

netidee

PROJEKTE

somes

Endbericht | Call 20 | Projekt ID 7723

Lizenz CC BY

Inhalt

1 Einleitung **3**
2 Projektbeschreibung **3**
3 Verlauf der Arbeitspakete **4**
3.1 Arbeitspaket 1 - Detailplanung und Formales am Projektstart 4
3.2 Arbeitspaket 2 - Somes v1 – Open Source Release & stabiler Funktionsumfang 4
3.3 Arbeitspaket 3 - Partizipation 5
3.4 Arbeitspaket 4 - KI-Funktionalität 6
3.5 Arbeitspaket 5 - Extensibility und EU-Erweiterung 7
3.6 Arbeitspaket 6 - Datenqualität und Datenquantität 8
3.7 Arbeitspaket 7 - Mobile Applikation 8
3.8 Arbeitspaket 8 – Dokumentation und Formales am Projektende 9
4 Umsetzung Förderauflagen **9**
5 Liste Projektendergebnisse **9**
6 Verwertung der Projektergebnisse in der Praxis **11**
7 Öffentlichkeitsarbeit/ Vernetzung **11**
8 Eigene Projektwebsite **12**
9 Geplante Aktivitäten nach netidee-Projektende **13**
10 Anregungen für Weiterentwicklungen durch Dritte **13**

1 Einleitung

Mit diesem Projekt konnten wir unsere Plattform “somes.at” auf das nächste Level heben, in dem wir die Architektur modernisiert, Standards etabliert, das Frontend deutlich niedrigschwelliger gestaltet, Beteiligungsfunktionen eingebaut, KI plattformweit eingesetzt, das Europäische Parlament überwachbar gemacht und die Entwicklung der somes Mobile App v1 abgeschlossen haben.
Wir waren überaus glücklich, uns dem langfristigen Ziel “vom Schulprojekt zur internationalen Musterlösung” annähern zu dürfen und freuen uns auf weitere Entwicklungen.

2 Projektbeschreibung

somes ist eine Plattform für politische Transparenz und Beteiligung. Der Name steht für “social frames”, gebildet aus “so” und “mes”. Weil Kommunikation nie wertneutral ist, wollen wir damit faire und verantwortungsvolle Rahmen anbieten, die politischen Diskurs ermöglichen, ohne ihn durch manipulative Sprache zu verzerren.

Projektziel war es, parlamentarische und regierungsspezifische Aktivitäten, Prozesse und Informationen so niederschwellig zugänglich zu machen, dass nachvollziehbar wird, wofür Abgeordnete tatsächlich stimmen und sprechen. Die Rohdaten wären dazu vorhanden, sind aber unstrukturiert, teils lückenhaft und ohne Aufbereitung schwer verwendbar, weshalb wir u.a. Redetexte, namentliche Abstimmungen und Abwesenheiten aus den stenographischen Protokollen extrahieren, parlamentarische Gegenstände thematisch einordnen, automatisiert auswerten und zusammenfassen.

Zielgruppen sind politikinteressierte Privatpersonen, die u.a. in der Informationsflut nicht mehr erkennen, wer sich für das einsetzt, wofür sie oder er sich ausgesprochen hat, Journalist*innen und NGOs, denen z.B. eine strukturierte Datengrundlage und Vergleichsmöglichkeiten zwischen Abgeordneten fehlen, sowie Politikwissenschafter*innen, die politisches Verhalten quantifizierbar und themenzentriert über mehrere Gesetzgebungsperioden analysieren wollen. Für alle drei stellen wir belegbare Zusammenfassungen, filterbare Suchen, Statistiken sowie Richtungs- und Haltungsanalysen bereit. Dazugekommen sind Abgeordnete, die über das neue Fragesystem selbst antworten, und nachnutzende Systeme, die über API bzw. unseren MCP-Server zugreifen.

Projektergebnis ist die zweisprachige Plattform somes.at mit getrennten Bereichen für Österreich und das EU-Parlament samt Mobile App. Der Datenbestand umfasst Abstimmungen, Verhandlungsgegenstände, Verordnungen, Ministerialentwürfe, Reden, Abwesenheiten und Abgeordnetenprofile, zugänglich über eine Rust-API mit versionierten Endpunkten und stark filterbarer Suche. KI-seitig erzeugen wir sprechende Titel, Themen, Zusammenfassungen in verschiedenen Detailgraden und Schwerpunkte mit Referenz auf die Stelle im Gesetz bzw. in der Rede, verlinken automatisiert relevante Medienberichte und betreiben Richtungsanalysen samt Orientierungsquiz.

Mit Abgeordnetenfragesystem, Stimmungsbarometer und Volksbegehren-Übersicht haben wir die Beteiligungsfunktionen umgesetzt und den Code offen auf github.com/som-es veröffentlicht.

3 Verlauf der Arbeitspakete

3.1 Arbeitspaket 1 - *Detailplanung und Formales am Projektstart*

Am Projektstart wurden alle erforderlichen Formalitäten nach Plan abgeschlossen.

3.2 Arbeitspaket 2 - *Somes v1 – Open Source Release & stabiler Funktionsumfang*

In AP2 wurden die Codebasen modernisiert, neue Funktionen hinzugefügt und das Frontend neu gestaltet. Primäres Ziel war es, die (bestehenden) Funktionen umfangmäßig und technisch zu stabilisieren, um daraus die v1 zu definieren. Wir sehen das Arbeitspaket als abgeschlossen an.

Frontend

- Migration und defacto rewrite von svelte 4 und skeleton ui toolkit zu svelte 5 und bits-ui
  Die kontinuierliche Weiterentwicklung des svelte Ökosystems hat diesen Schritt langsam aber sicher notwendig gemacht. Das neue reactivity System hatte uns anfangs etwas Schwierigkeiten bereitet, weshalb die Migration von besonders reaktiv verketteten Components und Pages mehr Zeit in Anspruch nahm, als ursprünglich geplant.
- Refactoring zu Server Side Rendering (SSR). Damit in Zukunft konkrete Gegenstände von Suchmaschinen und AI-Agents indexiert/aufgesucht werden können, setzen wir jetzt für einen Großteil der Funktionen auf SSR. Diese Änderungen bewirken zusätzlich, dass vieles deutlich interaktiver und flüssiger wirkt und ist (zB Suche nach Abstimmungen).
- Anpassung aller Components und Pages an ein modernes, niederschwelliges, professionelles Design mit Fokus auf Mobile und Progressive Disclosure. Zuerst in Figma visualisieren und später mündlich diverse Vor- und Nachteile aller Ansätze auszudiskutieren, war zwar aufwändig, hat sich allerdings unserer Meinung nach rentiert.
- Diverse neue Funktionen, die auch wegen des Redesigns entstanden sind
  a. Deutlich mehr Filterfunktionen
  b. Suche über alle Abgeordneten
  c. Umfangreiche Konfiguration der E-Mail-Benachrichtigungen
  d. Zwischenrufe, Ordnungsrufe, Abwesenheiten, eingebrachte Anträge bei Abgeordnetenprofilen
  e. Übersicht der letzten Nationalratssitzung
  f. Neuentwicklung der Statistiken
  g. AI-Toggle
  h. …

Backend

- nginx für SSR (via node) und API (via Rust/axum-rs) Routing
- Refactoring, Versionierung, Internationalisierung und Gruppierung, insbesondere der API-Routes. Statt flachen Routen gibt es endlich hierarchische Endpunkte.
    - `https://somes.at/api/at/v1/vote_results/search?is_finished=true&page=1&entries_per_page=1000&search=`
- Filterung für die meisten Attribute der im meilisearch abgespeicherten Daten. (Zugänglich über <endpoint art>/search). Dafür wurde ein Rust-Attribute-Macro entwickelt, welcher entsprechend konfigurierbare Filterdatentypen generiert. `legislative_initiative[ityp][in][0]=I&legislative_initiative[voting][in][0]=Law&eurovoc_topics[0][topic][cn]=Wirtschaft`
    - `"Ityp"` wäre eigentlich keine Liste, also könnte man nicht mit einer Anfrage
    - mehrere unterschiedliche Typen erhalten. Der Macro kann allerdings optional nach Angabe das Attribut zur Liste im Filtertyp umwandeln. Die obere Anfrage filtert nach Initiativen && Gesetzen mit dem Thema Wirtschaft.
- SQL-Views für die wichtigsten API-Models auf Basis von Postgresql-Composite-Types, wofür ebenfalls ein Macro entwickelt wurde, welcher Rust-Typen zu SQL transformiert.
- Alle sonstigen Routen für die neuen Funktionen wurden hinzugefügt, darunter ein Abgeordneten-Meilisearch-Index.
- Falls Aktualisierungen in der Datapipeline registriert werden, werden mittels redis-stream abonnierte Caches informiert, welche daraufhin aktualisiert werden können. So sparen wir Ressourcen.
- Funktionalität der E-Mail-Benachrichtigungen bei neuen Einträgen/Gegenständen und Abgeordnetenaktivitäten
- Viele Bugfixes, Generalisierungen, diverse erweiterte Speicherung von Daten aus AP6, bessere Multithreading-Nutzung, …

3.3 Arbeitspaket 3 - *Partizipation*

Neben verbesserter politischer Transparenz setzen wir auf Funktionen, welche die politische Beteiligung der Bürger*innen stärken sollen. Dafür planen wir die Entwicklung von einem Abgeordnetenfragesystem nach abgeordnetenwatch.de-Vorbild, ein simples Stimmungsbarometer für eingebrachte Anträge und Entwürfe sowie die Auflistung von Volksbegehren. Geschätzter Fertigstellungsgrad: 100%

Abgeordnetenfragesystem

- Aufbau der grundlegenden Architektur für E-Mail-Versand an Abgeordnete, korrekter Zuordnung der Antwort und Implementierung der API
- Berücksichtigung von Abgeordneten-Somes-Accounts → statt dedizierten Accounts setzen wir vorerst vollständig auf E-Mails.
- Aufsetzen von Testumgebung
- Designing in Figma und Visualisierung im Web-Frontend

- Vorbereitung von Fragen zur erstmaligen Belebung
- Moderationsmöglichkeit

Stimmungsbarometer

- Entwürfe auf einer Skala von “dagegen” bis “dafür” bewerten
- Aggregierter Wert inklusive Barchart
- API-Integration mit Accountzwang gegen Mehrfachangaben

Volksbegehren

- Scraping der aktuell zur Eintragung stehenden Volksbegehren auf der Seite des BMI
- Historische Eintragungswochen werden über die Wayback-Machine verarbeitbar gemacht
- Visualisierung bzw. Erinnerung mit allen unterstützbaren Volksbegehren, dass gerade eine Eintragungswoche stattfindet → auf der /home bzw. Neuigkeitenseite

3.4 Arbeitspaket 4 - *KI-Funktionalität*

Um insbesondere die Niederschwelligkeit der Informationen zu gewährleisten, setzen wir auf KI. Im AP 4 werden Informationen zusammengefasst, Änderungen in der Architektur vorgenommen und automatisierte Analysen durchgeführt. Geschätzter Fertigstellungsgrad: 93%

Wir konnten eine Widerspruchserkennung umsetzen, die auch ca. der Beschreibung im ursprünglichen Netidee-Antrag entspricht. Der Ansatz bzw. die Funktionalität hat sich jedoch als ungenau bzw. wenig erkenntnisreich herausgestellt, weswegen wir keine Visualisierung und endgültige Pipeline entwickelt haben. Eine Verordnung/einen Ministerialentwurf als Widerspruch zu vorher definierten Standpunkten einer Partei zu klassifizieren, führt im Endeffekt zu einem Vorwurf, welcher besonders validiert und begründet sein muss. Deshalb haben wir Kontakt mit dem Projekt "politikmonitor.at" aufgenommen und um Kooperation angefragt, welche deutlich mehr Wahlversprechen/Standpunkte händisch/halbautomatisch kodieren und schließlich jeweils den Umsetzungsstand einzeln validiert begründen. Leider wurde uns aus zeitlichen Gründen abgesagt.

Architektur

1. Aufbau der python SomAI-API. Zuständig zur Ansteuerung von Inferenz (LLM, Embedding-Models), speziellen Berechnungen (zB cosine similarity) und vor Netidee entwickelten Scripts/Libraries (zB. Abgeordnetenvideovorstellungsextraktion :-) (existiert nicht mehr wegen rechtlichen Gründen))
   Damit wird eine zentrale Steuerung über unsere Rust-basierte Datenpipeline ermöglicht, welche zugleich ein passendes AI-Task-Scheduling implementiert, um unseren lokalen Build & AI-Server (BOSGAME M5 128GB) nicht zu überlasten.

umgesetzte Funktionen

1. Zusammenspiel von allen Komponenten

2. Generierung von Titeln, Themen (falls sie fehlen), Zusammenfassungen in unterschiedlichen detailgraden, Schwerpunkte mit Referenzen zum Gesetz für verbesserte Nachvollziehbarkeit
3. Ehemals python-basierte Abgeordnetenprofilanalyse (Links ⇔ Rechts nach Thema, libertär ⇔ autoritär, sozialistisch ⇔ kapitalistisch) läuft jetzt ebenfalls zentral über unsere Rustdatenpipeline
4. Maschinenlesbare Extraktion/Verwandlung von parlamentarischen Anfragen und Antworten, die als PDFs bereitgestellt sind (mit Demo-Daten versteht sich…)
5. Politisches Orientierungsquiz mit Abgeordnetenähnlichkeitsabgleich hinzugefügt
6. Reden werden verarbeitet und zusammengefasst
    - Visualisierung von Schwerpunkten inklusive Zitat und farblicher Hervorhebung in Primärquelle
    - Behandelt eine Rede tatsächlich einen bestimmten Antrag, und wenn ja, welche Schwerpunkte genau? (Darstellung mit Kreisen neben Antragsschwerpunkten)
7. Relevante Medienberichte verlinken
    - Sitemaps für ausgewählte Medienhäuser (URLs müssen zumindest Schlagwörter beinhalten) wodurch wir rechtlich sauber arbeiten können
        - ORF bietet keine Sitemap an (zumindest konnten wir keine finden)
        - Kronen Zeitung verlinkt lediglich per ID
    - URLs werden in Vektordatenbank gespeichert, dann mittels Antragsstitel + Zeitraum retrieved und anschließend mit detaillierter Zusammenfassung gereranked
8. Bestehendes Reden-Abgeordneten-RAG-System verbessert
    - Quellenangaben ergänzt
    - weitere politische Fragen hinzugefügt (zu jedem Antrag werden automatisiert Fragen erstellt, wir haben im Anschluss die besseren aus der aktuellen Legislaturperiode manuell ausgewählt)
    - System zugänglich gemacht über einen neuen MCP-Server (inklusive API + Tokengenerierungsmöglichkeit im Frontend)
        - Ursprünglich wollten wir zusätzliche Informationen im Abgeordnetenkontext ergänzen. Da das Hosting langfristig allerdings zu teuer wäre (bzw. mit der aktuellen Hardware maximal 2,3 Nutzer möglich wären), haben wir uns entschieden, das günstigere RAG-System stattdessen via MCP-Server zugänglich zu machen

3.5 Arbeitspaket 5 - *Extensibility und EU-Erweiterung*

Implementierung von EU-Abstimmung und Abgeordneten-Extraktionsscrapern. Umlegen der Daten auf unser Datenschema. Zusätzliche Maßnahmen für Multi-Parlament-Unterstützung durchführen. Geschätzter Fertigstellungsgrad: 100%

Wir extrahieren die Daten des Open Data Portals des Europäischen Parlaments und Teile der EU-Parlamentsseite, um grundsätzliche Prozeduren, Abstimmungen, Plenarsitzungen, Reden und Sitzplan verarbeiten zu können. Im Anschluss speichern wir die gesammelten Daten in einer zweiten EU-Datenbank mit identischem Schema. Die Trennung der Parlamente in API und Frontend erfolgt über die URL mittels /at vs /eu.
Die Korrektheit der Daten validieren wir mit Unit-Tests sowie einen Abgleich mit der bestehenden Initiative HowTheyVote.
In diesem Arbeitspaket wurden zusätzlich auch allgemeine UX-Verbesserungen und Bugfixes durchgeführt

Erste Grundlagen wurden bereits in AP 2 geschaffen. Wegen Priorisierung (weniger Kontextwechsel) auf AP 4 und AP 6 verspäteter Start im Juli 2026.

3.6 Arbeitspaket 6 - *Datenqualität und Datenquantität*

(Laufende) Sicherstellung und Verbesserung der Qualität der gesammelten (österreichischen) Daten. Zusätzlich werden mehr Daten extrahiert. Geschätzter Fertigstellungsgrad: 98% → da wir immer wieder auf Edge-Cases treffen, die oft (aber eben nicht immer) auf uns zurückzuführen sind, könnte hier noch deutlich mehr Zeit investiert werden.

Verbesserungen

1 Die Anzahl der Reden im Datenbestand hat sich mehr als verdoppelt. (Jetzt: 232966 vs 111342) während sich die erfolgreiche Redetextextraktionsqoute auf 95% erhöht hat. Wir schätzen, dass ein großer Teil der restlichen 5% wegen Fehlern in den Rohdaten von parlament.gv.at zustande kommt. Deshalb haben wir eine Liste von menschlich überprüften Debatten, welche in unserer Datenbank keinen Redetext hinterlegt haben, erstellt und diese der Parlamentsdirektion gemeldet.
Ein großer Teil der neuen Reden stammt von der Erweiterung auf den Bundesrat.

1. Datenmäßige Vorbereitung für eine spätere Erweiterung auf den Bundesrat. (Extraktion von Abwesenheiten, den namentlichen Abstimmungen, Reden/Debatten)
2. Extraktion der parlamentarischen Anfragen und Antworten (für Demozwecke)
3. Hinzufügen von vielen weiteren Unit-Tests für alle Extraktionsarten
4. Erste Meldung von Fehlern in den Rohdaten durchgeführt
    4.1. Hinweise auf fehlerhafte Protokollierung bei einer namentlichen Abstimmung
    4.2. Falsche Klubnamen bei Abstimmungen
    4.3. fehlende Verlinkung zu Protokoll für namentliche Abstimmungen (jetzt gibt es immer eine Verlinkung)
    4.4. Sonstige kleinere fehlerhafte Einträge
5. Extraktion von Startzeiten von Reden, Anpassung an unangekündigte Parlaments-API Änderungen, Verbesserte Cachingalgorithmen, größere Abdeckung von Extraktion der namentlichen Abstimmungen und Abwesenheiten, sonstige Bugfixes, …

3.7 Arbeitspaket 7 - *Mobile Applikation*

Neuentwicklung der Basis für eine demoreife Somes Mobile App auf Grundlage der Webapp. Geschätzter Fertigstellungsgrad: 100%

Da die Mobile App ähnlich aussehen soll und wir die Webapp auch mobilorientiert entwickeln, wurden Designs in Figma für die Mobile App unter AP 2 durchgeführt.

Unser vorgenommenes Ziel, die Mobile App auf den Stand der Webapp vom Dezember 2025 zu bringen, konnten wir bereits “in-house” vollständig umsetzen. Mit dem restlichen Budget konnten wir die Mobile App noch intensiver testen und neuere Funktionen, die über den Netidee-Zeitrahmen implementiert wurden, nachliefern. Wegen gesundheitlichen Problemen wurde ein externer Entwickler mit der Weiterentwicklung beauftragt. Demnach bietet die Mobile App zum Abschluss auch die EU-Erweiterung, das Abgeordnetenfragesystem sowie weitere KI-Funktionen.
Nach Durchlaufen der Google Play Store Testphasen, ist die Android App seit dem 19.10.2026 für alle verwendbar.

3.8 Arbeitspaket 8 – *Dokumentation und Formales am Projektende*

Während des Projektausklangs wurden alle erforderlichen Formalitäten abgeschlossen. Zusätzlich wurde die ursprüngliche OpenAPI-Dokumentation reaktiviert.

4 Umsetzung Förderauflagen

Es wurden keine speziellen Förderauflagen festgelegt.

5 Liste Projektendergebnisse

1	*Projektzwischenbericht*	*CC BY 4.0*	*netidee.at/somes*
2	*Projektendbericht*	*CC BY-SA 4.0*	*netidee.at/somes*
3	*Entwickler_innen-DOKUMENTATION*	*CC BY-SA 4.0*	*netidee.at/somes*
4	*Anwender_innen-DOKUMENTATION*	*CC BY-SA 4.0*	*netidee.at/somes*
5	*Veröffentlichungsfähiger Einseiter / Zusammenfassung*	*CC BY-SA 4.0*	*netidee.at/somes*

6	*Externkommunikation*	*CC BY-SA 4.0*	*Siehe Projektendbericht*
7	*Abgeordnetenähnlichkeitssystem*	*AGPL 3.0*	*https://github.com/som-es/dataservice* *https://github.com/som-es/somes* *Zugänglich unter: http://somes.at/orientation*
8	*Somes Mobile Applikation zur Interaktion mit den gesammelten Daten für mobile Endgeräte*	*GPL 3.0*	*https://github.com/som-es/somes-mobile-app* *https://github.com/som-es/somes-ts*
9	*KI-Funktionen Teil 1* *Verbessertes RAG, Redeverarbeitung, SomAI-API*	*APL 3.0*	*https://github.com/som-es/somai-api* *https://github.com/som-es/dataservice* *https://github.com/som-es/somes*
10	*Partizipationsmodul* *Volksbegehren, Stimmungsbarometer, Abgeordnetenfragesystem*	*AGPL 3.0*	*https://github.com/som-es/somes* *https://github.com/som-es/scraper*
11	*Gesetzestextverarbeitungsmodul* *Erweiterung des LLM-Kontext für gewisse Verordnungen*	*AGPL 3.0*	*https://github.com/som-es/dataservice*
12	*Quality-Of-Life Modul* *Erweiterte und verbesserte Suchfunktionen, Bugfixes, Home-Übersicht*	*AGPL 3.0*	*https://github.com/som-es/somes*
13	*Datenqualität, Datenquantität und EU-Erweiterung* *Überarbeitung und Erweiterung der Statistiken, Hinzufügen des Europäischen Parlaments, Extraktionen von Scraper verbessern*	*AGPL 3.0*	*https://github.com/som-es/somes* *https://github.com/som-es/scraper* *https://github.com/som-es/dataservice* *https://github.com/som-es/eurovoc*

14	*Parlamentarische Anfragen und Antworten zusammenfügen Gemeinderatsprotokollverlinkungen verbessern*	*AGPL 3.0*	*Mitarbeit/Unterstützung an https://codeberg.org/pib/pibtool https://github.com/some-es/somes https://github.com/some-es/dataservice https://github.com/some-es/scraper*
15	*KI-Funktionen Teil 2 Themenklassifikation, Medienberichte verknüpfen, Schwerpunkte von Anträgen und Reden*	*AGPL 3.0*	*https://github.com/some-es/somai-api https://github.com/some-es/dataservice https://github.com/some-es/sitemaps*

6 Verwertung der Projektergebnisse in der Praxis

Die Projektergebnisse werden nicht archiviert, sondern sind der laufende Betrieb von somes.at. Die Plattform ist weiterhin kostenlos, ohne Registrierung und ohne Werbung zugänglich, sowohl für Österreich (/at) als auch für das Europäische Parlament (/eu). Alle im Projekt umgesetzten Funktionen sind dort live und werden soweit möglich dauerhaft vom Verein weiterbetrieben.

Das Abgeordnetenfragesystem kann direkt in den weiteren Betrieb übergeführt werden. Nutzer*innen wird ermöglicht, öffentliche Fragen direkt an Abgeordnete zu stellen. Dadurch entsteht ein reger Austausch nach Vorbild von "abgeordnetenwatch.de".

Unsere Scraping-Library, welche Reden, alle verfügbaren österreichischen Abstimmungen und vieles mehr extrahiert, kann in andere politische Projekte integriert werden.

Die aufbereiteten Informationen ermöglichen es, politikinteressierten Personen die Aktivitäten des Nationalrates näher zu bringen. Nutzer*innen erhalten die Möglichkeit, einfacher nachzuvollziehen, ob sich die Abgeordneten dafür einsetzen, wofür sie sich ausgesprochen haben, während Journalismus und Politikwissenschaft einfacher auf ältere, aufbereitete Informationen zurückgreifen kann.

7 Öffentlichkeitsarbeit/ Vernetzung

A/N … Aufwand / Nutzen
Neben der verpflichtenden Veröffentlichung von Netidee-Blogs (A/N: mittel / mittel), wurden folgende Maßnahmen gesetzt:

Im Rahmen unseres Vereins haben wir einen Feedbackfragebogen über unsere Projektwebsite entworfen, womit wir uns Rückmeldungen von unseren Mitgliedern eingeholt haben. (A/N: gering / groß) Zusätzlich wurde von uns eine Führung mit Paul Stich durch das Parlament organisiert. (A/N: mittel / mittel bis groß)

Da gemeindetransparenz.at ähnliche Ziele wie somes verfolgt, allerdings auf Gemeinde- statt Parlamentsebene, haben wir deren Betreiber kontaktiert, woraus einige Wochen später ein informativer Austausch im Café Westend wurde. (A/N: mittel / mittel)

Um in Kontakt zu treten und Unterstützung anzubieten, haben wir die Organisatoren des "Transparenz im Parlament"-Volksbegehren kontaktiert und uns kurz ausgetauscht. (A/N: gering / gering mit Potential für groß)

Nach dem Bundeswettbewerb für KI-Finale im Oktober 2025 wurde im Mai 2026 die Veranstaltung "Next Generation AI" abgehalten, wo somes kurz vor Schüler*innen und dem Innovationsminister vorgestellt werden konnte. (A/N: mittel / mittel)

Um auf Fehler im Datenbestand von parlament.gv.at aufmerksam zu machen, haben wir die Open-Data-Abteilung der Parlamentsdirektion kontaktiert. Daraufhin folgte ein wichtiges Telefonat mit dem Open-Data-Beauftragten Simon Hofer. Einige Wochen später waren alle Probleme nachhaltig gelöst. Danke! (A/N: mittel - groß / sehr groß)

Simon Hofer schlug vor, dass wir beim ADV DX Award einreichen. Wir sind der Empfehlung gefolgt und konnten mit unserem Projekt den Talent Award gewinnen. Bei der parallel stattfindenden Data Excellence Konferenz konnten wir hochkarätige Kontakte knüpfen (und sehr gut essen). (A/N: mittel / groß)

Wir haben die Verantwortliche des Politikmonitors kontaktiert und um Kooperation angefragt. (A/N: gering / gering mit Potential für groß).
Am 8. September 2026 haben wir auf Vorschlag von Simon Hofer am Open Data Meetup "Wien und Wein" teilgenommen und uns mit anderen Teilnehmenden ausgetauscht. (A/N: mittel / mittel)

Wir empfehlen, Projekte und Personen direkt zu kontaktieren und anzusprechen. Es können sich immer tolle unvorhersehbare Gelegenheiten ergeben.

8 Eigene Projektwebsite

Alle Ergebnisse des Projekts fließen in unsere Plattform *somes.at*.

9 Geplante Aktivitäten nach netidee-Projektende

Wir führen die Pflege, Betrieb, Finanzierung und Weiterentwicklung der Plattform in Zukunft fort. Dazu gehören u.a:

- die Moderation des Abgeordnetenfragesystems
- die kontinuierliche Verbesserung der Datenqualität
- etwaige Fehlerbehebungen und Verbesserungen
- den Betrieb des lokalen KI-Servers
- die Übernahme der Hosting-Gebühren mittels Wettbewerbsgewinnen

Aktuell begleiten wir eine HTL-Diplomarbeit, welche ältere stenographische Protokolle aus dem Nationalrat in ein maschinlesbares und für somes nutzbares Format umwandelt.

Wir planen bei nlnet, dem europäischen Pendant zu netidee, einzureichen, um eine neue Architektur der Datenspeicherung, weitere EU-Funktionen und die Qualität sowie Quantität der Daten auszubauen.

Unser weiteres langfristiges Ziel ist es, die Weiterentwicklung von Transparenzplattformen nachhaltig zu ermöglichen und zu sichern, indem wir kostenpflichtige Angebote für professionelle Nutzung implementieren und integrieren.

Im Rahmen unseres Vereins möchten wir weiterhin Veranstaltungen zum Thema Transparenz und Demokratie organisieren.

10 Anregungen für Weiterentwicklungen durch Dritte

- Die bestehende Grundlage des Bundesrates nutzen und ausbauen
- Gesetzgebungsprozess stärker integrieren und visualisieren
    - Ausschüsse
- Mehrere Abstimmungsergebnisse pro Antrag/Prozedur darstellbar machen (besonders für EU relevant)
- Mehrere Sprachen neben Deutsch und Englisch hinzufügen
    - Weitere Parlamente integrieren
- Weitere Auswertungen der gesammelten Daten z.B. in Form von Statistiken
- Untersuchungsausschüsse aufbereiten
- Ähnliche, zuvor bereits eingebrachte, Anträge auflisten
- WAF verarbeiten
- Beifälle in Reden extrahieren
- Fragestunden aufbereiten
- Interaktivere Zusammenfassung und Aufbereitung der Plenarsitzungen
- Änderungsanträge und Hauptgegenstand kombinieren