) knüpfte ich Kontakte zur Uni Saarbrücken und konnte ein studentisches Team (fünf Studierende) für unser Projekt gewinnen. Matthias entwickelte mit mir gemeinsam drei Arbeitspakete für den Zeitraum November bis Ende Januar, die die Studierenden im Rahmen ihres Software-Engineering-Projekts (<https://cms.sic.saarland/se/>) für Ecocurious umsetzen

konnten. Er plante das Backend, definierte die Schnittstellen, wählte die Tools und Bibliotheken aus und kontrollierte den Arbeitsprozess des Saarbrücker Teams. Wir definierten zahlreiche Software-Anforderungen für die drei Tasks Sensor/Raspberry Pi, Server und Webseite, die vom Team größtenteils auch so umgesetzt wurden, beispielsweise:

- **Trigger:** Im Sensor haben wir eine akustische Detektionsschwelle eingerichtet, die definieren soll, ob ein Signal als Flügelschlag wahrgenommen und daraufhin aufgezeichnet wird. Das Gerät soll nur dann aufzeichnen, wenn ein Insekt durch den Sensor fliegt, um möglichst wenig Datenmüll zu generieren, muss dann aber sofort reagieren. Da das Signal sehr kurzlebig ist (200-800 ms), reichen kleine Verspätungen aus, um ein Insekt nicht zu detektieren.
- Der Sensor soll Daten über LoraWAN übertragen können (nur Zahlen, keine Bilder) und unterstützt MQTT optional zu HTTP.
- **Lightweight:** Da die Audioaufnahme, Audioverarbeitung als auch die Klassifizierung der Insekten auf einem Gerät unter limitierten Hardware-Ressourcen eines Einplatinencomputers laufen muss, müssen ressourcenintensive Prozesse wie Datenspeicherung, Modellgenerierung und Weiterverarbeitung ausgelagert werden. Die WAV-Dateien werden deshalb auf eine SD-Karte geschrieben, die regelmäßig ausgetauscht wird. Steht eine Internetverbindung zur Verfügung, werden die Dateien auf einen Server geschrieben und dauerhaft archiviert. Dies haben wir so umgesetzt.
- **Schreibzyklen:** Da der Speicher des Raspberry Pi's über eine limitierte Anzahl an Schreibzyklen verfügt, speichern wir temporäre Daten im Normalbetrieb in einer RAM-Disk.
- Das Backend besteht aus mehreren Komponenten (Datenbankserver, SensorThingsAPI, Userschnittstelle zur Verwaltung, Schnittstellen für die Visualisierung). Diese Komponenten liegen in Docker-Containern vor und können per Docker-Compose gestartet werden.
- **Webseite:** Über den noch einzurichtenden öffentlichen Server soll jeder User seinen Sensor anmelden können. Bisher ist dies nur dem Entwicklerteam zugänglich. Die Userverwaltung soll einfach zu bedienen sein und Datenschutz garantieren (<https://insectcounter.zapto.org/>). Die browserbasierte Map kann mehrere Sensoren auf einmal visualisieren und komplexere Visualisierungen anbieten (über eigenen JavaScript-Code). Sie hat per SensorThings-API Zugriff auf die Daten.
- **TensorFlow (TF):** Wir haben das Modelltraining mit TF und die Konvertierung in TF Lite für den Raspberry begonnen.
- **Erweiterbarkeit:** Da Machine-Learning-Modelle nur so gut sind wie ihre Trainingsdaten, wollen wir das bestehende Modell mit neuen Daten weiterentwickeln können, ohne dass dafür eine Softwareanpassung nötig ist. Im Idealfall sollte das alleinige Hinzufügen der Daten in einen Ordner genügen, um das Modell zu erweitern. Dies wurde implementiert.
- Für die wissenschaftliche Auswertung wurde auf dem Sensor auch die Möglichkeit implementiert, Umgebungsdaten wie Temperatur, Luftfeuchte etc. abzufragen und mit den Ergebnissen der Klassifizierung zu übertragen.
- **Sprache:** Um die Erweiterbarkeit des Projektes zu vereinfachen, soll die Anzahl der verwendeten Programmiersprachen möglichst gering gehalten werden. Daher ist ein Großteil in Python programmiert worden, auch um Hardwarekompatibilität zu garantieren.

Wichtig war es uns, möglichst viele standardisierte Komponenten zu verwenden. Im Sensor etwa wurde der Fokus auf die Verwendung von bekannten Bibliotheken zur Audibearbeitung und Klassifizierung gesetzt. Auch im Backend wurde Wert auf Standards gesetzt. Daher fiel die Wahl hier auf GOST, eine Referenzimplementierung der OGC SensorThingsAPI. Diese Schnittstelle erfreut sich wachsender

Verbreitung im Open Data Umfeld. Sie erlaubt es, Daten sowohl zeitlich als auch räumlich abzufragen, beispielsweise, um die Vorkommen einer Insektenart A im Naturschutzgebiets XY mit der Umrandung Z (GeoJSON Objekt) anzuzeigen. Durch die Verwendung von Docker-Containern ist der Nachbau auch für Laien auf einfache Weise möglich.

Die Zusammenarbeit mit den Studierenden verlief rein virtuell über Slack, Mail, Videokonferenz und Telefon. Sie war aufgrund der Komplexität des Projekts, der unterschiedlichen Backgrounds der Studierenden und den Vorgaben der Hochschule sehr zeitintensiv in der Kommunikation.

Derweil arbeitete **Team Zwei in Berlin** (Stefan, Levin, Oskar und Roman für die Hardware) mit der rein optischen Messvariante (fotografisch/Computer Vision). Nachdem Roman ein Terrarium gebaut hatte, zogen dort Heuschrecken und Heimchen ein. Weil es das Ziel des zu entwickelten DIY-Devices ist, mit einer kostengünstigen Technik zu arbeiten, nutzten wir eine PiCam zum Fotografieren. Dabei wollten wir sicherstellen, dass die sich bewegenden Insekten in einer Bildfolge nur einmal gezählt werden. Darum verwendeten Stefan und Levin die Open-Source-Überwachungssoftware *Motion* (><https://motion-project.github.io/>). Auch hier bauten sie einen Trigger ein: Aufgezeichnet wird nur, wenn ein Insekt sich bewegt (Datensparsamkeit). *Motion* kann auch mit hochwertigeren Kameras arbeiten und ist daher für die spätere Erweiterung des Setups optimal.

Die mit dem Raspberry Pi generierten Bilddaten werden anschließend in einer zentralen Datenbank hochgeladen, die zunächst auf unseren Servern laufen soll (noch gibt es keinen Projektserver). Diese Datenbank muss später an die Struktur der Europäischen Datenbank Fauna Europaea (<https://fauna-eu.org/>) angepasst werden, um wissenschaftlich nutzbar zu sein.

Da es Winter wurde, mangelte es auch den Berlinern an einheimischem Artenreichtum für ein sinnvolles Modelltraining der KI. Darum haben wir uns entschieden, zuerst mit einem anderen Task zu beginnen: Einer Webplattform fürs gemeinsame Datenmanagement (Arbeitstitel: *Insects Monitoring Lab*, <http://195.201.97.57/>, <https://github.com/ecocurious/insect-webtool>). Die Grundidee ist dabei, die Insekten-Bestandsdaten in einer zentralen Datenbank mit der Öffentlichkeit zu teilen. Die an verschiedenen Orten generierten Daten sollen dabei nicht nur von Citizen Scientists, sondern auch von Wissenschaftlern und Naturschützern genutzt werden. Die breite Zielgruppe von technisch versierten bis nicht-versierten Usern ist dabei maßgeblich für die Usability der Plattform.

Für das Team Berlin (das ja eigentlich die KI bearbeiten wollte) bedeutete dies, sich in neue Themenfelder einzuarbeiten. Die Backendentwicklung der Plattform war dabei unproblematisch, da Stefan und Levin ihre Erfahrungen in der Strukturierung von Webshops einbringen konnten. Allerdings mussten sie sich in die Frontendentwicklung erst neu einarbeiten, was die Umsetzung der einzelnen Aufgaben verzögerte. Bis zum *Demo Day* bauten sie eine Plattform mit diesen Funktionalitäten:

- Übertragung von dezentral gesammelten Bildern (Raspberry Pi) in eine zentrale Datenbank
- Suchfunktion für diese Bilder entwickeln
- Erstellung von Datensätzen zum Trainieren von ML-Algorithmen
- Annotation der Bilder (Markierung und Kategorisierung von Insekten)
- Python Client, der den Download von Datensätzen und den Upload von annotierten Daten vereinfacht
- Visuelle Evaluation von maschinell (KI) klassifizierten Bildern
- Darstellung von Zeitreihen (Entwicklung der Insektenpopulation über die Zeit)

Ins Frontend baute das Team Berlin deshalb folgende Funktions-Elemente ein:

Live-View: Um später jeden autonom im Freiland messenden Sensor überwachen zu können, wurde eine Live-View-Ansicht eingebaut (zunächst für nur einen Sensor).

Browser: Der User kann durch die Fotos browsen, sich einen Zeitraum auswählen und die Bilder zur Offline-Bearbeitung als Datenpaket auf seinen Rechner herunterladen. Er kann sie auch vom Raspberry Pi direkt hochladen (Transfer in beide Richtungen). Er kann sich außerdem in einer Visualisierung anzeigen lassen, wieviel Insekten einer Art in den ausgewählten Datenmengen vorkommen.

Annotation: In dieser Ansicht können die Insekten-Arten in den einzelnen Bildern annotiert werden. Die Annotationen sind farblich codiert und damit einem User zugeordnet. Das hat den Vorteil, unseriöse Tagger herausfiltern zu können und so die Datenqualität zu verbessern.

Die Annotationen trainieren das KI-Modell YOLO (You Only Look Once/ <https://pjreddie.com/darknet/yolo/>), das derzeit auf dem Desktop läuft. Das Team verwendete YOLO, weil hier zwei Aufgaben zugleich erledigt werden mussten: Einmal die Frage, wieviel Insekten auf dem Bild zu sehen sind (=Objekterkennung), und gleichzeitig die Frage, welche Insekten es sind (=Klassifizierung). YOLO ist sehr gut geeignet, diese Fragen gleichermaßen beantworten zu können.

Mit den Heuschreckenarten aus dem Terrarium labelte Levin 400 Bilder und erstellte so erste Trainingsdaten für das KI-Modell. Er stellte fest, dass die Heuschrecken mithilfe von YOLO zuverlässig erkannt und die vom Team Berlin entwickelte Plattform *Insects Monitoring Lab* grundsätzlich funktioniert. Allerdings sind viele Funktionen der Plattform noch nicht voll ausentwickelt. Eingebaut werden soll noch die Unterstützung von mehrschrittigen Annotationsprozessen, die Strukturierung der Label-Hierarchie in der Datenbank wie bei Fauna Europaea, die Nutzbarkeit mehrerer Raspberry PI-Geräte, Optimierung des User Managements und das Management von KI-basierten und gleichzeitig User-basierten Annotationen. Um das *Insects Monitoring Lab* voll funktionsfähig und stabil zu machen, wären weitere Arbeitsschritte im Bereich Frontend und Backend im Umfang von mindestens 300 Arbeitsstunden nötig.

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?

Die gesamte Idee, Insekten digital zu zählen und zu klassifizieren ist ein Bottom-up-Citizen-Science Projekt (bürgergetrieben und nicht durch eine Hochschule gestartet). Über die noch zu gestaltende Homepage zum Projekt und insbesondere über die *Insects Monitoring Lab* Plattform wollen wir verschiedene Interessengruppen vernetzen und ihre Zusammenarbeit unterstützen. Dabei handelt es sich um

- Maker, also Entwickler von Hardware-, Software- und IoT-Lösungen. Sie profitieren von unseren entwickelten Tools für den Raspberry PI.
- Data Scientists, die KI zur Klassifikation entwickeln. Sie profitieren von unseren dokumentierten Forschungsergebnissen zum Machine Learning Framework YOLO auf Insekten-Daten

- Biologen können von einem integrierten Daten-Annotations-Tool profitieren, um ihr Wissen mit Kolleg/innen und Bürgern teilen zu können. Sie können außerdem die analysierten Daten zum Insekten-Bestand für ihre Forschung nutzen.
- Bürger können den Insektenbestand auf ihren ausgewählten Flächen messen, sich darüber austauschen und dabei einiges über Forschung, eine umweltbezogene Anwendung der KI und das Insektenverhalten von beobachteten Arten lernen. Sie können außerdem lernen, was passiert, wenn man Insekten im Garten das richtige Futter anbietet: Nektar- und pollenreiche einheimische Blüten beispielsweise.

Welche Bezüge gibt es zu den Themenfeldern und Zielen des Software Sprints?

Unter dem Themenfeld *Civic Tech* ermöglichen wir interessierten Bürgern, ein kostengünstiges technisches Gerät auf Raspberry Pi-Basis aus Bauteilen in Workshops selbst zusammenzubauen, eigene Daten zu Insekten-Beständen aufzuzeichnen und diese mit einer Community zu teilen. Alle Roh-, und aufbereiteten Daten und Analysen stehen der Öffentlichkeit auf der Plattform zum Anschauen, Auswerten und zum Download zur Verfügung.

Welche zusätzlichen Erkenntnisse hast Du aus der Projektarbeit gewonnen, auch im Hinblick auf die Begleitung durch die Open Knowledge Foundation?

Als Projektleiterin habe ich sehr vom angebotenen Coaching profitiert, da es nicht immer einfach ist, mehrere Arbeitsgruppen aus der Ferne zu steuern, ohne bei Problemen mal eben ein physisches Treffen einberufen zu können. Darum lief die Zusammenarbeit im Team Stuttgart/Tübingen am besten, da wir uns hier auch kurzfristig treffen konnten. Es war für mich außerdem sehr wichtig, sich mit den anderen teilnehmenden Projektgruppen im Rocket Chat und telefonisch auszutauschen und Tipps im Hinblick auf Projektsteuerungstools, Konferenzteilnahmen und best-practice zu bekommen.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Welcher Nutzen ergibt sich für die Zielgruppe aus den Ergebnissen Deines Projekts?

Siehe oben

Welche weitergehenden Effekte ergeben sich aus der Open-Source-Stellung der Ergebnisse?

Maker und Entwicklerinnen werden aufmerksam und machen mit. Sie sehen, was fehlt und können ihre Ideen einbringen. Das Projekt ist ein Community-Projekt. Wir brauchen weitere Mitmacher, denn das Projekt soll weiter wachsen.

Gibt es Ideen für die Weiterentwicklung Deiner Lösung und Pläne zu deren Umsetzung?

Wir möchten in Zukunft mit Hochschulen im Bereich IT, KI, IoT und Sensorik zusammenarbeiten, ebenso mit Biologen/Entomologen, Vereinen und mit lokalen Umweltgruppen wie NABU, BUND etc.

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

Wir alle haben gelernt, wie wir konkrete technische Herausforderungen lösen können. Ein ganz wichtiger Lerneffekt war, die Perspektiven von verschiedenen Nutzern, insbesondere Biologen, zu hören und einzubinden. Projektleitung: ich habe gelernt, dass die Tools im Projektmanagement immer auf die Bedürfnisse aller abgestimmt werden müssen, sonst funktioniert es nicht. Das ist teamspezifisch, oft sogar personenspezifisch und muss angepasst werden. Eine generelle Lösung (jeder muss Tool XY als Haupt-Kommunikationskanal benutzen) gibt es leider nicht.

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 den Wintermonaten gab es kaum einheimische Insekten im Freiland und deshalb fehlten uns KI-Trainingsdaten, um anschließend YOLO zu trainieren. Team Berlin hat daher Fokus auf die Plattform *Insects Monitoring Lab* gelegt, die dann in der kommenden Insektensaison die Entwicklung und Skalierung des Insekten-Monitorings unterstützen kann. Bei der Entwicklung des *Insects Monitoring Lab* gab es für das Berliner Team ebenfalls Sackgassen, beispielsweise die Verwendung von GraphQL als Backend API. Dabei hat sich herausgestellt, dass sie die API nur schwer ins Frontend integrieren können. Daher haben sie auf eine Websocket-basierte Frontend-Backend-Kommunikation umgestellt, und dies klappte gut.

Als Sackgasse entpuppte sich der Versuch, existierende Datensätze von Insekten (iNaturalist-Foto-Daten auf der ML-Competition-Plattform Kaggle (<https://www.kaggle.com/>) zu verwenden und damit ein KI-Modell zu trainieren. Die Fotos sind perspektivisch zu nah aufgenommen (teilweise Makroaufnahmen) und funktionieren deshalb in unserem Setting (totalere Bildperspektive, andere Bildqualität) nicht. Darum ist der Ansatz, komplett mit eigenen Bildern zu arbeiten, hier der sinnvollere.

Im Team Stuttgart/Tübingen haben wir an der Wingbeats-Signalverarbeitung gearbeitet und mit Unterstützung der Saarbrücker Studierenden den kompletten Daten-Workflow für Wingbeats prototypisch abbilden können. Bereits vorhandene Wingsbeats-Aufnahmen (ebenfalls auf Kaggle) bezogen sich auf Tierarten, die bei uns nicht heimisch sind. Teilweise mussten die Audioformate angepasst (alle auf eine Länge gekürzt etc) werden.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

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

Momentan bei Github und auf der Ecocurious-Homepage (eine neue Seite bzw. Unterseite ist geplant)

<https://ecocurious.de/projekte/insect-counts/>

<https://github.com/ecocurious/insect-webtool>

<https://github.com/ecocurious/insects-count>

<https://github.com/ecocurious/insectcounter>

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 Gesamtidee ist so umfangreich, dass wir diese Prototyp-Phase mit einem definierten und vorzeigbaren Zwischenstand abgeschlossen haben.

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 gibt es mehrere akademische Studien, die die Wingbeats-Messmethode nutzen und auf denen unser Vorgehen teilweise basiert (Quellen siehe unten). Diese Paper ermutigten uns, mit dieser Messmethode weiterzuarbeiten, da sie einerseits störungsarm und freilandtauglich, andererseits sehr kostengünstig ist, was die Hardware-Komponenten angeht. Die Messmethodik ist in außerdem technischen Laien einfach zu vermitteln und für Citizen-Science-Anwendungen daher sehr geeignet.

Quellen sind u.a.:

<https://www.kaggle.com/potamitis/wingbeats/kernels>

<https://ieeexplore.ieee.org/document/7091854>

<https://ieeexplore.ieee.org/document/8553542>

<https://www.hindawi.com/journals/js/2018/3949415/>

http://www.cs.ucr.edu/~eamonn/CE/Bayesian%20Classification%20withInsect_examples.pdf

<https://academic.oup.com/sysbio/article/68/6/876/5368535>

Richtlinie zum „Software-Sprint“

FutureCityProjects

Schlussbericht

Zuwendungsempfänger:

FutureCityProjects – Norbert Rost & Jakob Schumann GbR

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S09 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Städte sind Ressourcenfresser. Stadtbewohner*innen werden bislang nur *unzureichend in die nachhaltige Stadtentwicklung eingebunden*. Sie würden Stadt gern mitgestalten, wissen aber oft nicht wie. Für Kommunen stellen diese Menschen ein *enormes Potenzial* als Wissensträger und Stadtgestalter dar, welches bislang aber nur unzureichend genutzt wird.

Bürger*innen können eingebunden werden, indem sie Projekte ersinnen und umsetzen, die die Städte nachhaltiger machen. Solcherart bürgerschaftliche Projektentwicklung mit Nachhaltigkeits-Fokus wurde im Rahmen des Zukunftsstadt-Städtewettbewerb von 2015-2018 in Dresden erprobt.¹ Dresden gehört zu den Finalisten des Wettbewerbs und setzt 8 bürgergetragene Projekten seit 2019 in der dritten Zukunftsstadt-Phase wissenschaftlich begleitet um.

Eine zentrale Erkenntnis aus dem zurückliegenden Zukunftsstadt-Prozess war, dass eine größere Skalierung der Bürger*innen-Teilnahme nur möglich ist, wenn es eine digitale Plattform gibt, über die Bürger*innen Projekte entwickeln können. Erste, eher unzusammenhängende Bausteine dafür wurden im Zukunftsstadt-Prozess erschaffen, sie waren in ihrer ad-hoc-gestalteten Form aber nur begrenzt produktiv einsetzbar.

Dem Projekt FutureCityProjects liegt die Motivation zugrunde, die Erkenntnisse aus dem Zukunftsstadt-Prozess weiterzunutzen und für andere Kommunen und Bürger*innen anderer Städte nutzbar zu machen. Damit soll ein Beitrag geleistet werden zu mehr Partizipation in Zeiten einer Demokratiekrise und zu nachhaltiger urbaner Entwicklung entsprechend der SDGs².

1 www.zukunftsstadt-dresden.de

2 SDG: sustainable development goals, Ziele für nachhaltige Entwicklung

Ziel des Vorhabens „FutureCityProjects“ war und ist, eine Projekt-Entwicklungs-Plattform zur Verfügung zu stellen, die Bürger*innen befähigt, **Projekte co-produktiv schrittweise** bis zur Umsetzungsreife zu entwickeln und antrags- oder umsetzungstauglich zu machen: inklusive Maßnahmen-, Zeit- und Finanzplan. Die Online-Plattform hilft dabei nicht nur, alle relevanten Informationen für ein Projekt zu strukturieren, sie ist auch eine **Lern-Plattform für unerfahrene Projekt-Macher*innen**, weil sie die Nutzer*innen in einem Schritt-für-Schritt-Prozess durch den Projektentwicklungsprozess leitet.

Das Projekt sollte als Webapplikation umgesetzt werden, um auf vielen Geräten nutzbar und für alle zugänglich zu sein. Es sollte auf responsives Design und Accessibility geachtet werden. Die Applikation basiert auf einer Client-Server-Struktur, wobei der API-Server planmäßig auf PHP & Symfony und der Client auf Javascript und NodeJS basiert. Sie ist damit auf vielen Servern aufwandsarm einsetzbar, und kann auch relativ aufwandsarm in Container-Umgebungen³. Um die Weiternutzung der entstehenden Daten zu ermöglichen und die Darstellung der Projekte nicht an die Applikation zu binden werden die Projektinformationen über eine JSON-REST-API verfügbar gemacht. Bei der Entwicklung sollte außerdem soweit möglich auf erprobte OpenSource-Bibliotheken und -Tools zurückgegriffen um den Einstieg für zukünftige Weiterentwicklung durch andere leicht zu gestalten und Neuentwicklungsaufwand zu minimieren.

Drei Arbeitsphasen waren vorgesehen:

In der **Konzeptionsphase** werden die Erfahrungen aus dem Zukunftsstadt-Projekt strukturiert und in ein Lastenheft formuliert inkl. Use-Cases, Datenkatalog, Workflow-Design, Funktionsbeschreibungen.

In der **Programmierphase** werden die Anforderungen des Lastenhefts in Software umgesetzt. Aufscheinende Programmier-Probleme werden über eine Anpassung der Konzeption gelöst.

In der Phase der **Usability-Optimierung** werden grafische Elemente erstellt und HTML/CSS so angepasst, dass für die Nutzer eine moderne, selbsterklärende und leichtgängige Nutzungserfahrung entsteht.

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

Das Projekt zielt auf zwei Zielgruppen:

- Zielgruppe A: Die Bürger*innen der Städte.
- Zielgruppe B: Die Stadtverwaltungen.

Städte suchen zunehmend nach Wegen zur Bürgerbeteiligung und nach smart-city-Anwendungen. Das vorliegende Tool soll den Stadtverwaltungs-Abteilungen helfen, mit knappen Personalressourcen nachhaltige Beteiligungsprozesse umzusetzen und Bürgerprojekte zu unterstützen.

Die entstandene Software steht als OpenSource unter MIT-Lizenz zur Verfügung. Sie ist als innovatives Werkzeug an der Schnittstelle zwischen Kommunalverwaltungen und Bürgerschaft einsetzbar. Sie adressiert die aktuell dringliche gesellschaftliche Fragestellung, wie angesichts globaler Umwelt- und Klimakatastrophen die lokale Ebene adressiert und eingebunden werden kann.

³<https://www.dev-insider.de/was-sind-docker-container-a-597762/>

Das Tool ist der Kategorie „Civic Tech“ zuzuordnen: Gesellschaftliche Teilhabe, Transparenz und Bürgerbeteiligung.

Ausführliche Darstellung der Ergebnisse

Zentrales Ergebnis ist die Bereitstellung der Software unter MIT-Lizenz bei GitHub:

- <https://github.com/FutureCityProjects/>

Die avisierten Meilensteine wurden erreicht, die Entwicklungsphasen erfolgreich abgeschlossen. Nicht alle angestrebten Einzelfunktionen konnten umgesetzt werden. Dennoch ist das Ergebnis in sich funktions- und einsatzfähig.

Neben der Software selbst wurden bei den Beteiligten Erkenntnisse erzielt:

- Erneutes Lernen einer Antragstellung ans BMBF/Prototypefund
- Übliche Stolperfallen beim Open-Source-Einsatz sowie Umgang mit React & Redux
- Wie „digitalisiert“ man ein erprobtes analoges Bürgerbeteiligungsverfahren?
- Wie setzt man ein Lastenheft/Pflichtenheft bei der prototypischen Softwareentwicklung ein?

Für den Einsatz eines Pflichtenheftes wurde eine allgemein einsetzbare, übertragbare Pflichtenheft-Struktur erarbeitet.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Die Zielgruppe „Kommunen“ kann ab sofort auf eine Open-Source-Softwarelösung zurückgreifen, um Beteiligungsverfahren nach dem Muster des Dresdner „Zukunftsstadt“-Prozesses umzusetzen. Die Software ist relativ leicht installierbar und niederschwellig bedienbar, so dass über den kommunalen Einsatz die Zielgruppe „Bürgerinnen und Bürger“ einbindbar ist. Die Antragshürden in bestehenden Förderprogrammen können mittels der Software gesenkt werden, neue Förderprogramme bürgernäher aufgesetzt werden. Zudem kann die Software und die durch sie entfaltete Plattform dazu genutzt werden, Bürger*innen in die nachhaltige Stadtentwicklung einzubinden.

Die Zuwendungsempfänger wollen die Software gern weiterentwickeln und einsetzen. Daher suchen sie aktuell nach Kommunen als Partner sowie nach Investoren, um fehlende Bausteine zu ergänzen und die Software noch leichter einsatzfähig und bedienbar zu machen sowie auf konkrete Anforderungen möglicher Nutzer anzupassen. Ein Förderantrag auf Innovationsförderung beim Amt für Wirtschaftsförderung Dresden wurde eingereicht, nach weiteren Fördermöglichkeiten wird gesucht.

Prinzipiell können durch die Veröffentlichung des Quelltextes problemlos (Sicherheits-)Audits vor dem Einsatz im kommunalen Umfeld vorgenommen werden und andere Akteure weitere Komponenten zur Gesamtlösung beisteuern die das Einsatzfeld erweitern was wiederum allen Nutzern zu gute kommt. Bisher wurde diese Möglichkeit jedoch noch nicht genutzt.

Persönlich haben wir viel über den aktuellen technischen Stand von Softwareentwicklung gelernt, was in weiteren Projekten anwendbar sein wird.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Die Ausgangsplanung konnte grundsätzlich umgesetzt und die angestrebten Ziele erreicht werden.

Es gibt einige wenige Bausteine, die mangels Zeit nicht umgesetzt werden konnten:

- Team-Einbindung: derzeit können Projekte nur von einer Einzelperson statt eines Teams bearbeitet werden
- Jury-Verfahren: die Bewertung geplanter Projekte durch eine Jury ist technisch noch nicht möglich

Hintergrund war, dass die verwendeten Open-Source-Bausteine teilweise schlecht dokumentiert sind und wenige Anwendungsbeispiele vorhanden waren, sowie Bugs aufwiesen. Die Einarbeitung und die darauf aufbauende Herstellung der Basisfunktionalität dauerte länger als geplant. Außerdem beanspruchte die Erstellung des Pflichtenhefts eine größere Zeit als geplant. Dadurch fehlte die Zeit für die oben genannten Bausteine.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Als zentrale Informationsseite dient diese URL:

- <https://zukunftsstadt.de/leistungen/futurecityprojects>

Dort verlinkt sind weiterhin:

- eine Präsentation über die Funktionalität der Software: https://zukunftsstadt.de/wp-content/uploads/2020/02/FutureCityProjects_ProjektWerkStadt_fuer_Buergerprojekte.pdf
- der Link zum GitHub-Repository: <https://github.com/FutureCityProjects/>
- eine laufende Instanz der Software zum Ausprobieren und Testen für jedermann: <https://futurecityprojects.zukunftsstadt.de/>

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Die Ursprungsplanung konnte grundsätzlich wie angedacht realisiert werden. Kleinere Teil-Bausteine konnten nicht realisiert werden, sie sind im Kapitel „Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben“ aufgeführt. Dafür ergaben sich während der Konzeptionsphase noch weitere, vorher ungeplante Features die umgesetzt werden konnten.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Während der konkreten Umsetzungsphase gab es keine Entwicklungen anderer Stellen, die relevanten Einfluss auf unsere Arbeit und unsere Arbeitsergebnisse hatten. Es gab unsererseits Aktivitäten, die Software wie auch das zugrundeliegende Bürgerbeteiligungsverfahren an mögliche Partner, insbesondere Kommunen, heranzutragen. Die Resonanz war bislang verhalten, was auch daran liegt, dass der Prototyp erst zum Ende der Förderphase planmäßig vorliegt und erst jetzt eine gute, plastische Kommunikation mit diesen Akteuren möglich ist.

Richtlinie zum „Software-Sprint“

Festival Grid

DLR Projektträger Berlin

Eingegangen am:

31. März 2020

Framework zur Echtzeit-Evaluierung des Energieverbrauchs auf Kulturveranstaltungen und Festivals

Eingangsnr.: 1375 

Schlussbericht

Zuwendungsempfänger:

Michael Zöllner

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS17S10 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Festival Grid ist ein modulares Framework zur Echtzeit-Evaluierung des Energieverbrauchs auf Kulturveranstaltungen und Festivals mit Hilfe von günstigen Hardware-Elementen und OpenSource Software. Durch die Echtzeit-Messung aller Verbraucher-Kompartimente eines Events soll das Bewusstsein für den ökologischen Impact solcher Veranstaltungen geschäft werden. Veranstalter*innen sollen durch detaillierte Statistiken und Echtzeitregulierung darin unterstützt werden, ihre Veranstaltungen klimaneutral zu konzipieren. Besucher*innen wollen wir schon auf dem Event aufzeigen, wie viel Energie die jeweiligen Angebote verbrauchen.

Die wichtigsten Meilensteine (Details siehe „Ausführliche Darstellung der Ergebnisse“):

- Django Server, Rest API, Django Admin, Topologie Admin, Regeleditor
- Geräte-Schnittstellen Plugin Interface, Sonoff-Tasmota Plugin
- Cordova App für Android Smartphones
- Graphische Darstellung der Verbrauchsdaten
- Dokumentation

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

Open-Air Events sind Orte, an denen enorm viel Strom verbraucht wird – häufig aus schlechten, fossilen Energiequellen, die dem Klima schaden. Festival Grid ist vornehmlich ein Tool für Veranstalter*innen von Festivals und Kulturveranstaltungen. Tatsächlich kann die Software aber in allen Anwendungsfeldern Verwendung finden, in denen eine vorrübergehende elektrische Infrastruktur betrieben wird, beispielsweise der Filmproduktion.

Festival Grid unterstützt darin, die Infrastruktur zu planen, den Betrieb zu überwachen, sämtliche Verbrauchsdaten zu erfassen und übersichtlich darzustellen.

Als Open Source Software kann Festival Grid leicht erweitert werden, so kann die Unterstützung für weitere Messhardware in Form von Plugins mit wenigen Zeilen Python Code erreicht werden.

Ausführliche Darstellung der Ergebnisse

Der Prototyp wurde erfolgreich implementiert. Alle angestrebten Milestones wurden erreicht, teilweise wurden aber einzelne Funktionen abgeändert oder gänzlich gestrichen, dafür weitere - in der anfänglichen Planung nicht vorgesehenen Funktionen - hinzugefügt.

Es wurden folgende Meilensteine implementiert:

- Django Server, Rest API
 - Hier werden die erfassten Verbrauchsdaten gespeichert und ausgewertet. Der Server kann mittels Docker leicht auf Linux-kompatibler Hardware installiert und betrieben werden.
- Django Admin, Topologie Admin Leaflet (Kartendarstellung)
 - Das Konfigurations-Interface von Festival Grid ist webbasiert. Betreiber*innen von elektrischer Infrastruktur können hier alle Messgeräte konfigurieren und die eingehenden Verbrauchsdaten in tabellarischer Darstellung, sowie als Graph, live einsehen. Es wurde eine „Scan“ Funktion implementiert, mit der Geräte automatisch im Netzwerk gefunden werden können. Gefundene Geräte können dann automatisch provisioniert werden (d.h. so konfiguriert, dass sie verwendbare Verbrauchsdaten erfassen und Festival Grid mitteilen).
 - Die Kartendarstellung wurde umfangreicher als ursprünglich geplant. Hier können alle Verbrauchspunkte („Nodes“) angelegt und auf einer Karte (Open Street Map oder Satellitenbilder) platziert werden. Kabelstrecken zwischen den Nodes können ebenso angelegt werden wie optionale Hintergrundgrafiken, beispielsweise für Innenräume.

- Verbrauchsregeln
 - Im Admin können Schwellenwerte für Warnung und Notabschaltung jeder Node konfiguriert werden.
- Geräte-Schnittstellen Plugin Interface (Stromverbrauch und Schalterfunktion)
 - Festival Grid ist grundsätzlich mit aller Messhardware kompatibel, die über gängige Netzwerke erreichbar ist (UDP, TCP/IP, HTTP(S), etc.). Es wurden zwei Plugin-Erweiterungen implementiert, auf denen neue Plugins aufsetzen können: HTTP Polling und MQTT. Desweiteren gibt es einen Provider von Zufallswerten für Demo-Zwecke.
- Implementation einer Geräte-Schnittstelle (Sonoff-Tasmota)
 - Hunderte mit der Open Source Firmware Tasmota kompatible Geräte werden von Festival Grid unterstützt. Eine Liste findet sich unter <https://templates.blakadder.com>, die Projekt Webseite unter <https://github.com/arendst/Tasmota>. Weitere Plugins können leicht hinzugefügt werden.
- Cordova App für Android Smartphones
 - Die App ermöglicht es Betreiber*innen vor Ort die aktuellen Verbrauchsdaten auf dem Smartphone einzusehen. Sollten im Web Admin konfigurierte Schwellenwerte überschritten werden, werden diese mit Alarmsound als Notification dargestellt. Festival Grid konnte einige Pull Requests zu der - von uns für Push Notifications verwendeten Software Gotify - besteuern und dort Funktionen hinzufügen, die für unsere Verwendungsart unabdingbar waren.
- Graphische Darstellung der Verbrauchsdaten
 - Hier kommt die Open Source Software Grafana zum Einsatz. Alle Konfigurationen (Darstellung, Datenquelle, etc.) werden vom Django Server automatisch vorgenommen. Desweiteren werden auch Graphen zur Serverüberwachung (Arbeitsspeicher, CPU-Auslastung, etc) erstellt.
- Dokumentation
 - Mit Sphinxdoc wird anschauliche Dokumentation generiert. Alle Funktionen wurden beschrieben. Ein Teil der Dokumentation sind Tutorials zum Aufsetzen des Systems. Der Programmcode wurde ebenfalls dokumentiert, auch diese Dokumentation wird mit Sphinxdoc generiert (z.B. die Beschreibung der Plugin Schnittstelle).

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Festival Grid sollte dieses Jahr auf mehreren Festivals und Veranstaltungen zum Einsatz kommen. Es bleibt allerdings unsicher, ob 2020 überhaupt noch Veranstaltungen stattfinden können. Neben festen Zusagen hatten wir eine Vielzahl von Anfragen großer und kleiner Festivals. Das Thema Ökobilanz und generell das Bedürfnis Verbrauchsdaten zu erheben ist groß. Wir versprechen uns von Festival Grid ein positives Einwirken auf die Planung und Durchführung von Festivals.

Naheliegende Weiterentwicklungen sind die Implementation weiterer Plugins für Geräte, die Unterstützung weiterer Netzwerksituationen (Ipv6, LoRa), sowie die Erweiterung von Dokumentation und Homepage.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Die implementation von Warnungsmeldungen an das Smartphone hat sich als wesentlich komplexer herausgestellt. Dies lag vor allem an den Einschränkungen des Android Smartphone-Betriebssystems. Push Notifications im lokalen Netzwerk, ohne die Cloudinfrastruktur der Hersteller, ist nicht vorgesehen. Hier wurde einiger Programmcode geschrieben, der verworfen werden musste.

Schlussendlich konnten aber beide Ziele – dass Festival Grid auch ohne Internetverbindung funktioniert, und dass möglichst keine closed-source Lösungen verwendet werden sollen – erreicht werden. Im Verlauf der Implementierung wurde allerdings einige schon erstellten Funktionen wieder entfernt. Generell konnten dann – wo möglich – existierende Open Source Projekte verwendet und keine eigenen Lösungen implementiert werden. Im Falle der Push Notifications fiel die Wahl auf Gotify, im Falle der visuellen Darstellung der Verbrauchsdaten auf Graphana.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Wir betreiben eine Homepage unter <https://festivalgrid.org>, dort finden sich weiterführende Links zu Quellcode und Dokumentation. Letztere wird in den nächsten Monaten sukzessive erweitert, insbesondere mit Erfahrungsberichten und Tutorials für Deployments an verschiedenen Orten.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Der zeitliche Aufwand war soweit korrekt veranschlagt. Die veranschlagte Stundenanzahl wurde leicht überschritten.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Die Anbindung an das Leipziger Kulturkollektiv „Keine Fische aber Grethen“ hat vor allem in der Planungsphase geholfen, die Ausrichtung des Projektes zu schärfen. Dies geschah insbesondere durch Gespräche mit befreundeten Festivalcrews und der Anpassung der Funktionalitäten um auf spezifische Bedürfnissen an die elektrische Infrastruktur einzugehen.

Richtlinie zum „Software-Sprint“

PSE – Portal Sozialbilanz Elektronik

Schlussbericht

DLR Projektträger Berlin
Eingegangen am:

Zuwendungsempfänger:

25. März 2020

Beschke und Fritsch GbR

Eingangsnr.: 1284 

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S11 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Auf allen Stufen der Lieferkette von Elektronikprodukten kommt es zur Verletzung von Menschenrechten. Aus der Vereinsarbeit des FairLötet e.V. entstand die Erkenntnis, dass gerade kleinere Elektronikhersteller*innen zwar für diese sozialen Risiken sensibilisiert sind, ihnen jedoch konkrete Handlungsoptionen fehlen. Gleichzeitig fehlt für klassische Methoden des Nachhaltigkeitsmanagements wie Lifecycle Assessments (LCA) in der Regel das Budget.

Mit dem Projekt Portal Sozialbilanz Elektronik soll ein niedrighschwelliger Einstieg in die Nachhaltigkeitsanalyse ermöglicht werden. Hersteller*innen sollen selbst die risikoreichsten Komponenten ihrer Geräte identifizieren und gezielt Handlungsalternativen entwickeln können. Dadurch soll die Sichtbarkeit des sozialen Nachhaltigkeitsmanagements in der Elektronikbranche gestärkt und der Stellenwert von Sorgfaltspflicht und sozialer Verantwortung erhöht werden.

Das Ziel des Projekts war die Erstellung eines Web-Portals, welches Anwender*innen ermöglicht, die Fairness von Elektronikprodukten analysieren zu lassen. Folgende Komponenten waren zur Umsetzung geplant:

(1) Ein grafisches Interface bereitet die Sozialanalyse verständlich auf und liefert eine Einordnung der berechneten Risiken. Zudem werden Bauteile hervorgehoben, welche besonders hohes Risiko beitragen.

Als Berechnungsgrundlage dient eine **(2) Modellierung des Elektronikprodukts durch generische Bauteile**. Diese bilden die Zusammensetzung der Bauelemente aus Rohstoffen ab. Weiterhin werden Statistiken über die Herkunft der einzelnen Rohstoffe sowie die damit jeweils verbundenen sozialen Risiken verarbeitet, welche aus frei verfügbaren Publikationen entnommen werden können.

Für Nutzer*innen wird eine **(3) grafische Schnittstelle zur Eingabe der Teileliste eines Elektronikprodukts** bereitgestellt, welche als Webanwendung bedient wird. Auf Basis dieser Teileliste wird mittels der quelloffenen Software OpenLCA **(4) eine Analyse der mit den einzelnen Teilen in Verbindung stehenden sozialen Risiken** erstellt. Diese Analyse wird der Nutzer*in wiederum im Browser ansprechend und verständlich präsentiert.

Für die interne Verwendung entsteht zudem **(5) ein Eingabewerkzeug**, mit welchem das Projektteam sowie Engagierte des FairLötet e.V. die Datenbasis des Portals erweitern und pflegen können.

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

Primäre Zielgruppe für das Portal Sozialbilanz Elektronik sind Elektronikhersteller*innen, insbesondere kleinere Unternehmen und Startups. Sie können mit dem Werkzeug besonders risikobehaftete Komponenten identifizieren und daraus folgend mögliche Maßnahmen, wie Änderungen beim Einkauf von Komponenten oder Designalternativen bei der Ausgestaltung der Produkte, prüfen.

Diese Zielgruppe profitiert von der Entwicklung des Portals dadurch, dass Methoden des Nachhaltigkeitsmanagements auch für Unternehmen kostengünstig verfügbar gemacht werden, für welche ihr Einsatz ansonsten aus Kosten- oder Auslastungsgründen nicht in Frage käme. Zudem werden diese Methoden für Unternehmen zugänglich gemacht, welche über keine einschlägige Expertise verfügen.

Im Rahmen der Vereinsarbeit des FairLötet e.V. und darüber hinaus kann das Werkzeug zudem für Bildungszwecke eingesetzt werden, z.B. im Hochschulkontext oder auf Veranstaltungen mit Elektronik- oder Nachhaltigkeitsbezug. Der FairLötet e.V. erreicht durch Bildungsveranstaltungen bisher insbesondere Studierende mit Schwerpunkt auf Nachhaltigkeitsmanagement, welche durch die Veranstaltungen vertiefte Einblicke in die Besonderheiten von Lieferketten von Elektronikprodukten erhalten.

Ausführliche Darstellung der Ergebnisse

Alle geplanten Meilensteine wurden erreicht. Das Ergebnis gliedert sich in eine Frontend-Komponente, die als Browser-Anwendung in Javascript entwickelt wurde, und eine in Python geschriebene Backend-Komponente, die benötigte Daten und Berechnungsergebnisse bereitstellt.

Die Frontend-Komponente bietet Funktionalitäten für Anwender*innen und Administrator*innen. Anwender*innen (Also Vertreter*innen der Zielgruppe des Projekts) können darüber Elektronikprodukte anlegen und konfigurieren und erhalten eine Übersicht und Visualisierung der ermittelten sozialen Risiken in Bezug auf verbaute Komponenten, der darin enthaltenen Rohstoffen und der Herkunftsländer der Rohstoffe. Über einen Admin-Bereich können Administrator*innen diese Daten bearbeiten und Ergänzen. Die Frontend-Komponente deckt damit die Meilensteine (1), (3) und (5) ab.

Die Backend-Komponente speichert die zur Berechnung notwendigen Daten und Produktmodelle (Meilenstein 2). Weiterhin bietet es, ergänzend zur grafischen Eingabemöglichkeit der Frontend-Komponente, für fortgeschrittene Anwender*innen die Möglichkeit, umfangreiche Datenbestände über CSV-Dateien zu aktualisieren (Meilenstein 5). Abweichend von der ursprünglichen Planung, stellte sich heraus, dass für die Umsetzung der für die Analyse notwendigen Berechnungen (Meilenstein 4) keine Anbindung an OpenLCA notwendig ist. Die Berechnungsfunktionen wurden stattdessen als Modul der Backend-Komponente implementiert.

Besonders wertvoll für die Erreichung der Projektziele war das von der Open Knowledge Foundation vermittelte Usability-Coaching. Durch diese Unterstützung war es uns möglich, die Usability der Frontend-Komponente wesentlich zu verbessern.

Ergänzend zur Softwareentwicklung wurden auch Ergebnisse in der Öffentlichkeitsarbeit erzielt. Das Projekt wurde vom Deutschlandfunk interviewt (https://www.deutschlandfunk.de/36c3-fairtronics-wie-elektronik-sozial-und-oekologisch.684.de.html?dram:article_id=466884) und konnte Zwischenergebnisse auf dem 36C3-Kongress (<https://media.ccc.de/v/36c3-10592-fairtronics>) und der Night of Open Knowledge (<https://nook-luebeck.de/talks/wie-fair-ist-mein-geraet/>) vorstellen.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Ein wichtiger Anwendungsfall für das Werkzeug ist das Prüfen von Handlungsalternativen beim Entwurf von Elektronikprodukten. Elektronikhersteller, insbesondere kleinere Unternehmen und Startups, können mit dem Tool besonders risikobehaftete Komponenten identifizieren und mögliche Maßnahmen (z.B. alternatives Sourcing, Designalternativen) prüfen.

Während der Förderphase kamen bereits Kontakte zu einzelnen Unternehmen zustande, welche im Lieferkettenmanagement oder dem Handel mit Elektronikprodukten involviert sind. Diese Kontakte werden wir im Hinblick auf eine Anwendbarkeit unserer erarbeiteten Methoden weiter verfolgen. Durch die Zusammenarbeit mit Plattform- bzw. Dienstleistungsanbietern könnte die Sichtbarkeit unseres Projekts erhöht und viele weitere Unternehmen zur Anwendung von sozialen Risikoanalysen ermuntert werden.

Im Rahmen der Vereinsarbeit von FairLötet und darüber hinaus kann das Werkzeug zudem für Bildungszwecke eingesetzt werden. Ein Ziel des Vereins ist, nunmehr Risikoanalysen zu konkreten Produkten zu erstellen und zu veröffentlichen.

Zusammenfassend wurde mit der Durchführung des Projekts ein Grundstein gelegt, die Thematik der sozialen Nachhaltigkeit auch in Zukunft vertieft zu verfolgen und Softwarelösungen in diesem Themenbereich zu entwickeln.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Ein Lösungsansatz, welcher nach näherer Analyse nicht weiter verfolgt wurde, war die Anbindung von OpenLCA als Berechnungsmodul. Die Komplexität der Anbindung der externen Software hätte hier den Aufwand überstiegen, die benötigten Berechnungsmethoden selbst zu implementieren.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Zur Präsentation der Projektergebnisse wurde die Website <https://fairtronics.org/> angelegt. Dort wird der Hintergrund und die Methodologie des Projekts erläutert. Zudem kann die Software im Rahmen einer Demo ausprobiert werden.

Der Quellcode steht auf der Plattform Gitlab unter <https://gitlab.com/fairloetet/pse/> zur Verfügung.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Das Projekt konnte im Rahmen des geplanten Kostenumfangs bearbeitet werden.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Neben der im Rahmen des Prototype Fund geleisteten Arbeit gab es Beiträge von weiteren Personen, welche teils ehrenamtlich geleistet und teils durch den FairLötet e.V. gefördert wurden. Hierbei sind insbesondere Beiträge in den Bereichen Frontend-Entwicklung, UI-Design, Datenrecherche, DevOps sowie Webdesign zu nennen.

Durch diese Beiträge konnte insgesamt ein größeres Augenmerk auf User Experience-Design und eine hochwertige Präsentation des Ergebnisses gelegt werden. Durch die Integration aller Beiträge wurde zugleich der Grundstein für die zukünftige Zusammenarbeit an dem Projekt gelegt.

osemData – Modus Datenanalyst für die openSenseMap

Schlussbericht

DLR Projektträger Berlin
Eingegangen am

Zuwendungsempfänger:

11. März 2020

Umut Tas

Eingangsnr.: 0556 NM

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S12 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Die Motivation hinter dem Projekt war es die Citizen Science Daten, die seit Jahren auf der openSenseMap gesammelt werden zugänglicher zu machen. Jeder Benutzer soll die Möglichkeit haben so einen Mehrwert aus den Daten zu ziehen und Theorien wie unser Handeln die Umwelt beeinflusst zu überprüfen. Dies war bis jetzt Benutzern vorbehalten die die Daten weiterverarbeiten konnten. Zudem sollte ein Augenmerk darauf gelegt werden das es den Benutzern Spass macht sich die Daten anzugucken und Hypothesen und Theorien über Umwelteinflüsse zu überprüfen. Die Entwicklung der neuen Plattform basiert auf Umfragen unter den Benutzern der openSenseMap.

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

Die Zielgruppe sind alle Benutzer der openSenseMap, die sich in drei Gruppen aufteilen lassen. Zum einen die Interessierten Bürger, die gerne mehr über Ihre Umwelt wissen möchten und Teil der Citizen Science Kampagne sind. Zudem wird die openSenseMap viel in der Bildung genutzt. Schüler lernen wie sie IoT-Geräte benutzen und Daten im Internet teilen. Jetzt haben Sie zusätzlich ein Werkzeug um die gesammelten Daten besser zu verstehen. Die dritte Zielgruppe wären Wissenschaftler, denen wir es ermöglichen wesentlich einfacher anomalien oder interessante Umweltphänomene zu entdecken und genauer zu untersuchen um so wissenschaftliche Fragestellungen beantworten zu können.

Ausführliche Darstellung der Ergebnisse

Es wurden zwei konkrete Ergebnisse während des Prototypenfundes erzielt.

Zum einen ein Aufsetzen der Serverinfrastruktur auf einem OpenStack System (gehostet bei der Universität Münster) welches eine langfristige und zuverlässige Bereitstellung der Daten garantiert. Alle Skripte (Terraform, Ansible, Prometheus, etc..) wurden als OpenSource bereit gestellt und ermöglichen jedem eine Kopie der Infrastruktur selbst zu hosten.

Zum anderen wurde eine Plattform entwickelt die eine ansprechende Darstellung der Daten ermöglicht. Hierzu wurden eine Anpassungen an der API und dem Datenmodell der openSenseMap implementiert. Der Großteil der Arbeit floss in eine Frontend-Applikation die auf dem Angular Framework basiert und die am dringlichsten gewünschten Funktionen der Benutzer implementiert, wie z.B. Vergleiche, Zeitliche Entwicklungen oder räumliche Zusammenhänge.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Die Zielgruppen und Nutzen wurden bereits auf Seite 1 erläutert.

Da die Ergebnisse noch nicht produktionsreif sind und momentan nur eine „eingeschränkte“ Version online verfügbar ist wird die Weiterentwicklung des Prototypen zu einem fertigen Produkt die erste Weiterentwicklung sein.

Anschliessend sind weitere Arbeiten geplant die nicht mehr in den Rahmen dieses Prototypenfundes gepasst haben, vorallem die Vereinheitlichung der Daten und anschliessend ein System um die Daten zu Klassifizieren um es den Benutzern zu ermöglichen genauere Analysen zu treffen (z.B. benötigen einige Fragestellungen Daten mit höherer Genauigkeit als andere, wir möchten allen Benutzern die Möglichkeit geben die Daten entsprechend zu filtern). Die Grundlage um diese beiden Probleme anzugehen wird ein Sensor-Wiki sein, welches Phänomene, Einheiten und Sensoren semantisch definiert.

Zudem befinde ich mich im Gespräch mit anderen Plattformen die Umweltdaten anbieten um die Ergebnisse dieser Förderung dort zu Nutzen (e.g. Luftdaten.info, Sensor.Community).

Für meine persönliche Entwicklung war die Arbeit an diesem Projekt sehr positiv. Da ich eher als Frontend-Entwickler gearbeitet habe in meiner Laufbahn, war die Entwicklung der Infrastruktur super interessant, anspruchsvoll und im endeffekt aber auch sehr Lehrreich. Zudem kam ich durch die Treffen bei der OKFN in Kontakt mit Personen die an ähnlichen Projekten arbeiten und vorallem auch ähnliche Ideologien mit mir teilen, die ich vermutlich sonst niemals getroffen hätte.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Es gab zwei grössere Ansätze die ich nicht weiter verfolgt hab während der Förderung (die für das Projekt aber weiterhin wichtig sein werden in der Zukunft).

Zum einen die Umstellung der Datenbank auf eine Time-Series Datenbank mit der es wesentlich effizienter wäre die Daten abzurufen. Diesen Ansatz habe ich nicht weiter verfolgt, weil es einfach zu viel Arbeit wäre und den Rahmen des Prototypenfundes gesprengt hätte. Stattdessen habe ich versucht die Menge der Daten die für die Visualisierungen gebraucht wird zu reduzieren.

Ein zweites Arbeitspaket das nicht weiter verfolgt wurde waren 3D-Visualisierungen, die zwar sehr gut aussehen und vielen Benutzern Spass machen, aber zu wenig Mehrwert lieferten um den Aufwand zu rechtfertigen. Die reine Darstellung der Daten war mit einfacheren Methoden klarer und besser realisierbar für alle Endgeräte.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

<https://sensebox.de> – Das Projekt aus dem die openSenseMap entstanden ist, hier findet man allgemeine Informationen zur CitizenScience Kampagne.

<https://docs.opensensemap.org> – Beschreibung der API

<https://data.opensensemap.org> – Prototyp des während der Förderung entwickelten Frontends

<https://github.com/sensebox/openSenseMap-infrastructure> – Die Infrastruktur bei Github

<https://github.com/sensebox/openSenseMap-api> – Die API bei Github

<https://github.com/sensebox/openSenseMapX> – Das Frontend bei Github

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Es kam ein Arbeitspaket unerwartet dazu, dafür habe ich auf ein anderes verzichtet. Im Projektplan hatte ich von einer „Bereinigung der Daten“ und Anwendung von statistischen Methoden um diese besser nutzbar zu machen gesprochen. Dieses Arbeitspaket ist aufgrund der Infrastruktur aus dem Projektplan gefallen, weil die bestehende Infrastruktur so leider nicht weiterverwendet werden konnte mussten wir den Umzug zum OpenStack der Universität Münster implementieren. Ohne Server von denen man die Daten abrufen kann wären die Besten statistischen Methoden nichts Wert gewesen.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

-

30. März 2020

JMAP – Verbesserte Energieeffizienz für K-9 Mail

Eingangsnr.: 1352 

Schlussbericht

Zuwendungsempfänger:
Christian Ketterer

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S13 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Aufgabenstellung und Motivation

JMAP (JSON Meta Application Protocol) ist ein neues, offenes Netzwerkprotokoll für den Zugriff auf E-Mail-Postfächer und den Versand von E-Mails, das von der IETF JMAP Working Group entwickelt wurde. Ein wichtiges Ziel gegenüber Vorgängerprotokollen ist eine erhöhte Energieeffizienz durch die Reduzierung von Netzwerkverkehr und Flexibilität bei der Benachrichtigung über neue E-Mails.

Ziel dieses Projekts war es Unterstützung für das JMAP-Protokoll zu K-9 Mail, einer Open-Source Email-App für Android, hinzuzufügen. Damit soll sichergestellt werden, dass K-9 Mail sowohl in Bezug auf Funktionalität als auch Energieeffizienz mit proprietären Lösungen mithalten kann.

Der Plan für dieses Projekt umfasste grob folgende Punkte:

- Aufsetzen einer Testumgebung
- Erstellen von Testprogrammen zur Implementierung der einzelnen Funktionen (Abruf der Ordnerliste, Abruf der Liste der E-Mails in einem Ordner, Abruf einzelner E-Mails, Versenden einer E-Mail usw.)
- Integration dieser Funktionalität in K-9 Mail
- Implementierung sekundärer Funktionalität (E-Mails als gelesen markieren, löschen, verschieben)
- Implementierung des Push-Mechanismus zur energiesparenden, zeitnahen Benachrichtigung über neue E-Mails
- Erstellen von Dokumentation zur Benutzung von K-9 Mail mit JMAP-Servern

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

Zielgruppe sind Anwender von K-9 Mail, die per JMAP auf den E-Mail-Server ihrer Wahl zugreifen möchten. Durch die Verwendung von JMAP profitieren sie von der erhöhten Energieeffizienz, was eine längere Nutzung ihres Smartphones mit einer Akkuladung ermöglicht. Weiterhin reduziert sich in vielen Fällen die Menge der übermittelten Daten. Je nach Mobilfunktarif reduziert sich dadurch der Preis oder es steht mehr Datenvolumen für andere Anwendungen zur Verfügung.

Ausführliche Darstellung der Ergebnisse

Die Entscheidung welche Server-Software für die Testumgebung benutzt werden sollte, hat einige Zeit in Anspruch genommen. Letztendlich wurde aber die Software „Cyrus IMAP“ verwendet.

Für die Implementierung des JMAP-Protokolls wurde die Software-Bibliothek „Java JMAP Library“ verwendet. Damit die Bibliothek für das Projekt nutzbar war, mussten einige Änderungen daran vorgenommen werden. Die Kommunikation und Zusammenarbeit mit dem Autor der Bibliothek hat sehr gut funktioniert.

Mit Hilfe der Bibliothek wurden Testprogramme zum Abrufen der Ordnerliste, zum Abruf der Nachrichtenliste und zum Abruf von Nachrichten erstellt. Die Testprogramme wurden vorwiegend dazu verwendet um die „Java JMAP Library“ zu testen und zu ergründen ob Änderungen an der Bibliothek notwendig sind. Es wurden einige Änderungen durchgeführt, auch kleinere Fehler gefunden und behoben.

Ein weiterer wichtiger Punkt der Testprogramme war das Sammeln von Praxiserfahrung mit dem Protokoll. Das Experimentieren ist mit kleinen Testprogrammen deutlich einfacher als mit Teilen einer relativ komplexen Applikation wie K-9 Mail.

Im nächsten Schritt wurde das :backup:jmap Modul in K-9 Mail erstellt. Dieses Modul benutzt dieselbe „Backend“-Schnittstelle, die auch von anderen Protokoll-Implementierungen in K-9 Mail (IMAP, POP3, WebDAV) verwendet wird. Da sich das JMAP-Protokoll etwas von den anderen Protokollen unterscheidet, waren kleinere Änderungen an dieser Schnittstelle notwendig.

Es wurde folgende Funktionalität in K-9 Mail integriert:

- Abruf der kompletten Liste von Ordnern auf dem Server
- Abruf der Änderungen in der Liste von Ordnern
- Abruf der kompletten Liste von E-Mails in einem Ordner
- Abruf der Änderungen an E-Mails in einem Ordner (neue/gelöschte E-Mails)
- Abruf der Änderungen von E-Mails („E-Mail gelesen“-Status usw.)
- Ändern des „gelesen“-Status
- Löschen von E-Mails
- E-Mails kopieren und verschieben
- E-Mails hochladen (z.B. Entwurf auf dem Server speichern)

Da ich für diese Meilensteine leider mehr Zeit als ursprünglich geplant benötigt habe, blieb keine Zeit mehr das Senden von E-Mails und Unterstützung für Push-Benachrichtigungen zu implementieren.

Nach 10 Jahren Arbeit an Open-Source-Software in meiner Freizeit, hat es mir diese Projektförderung das erste Mal ermöglicht in Vollzeit an Open Source zu arbeiten. Das hat die Art meiner Beiträge zu Open-Source-Projekten geändert und davon hat nicht nur das geförderte Projekt profitiert. Insgesamt habe ich diese Zeit sehr genossen und plane dies fortzusetzen. Im Moment verfolge ich mehrere Ansätze zur Finanzierung dieses Vorhabens.

Die Betreuung durch das DLR und die Open Knowledge Foundation haben das Projekt relativ reibungslos gestaltet, wofür ich sehr dankbar bin.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Im Rahmen dieses Projektes wurde K-9 Mail um grundlegende Funktionalität zur Unterstützung des JMAP-Protokolls erweitert. Diese Funktionalität ist allerdings nicht so weit fortgeschritten, dass ich sie für Endanwender freischalten will. Ich plane aber auch nach dem Förderzeitraum an diesem Projekt zu arbeiten, um die fehlende Unterstützung für Push-Benachrichtigungen und einer Bedienoberfläche zur Konfiguration von JMAP-Servern hinzuzufügen.

Die Veröffentlichung unter einer Open-Source-Lizenz bedeutet, dass andere durch das Lesen des Quellcodes von dem Projekt lernen können. Gleichzeitig ermöglicht es anderen bei der Implementierung der fehlenden Funktionalität helfen zu können.

Die Verfügbarkeit der JMAP-Software-Bibliothek unter einer Open-Source-Lizenz war eine große Hilfe für mich. Gleichzeitig führt meine Arbeit an dieser Bibliothek dazu, dass Fehlerbehebungen und Verbesserungen nicht nur K-9 Mail, sondern auch anderen Benutzern dieser Bibliothek zugute kommen.

Durch die Arbeit an dem Projekt habe ich ein gutes Verständnis über das JMAP-Protokoll erlangt. Interessant war es auch die Diskussionen auf der öffentlichen Mailingliste der JMAP Working Group während der Entwicklung des JMAP-Standards zu verfolgen. Viele Design-Entscheidungen gehen auf solche Diskussionen zurück und sind sonst nirgendwo dokumentiert.

Arbeiten, die zu keiner Lösung geführt haben

Zuerst habe ich versucht Apache James als Server-Software für die Testumgebung zu verwenden. Leider unterstützte die Software nur eine Vorabversion des JMAP-Protokolls, das sich bei näherer Betrachtung als zu unterschiedlich zu der finalen Version herausgestellt hat. Den Versuch die Implementierung zu ändern, so dass die finale Protokollversion unterstützt wird, habe ich abgebrochen, da sich die Arbeiten als deutlich zu umfangreich für den Projektzeitraum herausgestellt haben.

Letztendlich wurde die Entwicklungsversion von Cyrus IMAP als Server-Software für die Testumgebung verwendet. Da es sich um die bislang einzige ernstzunehmende Server-Implementierung handelt, kann Cyrus IMAP als de facto Referenzimplementierung angesehen werden.

Öffentliche Informationen über das Projekt

Die JMAP-Unterstützung in K-9 Mail ist noch nicht reif für den Einsatz durch Endanwender. Auch weil es noch keine anwenderfreundliche Benutzerschnittstelle zum Einrichten von JMAP-Konten in K-9 Mail gibt, existiert noch keine Dokumentation für Anwender.

Der Quellcode des Projekts befindet sich im regulären K-9 Mail Repository auf GitHub. Die Implementierung besteht zum Großteil aus dem Modul `:backend:jmap`. Damit die JMAP-Unterstützung getestet werden kann, wurde das App-Modul `:app:k9mail-jmap` erstellt. Für die noch ausstehende Arbeit wurde ein Milestone im K-9 Mail Issue Tracker erstellt.

Links:

- K-9 Mail auf GitHub: <https://github.com/k9mail>
- `:backend:jmap`: <https://github.com/k9mail/k-9/tree/master/backend/jmap>
- `:app:k9mail-jmap`: <https://github.com/k9mail/k-9/tree/master/app/k9mail-jmap>
- JMAP Milestone: <https://github.com/k9mail/k-9/milestone/26>
- Übersicht zum JMAP-Protokoll: <https://jmap.io/>

Anmerkungen zur Arbeits- und Kostenplanung

Da es sich bei JMAP um ein sehr neues Protokoll handelt und es zu Beginn so gut wie keine benutzbaren Implementierungen gab, war der Unsicherheitsfaktor bei diesem Projekt relativ hoch. Speziell das Aufsetzen der Testumgebung benötigte mehr Zeit als ursprünglich angenommen. Das Erweitern einer bereits existierenden JMAP-Software-Bibliothek, statt eine komplett neue Protokollimplementierung zu programmieren, sparte etwas Zeit. Die Integration der Funktionalität in K-9 Mail erforderte mehr Änderungen an der Architektur von K-9 Mail als vermutet, so dass auch hier mehr Zeit benötigt wurde als ursprünglich geplant.

Insgesamt bin ich mit den im Projektzeitraum erreichten Ergebnissen zufrieden.

Ergebnisse bei anderen Stellen

Daniel Gultsch hat die „Java JMAP Library“ (<https://github.com/iNPUTmice/jmap>) unter einer Open Source Lizenz zur Verfügung gestellt und explizit dazu eingeladen diese für das Projekt einzusetzen. Ich habe von dieser Einladung gerne Gebrauch gemacht und die Bibliothek zusammen mit ihm erweitert und Fehler behoben. Daniel hat auch einen Zugang zu seinem JMAP-Server für Testzwecke zur Verfügung gestellt.

Insgesamt war die Zusammenarbeit mit Daniel Gultsch und auch den Entwicklern des Apache James Projekts sehr angenehm. Unser gemeinsames Interesse am Thema E-Mail wird höchstwahrscheinlich dafür sorgen, dass wir auch in der Zukunft zusammen arbeiten werden.

Meine Beiträge zur Java JMAP Library: <https://github.com/iNPUTmice/jmap/commits?author=cketti>

Richtlinie zum „Software-Sprint“

GrowF – Growing Futures

Schlussbericht

Zuwendungsempfänger:

Andreas Gmeiner und Nic Dürr, Growing Futures

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S14 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Klimawandel, Nitratbelastung in Gewässern und das Artensterben sind zu Teilen auf die derzeitige agro-chemische und industrielle Form der Landwirtschaft zurückzuführen. Der Preisdruck auf dem Markt hindert viele Landwirte daran ihre Anbaumaßnahmen so auszugestalten, dass obige Herausforderungen adressiert werden können. Innerhalb von Solidarischen Landwirtschaften wird durch die feste Zusage der Finanzierung den Bauern und Bäuerinnen die Möglichkeit gegeben diesen Herausforderung trotz allem zu begegnen. Dieses Konzept ist sozioökologisch tragfähig und zeigt Wege in die Zukunft auf. Durch die geringe Flexibilität für Abnehmer bedienen SoLawis jedoch bisher nur eine Nische im Ernährungssystem. „Growing Futures“ ermöglicht es SoLawis, Gärtnern und Gärtnerinnen ein größeres gesellschaftliches Segment anzusprechen, ihren Markt zu vergrößern, um dadurch die Natur, die Artenvielfalt und das Klima zu schützen.

Dies lösen wir konzeptuell dadurch, dass bei individualisierten SoLawis die Ernte nicht zu gleichen Teilen auf die Mitglieder aufgeteilt wird, sondern die Planung und Aufteilung der Ernte auf Grundlage der Präferenzen des jeweiligen Mitglieds beruht. Die Rückmeldung der Mitglieder fließt zudem in die Anbauplanung der SoLawi ein, um schon im Vorfeld zu prognostizieren, welche Mengen angebaut werden müssen, um die Mitglieder möglichst gut nach ihren Präferenzen zu versorgen.

Um dies zu erreichen wurden zwei Applikationen entwickelt, die sich auch in den Meilensteinen widerspiegeln. Im Meilenstein 1 wurde ein Schwerpunkt auf bidirektionalen Informationsfluss zu den Mitglieder einer Solawi gelegt. Im Meilenstein 2 ging es um die Verwaltung der Ernte und von Anteilen sowie der individualisierten Bestückung der Anteile. In Meilenstein 3 wurde ein Algorithmus erstellt, um die Berechnung einer rudimentären Anbauplanung für die kommende Saison auf Grundlage der Präferenzen der Mitglieder zu ermöglichen.

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

Wir erwarten, dass durch die Einführung von individuellen Anteilsausgaben durch „Growing Futures“ die sog. „Retention Rate“, also der Prozentsatz der Mitglieder die nach einer Saison bei der SoLawi bleiben, steigt. Zudem erwarten wir, dass landwirtschaftliche Betriebe oder Initiativen denen bisher der Aufbau einer solidarischen Landwirtschaft zu komplex war sich dazu entscheiden eine SoLawi zu gründen, da durch die Software-Unterstützung die Einstiegshürde gerade für Gärtner, die in SoLawis nicht selten mehr als 80 verschiedene Kulturen anbauen, sinkt.

Eine niedrige Retention Rate hindert SoLawis daran zu skalieren, sowohl in der Tiefe (also mehr Mitglieder pro SoLawi) als auch in der Breite (also mehr SoLawis in einem Einzugsbereich). Unzufriedene Mitglieder, die aus einer SoLawi ausscheiden haben eine sehr geringe Wahrscheinlichkeit erneut in die gleiche oder eine andere Solawi einzusteigen.

Indem die Anteile individualisiert werden und ihr Feedback auch ankommt, erwarten wir einen signifikanten Anstieg der Zufriedenheit der Mitglieder und somit der Retention Rate. Dadurch ermöglichen wir die Skalierung und die Neugründung von SoLawi-Initiativen, um größere Teile der Bevölkerung in dieses alternative Ernährungssystem einzubinden. Nur dadurch bekommen wir die Möglichkeit dieses Ernährungssystem, in welchem externe Kosten von landwirtschaftlichen Produktionsprozessen internalisiert werden, zu einer signifikanten Größe anwachsen zu lassen. Im Falle von SoLawis bedeutet dies, dass durch die grundlegend agrarökologische Ausrichtung von SoLawis die ökologische, ökonomische und soziale Nachhaltigkeit im landwirtschaftlichen Sektor steigt. Durch die festen Beiträge, die Mitglieder an die SoLawi entrichten, ist es u.A. möglich ambitionierte agrarökologische Produktionsstandards einzuhalten, faire Löhne zu bezahlen und Landwirtschaft als eine gesamtgesellschaftlich vorteilhafte Branche weiter zu entwickeln. Auf diese Weise tragen SoLawis aktiv dazu bei Klimawandel, Artensterben, Nitratbelastung und Pestizidbelastung zu verringern.

Ausführliche Darstellung der Ergebnisse

Im Laufe des Projektablaufes wurden drei Meilensteine angestrebt. Die „Mitglieder-App“, die „Gärtner-App“ und die „Anbauplanung“.

Mitglieder-App

Die Mitglieder-App ist eine Mobile-First-App, die es Gärtnern ermöglicht die (individuellen) Anteile des kommenden Ausgabetermins an die Mitglieder der SoLawi zu übermitteln. In der Mitglieder-App ist es zudem den Mitgliedern möglich, dass sie sich zu den vergangenen Lieferungen rückmelden, um dem Gärtner Feedback zu Qualität und Quantität zu geben. Zudem beinhaltet die Mitglieder-App die Präferenzaufnahme für einzelne Produkte, die der Gärtner anbaut und welche als Grundlage für die individualisierte Anteilsausgabe verwendet wird. In dieser Präferenzaufnahme ist es möglich anzugeben, von welchen Gemüsesorten man welche Mengen erhalten will und wie viele Personen vom SoLawi-Anteil essen.

Gärtner App

Die Gärtner-App ist das Herzstück von Growing-Futures. In dieser App können die Gärtner Mitglieder, Anteile, Ausgabestellen, Rückmeldungen, Präferenzumfragen, ihr Ernte-Inventar und die

Ausgabemenge verwalten. e nach Funktionalität entweder auf dem Desktop oder in einer Mobile-App.

Für die softwaregestützte Ausgabeplanung gibt es im Tool verschiedene Optionen. In einer klassischen SoLawi wird geerntet was reif ist und die Mitglieder bekommen je nach ihrem Mitgliedsmodell eine feste Menge an Gemüse zugeteilt. Dies zu definieren ist aufgrund der Erntemenge auch bei Growing Futures möglich. Growing Futures ermöglicht es jedoch auch/zusätzlich individuell zusammengestellte Anteile für die einzelnen Mitglieder zu gestalten.

Diese individualisierte Ausgabeplanung hat sich im Laufe der Entwicklung in zwei unterschiedliche Modelle der individuellen Anteilsausgabe aufgeteilt: Zum einen die auf Grundlage des auf dem Luzernenhof verwendeten Verfahrens der "freien Entnahme" und eines auf Grundlage des Prognose und Planungsalgorithmus. Zusätzlich zu diesen beiden Ausgabeverfahren wird noch eine klassische "anteilsbasierte Ausgabe" und eine "Lieferung" als mögliche Ausgabe angeboten um ein möglichst breites Spektrum an SoLawi-Prozessen abzudecken.

- 1) Bei der "freien Entnahme" kann der Gärtner die vorhandenen und beerntbaren Kulturen hinterlegen. Diese werden aggregiert mit einer subjektiven Einschätzung der potentiellen Erntemenge (viel, mittel, wenig) sowie ein Maximum pro beerntbarer Kultur pro Portion. Einige Tage vor der Ernte wird für die Mitglieder in der Mitglieder-App ein Bestellformular freigeschaltet, in welchem sie eine beliebige Menge der einzelnen Gemüse ordern können. Dieses Ausgabeverfahren ermöglicht Mitgliedern zwar die größtmögliche Flexibilität jedoch besteht die Gefahr, dass Überproduktion nicht ausgegeben werden kann oder, dass bestellte Mengen nicht geliefert werden können.
- 2) Beim "prognosebasierten Verfahren" werden dem Gärtner aufgrund der Präferenzen der einzelnen Mitglieder "Portionen" für die einzelnen Gemüse vorgeschlagen, die den Bedürfnissen der Mitglieder entsprechen. Wenn diese Mengen nicht erntbar sind, kann der Gärtner auch eine vorhandene geerntete Menge durch den Algorithmus an die Mitglieder aufteilen lassen, um eine möglichst präferenzorientierte Ausgabe zu ermöglichen. Diese Ausgabeart hat den Vorteil, dass weder Überschüsse noch Mindermengen ein Problem für die Ausgabe sind, da alles verteilt wird. Es zieht jedoch die allgemeinen Präferenzen der Mitglieder in betracht und nicht den Bedarf zum Zeitpunkt der Ausgabe.
- 3) Bei der "klassischen" Ausgabestrategie können verschiedene Größen an Anteilen definiert und frei konfiguriert werden.
- 4) Bei einer "Lieferung" ist es nur möglich eine gewisse Menge verschiedener Produkte an einen Bezugspartner zu liefern (dies ist u.U. relevant, wenn eine SoLawi noch Restaurants oder Läden beliefert).

Anbauplanung

Die Anbauplanung ist der Meilenstein, der nur in seinen Grundzügen implementiert wurde. Die Solawi-Community diskutiert seit einiger Zeit über die Entwicklung einer Anbauplanung, die den Anforderungen von SoLawis gerecht wird. Unser System liefert derzeit als "Anbauplanung" vorgeschlagene Mengen und Anbauzeitpunkt für den Gärtner, ohne jedoch die Situation des Hofes in seiner Gänze zu berücksichtigen. In der Zwischenzeit hat sich ein Entwickler gefunden, der die Ausgabeplanung weiterführen will.

Weiterentwicklung

Für die Weiterentwicklung gibt es ein gut gefülltes Backlog. Derzeit ist die Applikation auf einem Stand, der zeigt, dass das Verfahren funktioniert und eingesetzt werden kann. Für den täglichen Gebrauch müssen jedoch noch einige Features verbessert und neu implementiert werden. Die derzeit definierten Features, die in den folgenden Monaten implementiert werden sollen, sind jedoch für ein 1.0 Release vollständig.

Zudem soll in einem nächsten Schritt in einer dritten individualisierten Ausgabestrategie des "prognosebasierten Verfahrens" dem Modell der "freien Entnahme" nachgeschaltet werden, um auch die Ausgabe von Überproduktion sicherzustellen.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Anwender der Software "Growing Futures" sind Solidarische Landwirtschaften (SoLawis), insbesondere die Gärtner, Personen in der Verwaltung und Mitglieder. In einer SoLawi finanziert eine Gruppe von Personen einen Hof in seiner Gänze und erhält im Gegenzug alle landwirtschaftlichen Produkte, die auf diesem Hof erwirtschaftet werden. Mitglieder verpflichten sich für einen gewissen Zeitraum (i.d.R. eine Saison) die SoLawi zu unterstützen.

Derzeit werden durch die Software neben der Kernfunktionalität der individuellen Anteilsausgabe auch verschiedene SoLawi-spezifische Prozesse abgebildet. Diese unterstützen die Gärtner*innen in ihrer täglichen Arbeit, insbesondere im Bereich der Ernte, Planung und Verteilung der Produkte.

Für SoLawis führt die Digitalisierung der Ernteaufnahmeprozesse sowie die Möglichkeit des Live-Feedbacks durch Mitglieder zu einer höheren internen wie externen Transparenz. Innerhalb von Solawis sind Gärtner oft verunsichert darüber, ob sie die Anteile richtig bestücken, ob die Mitglieder zufrieden sind mit den ausgegebenen Anteilen. Zudem ist es schwierig aufgrund der starken Unterschiede in Präferenzen aufgrund von Stichproben eine allgemeingültige Weiterentwicklung des Anbauplanes zu definieren. Die Auswertung der jährlichen Fragebogen führt zu widersprüchlichen Ergebnissen, die in traditionellen SoLawis nicht umgesetzt werden können (z.B. wollen 50% der Mitglieder mehr Grünkohl, die anderen 50% aber weniger).

Die Möglichkeit die Software als Open-Source Lösung anzubieten gibt der SoLawi-Community, die mit Juntagrigo und OpenOlitor schon zwei aktive Open-Source Projekte zur Verwaltung von Solawis hat, die Möglichkeit auch zu "Growing Futures" beizutragen.

Ideen zur Weiterentwicklung der Lösung gibt es derzeit mehrere. Konkrete Planungen beschränken sich derzeit darauf den Prototypen zu einem funktionsfähigen Produkt auszubauen, welches direkt verwendet werden kann, ohne dass ein großer Aufwand für Anpassung betrieben werden muss. Eher langfristig wäre es möglich aus Growing Futures ein SoLawi-Verwaltungstool zu machen, da die derzeitigen Tools häufig nicht den Anforderungen der SoLawis entsprechen.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Es gab keine Punkte für die keine Lösung gefunden wurden, nur bestimmte Bereiche, die sich als komplexer dargestellt haben als erwartet. Hier mussten wir aber nur lange genug analysieren und iterativ weiterarbeiten bis sich zumindest eine heuristische Lösung ergeben hat.

Der Verteilalgorithmus ist ein komplexes Optimierungsproblem mit vielen Unbekannten, im Laufe der Entwicklung wurden hier mehrfach Lösungsansätze entwickelt, die zwar mathematisch sinnvoll erschienen, jedoch unter realen Bedingungen nicht verargumentierbare Ergebnisse geliefert haben. Dies war hauptsächlich auf fehlende constraints bei den Produkten zurückzuführen, da diese jedoch von den Gärtnern gepflegt werden müssten, haben wir uns dagegen entschieden dies zu verwenden, da es die Anwendungsfreundlichkeit massiv reduziert hätte.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Die einzelnen Bausteine des Projektes können mit einer Übersicht über die Funktionalität auf <https://gitlab.com/growing-futures> eingesehen werden.

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Der Aufwand der Teilaufgaben hat sich weitestgehend wie antizipiert verhalten, wobei die Gärtner-App durch das hinzukommen weiterer Prozesse, die für eine Nutzung essentiell nötig sind, etwas mehr Zeit in Anspruch genommen hat als anfangs angenommen. Nichtsdestotrotz sind wir weitestgehend im Plan geblieben und sind mit dem Ergebnis zufrieden.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Durch die Vorstellung unseres Tools standen wir u.A. mit mehreren Gärtnern von SoLawis und auch mit dem gemuese-anbauplaner.de sowie anderen NGOs im Gespräch. Gerade durch den gemuese-anbauplaner wurden wir in unserer Entscheidung die Anbauplanung den Gärtnern oder spezialisierten Tools zu überlassen bestärkt, da das Domänenwissen hier enorm ist und sich sehr stark von Betrieb zu Betrieb unterscheidet.

Durch das ausführliche Testen von OpenOlitor und Juntagrigo wurden wir zudem von vielen Personen aus dem SoLawi-Umfeld animiert unser Tool technisch so aufzusetzen, dass es mittelfristig auch zu einem leichtgewichtigen Mitgliederverwaltungstool ausgebaut werden kann, wie oben im Weiterentwicklungspotential beschrieben.

Richtlinie zum „Software-Sprint“

SLI – System.Luftdaten.info

DLR Projektträger Berlin

Eingegangen am:

Schlussbericht

14. Juli 2020

Zuwendungsempfänger:

Eingangsnr.: 3619/B6

Systems.Luftdaten.info

Rajko Zschiegner (DevOps), David Lackovic (Developer) und Lukas Mocek (International Community & Partnership Development)

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S15 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)?

Motivation:

Wir - das Kernteam bestehend aus Rajko Zschiegner (DevOps), David Lackovic (Developer) und Lukas Mocek (International Community & Partnership Development) sind seit Jahren mit dem Projekt zur Luftqualitätsmessung und jetzt mit der daraus resultierenden globalen Plattform für Umweltdaten verbunden. Wir haben durch unsere gemeinsame Arbeit viele auch internationale Beziehungen und Freundschaften geknüpft. Daraus ist ein Netzwerk an Freiwilligen und eine internationale Community entstanden, die dem Projekt die Skalierung und nötige Anpassung an Sprache, Kultur und lokale Gegebenheiten erst ermöglicht. Wir sind persönlich an den Aufgaben gewachsen und haben uns inhaltlich gegenseitig in diesem für uns alle interessanten und wichtigen Thema unterstützt und vorangebracht. Der gemeinsame Austausch und das ständige weiterbilden und die Erfahrung was mit offenen Daten an anderen Stellen und von anderen Gruppen alles umgesetzt werden kann ist der Antrieb der durch diese Förderung umgesetzten Bemühungen diese besser miteinander zu verbinden und einen besseren Austausch zu ermöglichen.

Problem:

In den letzten drei Jahren hatten wir verstärkten Austausch mit den lokalen Projekt-Gruppen in anderen Ländern und haben viele interessante und gute Ansätze und Lösungen die, durch die von unserer Plattform bereitgestellten offenen Daten, ermöglicht wurden. Nur brauchte es eine erste Sichtbarkeit für diese Communities und Projekte und vor allem eine Technische Grundlage auf die alle zugreifen und aufbauen könnten. Es brauchte eine Infrastruktur, die Daten annehmen, speichern, bereitstellen und grafisch darstellen kann. Hierzu musste diese skalierbar und länderspezifisch

anpassbar sein. Wenn dieses Grundgerüst stand mussten weitere Anwendungsfelder abgedeckt werden: Integration weiterer Sensor-Typen und bessere Formen der Visualisierung.

Vorgehensweise zur Problemlösung - Anhand der Milestones:

Milestones 1: Recherche bzgl. best practise im Bereich der mobilen Datenerhebung

Hier konnten wir Unterstützung und eine erste Einschätzung guter Ansätze von der Bundesanstalt für Materialforschung und -prüfung <https://www.bam.de> erhalten. BAM ist hierbei führend auf dem Gebiet der mobilen Datenerhebung und hat zahlreiche Publikationen zu ihren Umsetzungen.

Milestones 2: Konzepterstellung der benötigten API-Anpassungen

Bezüglich der API konnten wir uns mit dem Joint Research Center in Italien - JRC <https://ec.europa.eu/jrc/en/about/jrc-site/ispra> und dem Open Geo Consortium - OGC <http://www.opengeospatial.org> austauschen und haben so von existierenden Datenstandards und neuen API Formaten erfahren.

Das JRC ist Europas leitende Forschungseinrichtung und arbeitet für die Europäischen Kommission.

Das OGC Entwicklung die raumbezogener Informationsverarbeitung weiter und ist auf Geodaten spezialisiert. Hierzu wurden uns weitere Kontakte zu europäischen Geoportalen und EO - Earth Observation Portalen und Initiativen vermittelt.

Milestones 3: Entwicklung eines Prototypen mit Anpassung der aktuellen API

Während der Projektlaufzeit haben wir herausgefunden, dass es globale Initiativen wie z.b. ein Projektteam rund um die EarthDayChallenge aus Washington D.C. in den USA gibt welche sich mit einem ähnlichen Ansatz beschäftigen und haben über gemeinsame Kontakte beim JRC und OGC einen Austausch gestartet. Hierzu haben wir zwei Erfahrungen gesammelt.

Erstens: In der grundsätzlichen API Anpassung sind unsere Ansätze gleich, jedoch möchte die EDC Gruppe mit ihrer Version viel mehr Einsatzfelder abdecken. Daher ist ein komplettes Zusammenbringen der Ansätze nicht möglich. Jedoch konnte durch den regelmässigen Austausch die Entwicklungszeit bei beiden Gruppen verkürzt werden.

Zweitens: Der gemeinsame Austausch hatte dazu geführt, dass unser komplette Datensatz in die neue EDC platform mit aufgenommen wird und somit eine bessere globale Sichtbarkeit erfährt.

Milestones 4: Implementierung der Erfassung von mobilen Sensoren im Registrierungsportal

Hierbei sind wir kurz vor einer ersten Pilotphase in der mobile Sensoren an Stadtbussen angebracht werden. Der erste Testlauf wird voraussichtlich in Stuttgart starten und ein weiterer kann in Nürnberg durchgeführt werden.

Milestones 5: Dokumentation der neuen API

Bei der Dokumentation arbeiten wir mit Experten vom JRC, OGC und von der INSPIRE platform zusammen.

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?

Kurzfristig ist die erste Zielgruppe klein und sehr speziell, da es sich um Experten handelt, die daran arbeiten existierende Datensätze in bekannte Datenportal einzubinden und zugänglich zu machen. Ist diese Arbeit jedoch abgeschlossen, können die Datensätze in standardisierten Formaten unter Berücksichtigung von Interoperabilitäts-Standards bei mehr Projekten berücksichtigt und eingebunden werden. Das hilft der Verbreitung und weitreichenderen Nutzung der von Bürgern generierten offenen Daten.

Hinzu kommt, dass die neuen Datenformate und API's mehr Anwendungsfelder abdecken und eine Anreicherung der Metadaten ermöglichen. Das war für uns ein entscheidender Punkt.

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?

Erkenntnisse: Community Aktivitäten brauchen Zeit:

Da wir das Sensor Netzwerk bereits seit 5 Jahren betreiben ist uns dieser Punkt eigentlich bekannt. Jede Anpassung braucht ihre Zeit und speziell in einem Community Projekt müssen breite Zeiträume eingeplant werden bis das Gesamtverständnis sich etabliert und eine ausreichende Akzeptanz gewährleistet ist.

Die Corona Krise hat hier eine stärkere Anpassung an alle bisherigen Vorgehensschritte erfordert. Wir hatten hierzu mehrere speziell dafür angesetzte Diskussionsrunden angesetzt um sicherzustellen, dass wir mit unseren Anforderungen und umgesetzten Änderungen uns von der Lebenswirklichkeit der internationalen Community nicht entfernen.

Hier haben sich zwei Ansätze als besonders Hilfreich erwiesen:

1. Wir haben in einem ersten Schritt, die Regelmäßigkeit der Kontaktmöglichkeiten mit dem Kernteam verstärkt kommuniziert. Damit haben wir eine bessere Erreichbarkeit und Sichtbarkeit signalisiert. Diese war bis dahin im monatlichen Rhythmus in Form eines 'Monthly Community Calls'. Das war zwar neu, jedoch durch die vielen kurzfristigen Anfragen bei weitem nicht genug. Somit mussten wir uns anpassen wie folgt beschrieben.

Mit der Zeit haben wir erkannt, dass eine Erhöhung der Häufigkeit der regelmässigen 'Community Calls' auf einen wöchentlichen Takt es mehr Teilnehmern die Möglichkeit geboten hat sich zu beteiligen und folge dessen in die Entwicklungen einzubringen. Und, dass es die einzelnen Calls entschleunigt, da sich die zu diskutierenden Themen nicht auf einen Termin im Monat angestaut haben, sondern jede Woche kleinere Schritte besprochen, abgearbeitet, integriert und bewertet werden konnten.

Somit haben wir die im Januar 2020 festen 12 Monatstermine für das gesamte Jahr, in wöchentliche calls umfunktioniert. Jeder Mittwoch um die gleiche Zeit, im gleichen Kanal können Community Mitglieder nicht nur mit dem Kernteam in Austausch treten, sondern sich untereinander über ihre Sichtweisen und Ergebnisse austauschen.

2. Ein weiterer erfolgreicher Schritt war das Übergeben der Leitung von Teilprojekten an Community Mitglieder in anderen Ländern wie z.B.: Koordination einer europaweiten NO2 Campagne mit 10 teilnehmenden Städten in 10 europäischen Ländern. Hierbei wurde eine für die Community neue Messmethode mit Passivsammlern angewandt. Diese ist bei den offiziellen Messstellen etabliert. Hier hat sich über einen Kontakt die Möglichkeit zum Wissensaustausch ergeben und diesen konnten wir durch Unterstützung einer ersten Community Gruppe aus Bulgarien in eine eignen Kampagne

einbringen. Das hat die weiteren Kampagnenteilnehmer ermutigt, da sie auf kurzem Weg auf die Expertise zugreifen konnten und mit Leuten zu tun hatten, die sie kennen.

Co-creation Prozess:

Jegliche Veränderungen und Anpassungen müssen mit der Community rechtzeitig kommuniziert werden. Beim Einbringen der Community in den Erstellungsprozess ist es besonders wichtig diese Beteiligung so früh wie möglich zu ermöglichen. Hierzu eignen sich Country Leads und Local Heroes, die seit Jahren mit dem Projekt verbunden sind als geeignete Multiplikatoren. Schliesslich gilt es die drei folgenden Schritte anzuwenden:

- Übersetzung der Nachricht in die jeweilige Landessprache.
- Anpassung an den kulturellen Kontext. - Jedes Land ist an einem anderen Punkt und nimmt die unterschiedlichen Kategorien als große bzw. kleine Änderung auf.
- Lokale Anpassung. - Jede Region und jede Stadt ist anders. Eine lokale Anpassung ist ein wichtiger Schlüssel um die Akzeptanz bei den Teilnehmern nicht nur zu erhöhen, sondern erst zu ermöglichen. Hier ist das Einbringen von spezifischem Wissen wie z.B. was ist wann passiert und wie ist es mit aktuellen Ereignissen wie z.B. Institutionelle Kampagnen verbunden, da sich Bürger informieren und den Zusammenhang verstehen wollen.

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?

Wissenschaftler die mit Umweltdaten arbeiten, Daten-Journalisten, Luftqualitätsexperten, interessierte Bürger sind die uns bekannten Nutzer.

Der Open Source Ansatz lässt sich an der schnellen Umsetzung der Internationalisierung gut verdeutlichen. Wir haben die Webseite und den Dienst in kürzester Zeit mit der Unterstützung in neun Sprachen anbieten können. Aktuell arbeiten wir an der Einbringung von Windstößen in der globalen Visualisierung. Diese ist uns von der Community aufgezeigt worden. Jetzt gilt es das auch performant und skalierbare einzubringen.

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

Ja. Das ständige Lernen und den Austausch mit den Institutionen und den Wissenschaftlern ermöglicht uns einen neuen Blick darauf was noch alles möglich ist, da wir die meiste Zeit die Anfragen von Bürgern und ihren lokalen Bedürfnissen beantworten und bearbeiten ist es interessant zu sehen, was ein globales Sensor Netzwerk gemeinsam umsetzen kann.

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?

Ja, wir haben zu Beginn die Auswahl des neuen Frameworks mehrmals geändert, da es zu diesem Zeitpunkt viele Neuentwicklungen gegeben hat, die sich als besser geeignet herausgestellt haben. Wir sind mit der finalen Version weiterhin sehr zufrieden.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

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

Der am meisten genutzte Dienst ist die Live Map unter: <https://maps.sensor.community>

Quellcode: <https://github.com/opendata-stuttgart/>

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 wir die Plattform in diesem Jahr verstärkt präsentieren wollten hatten wir viele Veranstaltungen, Messen, Wissenschaftliche Konferenzen und Vorträge eingeplant und vorbereitet. Mit der Corona-Krise, den damit verbundenen Reisebeschränkungen und der Absage von allen größeren Veranstaltungen ist viel Zeit der Vorbereitung nicht zum Tragen gekommen. Das hat unser Gesamtkonzept der geplanten Zusammenarbeit für das gesamte Jahr beeinflusst. Wir mussten uns neu ausrichten und auch für alle Meetups / Treffen / Workshops der lokalen Gruppen online Treffen organisieren.

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 zuvor Beschrieben gab und gibt es weiterhin Anpassungen an unser Vorgehen durch die Zusammenarbeit und dem regelmässigen Austausch mit JRC, OGC, EDC.

Lukas Moak

MliftW – Musiklizenzierungs- Infrastruktur für das Web

Schlussbericht

Zuwendungsempfänger:

Alexander Blum, Thomas Mielke, Christoph Scheid, Udo Spallek und Sarah Stoffels

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS19S16 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Wir wollten eine öffentlich zugängliche API (Application Programming Interface = Softwareschnittstelle) entwickeln, mit der Lizenznehmer_innen (also z.B. Musikplattformen) auf zuverlässige, unbürokratische, kleinteilige und sichere Weise einen niederschweligen Zugang zu Lizenzinformationen musikalischer Werke bekommen, wobei diese bei Bedarf auch gleich für verschiedene Nutzungen (z.B. Streaming) lizenziert werden können.

Die Motivation entstand dadurch, dass es für Musik-Lizenzgeber wie z.B. Verwertungsgesellschaften keine passenden Lösungen am Markt gibt – erst recht keine Open-Source lizenzierte. Das Beispiel des Internets, dessen Infrastruktur zum größten Teil aus Open-Source-Software besteht, zeigt die Wichtigkeit der Verfügbarkeit freier Software für den Aufbau eines funktionierenden Lizenzierungssystems. Der momentane Zustand der Musiklizenzierung ist ein Flickenteppich an nationalen monopolistischen Verwertungsgesellschaften, die bilaterale Vereinbarungen treffen und letztlich doch Pauschallizenzierungen anbieten, weil sie jeweils mit der Komplexität überfordert sind und ein offeneres System ihre Monopolisten-Stellung gefährden würde. Es gibt Initiativen der Verwertungsgesellschaften wie DDEX, die technisch nicht besonders ambitioniert sind, und es gibt eher unabhängige Initiativen wie die Open Music Initiative, die momentan zu sehr mit sich selbst beschäftigt sind, um aktiv hilfreich sein zu können. Nichtsdestotrotz konnte unser Projekt auf den Vorarbeiten der Open Music Initiative aufbauen, deren Use-Case-Katalog wir intensiv studierten, und auch die DDEX-Standards konnten uns ein wenig Orientierung geben.

Unsere Meilensteine lassen sich grob in *Werke-API*, *Nutzungserfassung-API* und *Lizenzierungs-Kern* unterteilen.

Nicht zuletzt bildete die Kontroverse um die EU-Urheberrechtsrichtlinie in 2018/19 den Antrieb für uns, sich den komplexen Herausforderungen dieses Vorhabens zu stellen.

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

Wer ist die Zielgruppe für Deine Lösung?

Primär ist unser Fokus auf Verwertungsgesellschaften und anderen Rechteverwertungsunternehmen (Verlage) im Musikbereich gerichtet. Tatsächlich sind die beteiligten Entwickler_innen seit Jahren ehrenamtlich beim Aufbau der neuen Musik-Verwertungsgesellschaft Cultural Commons Collecting Society SCE und konnten mit der Entwicklung des Lizenzierungs-Prototypen auf dem bestehenden freien Repertoire-Verwaltungssystem der C3S SCE aufsetzen.

Wie profitiert sie von den Ergebnissen Deines Projekts?

Der Nutzen einer freien Softwareplattform mit offenen Schnittstellen für Verwertungsgesellschaften und andere Rechteverwertungsunternehmen ist offensichtlich. Wir erwarten nicht, dass etablierte Verwertungsgesellschaften in absehbarer Zeit von ihren erprobten proprietären Eigenentwicklungen ablassen werden. Die EU-Richtlinie 2014/26 über die kollektive Wahrnehmung von Urheber- und verwandten Schutzrechten und die Vergabe von Mehrgebietslizenzen für Rechte an Musikwerken für die Online-Nutzung im Binnenmarkt bricht mit den bisherigen verkrusteten nationalen Strukturen und erleichtert es neuen Akteuren, wie der C3S, im Feld der der kollektiven Rechtewahrnehmung Fuß zu fassen. Für diese ist unsere freie Softwareplattform und deren Implementierung von offenen Schnittstellen eine erhebliche Erleichterung bei der Aufnahme des Betriebs.

Neben Organisationen zur kollektiven Rechtewahrnehmung sind sekundär Musikrechteinhaber, die Mitglieder von Verwertungsgesellschaften sind, wie etwa die Genoss_innen der C3S, Nutznießer – und auch, wenn auch nur vermittelt, Lizenznehmer_innen und schließlich Musikkonsument_innen, die sich sicher sein können, dass ihr Geld auch genau die Künstler_innen erreicht, deren Werke sie nutzen.

Welche Bezüge gibt es zu den Themenfeldern und Zielen des Software Sprints?

Die Bezüge zum Themenfeld *Infrastruktur* und zum Schwerpunkt *Commit – System erneuern* liegen nach dem bisher gesagten auf der Hand.

Ausführliche Darstellung der Ergebnisse

1. Werke-API (Fokus: Urheberrecht für Musik)

1.1) Entwicklung einer maschinenlesbaren API-Spezifikation zur Abfrage durch potenzielle Lizenzzahlungsempfängern

Die Entwicklung der *REST-API* erfolgte auf Basis der *OpenAPI*-Spezifikation Version 3.0.3. Es wurden umfangreiche Use-Case-Sammlungen und User-Stories einbezogen sowie unsere eigene Expertise aus der Arbeit an der C3S SCE. Für die Abfragen von Lizenzinformationen ist zunächst eine Identifikation des jeweiligen Werkes notwendig. Dazu wurde eine multifaktorielle Abfragemöglichkeit geschaffen. So können Werke über verschiedenste Identifikationsmerkmale wie ISRC, ISWC, Harry Fox Agency Code, Künstler/Titel und Audio-Fingerprint identifiziert werden, wobei

in der Ergebnismeldung eine Score zurückgeliefert wird, die die Gewissheit repräsentieren soll, mit der das gesuchte Werk tatsächlich gefunden wurde.

Zu den Gegenseitigkeitsverträgen unter Verwertungsgesellschaften sind wir im Rahmen dieses Meilensteins aus Kapazitätsgründen nicht gekommen.

1.2) Umsetzung der maschinenlesbaren API zur Abfrage durch potenzielle Lizenzzahlungsempfängern

Die API wurde auf Basis des Web-Service-Frameworks *Pyramid* und des ERP und Datenbankabstraktions-Frameworks *Tryton* umgesetzt und setzte an der bestehenden C3S-Werkeverwaltung an.

Es wurde zunächst ein System zur flexiblen Verwaltung von Identifikatoren geschaffen. Es besteht damit die Möglichkeit beliebig viele ‚Identifikationsräume‘ anzulegen, z.B. für jeden Musikverlag einen, wobei die Identifikatoren die verlagseigene Katalognummer repräsentieren. Außerdem musste eine Struktur geschaffen werden, um Rechteinhaber_innen (Urheber und deren Erben, Verlage, etc.) in historisierter Weise zu repräsentieren.

Kaskadierende Abfragen an andere Verwertungsgesellschaften und Dienstanbieter konnten im Rahmen des Projekts aus Kapazitätsgründen und wegen mangelnder Resonanz potenzieller Partner nicht realisiert werden.

1.3) Umsetzung einer maschinenlesbaren API zur Validierung von Audio-Fingerprints für andere Dienstanbieter

Das von uns favorisierte Audio-Fingerprint-System EchoPrint wurde mit der Integration des Service als *Docker-Container* organisch in das Lizenzierungs-Softwareprojekt eingebunden und es wurde mit automatisierten Tests abgesichert und in unser DevOps-Environment *Jenkins* einbezogen.

1.4) Erfüllung der technischen Voraussetzungen für DSGVO Compliance

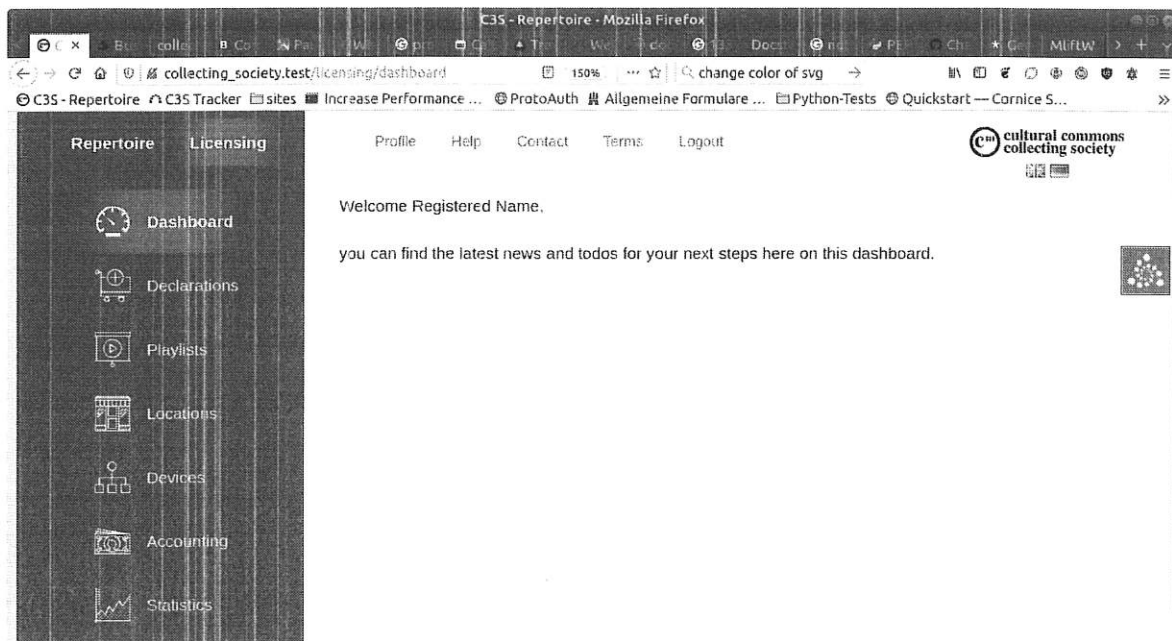
2. Nutzungserfassung-API

2.1) Maschinenlesbare API zur Nutzungserfassung

- via Audio-Fingerprints
- via Künstler/Titel/Stil

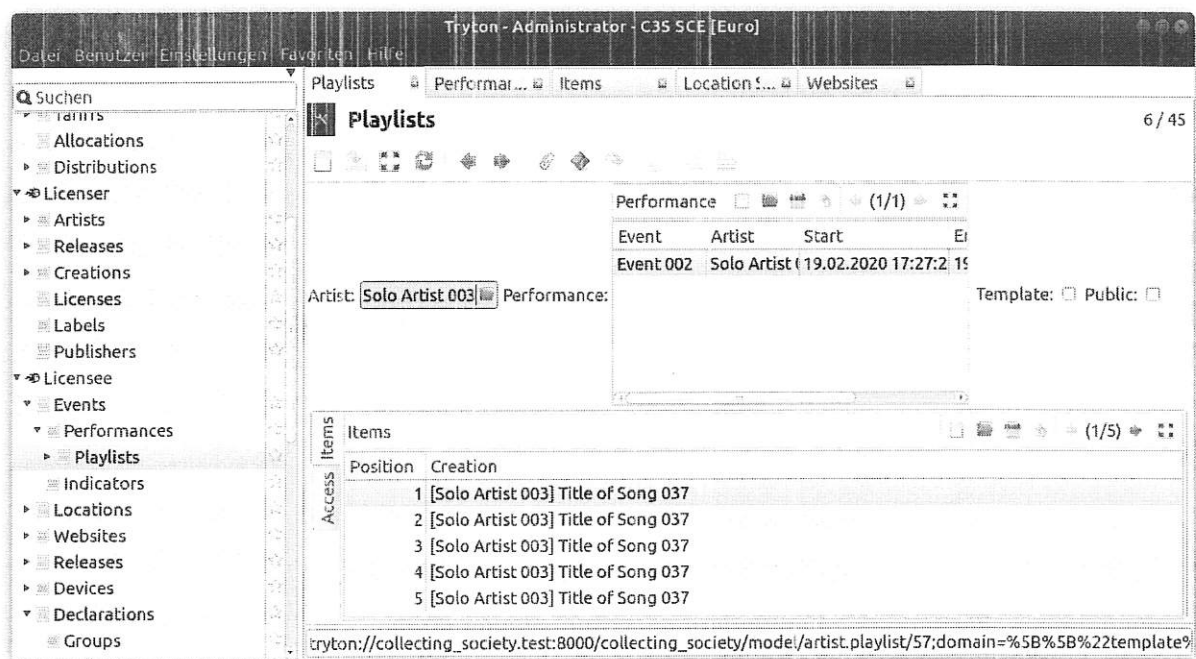
2.2) Menschenlesbare GUI zur Nutzungserfassung

Webbasierte Formulare, z.B. zur Erfassung von Spielfolgebögen, konnten aus Kapazitätsgründen wenige realisiert werden.



Dashboard des Lizenznutzungs-Bereichs

Durch die Nutzung des *Tryton-Frameworks* haben wir jedoch die Möglichkeit auf alle Datenstrukturen über die GUI des *Tryton-Clients* zuzugreifen. Diese Ansicht wird im Betrieb den Mitarbeiter_innen der nutzenden Organisationen vorbehalten sein:



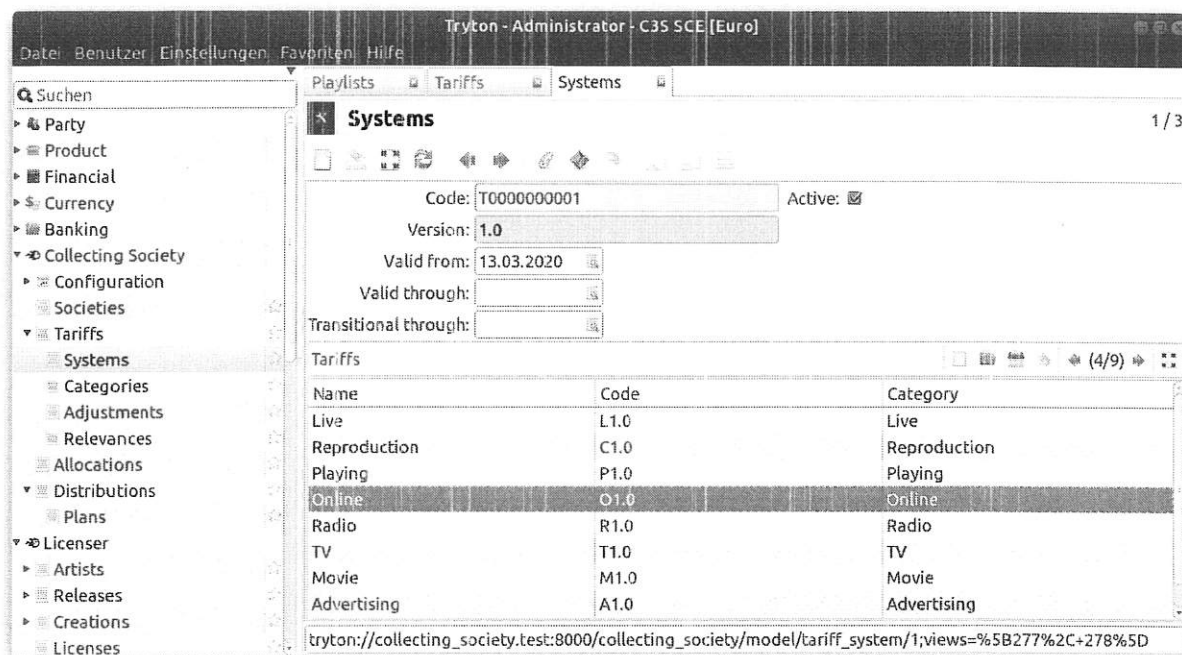
Tryton-Client: Ansicht zur Bearbeitung von Spielfolgelisten bei Veranstaltungen

3. Lizenzierungs-Kern

3.1) Abrechnungsmodul

Abrechnungen erfolgen auf zweierlei Weise: Via Tarifplan für die Lizenznehmer_innen und via Verteilungsplan für die Lizenzgeber_innen. Dabei definiert nach der Anmeldung (Declaration) einer Nutzung der entsprechenden Tarif (z.B. Live-Event oder Online-Podcast) gemäß den Tarifindikatoren

(z.B. Anzahl der Besucher_innen einer Veranstaltung, Nachlässe, etc.) und entsprechend der aufgeführten Werke (Playlist) die in Rechnung gestellte Geldsumme. Nach erfolgtem Inkasso wird der Betrag entsprechend der Playlist und dem Verteilungsplan an die Beteiligten Künstler_innen verteilt.



Tryton-Client: Tarifsystem-Ansicht

Des weiteren haben wir eine buchhaltungstechnische Abbildung der Vorgänge konzipiert und uns für die Klärung offener steuerrechtlicher Fragen mit entsprechenden Fachleuten kurzgeschlossen.

Eine Berücksichtigung mehrerer Verwertungsgesellschaften bei der Verteilung war aus Kapazitätsgründen im Rahmen des Projekts nicht leistbar.

3.2) Reporting für Rechteinhaber_innen (sofern eigenes VG-Repertoire betroffen ist)

Eine Reporting für Rechteinhaber_innen war konnten wir aus Kapazitätsgründen im Rahmen des Projekts nicht umsetzen.

3.3) Verwertungsgesellschaften (bei Gegenseitigkeitsverträgen)

Eine kaskadierende Verteilung unter Einbeziehung mehrerer Verwertungsgesellschaften konnte im Rahmen des Projekts aus Kapazitätsgründen und wegen mangelnder Resonanz potenzieller Partner nicht realisiert werden.

Unser Team hat sich während der Projektdauer intensiv mit der Welt der Lizenzierung im Musikbereich auseinandergesetzt und wertvolle Erkenntnisse gewonnen, ohne die die Konzeption der offenen API kaum möglich gewesen wäre. Eine wichtige Erkenntnis war, dass eine auch noch so elaborierte Spezifikationsarbeit an der API in der Umsetzung als Software-Prototyp einen noch einmal wesentlich höheren Aufwand bezüglich der dahinter liegenden Datenmodelle impliziert. Diesen haben wir deutlich unterschätzt – aber möglicherweise ist solch eine Unterschätzung die Voraussetzung dafür, sich mir einem so kleinen Team wie dem unsrigen, an eine solche Aufgabe zu

wagen. Somit bleibt festzuhalten: Auch wenn wir nicht alle Milestones erreicht haben, war das Projekt in diesem Sinne erfolgreich.

Die Begleitung durch die OKF war motivierend und rechtfertigte den ‚zusätzlichen‘ Zeitaufwand. Gelegentliche Kontakte zu anderen Projekten waren inspirierend.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Der Nutzen für die Zielgruppe wurde auf Seite 2 bereits ausführlich geschildert. Die Open-Source-Stellung des Software-Projekts hat keine unmittelbaren Effekte, ist aber Voraussetzung für den Aufbau einer intelligenteren Infrastruktur.

Auf lange Sicht ist auch die Evolution des Systems zur Nutzung durch einzelne Urheber_innen denkbar, was tendenziell zu einer Veränderung der Rolle von Verwertungsgesellschaften weg von kollektiver Rechtswahrnehmung mehr hin zu Dienstleistungen für Rechteinhaber mit sich bringen könnte. Dies ist aber zum jetzigen Zeitpunkt reine Spekulation.

Eine andere mögliche Entwicklungsrichtung ist die Nutzbarmachung für andere als Musikwerke. Wir haben das bei der Konzeption von Anfang an Hinterkopf gehabt und versucht, dort, wo es nicht zu viel Aufwand war, die nötige Abstraktionsleistung zu bringen. So ließe sich das Audio-Fingerprinting möglicherweise gut durch ein Video- oder Bild-Fingerprintingverfahren ersetzen.

Selbstverständlich gab es Lerneffekte beim Einsatz der vielen Tools, Standards und Frameworks, die wir für das Projekt nutzen. Prominent dabei ist sicher die OpenAPI-Spezifikation anzuführen. Auch der Umgang mit der Docker-Containerisierungslösung oder die Anwendung automatisierter Dokumentation mit *Sphinx* ist hier nennenswert. Neben dem Technischen ist aber auch Erfahrung bei der Selbstorganisation kleiner Teams gesammelt worden.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Es war ursprünglich geplant die Spezifikation der API zusammen oder zumindest im Kontakt mit Partnern im Bereich der Musikkonzessionierung durchzuführen. Dies war nicht erfolgreich. Wir haben uns deshalb auf die Arbeiten beschränkt, die wir aus uns verfügbaren Informationsquellen sinnvoll definieren konnten. Siehe Abschnitt „Ausführliche Darstellung der Ergebnisse“.

Außerdem konnten wir nicht in dem Maße auf Spezialisten in unserem Team zurückgreifen, wie wir es ursprünglich geplant hatten. Einige für diese Spezialisten bestimmte Arbeiten wurden von weniger mit der Materie bekannten Teammitgliedern durchgeführt

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

Projekthomepage:

<http://MliftW.c3s.cc>

GitHub:

https://github.com/C3S/collecting_society_docker

https://github.com/C3S/collecting_society

https://github.com/C3S/collecting_society_web

https://github.com/C3S/portal_web

https://github.com/C3S/collecting_society_worker

Sphinx-Dokumentation:

<https://files.c3s.cc/csdoku/html/>

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Wir haben zu den Milestones diametral elf Arbeitspakete für den Projektzeitraum definiert. Aus dieser Tabelle geht hervor, inwiefern wir die Arbeitsplanung einhalten konnten:

Arbeitspaket	Geplante Stunden	Geleistete Stunden
Spezifikation	240	287
Systemadministration	40	10
Datenbankmodellierung	230	433
Web/API	200	147
Fingerpr	30	29
Doku	30	28
Tests	90	48
Recht	15	6
Text	10	0
Orga	25	92
Netzwerk	40	21
Summe	950	1101

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

Es gab keine Entwicklungen anderer Personen oder Institutionen, die nennenswerten Einfluss auf unsere Arbeiten oder Zielsetzungen hatten.