



hoaxly

Fachlicher Zwischenbericht

prj2035\_Call11\_ZB

**a:o** acolono GmbH  
**meet** Fillgradergasse 12-14/1A  
1060 Wien, Österreich  
**talk** +43 1 3051030  
**find** <https://acolono.com>  
**write** [hello@acolono.com](mailto:hello@acolono.com)



# Fachlicher Zwischenbericht

Aktivitäten & Fortschrittsbericht auf Basis der Arbeitspakete im excel/Förderrate 1.

## AP 1

### **Projektmanagement einrichten (Jira), Team definieren, Userstories und Backlog schreiben, Regelmässig Sprint-Planung, Regelmässig Retrospektiven**

Die Userstories wurden sehr gewissenhaft erstellt. Durch das genaue Durchdenken aller Features sind nicht wenige Punkte aufgetaucht, die für das Konzept und die technische Umsetzung relevant waren.

Als Backlog wurden Tasks im Jira angelegt, mit jeweils einem Verweis auf die relevante Userstory.

Die Arbeiten wurden in monatlichen Sprints organisiert. Regelmässig gab es ein Retrospektiv-Meetings, wo die Arbeit präsentiert wurde und Probleme sowie Fortschritte diskutiert wurden. Während der Sommermonate wurden diese Retrospektiv-Meetings aufgrund der wenigen Fortschritte jedoch nur spärlich abgehalten.

## AP 2

### **Erstellung technisches Konzept, Erstellung inhaltliches Konzept, Erstellung grundlegendes Design**

Neben dem technischen Konzept wurde konzeptionell sehr viele Überlegungen angestellt. Macht ein Tool wie hoaxly überhaupt weiterhin Sinn? Es schießen ja aktuell sehr viele Tools und Projekte rund um Fake-News aus dem Boden. Wir konnten diese Frage für uns jedoch mit einem klaren Ja beantworten. Oberstes Ziel muss sein mit hoaxly die Medienkompetenz zu stärken, hier sollten die hoaxly Anwendungen also vor allem eine pädagogische Rolle spielen. Außerdem sehen wir auch einen steigenden Bedarf bei der Zielgruppe der JournalistInnen, vor allem an einer Meta-Suchmaschine wie hoaxly es ist.

Eine Liste von Richtigstellungsplattformen, mit u.a. Analyse der Datenstruktur, Auszeichnungssprache (zB. ClaimReview), verwendetes CMS, Bewertungssystematik und Verpflichtung zu einem "Code of Principles" ist hier veröffentlicht:

[https://drive.google.com/open?id=1-jm8Rs-O\\_I9-dgBqB0Xz05Aqv2T4BI6bi9g-aR1Zzh4](https://drive.google.com/open?id=1-jm8Rs-O_I9-dgBqB0Xz05Aqv2T4BI6bi9g-aR1Zzh4)

Diese Liste ist für uns Ausgangspunkt dafür zu entscheiden, welche Daten welcher Plattformen in die hoaxly Datenbank aufgenommen werden bzw. welche Quellen von uns für diesen Zweck genauer überprüft werden und wird laufend erweitert.



## AP 3 + AP 9

### Datenbank erstellen und aufsetzen, API-Dokumentation erstellen Importer und Scraper für Datenquellen

Basis unserer technischen Lösung ist Elasticsearch und Kibana. Das Scraping erfolgt über Scrapy (<https://scrapy.org/>) und Portia. Portia (<https://scrapinghub.com/portia/>) ermöglicht die Erstellung von Scraping Templates ohne Programmieren zu müssen.

Aktuell importieren wir erfolgreich HTML, RSS-Feeds und strukturiertes Markup wie schema.org. Der für Richtigstellungen vorhandene ClaimReview schema.org Standard wird leider aktuell nur von sehr wenigen Seiten unterstützt, vor allem nicht von europäischen Plattformen. Das würde den Import der Daten extrem erleichtern.

Die Optimierung der Scraper und der für die Anwendungen nötigen Objektstruktur in Elasticsearch hat uns viel Mühe bereitet (bzw. tut es noch immer). Die Importer geben sehr viele Daten zurück, vieles davon ist nicht brauchbar und muss gefiltert bzw. weiterverarbeitet werden. Unsere bis zu diesem Projekt nicht vorhandenen Kenntnisse in der Programmiersprache Python haben es uns auch nicht leicht gemacht.

Berechtigungslayer für die Datenbank-Abfragen: Für interne Tests können wir bereits über eine allgemeine API auf die Datenbank zugreifen, es ist jedoch notwendig, auch externen Personen Zugriff auf die Daten zu geben. Diese dürfen jedoch nur Lesezugriff haben und sollten nur ausgewählte Endpoints zur Verfügung haben. Bis auf diesen Punkt und die API Dokumentation ist AP 3 jedoch abgeschlossen.

Bei den Importern (AP 9) ist das folgende Stichwort aktuell zentral: "Friendly scraping". Für da Entwickeln haben wir immer nur einzelne Artikel importiert, da wir sonst riskieren von den Plattformen gesperrt zu werden (bzw. unsere IP). Am Server gilt es hier entsprechende Vorkehrungen zu treffen bzw. zumindest die Best Practices umzusetzen: Robots.txt Einstellungen der Plattformen überprüfen und akzeptieren, nur so viele Zugriffe wie unbedingt notwendig und Kontaktinformationen mitsenden. Die wichtigsten Plattformen werden wir jedoch sowieso vor den regelmässigen Scrape-Vorgängen kontaktieren und um Erlaubnis fragen müssen (Partnerschaften bieten sich hier an). Hierfür ist ein eigenes Info-Dokument in Arbeit (u.a. mit unseren Scraping-Guidelines, Kontaktinfos, unsere Ziele und vor allem die Info, dass wir nur Metainformationen in unsere Datenbank aufnehmen und entsprechend auf den Urheber verlinken).

Mehr Infos zum Thema u.a. hier:

<https://blog.scrapinghub.com/2016/08/25/how-to-crawl-the-web-politely-with-scrapy/>



## AP 4

### Webserver einrichten

Ein Webserver wurde bei hetzner.de gemietet und eingerichtet. Aktuell laufen u.a. eine ElasticSearch Instanz mit Kibana und die vorhandenen Scraping Tools (Scrapyd, Portia).

<https://kibana.hoax.ly>

<https://scrapyd.hoax.ly/>

Die Importer sind jedoch noch nicht scharf geschaltet. Siehe obigen Punkt zum Thema "friendly scraping".

Für das Deployment und Packaging verwenden wir Docker bzw. Ansible. Lokal können damit Entwicklungsumgebungen leicht aufgesetzt werden und Updates einfach auf den Server deployd werden.

## AP 5

### Browser Plugin programmieren

Ein grundlegendes Konzept wurde erstellt. Für die technische Umsetzung wurde die WebExtensions API auserkoren. Diese ermöglicht es Browser Extensions zu schreiben, die mit fast allen Browser Engines kompatibel sind. Um die Umsetzung richtig starten zu können fehlt jedoch noch der Berechtigungslayer für die Datenbank-Abfragen (API, AP3).

Ein gutes Beispielprojekt ist <https://github.com/bs-detector/bs-detector>, leider nicht mehr sehr aktiv.

## AP 6

### Chat-Bots programmieren

Das Konzept für den Chatbot ist fertig. Wie bei AP 5 fehlt für die Umsetzung jedoch noch der Berechtigungslayer für die Datenbank-Abfragen (AP3). Sobald diese Layer vorhanden ist werden wir mit Hilfe von Thomas Schranz (Mr. Chatbot himself hat sich erfreulicherweise bereit erklärt, uns hier zu unterstützen) die Bots fertig stellen. Wir rechnen mit ersten Testversionen im Jänner.



## AP 7

### Webseiten-Installationspaket programmieren

Unter dem Namen "Hoaxly CMS" ist eine Drupal Distribution in Arbeit, die folgende Funktionalitäten bietet:

- Einfache Installation
- Mehrsprachigkeit
- Vorkonfigurierter Inhaltstyp mit allen relevanten Feldern für Richtigstellungen
- Vereinfachte Übersichten und Verwaltungsseiten im Backend
- Einfache Suche nach Richtigstellungen im Frontend (wenn Zeit dafür vorhanden)
- Automatische Ausgabe der Daten im ClaimReview schema.org Standard
- Aktivierte REST API, um auf die Daten bei Bedarf einfach zugreifen zu können (zB zur Integration in die hoaxly Datenbank).
- Ansprechende Darstellung der Richtigstellungen im Frontend (wenn Zeit dafür vorhanden)
- Vordefinierte Benutzerrollen und Berechtigungen, besondere Workflows bei Bedarf.

Wie unsere Analysen der von den Plattformen eingesetzten technischen Lösungen zeigen (fast keine strukturierten Inhalte oder Einsatz von schema.org), gibt es hier großen Bedarf. Voraussichtliche Fertigstellung Version 1: Jänner 2018.

## AP 8

### Definition Qualitätskriterien für Datenquellen

Bei den Kriterien, welche Plattformen eingebunden werden sollen, haben wir uns entschieden, den Standard IFCN (<http://www.poynter.org/fact-checkers-code-of-principles/>) so gut wie möglich zu verwenden. Wie unser Analyse jedoch zeigt, gibt es eher wenige europäische Richtigstellungsplattformen, die sich diesen Richtlinien verbunden fühlen bzw. das auch kommunizieren. Vor allem im deutschsprachigen Raum ist uns mit <https://correctiv.org/echtjetzt/> nur eine einzige Plattform bekannt.

Das bedeutet, dass wir entweder Plattformen bitten müssen, dieses Audit durchzuführen. Oder wir führen selber Prüfungen durch, um transparent begründen zu können, dass eine Datenquelle unseren Anforderungen gerecht wird. Das ist ein gehöriger Mehraufwand, für den Erfolg der hoaxly Datenbank jedoch sicher von großer Bedeutung.

Für die Prüfung der Plattformen werden wir das Assessment Dokument von IFCN verwenden:

[https://drive.google.com/file/d/1oM9hZvCNk8jirwJ2madpCYY09o\\_igZQE/view?usp=sharing](https://drive.google.com/file/d/1oM9hZvCNk8jirwJ2madpCYY09o_igZQE/view?usp=sharing)



## AP 10

### **Durchführung von Tests und Feedback einholen, Programmierung der definierten Verbesserungen, Dokumentation finalisieren und überarbeiten**

User- und Usability Tests werden wir erst durchführen, wenn die Anwendungen fertig sind. Dokumentation für EntwicklerInnen ist bereits zum Teil vorhanden, für Endanwender stehen aktuell allgemeine FAQs zur Verfügung: <https://hoax.ly/faq.html>

## AP 11

### **Pressearbeit, Erstellung der finalen Webseite für das Projekt, Werbung**

Auch dieses Arbeitspaket ist erst relevant, wenn die ersten Anwendungen veröffentlicht wurden und benutzt werden können. Ab diesem Zeitpunkt wird mit der Öffentlichkeitsarbeit richtig begonnen.

Eine einfache Webseite ist jedoch bereits online: <https://hoax.ly/index.html>

Darüber hinaus ist hoaxly auch auf Facebook und Twitter vertreten:

[https://twitter.com/hoax\\_ly](https://twitter.com/hoax_ly)

<https://www.facebook.com/Hoaxly-162897680835820>

Beim netidee OpenSource Community Camp gewannen wir mit unseren Pitch den dritten Preis und erhielten dadurch eine Erwähnung auf <https://www.derbrutkasten.com/>.



Preisverleihung netidee OpenSource Community Camp (Foto: Arman Rastegar)

Besonders gefreut hat uns die darauf folgende Einladung zur Ars Electronica im September. Diese Chance mussten wir nutzen, um Feedback zu sammeln und Kontakte zu knüpfen. Hierfür haben wir eine eigene Anwendung programmiert, das hoaxly Fake News Game: <https://www.aec.at/ai/en/hoaxly/>

Das hoaxly Fake news Game ist eine interaktive Installation, in der Schlagzeilen nach dem vermuteten Wahrheitsgehalt bewertet werden. Die SpielerInnen lesen Schlagzeilen, treffen Bewertungen und erhalten umgehend Rückmeldung über den Ursprung der Nachricht. Die Analysen von Plattformen, die Falschmeldungen im Internet aufdecken, zum Beispiel Mimikama oder Snopes, erscheinen auf dem Bildschirm, inklusive Statistiken, wie oft Meldungen in sozialen Medien geteilt wurden. Das Game ist eine Kombination aus zwei netidee Projekten: OpenTrigger (Voting über kabellose Buttons) und hoaxly.

Hierfür wurden etwa 40 unterhaltsame Meldungen ausgewählt. Das Spiel ist bei den BesucherInnen sehr gut angekommen, insgesamt wurden bei der Ars Electronica 1,234 Einschätzungen abgegeben. Nur rund zwei drittel (63,45%) davon waren richtig eingeschätzt. Ein Indiz dafür, dass es nicht leicht ist, den Wahrheitsgehalt von Informationen einzuordnen.



Das hoaxly fake news game bei der Ars Electronica 2017 (Foto: Nico Griener)

Ein Blogpost über die Auswertung der Ergebnisse ist in Arbeit, hier noch eine Übersicht über die am meisten falsch oder richtig eingeschätzten Meldungen:

true: votecorrect: Descending

articleheading.keyword: Descending ↕	Count
AfD-Politiker Podeswa fordert Frauenverbrennungen gegen Klimawandel	36
Volvo: Kängurus lassen selbstfahrende Autos verzweifeln	31
Richter lobt Drogendealer für Geschäftsmodell	31
Polizei belagert 7 Stunden lang leeres Haus	30
Mehrere Waffen in Versteck auf Donauinsel gefunden	28
3 gleichnamige Kandidaten bei Rumänischer Wahl	26
Drug-sniffing lizard joins Arizona police force	21
Kondome mit Luthers Segen	21
Frau täuschte Blindheit vor - 28 Jahre lang	20
Fleischallergie nach Zeckenbiss	9

false: votecorrect: Descending

articleheading.keyword: Descending ↕	Count
Fleischallergie nach Zeckenbiss	33
Frau täuschte Blindheit vor - 28 Jahre lang	23
Kondome mit Luthers Segen	22
Drug-sniffing lizard joins Arizona police force	21
3 gleichnamige Kandidaten bei Rumänischer Wahl	17
Mehrere Waffen in Versteck auf Donauinsel gefunden	15
Polizei belagert 7 Stunden lang leeres Haus	13
Volvo: Kängurus lassen selbstfahrende Autos verzweifeln	12
Richter lobt Drogendealer für Geschäftsmodell	12
AfD-Politiker Podeswa fordert Frauenverbrennungen gegen Klimawandel	7

Abb.: hoaxly fake news game Auswertung: Die am meisten falsch oder richtig eingeschätzten Meldungen



Der Code hierfür wurde auf github veröffentlicht: <https://github.com/piccaso/HoaxBuzzer>  
Die Daten über die Einschätzungen der TeilnehmerInnen sind hier einsehbar:  
<https://drive.google.com/open?id=1Oq6o8gjCJKgJnnxDfWfmD9g9VxnXlCo>

## SOLL-IST (Abweichung-)Analyse

Es ist schwer zu übersehen, dass wir weit hinter dem ursprünglichen Zeitplan liegen. Laut Plan müssten wir das Projekt bereits fertig gestellt haben, wir sind jedoch erst bei geschätzt knapp über 50 %.

Was sind die Ursachen? Wir haben erstens die technische Schwierigkeit des Projektes stark unterschätzt. Im Detail zeigt sich, dass das Scrapen und die Normalisierung der Daten nicht so einfach ist, vor allem wenn man es richtig machen möchte.

Auch konzeptuell sind wir auf ein paar Hürden gestoßen. Hauptpunkt für die Verzögerung war jedoch die im Sommer bzgl. Ressourcen sehr angespannte Situation in unserer Firma (u.a. auch aufgrund der DrupalCon Vienna, einer großen Konferenz, auf die wir Jahre lang hingearbeitet haben und die nun im September statt gefunden hat). Die Situation hat sich jedoch nun entspannt und wir sind zuversichtlich, die hoaxly Anwendungen im ersten Quartal 2018 fertig stellen zu können.

Im August (und Großteils auch im September) wurden fast alle Stunden für das hoaxly Fake News Game für die Ars Electronica verwendet (in AP 11 gebucht).

Eine Änderung sind hier ursprünglich nicht eingeplante 24 Stunden für die Entwicklung des Tools von Florian Fida (extern zu € 35.-), diese wurden von Nico Grienausers Zeitbudget im AP 11 abgezogen (im Dokument gelb markiert bei den Arbeitspaketen). In der Gesamtanzahl der Stunden ändert sich nichts.

## Übersicht über die erreichten bzw. noch offenen Funktionalitäten aus Projektgesamtsicht

Erreicht:

- Server ist eingerichtet und läuft, hoax.ly domain registriert
- Inhaltl. und technisches Konzept ist fertig gestellt, inkl. Analyse von Richtigstellungsplattformen
- Datenbank ist funktionell. Einige Importer sind erstellt (u.a. für [climatefeedback.org](http://climatefeedback.org), [mimikama.at](http://mimikama.at), [snopes.com/](http://snopes.com/), [www.stopfake.org/](http://www.stopfake.org/), [cochrane.org](http://cochrane.org), ...)
- Qualitätskriterien für Datenquellen definiert
- FAQs für Endanwender: <https://hoax.ly/faq.html>
- Präsentation des Projektes auf diversen Events (govCamp, Ars Electronica, netidee OpenSource Community Camp)
- 3. Platz bei netidee OpenSource Community Camp Pitch
- hoaxly Fake News Game
- Basis für das Webseiten Installationspaket



- Userstories und Konzept für Anwendungen

Noch Offen:

- Fertigstellung Webseiten Installationspaket
- Berechtigungslayer Datenbank
- "Friendly Scraping" sicher stellen
- Fertigstellung Chatbots
- Fertigstellung Browser Extension
- Testen Anwendungen und Userfeedback einholen
- Dokumentation und veröffentlichungsfähiger Einseiter
- Öffentlichkeitsarbeit und Partner finden

## Sonstige Highlights (z.B. Lessons Learned, Vernetzungsaktivitäten, Öffentlichkeitsarbeit/ Publikationen)

Ö1 Interview: Beim letzten GovCamp Vienna gab es eine interessante Diskussion zum Thema "Wege aus der politisch motivierten Lüge", im Zuge dessen wir hoaxly präsentierten. <http://dialogplus.at/2016/12/12/wege-aus-der-politisch-motivierten-luege/>

Anschließend durfte Christian Ziegler zum Thema ein Ö1 Interview geben (für die Sendung Matrix). Das Interview ist leider inzwischen nicht mehr online.

Zusammenarbeit mit SuperGraph (Thomas Hoch von FH Hagenberg): Thomas Hoch hatte gemeinsam mit Hiu-wai Chan aus Hongkong ein Projekt auf der Ars Electronica präsentiert, das ebenfalls Fake News zum Thema hatte: The Murder of Jo Cox. Er ist Computer Scientist im Bereich Machine Learning und Data Mining. Für Februar ist ein gemeinsamer Workshop geplant.

Man sieht, das Projekt hat großes Potential. Wir freuen uns schon sehr darauf, die hoaxly Anwendungen in Aktion zu sehen und mit der Öffentlichkeitsarbeit richtig durchzustarten.

## Ausblick/Ideen

- Einbindung von Datenbanken wie <http://www.opensources.co/> oder <https://github.com/marktron/fakenews> (könnte die Nützlichkeit unserer Anwendungen deutlich erhöhen, weil zusätzl. vor bestimmten Seiten generell gewarnt werden könnte).
- Aufnahme von Zitaten in die Datenbank.
- Tool, um Meldungen live in realtime überprüfen zu können. Kann Sprache in Text fassen und mit Datenbank abgleichen. In Datenbank nicht vorhandene Aussagen können über Crowdsourcing überprüft werden.



- Nicht nur die Anzahl der Nennungen auf Twitter wird gespeichert. Auch die Accounts, die diese Meldungen teilen, werden gespeichert. So könnten Accounts eruiert werden, die überdurchschnittlich oft erwiesen falsche Meldungen teilen (zB Bots).
- Tool, wie <http://www.sharethefacts.org/>, nur OpenSource und mit standardisierten Rating values. Ev. Drupal und WordPress Modul, das so ein widget anbietet und so zur Verbreitung von ClaimReview schema.org beiträgt. Da hätte zur Folge, dass wir diese Seiten auch leichter crawlen können.