Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Kdo Cy / CEA
CoE Software Engineering

Richtlinien zur Veröffentlichung von Open Source Software im CEA

Guidelines und Prozesse

Version 1.0.0-2-g24d61c2, 16.11.2026

Tabelle 1. Referenzen

Erkennungszeichen	Titel
EMBAG	https://www.fedlex.admin.ch/eli/cc/2023/682/de
BK «OSS Hilfsmittel»	https://www.bk.admin.ch/de/oss-hilfsmittel

Inhaltsverzeichnis

1. Einleitung . . . 3
1.1. Übersicht . . . 3
1.2. Scope dieser Richtlinien . . . 3
1.3. Projekt Dokumentation . . . 4
2. Vorgehen Vorabklärungen . . . 4
2.1. Verteilung der Kosten . . . 5
2.2. Verantwortlichkeiten bei Veröffentlichung von OSS . . . 5
2.3. OSS-Community-Konzept . . . 5
2.4. Prozess Vorabklärungen . . . 6
3. Vorgehen Analyse und Aufbereitung . . . 8
3.1. Analyse Quellcode . . . 8
3.2. Lizenzwahl . . . 9
3.3. Open Source spezifische Dokumentation . . . 9
3.4. Prozess Analyse und Aufbereitung . . . 9
4. Vorgehen Publikation und Bekanntmachung . . . 11
4.1. Wahl der Plattform . . . 11
4.2. Abbildung der Organisationsstrukturen . . . 15
4.3. Publikation der Software . . . 15
4.4. Kommunikation . . . 16
4.5. Prozess Publikation und Bekanntmachung . . . 17

1. Einleitung

Das «Bundesgesetz über den Einsatz elektronischer Mittel zur Erfüllung von Behördenaufgaben» (EMBAG) verpflichtet Bundesbehörden, den Quellcode von Software, die sie entwickeln oder entwickeln lassen, offenzulegen, sofern keine sicherheitsrelevanten oder rechtlichen Einschränkungen bestehen. Das CEA verfolgt diesen gesetzlichen Auftrag und erarbeitet spezifische Richtlinien zur Veröffentlichung von Open Source Software (OSS) unter Berücksichtigung seiner besonderen Anforderungen an Sicherheit und Vertraulichkeit.

1.1. Übersicht

Das vorliegende Dokument dient als **Ergänzung der OSS Hilfsmittel** der Bundeskanzlei und berücksichtigt die besonderen Anforderungen des CEA. Während die bestehenden Leitfäden allgemeine Prinzipien und Prozesse für die Veröffentlichung von Open Source Software im Bundesumfeld definieren, adressiert diese Richtlinien-Erweiterung spezifische Sicherheits- und Vertraulichkeitsaspekte, die für das CEA von besonderer Bedeutung sind. Ziel ist es, eine konsistente Umsetzung der OSS-Strategie zu gewährleisten, ohne die operationelle Sicherheit zu gefährden.

Diese Richtlinien-Erweiterung basiert auf der Em002-2 Anleitung zur Veröffentlichung von Open Source Software der Bundeskanzlei und ergänzt diese um spezifische Vorgaben für das CEA.

Neben diesen **Richtlinien zur Veröffentlichung von Open Source Software im CEA** ist begleitend zu dieser Anleitung **ein Zusatzdokument** Rollen und Verantwortlichkeiten zur Veröffentlichung von Open Source Software vorhanden, welches die **vier bestehenden BK-Checklisten** ergänzt.

1.2. Scope dieser Richtlinien

Die Richtlinien gelten für alle zu veröffentlichenden Softwareprojekte des CEA, einschliesslich solcher, die intern entwickelt oder von externen Firmen im Auftrag erstellt wurden.

Explizit nicht behandelt werden zum aktuellen Zeitpunkt:

- Bestehende und zukünftige Open Source Projekte, die von CEA-Mitarbeitenden entwickelt werden, aber nicht offiziell unter CEA-Branding veröffentlicht werden (keine CEA-Attribution ist erwünscht)
- Open Source Projekte, die von CEA-Mitarbeitenden privat entwickelt werden
- Beiträge (Contributions) zu externen Projekten
- Förderung (Sponsoring) von externen Projekten
- Private, momentan nicht-öffentliche Projekte für die Zusammenarbeit mit Externen (Cloudsourcing)

Erstellt eine externe Firma eine Applikation für das CEA, welche ebenfalls als Open Source Software freigegeben werden soll, dann soll nach diesem Prozess vorgegangen werden:

1. die Software wird ins interne GitLab überführt
2. der hier beschriebene Prozess wird normal durchlaufen

Eine direkte Überführung in ein öffentliches Repository des CEA ist nicht vorgesehen.

1.3. Projekt Dokumentation

Eine Infoseite des Projektes ist im Confluence «CEA Intranet (INTERN)» für alle Mitarbeitenden des CEA einsehbar. Sie besteht aus:

- einer kurzen Beschreibung des Projektes
- der Kontaktinformation des/der Verantwortlichen
- dem ausgefüllten Formular Rollen und Verantwortlichkeiten zur Veröffentlichung von Open Source Software
- dem OSS-Community-Konzept

Die vertraulichen Dokumente werden nicht publiziert, sondern sind bei Bedarf erhältlich:

- alle ausgefüllten Checklisten der Bundeskanzlei
- die ausgefüllte Schutzbedarfsanalyse

2. Vorgehen Vorabklärungen

Referenz

- Em002-2 Anleitung zur Veröffentlichung von Open Source Software - 3 Vorgehen Vorabklärungen

Diese Richtlinien beschreiben den ersten Schritt, der notwendig ist für die Veröffentlichung als eigenständiges Open-Source-Projekt.

Ziel der Vorabklärungen ist es:

- Festhalten der grundlegenden Informationen zur Software und den Verantwortlichkeiten
- Bestimmung, ob sich der Aufbau einer aktiven Community lohnt
- Entscheidung, ob die Software publiziert werden kann

Tätigkeiten:

- Erstellen der Schutzbedarfsanalyse (SEPOS P041, siehe Verzeichnis `SEPOS_Schutzbedarfsanalyse`); bei erhöhtem Schutzbedarf wird die Software nicht veröffentlicht
- Durcharbeiten von Em002-2 Anleitung zur Veröffentlichung von Open Source Software - 3 Vorgehen Vorabklärungen
- Durcharbeiten von Em002-4 Leitfaden OSS-Community
- Ausfüllen von Rollen und Verantwortlichkeiten zur Veröffentlichung von Open Source Software
- Ausfüllen von Em002-2.1 Checkliste OSS Vorabklärung
- Ausfüllen von Em002-4.1 Checkliste OSS Community
- Erstellen des OSS-Community-Konzepts

Bei Unsicherheit oder bei einem besonders grossen Projekt wird der Entscheid im Führungsrapport (Fhr Rap) diskutiert und gefällt.

2.1. Verteilung der Kosten

Das CoE SW-Engineering übernimmt sämtliche Kosten, sofern es nicht spezielle Gründe gibt, die dagegensprechen.
Dies beinhaltet insbesondere:

- Lizenzkosten für die öffentlichen Git-Plattformen (GitLab / GitHub)
- IAM-Kosten (Google Workspace)
- Domain-Kosten (Infomaniak)

2.2. Verantwortlichkeiten bei Veröffentlichung von OSS

Die Verantwortlichkeiten richten sich nach folgendem RACI (Responsible/Accountable/Consulted/Informed):

Aufgabe	C CEA	DC CoE SW-Engineering	C Sektion (des Projektes)	Project Owner
Richtlinien und Prozesse entwickeln	A	R		
Skeleton-Repo erstellen / pflegen	A	R		I
IAM für CEA aufsetzen / pflegen	A	R		
Erstellen / pflegen von anonymen Usern	I	R	A	C
Projekt veröffentlichen und kommunizieren	I	I	A	R
OSS Budget verwalten und Zahlungen tätigen	A	R		

Falls ein Projekt keinen Owner mehr hat (z.B. weil dieser das CEA verlässt und kein Nachfolger besteht), wird es archiviert.

2.3. OSS-Community-Konzept

In den Dokumenten der Bundeskanzlei gibt es keine direkte, explizite Beschreibung eines OSS-Community-Konzepts. Allerdings behandelt das Em002-4 Leitfaden OSS-Community das Thema der Community-Struktur und Governance.

Ein effektives Open-Source-Community-Konzept ist essenziell für die nachhaltige Entwicklung und den Erfolg eines veröffentlichten Softwareprojekts. Es ist für jedes Projekt zu erstellen. Lohnt sich der Aufbau einer aktiven Community nicht, genügt eine Kurzfassung. Sie hält mindestens fest, wie mit Beiträgen, Anfragen und Forks umgegangen wird, sowie die Lizenz und die Kommunikationsmassnahmen.

Das Konzept sollte die folgenden Punkte umfassen:

- **Zielsetzung der Community**: Definition des Zwecks und der erwarteten Vorteile für alle Beteiligten.
- **Organisations- und Governance-Struktur**: Bestimmung der Besitzverhältnisse am Quellcode, Gremien, Entscheidungsprozesse und Steuerung der Community.
- **Roadmap und Change-Prozesse**: Regeln zur Priorisierung neuer Features und Verbesserungen sowie der Umgang mit vorgeschlagenen Änderungen.
- **Beitragsmanagement**: Festlegung der Regeln für externe Beiträge, Qualitätskriterien für Code-Reviews und den Merge-Prozess.
- **Support- und Wartungsstruktur**: Verantwortlichkeiten für Wartung, Weiterentwicklung und Sicherheitsupdates.
- **Community-Interaktion und Kommunikation**: Nutzung von Mailinglisten, Foren oder anderen Plattformen zur Zusammenarbeit sowie die Kommunikationsmassnahmen zur Veröffentlichung (siehe Abschnitt «Kommunikation»).

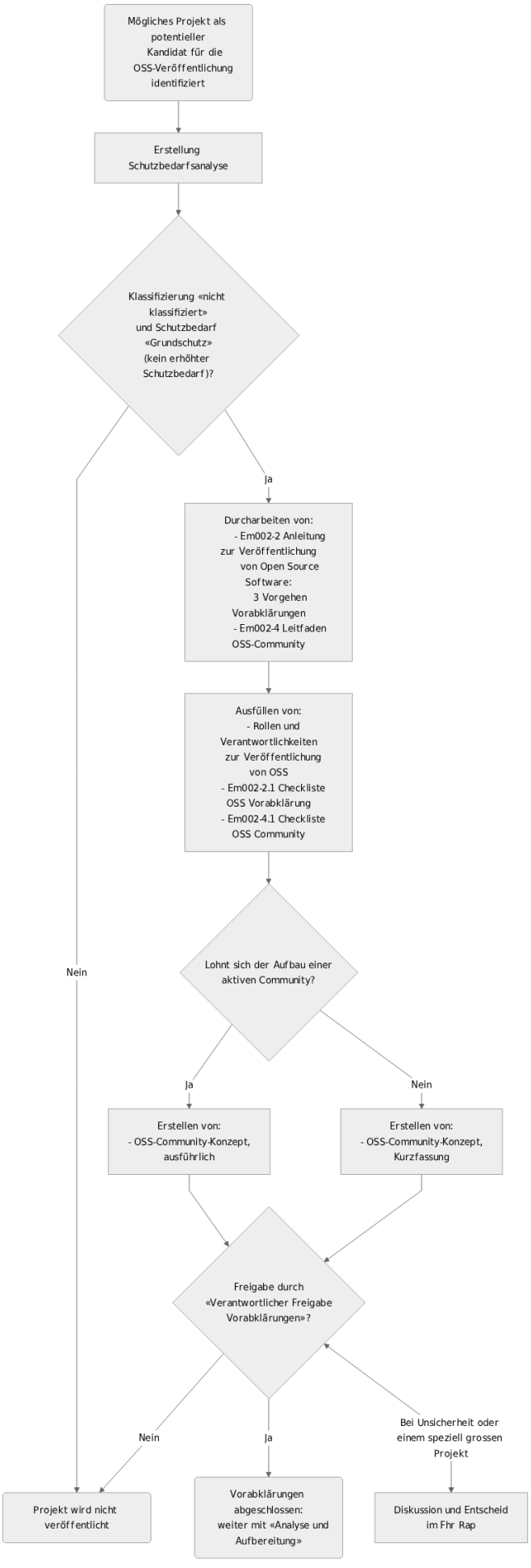
Insbesondere müssen die folgenden (GitLab-) Rollen definiert und Personen zugewiesen werden:

- **Owner(s)**: Langfristige Betreuung des Codes
- **Maintainer(s)**: Prüft und genehmigt Merge Requests

ANMERKUNG	Es muss zumindest ein verantwortlicher Owner vorhanden sein. Der **Project Owner** gemäss Formular «Rollen und Verantwortlichkeiten» ist Owner in GitLab, der **Project Reviewer** mindestens Maintainer.

2.4. Prozess Vorabklärungen

Der Prozess für die Vorabklärungen für die Publizierung eines Open-Source-Projektes ist nachfolgend abgebildet:


Mögliches Projekt als potentieller Kandidat für die OSS-Veröffentlichung identifiziert
Erstellung Schutzbedarfsanalyse
Klassifizierung «nicht klassifiziert» und Schutzbedarf «Grundschutz» (kein erhöhter Schutzbedarf)?
Ja
Nein
Durcharbeiten von:
- Em002-2 Anleitung zur Veröffentlichung von Open Source Software:
3 Vorgehen Vorabklärungen
- Em002-4 Leitfaden OSS-Community
Ausfüllen von:
- Rollen und Verantwortlichkeiten zur Veröffentlichung von OSS
- Em002-2.1 Checkliste OSS Vorabklärung
- Em002-4.1 Checkliste OSS Community
Lohnt sich der Aufbau einer aktiven Community?
Ja
Nein
Erstellen von:
- OSS-Community-Konzept, ausführlich
Erstellen von:
- OSS-Community-Konzept, Kurzfassung
Freigabe durch «Verantwortlicher Freigabe Vorabklärungen»?
Nein
Ja
Bei Unsicherheit oder einem speziell grossen Projekt
Projekt wird nicht veröffentlicht
Vorabklärungen abgeschlossen: weiter mit «Analyse und Aufbereitung»
Diskussion und Entscheid im Fhr Rap

Entscheid:
Grundsätzliches Einverständnis und Machbarkeit zur Veröffentlichung der Applikation als Open Source ist gegeben.

3. Vorgehen Analyse und Aufbereitung

Referenz

- Em002-2 Anleitung zur Veröffentlichung von Open Source Software - 4 Vorgehen Analyse und Aufbereitung

Ziel dieses Schrittes ist es:

- Die zur Veröffentlichung als Open Source notwendigen Arbeiten sind erledigt
- Der Quellcode und die Dokumentation sind überprüft
- Der Entscheid zur Lizenz ist gefällt
- Die Software ist bereit zur Veröffentlichung

Tätigkeiten:

- Durcharbeiten von Em002-2 Anleitung zur Veröffentlichung von Open Source Software - 4 Vorgehen Analyse und Aufbereitung
  (inkl. Anhang D und Anhang E)
- Ausfüllen von Em002-2.2 Checkliste OSS Analyse und Aufbereitung

Die Freigabe erteilt der **Project Reviewer**, eine zweite Person neben dem Project Owner (Vier-Augen-Prinzip; siehe Formular «Rollen und Verantwortlichkeiten», Abschnitt «Verantwortliche für Freigabe»).

3.1. Analyse Quellcode

Jedes Projekt, resp. der Project Owner, ist für die geforderte Code Quality und die Einhaltung dieser Richtlinien durch den Einsatz entsprechender Tools (z.B. Formatter, Linter, …) selbst verantwortlich. Bisher veröffentlichte Software des CEA soll aber als Referenz dafür dienen.

Referenz

- Em002-2 Anleitung zur Veröffentlichung von Open Source Software - Anhang E

3.1.1. Quellcode History

Die gesamte Repository-Vergangenheit (Commit-History) muss vor der Veröffentlichung gelöscht werden (d.h. `rm -rf .git`), um sensible Informationen zu entfernen.

3.1.2. Sensitive, schützenswerte oder vertrauliche Daten

Neben den in Em002-2 Anleitung zur Veröffentlichung von Open Source Software - Anhang E.2 erwähnten Informationen dürfen insbesondere keine Rückschlüsse auf interne Methoden, Quellen oder spezifische Fähigkeiten gezogen werden können. Im Code sollen keine konkreten Informationen über interne Infrastruktur, Mitarbeitende und Strukturen vorkommen.

3.1.3. Dokumentation überprüfen

Referenz

- Em002-2 Anleitung zur Veröffentlichung von Open Source Software - Anhang D

Anpassungen gegenüber den OSS-Hilfsmitteln der BK:

- Die Durchführung, resp. Einrichtung eines Bug-Bounty-Programmes ist optional
- Der Aufbau einer Demo-Instanz und des Deployments mittels CI/CD Pipeline sind optional

3.2. Lizenzwahl

Die Auswahl der Lizenzen für die Veröffentlichung von Software richtet sich nach Em002-3 Leitfaden OSS-Lizenzen der Bundeskanzlei. Die bevorzugten Lizenzen für das CEA sind:

- **Apache License 2.0**, «wenn kein Copyleft gewünscht ist» **(bevorzugt)**
- **GNU Affero General Public License (AGPL) v3**, «wenn eine nachhaltig offene Entwicklung angestrebt wird»

Von den bevorzugten Lizenzen kann begründet abgewichen werden. Die Begründung ist im Community-Konzept des Projekts festzuhalten (Übersichtstabelle, Zeile «Begründung Lizenzwahl»).

Bei Einsatz von Fremdcode sind die Nutzungsbedingungen auf Kompatibilität mit der vorgesehenen Lizenz zu prüfen und falls nötig Code zu ersetzen.

3.3. Open Source spezifische Dokumentation

Dokumentation ist ein integraler Bestandteil von Open Source Software. Diese ist zwingend in einer guten Qualität zu erstellen. Empfohlen wird hierbei Englisch als Sprache.

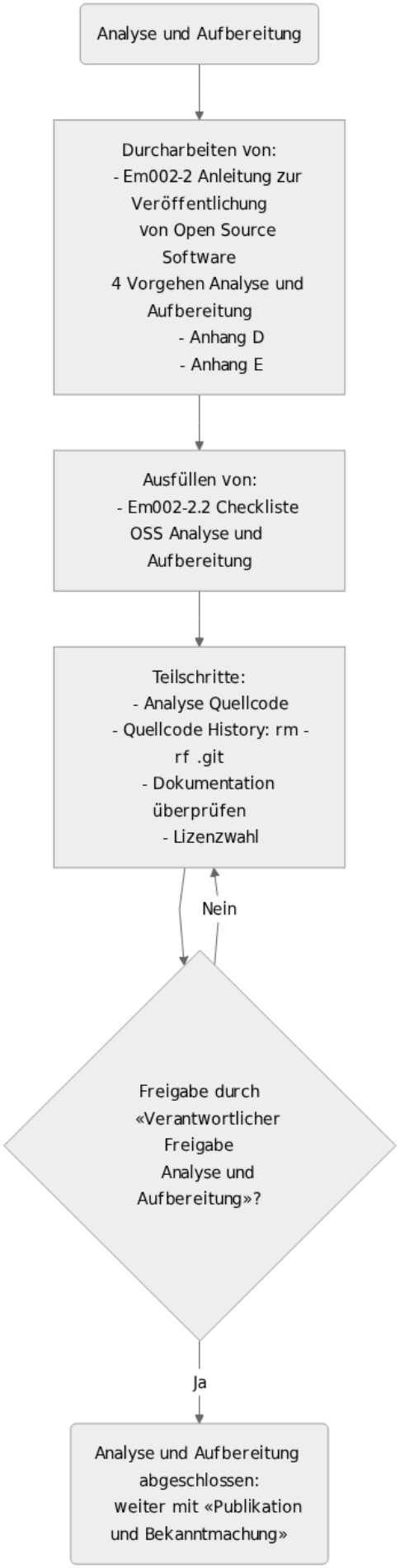
3.3.1. Skeleton-Repository

Im CEA-GitLab wird ein Skeleton-Repository unterhalten (`gitlab.com/swiss-armed-forces/templates/skeleton`). Dieses gilt als Vorlage für die anderen Repositories und regelt Standardeinstellungen und Mindestanforderungen.

Dieses Repository enthält die bei einer Veröffentlichung erwartete minimale Dokumentation. Sie beinhaltet die in Em002-2 Anleitung zur Veröffentlichung von Open Source Software - 4.3 Open Source spezifische Dokumentation genannten Dokumente (README, LICENSE, …) als Vorlage und als Mindestanforderungen an das Projekt.

3.4. Prozess Analyse und Aufbereitung

Der Prozess für die Analyse und Aufbereitung ist nachfolgend abgebildet:


Analyse und Aufbereitung
Durcharbeiten von:
- Em002-2 Anleitung zur Veröffentlichung von Open Source Software
4 Vorgehen Analyse und Aufbereitung
- Anhang D
- Anhang E
Ausfüllen von:
- Em002-2.2 Checkliste OSS Analyse und Aufbereitung
Teilschritte:
- Analyse Quellcode
- Quellcode History: rm -rf .git
- Dokumentation überprüfen
- Lizenzwahl
Nein
Freigabe durch «Verantwortlicher Freigabe Analyse und Aufbereitung»?
Ja
Analyse und Aufbereitung abgeschlossen:
weiter mit «Publikation und Bekanntmachung»

Entscheid:
Definitiver Entscheid zur Veröffentlichung der Applikation als Open Source.

4. Vorgehen Publikation und Bekanntmachung

Referenz

- Em002-2 Anleitung zur Veröffentlichung von Open Source Software - 5 Vorgehen Publikation und Bekanntmachung

Ziel dieses letzten Schrittes ist es:

- Die Software ist veröffentlicht und für Dritte gut auffindbar
- Die Kommunikation der Veröffentlichung hat stattgefunden
- Dem Aufbau einer Community steht nichts im Weg

Tätigkeiten:

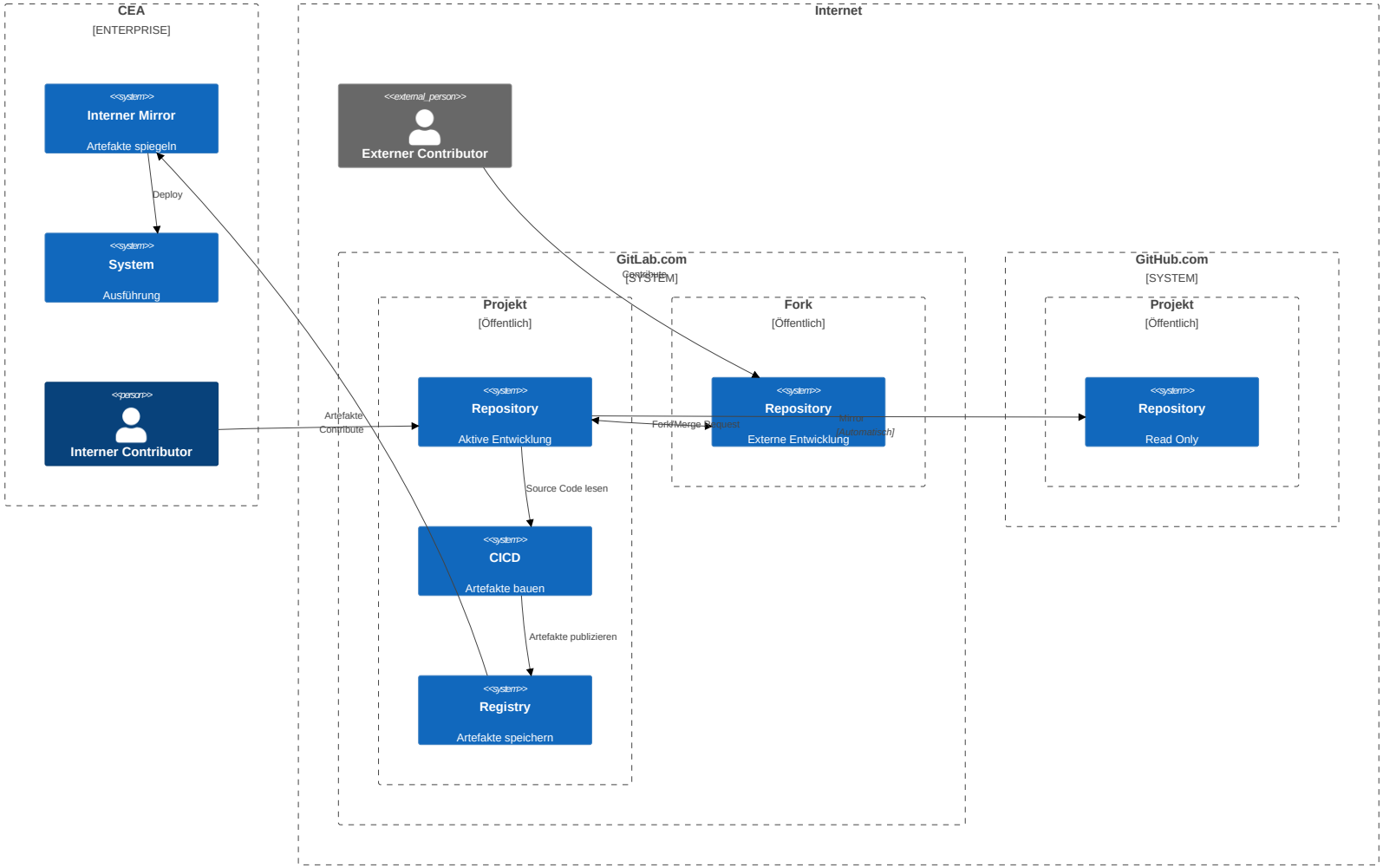
- Durcharbeiten von Em002-2 Anleitung zur Veröffentlichung von Open Source Software - 5 Vorgehen Publikation und Bekanntmachung
- Ausfüllen von Em002-2.3 Checkliste OSS Freigabe und Publikation
- Festlegen der Kommunikationsmassnahmen im OSS-Community-Konzept

4.1. Wahl der Plattform

Open Source Software soll primär auf offiziellen Plattformen veröffentlicht werden, die den Anforderungen der Bundesinformatik entsprechen (GitHub oder GitLab).

Das CEA nutzt primär GitLab.com zur Veröffentlichung von Software. Durch die grössere Verbreitung und Bekanntheit von GitHub.com bietet sich ein kontinuierlicher (read-only) Export dorthin an. Nachdem die Software auf GitLab.com veröffentlicht wurde, findet die weitere Entwicklung nur noch dort statt.

Variante: Öffentliche Entwicklung

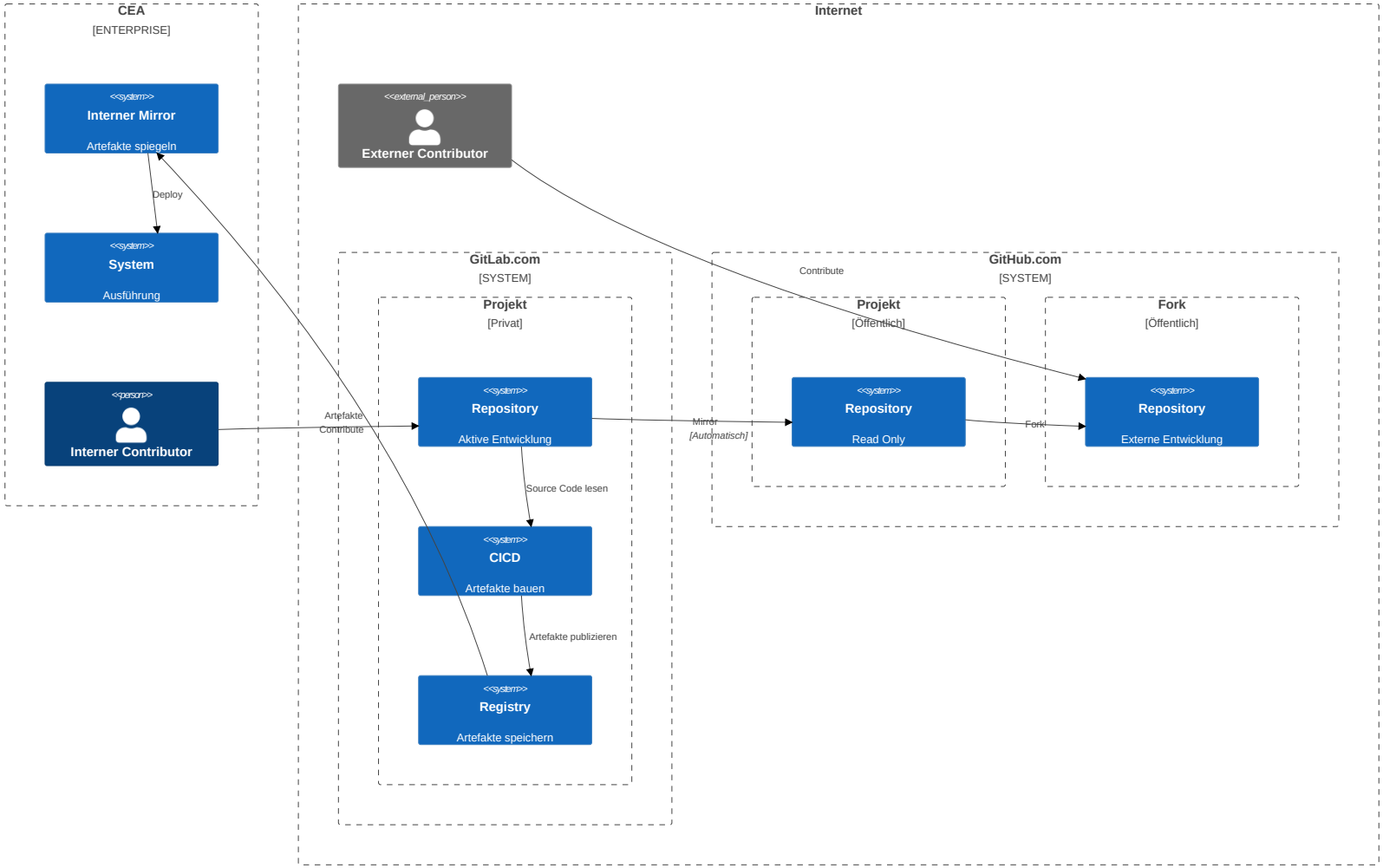


Sobald ein Projekt auf GitLab.com überführt wurde, wird es nur noch dort weiterentwickelt. Interne Changes, Test Cases oder Issues müssen von den CEA-Mitarbeitenden manuell verallgemeinert und ins Online-Repository transferiert werden, um keine Rückschlüsse auf Daten und Personen zu erlauben.

4.1.1. Variante ohne öffentliche Entwicklung

Für Organisationseinheiten, die ausschliesslich an der Veröffentlichung interessiert sind, ohne dass die Entwicklung öffentlich erfolgt, ist eine alternative Struktur erforderlich. Die gesamte Entwicklungsarbeit findet in einem nicht-öffentlichen Repository auf GitLab.com statt. Der read-only Repository-Mirror von GitHub dient primär der Erfüllung der durch das EMBAG verlangten Veröffentlichung.

Variante: Ohne öffentliche Entwicklung



Die Richtlinien in diesem Dokument, resp. in den Dokumenten der Bundeskanzlei gelten trotzdem vollumfänglich.

4.2. Abbildung der Organisationsstrukturen

Die Struktur und Organisation der Repositories unterscheidet sich je nach verwendeter Plattform. Sowohl GitHub als auch GitLab bieten leistungsfähige Funktionen zur Verwaltung von Open Source Projekten, jedoch mit unterschiedlichen Ansätzen:

- **GitHub**: Basiert auf Organisationskonten, in denen Repositories verwaltet und durch Teams organisiert werden können. Der Zugang kann granular gesteuert werden, mit speziellen Rollen für Maintainer, Contributor und Reviewer.
- **GitLab**: Ermöglicht eine tiefere hierarchische Struktur mit Gruppen und Untergruppen. Dies erlaubt eine stärkere organisatorische Einbindung von Projekten und bietet erweiterte CI/CD-Optionen.

Repositories sollten entsprechend der internen Struktur logisch gegliedert werden, um Wartung und Governance zu erleichtern. So werden die Organisationsstrukturen wie folgt abgebildet:

- **GitLab (Entwicklung)**: Hierarchische Gliederung
  - Swiss-Armed-Forces → Cyber-Command → CEA
  - https://gitlab.com/swiss-armed-forces/cyber-command/cea
- **GitHub (read-only Mirror)**: keine Gliederung möglich, nur Organisation
  - Swiss-Armed-Forces
  - https://github.com/Swiss-Armed-Forces/project (Organisation ist bereits bestehend)

4.3. Publikation der Software

Nachdem die Voraussetzungen gemäss Em002-2.3 Checkliste OSS Freigabe und Publikation geklärt sind, kann die Software publiziert werden.

Auf die Erstellung eines ISDS-Konzeptes wird explizit verzichtet, da die Risikoeinschätzung bereits im ersten Schritt **Vorgehen Vorabklärungen** erfolgt ist.

4.3.1. Anonymisierung der User

CEA-Mitarbeitende, die an öffentlichen CEA-Repos arbeiten, bleiben anonym. Sie melden sich mit ihrem anonymisierten Google-Workspace-Konto per Single Sign-on (SAML) bei GitLab an. Die Usernames dürfen der Öffentlichkeit keine Rückschlüsse auf die Identität der Mitarbeitenden erlauben. Innerhalb des CEA müssen die anonymisierten User den echten Benutzern zugeordnet werden können. Alle CEA-Accounts müssen 2FA haben.

Die anonymen Accounts dürfen nicht für private Zwecke verwendet werden. Andere geschäftliche Use Cases (z.B. die dienstliche Verwendung für Drittplattformen wie DockerHub) sind explizit erlaubt, solange kein Informationsabfluss entsteht. Das liegt in der Eigenverantwortung der Nutzenden.

Accounts von externen Firmen werden ebenfalls anonymisiert, wenn die Identität der Firma geschützt werden muss. Wenn dies nicht der Fall ist, können Angestellte von externen Firmen auch mit «normalen» Konten mitarbeiten. Über die Anonymisierung der Mitarbeitenden externer Firmen entscheidet der **C Sektion (des Projektes)**.

4.3.2. Verwaltung der Benutzer (IAM)

Das Identity and Access Management (IAM) wird mittels Google Workspace ermöglicht. Dazu ist **GitLab Premium** notwendig, das eine User-Verwaltung mit externem IAM überhaupt erst erlaubt. Eine zentrale Google-Workspace-Instanz, in der alle anonymen User gepflegt werden, wird für unterschiedliche GitLab-Gruppen verwendet.

Bei **GitHub** ist einzig eine «flache» Organisation möglich. Damit der Export von GitLab auf GitHub funktioniert, muss pro Projekt innerhalb der GitHub-Organisation ein User erstellt werden.

4.3.3. Durchführung der Publikation

Nachdem die (anonymisierten) Benutzer (siehe auch Rollen und Verantwortlichkeiten zur Veröffentlichung von Open Source Software - 2 Rollen und Verantwortlichkeiten) im öffentlichen GitLab kreiert worden sind, kann die Software publiziert werden.
Die folgenden Aufgaben sind abzuarbeiten:

1. Übertragen des lokalen GitLab-Repos auf GitLab.com (Sichtbarkeit vorerst privat)
2. Erstellen des GitHub-Repo (unter Swiss-Armed-Forces, vorerst privat)
3. Aufsetzen des Workflows für den Export in das read-only GitHub-Repo (dies ist in der Verantwortung des Project Owners)
4. Bei Bedarf: Aufsetzen eines lokalen Update-Jobs vom öffentlichen Repo

Die Repositories bleiben privat, bis der Verantwortliche für **Publikation und Bekanntmachung** die Freigabe erteilt hat (siehe Abschnitt «Prozess Publikation und Bekanntmachung»). Erst dann werden sie öffentlich geschaltet.

4.4. Kommunikation

Nach der erfolgreichen Veröffentlichung können weitere Kommunikationsmassnahmen innerhalb und ausserhalb des CEA initiiert werden. Diese fördern die Verbreitung und maximieren den Nutzen der Publikation.

Folgende Möglichkeiten bestehen:

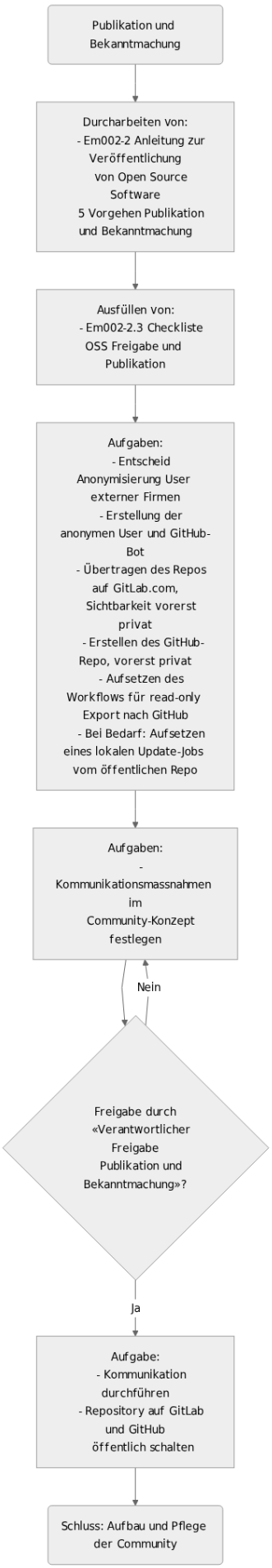
- CEA Intranet News
- Mitteilung an die Öffentlichkeit über die «Kommunikation Kommando Cyber»
- Interessierte Community direkt ansprechen:
    - Social Media Plattformen (X/Twitter, LinkedIn, …)
    - Entwickler-zentrierte Websites (Hacker News, Reddit, Product Hunt, …)
    - Publizieren von Podcasts oder YouTube Videos

Die für diese Software gewählten Massnahmen werden im **OSS-Community-Konzept** beschrieben (Abschnitt «Kommunikation und Sichtbarkeit»). Das Konzept aus den Vorabklärungen wird dazu ergänzt.

Sind diese Massnahmen durch den Verantwortlichen für **Publikation und Bekanntmachung** freigegeben, können sie umgesetzt werden.

4.5. Prozess Publikation und Bekanntmachung

Der Prozess für die Publikation und Bekanntmachung eines Open-Source-Projektes ist nachfolgend abgebildet:


Publikation und Bekanntmachung
Durcharbeiten von:
- Em002-2 Anleitung zur Veröffentlichung von Open Source Software
5 Vorgehen Publikation und Bekanntmachung
Ausfüllen von:
- Em002-2.3 Checkliste OSS Freigabe und Publikation
Aufgaben:
- Entscheid Anonymisierung User externer Firmen
- Erstellung der anonymen User und GitHub-Bot
- Übertragen des Repos auf GitLab.com, Sichtbarkeit vorerst privat
- Erstellen des GitHub-Repo, vorerst privat
- Aufsetzen des Workflows für read-only Export nach GitHub
- Bei Bedarf: Aufsetzen eines lokalen Update-Jobs vom öffentlichen Repo
Aufgaben:
- Kommunikationsmassnahmen im Community-Konzept festlegen
Nein
Freigabe durch «Verantwortlicher Freigabe Publikation und Bekanntmachung»?
Ja
Aufgabe:
- Kommunikation durchführen
- Repository auf GitLab und GitHub öffentlich schalten
Schluss: Aufbau und Pflege der Community