x

WORKSHOP

Luan Noah Lezama Koch

DECEMBER 11, 2024

CSBE

Workshop-IT-Applikation

.1. Zaitplan

Projektmanagement

Name	IST	SOLL	Status	Priorität
Erste Dokumentation		2 tage	In Bearbeitung	High
Arbeitsjournal		Täglich 30 min	Jeden Tag	High
Projektstatusberichte		Täglich 30 min	Jeden Tag	High
Definition der Projektziele		2 tage	Nicht gestartet	Medium
Erstellung des ursprünglichen Zeitplans		3 tage	Erledigt	Medium
Identifizierung von Rollen und Verantwortlichkeiten		1 tage	Nicht gestartet	Low
Erstellung des ursprünglichen Risikoplans		2 tage	Nicht gestartet	Low

Infrastruktur

Name	IST	SOLL	Status	Priorität
Forschung über Cloud-Anbieter		14 tagen	Nicht gestartet	High
Auswahl des Cloud-Anbieters		10 tagen	Nicht gestartet	High
Identifizierung der benötigten Dienstleistungen		10 tagen	Nicht gestartet	High
Entwurf einer Cloud-Architektur		30 tagen	Nicht gestartet	Medium
Dokumentation der Infrastruktur		10 tagen	Nicht gestartet	Medium

Architektur

Name	IST	SOLL	Status	Priorität
Überprüfung der architektonischen Anforderungen		10 tagen	Nicht gestartet	High
Auswahl der Architekturstruktur		14 tagen	Nicht gestartet	High
Evaluierung von Technologien für Komponenten		10 tagen	Nicht gestartet	High
Design der Komponenten Interaktion		18 tagen	Nicht gestartet	High

Planung von CI/CD		10 tagen	Nicht gestartet	Medium
Definition von Sicherheitsmassnahmen		10 tagen	Nicht gestartet	High
Backup und Wiederherstellungsstrategie		7 tagen	Nicht gestartet	High
Architektur Dokumentation		5 tagen	Nicht gestartet	Medium

Testkonzept

Name	IST	SOLL	Status	Priorität
Definition der globalen Testziele		5 tagen	Nicht gestartet	High
Ausarbeitung der Teststrategie		5 tagen	Nicht gestartet	High
Identifikation der Testobjekte		5 tagen	Nicht gestartet	High
Auswahl der Testarten		5 tagen	Nicht gestartet	High
Erstellung der Testfälle		10 tagen	Nicht gestartet	High
Planung der Testinfrastruktur		7 tagen	Nicht gestartet	Medium
Testdurchführung und Dokumentation		15 tagen	Nicht gestartet	High

Vorlage

Name	IST	SOLL	Status	Priorität
Technologie- und Framework-Auswahl		7 tagen	Nicht gestartet	High
Frontend-Grundstruktur erstellen		10 tagen	Nicht gestartet	High
Backend-Grundstruktur erstellen		10 tagen	Nicht gestartet	High
Integration von Entwicklertools		5 tagen	Nicht gestartet	Medium
Versionsverwaltung einrichten		3 tagen	Nicht gestartet	Medium
Modularisierung und Namenskonventionen		5 tagen	Nicht gestartet	Low
Funktionalitäten Dokumentation		2 tagen	Nicht gestartet	Low

Projektmanagement

Name	IST	SOLL	Status	Priorität	Von	Bis
Erste Dokumentation		2 tage	In Bearbeitung	High	2024-12-27	2024-12-29
Arbeitsjournal		Täglich 30 min	Jeden Tag	High		
Projektstatusberichte		Täglich 30 min	Jeden Tag	High		
Definition der Projektziele		2 tage	Nicht gestartet	Medium	2024-12-11	2024-12-13
Erstellung des ursprünglichen Zeitplans		3 tage	Erledigt	Medium	2024-12-14	2024-12-18
Identifizierung von Rollen und Verantwortlichkeiten		1 tage	Nicht gestartet	Low	2024-12-19	2024-12-19
Erstellung des ursprünglichen Risikoplans		2 tage	Nicht gestartet	Low	2024-12-20	2024-12-22
					2024-12-11 bis 2024-12-27	**2024-12-13 bis 2024-12-29**

Infrastruktur

Name	IST	SOLL	Status	Priorität	Von	Bis
Forschung über Cloud-Anbieter		14 tagen	Nicht gestartet	High	2025-01-06	2025-01-23
Auswahl des Cloud-Anbieters		10 tagen	Nicht gestartet	High	2025-01-24	2025-02-06
Identifizierung der benötigten Dienstleistungen		10 tagen	Nicht gestartet	High	2025-02-07	2025-02-21
Entwurf einer Cloud-Architektur		30 tagen	Nicht gestartet	Medium	2025-02-24	2025-04-04
Dokumentation der Infrastruktur		10 tagen	Nicht gestartet	Medium	2025-04-07	2025-04-18
					2025-01-06 bis 2025-04-07	**2025-01-23 bis 2025-04-18**

Architektur

Name	IST	SOLL	Status	Priorität	Von	Bis
Überprüfung der architektonischen Anforderungen		10 tagen	Nicht gestartet	High	2025-01-06	2025-01-17
Auswahl der Architekturstruktur		14 tagen	Nicht gestartet	High	2025-01-20	2025-02-02
Evaluierung von Technologien für Komponenten		10 tagen	Nicht gestartet	High	2025-02-03	2025-02-14
Design der Komponenten Interaktion		18 tagen	Nicht gestartet	High	2025-02-17	2025-03-06
Planung von CI/CD		10 tagen	Nicht gestartet	Medium	2025-03-10	2025-03-20
Definition von Sicherheitsmassnahmen		10 tagen	Nicht gestartet	High	2025-03-24	2025-04-04
Backup und Wiederherstellungsstrategie		7 tagen	Nicht gestartet	High	2025-04-07	2025-04-15
Architektur Dokumentation		5 tagen	Nicht gestartet	Medium	2025-04-16	2025-04-22
					2025-01-06 bis 2025-04-16	**2025-01-17 bis 2025-04-22**

Testkonzept

Name	IST	SOLL	Status	Priorität	Von	Bis
Definition der globalen Testziele		5 tagen	Nicht gestartet	High	2025-04-21	2025-04-25
Ausarbeitung der Teststrategie		5 tagen	Nicht gestartet	High	2025-04-28	2025-05-02
Identifikation der Testobjekte		5 tagen	Nicht gestartet	High	2025-05-05	2025-05-09
Auswahl der Testarten		5 tagen	Nicht gestartet	High	2025-05-12	2025-05-16
Erstellung der Testfälle		10 tagen	Nicht gestartet	High	2025-05-19	2025-05-30
Planung der Testinfrastruktur		7 tagen	Nicht gestartet	Medium	2025-06-02	2025-06-10
Testdurchführung und Dokumentation		15 tagen	Nicht gestartet	High	2025-06-11	2025-07-02
					2025-04-21 bis 2025-06-11	**2025-04-25 bis 2025-07-02**

Vorlage

Name	IST	SOLL	Status	Priorität	Von	Bis
Technologie- und Framework-Auswahl		7 tagen	Nicht gestartet	High	2025-07-03	2025-07-14
Frontend-Grundstruktur erstellen		10 tagen	Nicht gestartet	High	2025-07-15	2025-07-28
Backend-Grundstruktur erstellen		10 tagen	Nicht gestartet	High	2025-07-29	2025-08-11
Integration von Entwicklertools		5 tagen	Nicht gestartet	Medium	2025-08-12	2025-08-18
Versionsverwaltung einrichten		3 tagen	Nicht gestartet	Medium	2025-08-19	2025-08-21
Modularisierung und Namenskonventionen		5 tagen	Nicht gestartet	Low	2025-08-22	2025-08-28
Funktionalitäten Dokumentation		2 tagen	Nicht gestartet	Low	2025-08-29	2025-09-01
					2025-07-03 bis 2025-08-29	**2025-07-14 bis 2025-09-01**

Programminitialisierungsauftrag

Workshop

Klassifizierung	VERTRAULICH
Status	in Arbeit
Programmnummer	1

Programmleiter	Noah Lezama
Version	0.1
Datum	12. Dezember 2024
Auftraggeber	Noah Lezama
Autor/Autoren	Noah Lezama
Verteiler	

Änderungsverzeichnis

Tabelle 1 Änderungskontrolle

Version	Datum	Änderung	Autor
0.1	12.12.2024	Start	Noah Lezama

Beschreibung

Der Programminitialisierungsauftrag bildet die verbindliche Grundlage für die Freigabe der Phase Programminitialisierung. Er ist die Programmvereinbarung zwischen Programmauftraggeber und Programmleiter für die Initialisierung.

Ausgangslage

Ziel des Projekts ist die Analyse und Dokumentation einer bestehenden Webanwendung für die digitale Dokumentenverwaltung. Der Schwerpunkt liegt auf der Erstellung einer Projektdokumentation nach der HERMES-Methodik, einschliesslich einer technischen Vorlage zur Unterstützung künftiger Entwicklungen.

Ziele

Vollständige HERMES-Dokumentation erstellen

Die Anwendung soll gemäss den HERMES-Phasen dokumentiert werden, inklusive Testkonzept und Risikoanalyse.

Infrastruktur- und Architekturkonzept erarbeiten

Auswahl eines geeigneten Cloud-Anbieters und Erstellen eines Architekturplans für künftige Weiterentwicklungen.

Erstellung einer technischen Vorlage (Vorlage)

Entwicklung einer Grundstruktur (Frontend, Backend, Docker, CI/CD), die als Ausgangspunkt für eine Neuentwicklung oder Weiterentwicklung dienen kann.

Ziele der Phase Programminitialisierung

Nr.	Kategorie	Beschreibung	Messgrösse	Priorität*
1	*Projektmanagement*	*Erstellung der Projektdokumentation gemäss HERMES und zaitplan*	*Dokument abgeschlossen*	*M*
2	*Infrastruktur*	*Auswahl und Planung der Cloud-Infrastruktur*	*Cloud-Architektur definiert*	*M*
3	*Architektur*	*Design der Komponenten-Interaktionen*	*Architekturplan erstellt*	*M*
4	*Testkonzept*	*Definition und Planung der Teststrategie*	*Teststrategie genehmigt*	*M*
5	*Vorlage*	*Erstellung einer technischen Vorlage für zukünftige Projekte*	*Vorlage dokumentiert*	*M*
**** Priorität: M = Muss / 1 = hoch, 2 = mittel, 3 = tief***				

Tabelle 2 Auflistung der Ziele

Rahmenbedingungen

Anwendung der HERMES-Methodik

Einhaltung des Zeitplans (11-12-2024 bis 01-09-2025)

Kostenrahmen: 10,000 CHF

Ressourcenbedarf

Personalaufwand

Geschätzter Personalaufwand

Phase	Geplant(stunden)
Programminitialisierung	*160*

Tabelle 3 Mittelbedarf Personalaufwand

Sachmittel

Konferenzräume für Meetings

IT-Infrastruktur: Workstations, Cloud-Tools

Spezifische Software: Projektmanagement- und Testtools

Kosten

Phase	Geplant
Programminitialisierung	*10‘000*

Tabelle 4: Kosten

Termine

Nr.	Ergebnis	Termin
1	*Erste Dokumentation*	*27-12-2024*
2	*Stakeholder Liste*	*11-12-2024*
3	*Definition der Projektziele*	*11-12-2024*
4	*Auswahl des Cloud-Anbieters*	*24-01-2025*
5	*Design der Komponenten Interaktion*	*17-02-2025*
6	*Definition der Teststrategie*	*28-04-2025*
7	*Backend/Frontend Grundstruktur erstellen*	*29-07-2025*

Tabelle 5: Ergebnisse samt Termine

Personalressourcen

Rolle / Person	Monat 1	Monat 2	Monat 3	Monat 4	Monat 5	Bestätigung Vorgesetzter
Programmleiter	*100%*	*100%*	*100%*	*100%*	*100%*	*Vorgesetzter*
IT-Architekt	*50%*	*50%*	*50%*	*50%*	*50%*	

Tabelle 6: Personalressourcen

Kommunikation

Reporting während der Phase Initialisierung, Information Auftraggeber, Information der betroffenen Stellen und Stakeholder

Adressat der Information	Kommunikationsverantw.	Inhalt	Ziel	Mittel / Medium	Termin
Abteilungsleiter	*Noah Lezama*	*Projektstatus und Meilensteine*	*Transparenz schaffen*	*Meeting*	*17-01-2025*
Stakeholder	*Noah Lezama*	*Ergebnisse der Cloud-Auswahl*	*Entscheidung unterstützen*	*E-Mail*	*06-02-2025*

Tabelle 7: Kommunikation

Risiken

Risiken der Phase Programminitialisierung

Nr.	Risikobeschreibung	EW	AG	RZ	Massnahmen	Verantw.	Termin
R1	*Verzögerungen bei der Cloud-Auswahl*	*2*	*2*	*4*	*Frühzeitige Planung und Anbieterabgleich*	*Noah Lezama*	*23.01.2025*
R2	*Fehlende Ressourcen für Testplanung*	*2*	*3*	*6*	*Zusätzliche Kapazität einplanen*	*IT-Architekt (Noah Lezama)*	*25-04-2025*
Legende:	***EW=Eintretenswahrscheinlichkeit: 1 Niedrig / 2 Mittel / 3 Hoch;*** ***AG=Auswirkungsgrad: 1 Gering / 2 Mittel / 3 Gross;*** ***RZ=Risikozahl: RZ = EW x AG***						

Tabelle 8: Risiken – EW x AG = RZ

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
eCH Standard	Eidgenössischer E-Government Standard
HERMES	Vorgehensmethodik für Projekte und Programme HERMES 5 ist ein eCH Standard

Tabelle 9: Abkürzungen und Glossar

Programmrechtsgrundlagenanalyse

Workshop

Klassifizierung	intern
Status	in Arbeit
Programmnummer	1
Programmleiter	Noah Lezama
Version	0.1
Datum	18. Dezember 2024
Auftraggeber	Noah Lezama
Autor/Autoren	Noah Lezama
Verteiler	

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	18.12.2024	Start	Noah Lezama

Beschreibung

Die Programmrechtsgrundlagenanalyse beschreibt die für das Programm bestehende bzw. geltende Rechtsgrundlagen und den allfälligen Bedarf für deren Änderung.

Bestehende Rechtsgrundlagen

1 Datenschutz-Grundverordnung (DSGVO/GDPR)
Legt die Richtlinien fest, wie personenbezogene Daten gespeichert und verarbeitet werden müssen (Sicherheit, Einwilligung, Löschung etc.).
2 Schweizer Datenschutzgesetz (DSG)
Insbesondere relevant, wenn die Datenverarbeitung in der Schweiz stattfindet oder Schweizer Bürger betroffen sind. Die überarbeitete Fassung seit 2023 stärkt die Rechte der Betroffenen.
3 E-Government-eCH-Standards
Diese regeln die Interoperabilität und Transparenz bei elektronischen Behördendiensten sowie Projekten im öffentlichen Sektor in der Schweiz.
4 Vorschriften zur Cloud-Beschaffung
Bei der Nutzung von Cloud-Diensten (insbesondere wenn Server ausserhalb der Schweiz liegen) muss sichergestellt sein, dass die gesetzlichen Vorgaben zu Datenspeicherung, Vertraulichkeit und Datenhoheit eingehalten werden.

Identifizierte Lücken

Verwaltung von Berechtigungen
Aktuell keine klaren Vorgaben, wie vorübergehende Zugriffe oder externe Audits geregelt werden sollen.

Cloud-Konformität
Nicht alle potenziellen Cloud-Anbieter erfüllen zu 100 % die schweizerischen/europäischen Datenschutzanforderungen.

Interoperabilität
Keine klaren Regelungen, die eine problemlose Integration mit zukünftigen oder externen Systemen sicherstellen.

Vorschläge zur Deckung von Lücken

Zugriffs- und Berechtigungssystem
Entwicklung eines granularen Rechte- und Rollenmodells, das alle relevanten Datenschutz- und Sicherheitsanforderungen erfüllt.

Strikter Cloud-Compliance-Check
Erstellung einer detaillierten Checkliste zur Auswahl von Cloud-Anbietern (Standort der Datenzentren, Zertifizierungen wie ISO 27001, klare Haftungsklauseln bei Verstössen).

Interoperabilitätsrichtlinien
Implementierung von Schnittstellen (APIs) gemäss Standardisierungen (z.B. REST/GraphQL mit klar dokumentierten Schemas), um zukünftige Erweiterungen zu erleichtern.

Beurteilung der Konsequenzen

Verzögerungen bei der Umsetzung

Wenn Teile des Projekts aufgrund neuer Vorschriften angepasst werden müssen, kann sich die Gesamtplanung verzögern.

Erhöhte Kosten

Nachträgliche Anpassungen (z.B. bei Sicherheitslücken oder mangelnder Compliance) sind in der Regel teurer als ein durchdachtes Vorgehen von Anfang an.

Vertrauensverlust

Bei Datenschutz- oder Sicherheitsvorfällen kann das Image des Projekts nachhaltig geschädigt werden.

Empfehlung

Rechtliche Entwicklungen beobachten

Kontinuierliches Monitoring der DSG, DSGVO und eCH-Anforderungen durch einen Verantwortlichen.

Vertragliche Flexibilität

Bei Cloud-Providern auf Verträge achten, die Anpassungen ohne übermässige Vertragsstrafe ermöglichen.

Nachhaltige Investition in Standards

Von Beginn an in ein durchdachtes Sicherheits- und Berechtigungsmodell investieren, um spätere aufwändige Korrekturen zu vermeiden.

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
eCH Standard	Eidgenössischer E-Government Standard
HERMES	Vorgehensmethodik für Projekte und Programme HERMES 5 ist ein eCH Standard

Programmstudie

Projektname

Klassifizierung	VERTRAULICH
Status	in Arbeit
Programmnummer	1
Programmleiter	Noah Lezama
Version	0.1
Datum	18. Dezember 2024
Auftraggeber	Auftraggeber
Autor/Autoren	Noah Lezama
Verteiler	

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	18.12.2024	Start	Noah Lezama

Tabelle 4: Änderungskontrolle

Beschreibung

Die Programmstudie bildet die Grundlage für die Entscheidung, ob ein Programm gestartet wird oder nicht. Sie ist die Voraussetzung für die Erarbeitung des Programmmanagementplans und des Programmauftrags. Die Programmstudie beschreibt die Ziele, den Programmumfang mit möglichen Varianten und deren Bewertung.

Ausgangslage

Die Dokumentenverwaltung in der Organisation ist teilweise digitalisiert, jedoch mit deutlichen Ineffizienzen verbunden. Es existieren verschiedene, nur lose miteinander verbundene Ablagen teils in der Cloud, teils lokal, was die Suche und Wiederauffindbarkeit von Informationen erschwert. Außerdem fehlt ein übergreifendes System, das:

- Metadaten zu Dokumenten effizient verwaltet,
- Benutzerberechtigungen und Rollen klar regelt,
- eine automatische Protokollierung (Audits/Logs) ermöglicht,
- und sich mit externen Diensten (z. B. für Dokumentenfreigaben) integrieren lässt.

Um diese Defizite zu beheben, soll eine einheitliche Webanwendung eingeführt werden. Ziel ist es, den gesamten Dokumentenlebenszyklus zu zentralisieren und sowohl internen als auch externen Nutzenden eine sichere, performante und benutzerfreundliche Plattform anzubieten.

Programmvision

Die Vision besteht darin, die Dokumentenverwaltung zu vereinheitlichen, zu automatisieren und sicherer zu gestalten. Auf diese Weise sollen manuelle Aufwände reduziert, die Zusammenarbeit erleichtert und die Einhaltung von Gesetzen und Richtlinien (Datenschutz etc.) gewährleistet werden. Das Endergebnis soll ein skalierbares, modulares System sein, das auch für künftige Anforderungen offen bleibt.

Situationsanalyse

Die Organisation leidet unter verstreuten Dokumenten, geringer Nachvollziehbarkeit von Änderungen und unzureichenden Standards in den bestehenden Prozessen. In diesem Abschnitt wird die aktuelle Situation (IST) zusammengefasst, um die Grundlage für die anstehenden Anforderungen zu schaffen.

Geschäftsorganisation

Struktur: Jedes Fachteam nutzt eigene Lösungen wie Netzlaufwerke, private Cloudspeicher oder sogar isolierte, intern entwickelte Tools.

Hauptprozesse:

- Digitalisieren und manuelles Hochladen von Dokumenten in geteilte Ordner
- Freigaben und Revisionen via E-Mail-Kommunikation
- Lokale Archivierung und teils doppelte Ablage in externen Systemen

Beteiligte Akteure:

- IT-Abteilung: Bietet primär Support, kann jedoch unterschiedliche Systeme nicht einheitlich betreuen
- Fachabteilungen: Jede Abteilung hat eigene Strukturen und Ablage-Standards
- Leitung: Bekommt nur eingeschränkte Einblicke in aktuelle Dokumentenflüsse

Mengen und Häufigkeiten

Eingesetzte Sachmittel

Nr.	Beschreibung	Erläuterung
01	*Physische Server*	*2 Server im lokalen Rechenzentrum*
02	*PCs / Laptops*	*Ca. 50 Geräte im Einsatz*
03	*Cloudspeicher (div. Anbieter)*	*Versch. einzelne Konten bei Dropbox etc.*

Tabelle 2: Eingesetzte Sachmittel / Produkt oder IT System

Geschäftsvorfälle / Transaktionen

Geschäftsvorfall / Transaktion	Durchschnitt pro Zeiteinheit	Spitze pro Zeiteinheit	Minimum pro Zeiteinheit
Dokumenten-Uploads	*2.000*	*5.000*	*500*
Suchanfragen im System	*3.000*	*6.000*	*800*
Änderungen / Aktualisierungen	*500*	*1.000*	*100*

Tabelle 3: Geschäftsvorfall / Transaktion

Datenbestände

Objekttyp	Bestand	Mutationen pro Zeiteinheit	Zugänge pro Zeiteinheit	Abgänge pro Zeiteinheit
Dokumente	*50.000*	*2.000*	*1.000*	*200*
Benutzer	*1.500*	*50*	*30*	*10*
Kategorien/Metadaten	*200*	*20*	*10*	*5*

Tabelle 4: Datenbestände und Bewegungen pro Objekt

Informationssicherheit und Datenschutz

Es gibt keine zentralisierte Verschlüsselungs- oder Zugriffsverwaltung.

Hauptanforderungen:

1. Einhaltung relevanter Datenschutzgesetze,
2. Sicherstellung von Datenintegrität und -verfügbarkeit,
3. Einführung eines Protokoll- und Auditverfahrens für Änderungen.

Stärken-, Schwächen- und Ursachenanalyse

Stärken

Prosa Beschreibung inklusive Überlegungen, ob die Stärken künftig erhalten, oder gar gestärkt werden könnten.

Nr.	Beschreibung	Ursache	Erhaltungs-chancen *
01	*Qualifiziertes Personal im IT-Bereich*	*Kontinuierliche Schulungen, Erfahrung*	*Hoch*
02	*Teilweise bereits digitalisierte Dokumente*	*Einsatz von Scannern pro Abteilung*	*Mittel*
**** Erhaltungsschanzen: + = Stärkung möglich / H = Hoch / M= Mittel / N= Niedrig***			

Tabelle 5: Stärken und ihre Ursachen

Schwächen

Prosa Beschreibung inklusive Überlegungen, ob die Schwächen beseitigt werden könnten.

Nr.	Beschreibung	Ursache	Beseitigungsc hancen *
01	*Kein zentrales Repository für alle Dokumente*	*Zuviele verschiedene Cloud-/Lokalspeicher*	*Hoch*
02	*Freigabeprozesse ohne formale Nachvollziehbarkeit*	*Kommunikation überwiegend per E-Mail*	*Hoch*
03	*Schwierigkeiten bei zentralen Backups und Wiederherstellung*	*Keine standardisierten Abläufe*	*Mittel*
**** Beseitigungschancen: H = Hoch / M= Mittel / N= Niedrig***			

Tabelle 6: Schwächen und ihre Ursachen

Ursachen

Historisch gewachsene Insellösungen in den Abteilungen

Fehlende unternehmensweite Richtlinien zur Dokumentenverwaltung

Zu wenig Automatisierung und fehlendes Monitoring (Protokollierung)

Programmumfang

Das Vorhaben umfasst die Konzeption und Dokumentation einer integrierten Lösung zur Dokumentenverwaltung. Die wichtigsten Bausteine:

Frontend (Web): Ein benutzerfreundliches Portal für die Endnutzer.

Backend / API: Zentrale Services für Authentifizierung, Business-Logik und Dokumentenhandling.

Datenbank: Strukturiertes Speichern von Dokumenten und Metadaten (relational oder NoSQL).

Cloud-Infrastruktur (IaaS/PaaS): Ein Provider, der Hosting, Skalierung und ggf. weitere Dienste bereitstellt.

Sicherheit und Compliance: Verschlüsselung, Protokollierung, Auditing, Backup-Strategien.

Testkonzept: Umfangreiche Tests der Kernfunktionen wie Registrierung, Upload, Suche etc.

Programmziele

System-/ Produktziele

Nr.	Kategorie	Beschreibung	Messgrösse	Gewicht*	Zeitraum
S1	*Organisation*	*Dokumentenverwaltung zentral und für alle Abteilungen zugänglich*	*100 % der Dokumente in einem gemeinsamen Repo*	*Muss (M)*	*Ab Rollout*
S2	*Funktionalität*	*Mehrstufige Loginverfahren (E-Mail/SMS) ermöglichen*	*Erfolgreiche Authentifizierung in Tests*	*Soll (S)*	*Frühphase Entwicklung*
S3	*Qualität*	*Versions- und Änderungsnachverfolgung implementieren*	*Versionshistorie und Audit-Log einsehbar*	*Kann (K)*	*Erste Live-Version*
S4	*Sicherheit*	*Regelmäßige verschlüsselte Backups und verschlüsselte Datenablage*	*Nachweis der Backup-Läufe und Encryption*	*Muss (M)*	*Innerhalb erstes Quartal*
S5	*Benutzerfreundl.*	*Intuitive Weboberfläche*	*Zufriedenheit > 80 % bei Umfrage*	*Soll (S)*	*Vor dem Go-Live*
**** Gewicht: M = Muss / S = Soll / K = Kann***					

Tabelle 7: System / Produktziele

Programmvorgehensziele

Nr.	Kategorie	Beschreibung	Messgrösse	Gewicht*
T1	*Termin*	*Erste nutzbare Version innerhalb von 6 Monaten*	*Applikation produktiv verfügbar*	*Muss*
Q1	*Qualität*	*Sicherstellen, dass Berichte zu hochgeladenen Dokumenten fehlerfrei generiert werden*	*< 1 % Fehlerquote*	*Muss*
K1	*Kosten/Nutzen*	*Budget für Infrastruktur und Entwicklung nicht überschreiten*	*Geplantes Gesamtbudget einhalten*	*Soll*
** Gewicht: M = Muss / S = Soll / K = Kann*				

Tabelle 8: Programmvorgehensziele

Strategiebezug und Umsetzung von Vorgaben

Strategiebezug

Dieses Projekt unterstützt die Digitalisierungsstrategie der Organisation, in der Kostensenkung, verbesserte Nutzerfreundlichkeit und Transparenz zentrale Ziele sind. Durch den Einsatz moderner Technologien (Cloud, Containerisierung etc.) soll ein zukunftssicheres Fundament für weitere Digitalisierungsvorhaben geschaffen werden.

Umsetzung von Vorgaben und Rahmenbedingungen

Beachtung der geltenden Datenschutzvorschriften (z. B. DSGVO/GDPR, schweizerisches DSG).

Orientierung an ITIL-Prozessen für den IT-Betrieb.

Projektsteuerung nach HERMES-Methodik, inkl. klar definierten Phasen und Meilensteinen.

Grobanforderungen

ID	Anforderungen	Art [1]	Abnahmekriterium	Wichtigkeit [2]	Dringlichkeit [3]
A1	*Benutzerregistrierung*	*F*	*Nutzende erhalten nach Registrierung eine Bestätigung*	*5*	*5*
A2	*Anmeldung per E-Mail/SMS*	*F*	*Login unter 2 Sekunden möglich*	*5*	*4*
A3	*Dokumente hochladen*	*F*	*Erfolgreiche Tests mit mehreren Dateien in der Cloud*	*5*	*4*
A4	*Metadaten bearbeiten*	*Q*	*Aktualisierung sichtbar in < 2 Sek.*	*4*	*3*
A5	*Inbox für unbearbeitete Dokumente*	*F*	*Dokumente im Status „offen“ werden korrekt angezeigt*	*3*	*4*
A6	*Archivierung/Löschung*	*A*	*Nachweispflicht: Historie bleibt nachvollziehbar*	*5*	*3*
A7	*Volltextsuche*	*F*	*Trefferquote mind. 99 %*	*5*	*5*
A8	*Dokumente teilen (Link/Einladungen)*	*Q*	*Externe Zugriffe funktionieren gemäß Testszenarien*	*4*	*3*
A9	*Kategorienverwaltung*	*A*	*Erstellung neuer Kategorien und korrekte Zuordnung*	*3*	*3*
A10	*Audit und Protokolle aller Aktionen*	*S*	*Vollständige Logs im Audit-Modul abrufbar*	*5*	*4*

1) Art = Anforderungsart: G = Geschäftsorganisation, F = Funktional, Q = Qualität, S = Sicherheit, M= Migration, A= Architektur, B = Betrieb, K = Konformität (Gesetzgebung, Weisungswesen, Normen und Richtlinien)
2) Wichtigkeit: 5 = muss zwingend umgesetzt werden; 4 = sehr wichtig, 3 = wichtig, 2 = normal, 1 = nicht wichtig
3) Dringlichkeit: 5 = muss sofort umgesetzt werden, 4 = sehr dringend, 3 = dringend, 2 = normal, 1 = nicht dringend

Tabelle 9: Grobanforderungen

Lösungsbeschreibung

Der Lösungsentwurf umfasst mehrere Kernkomponenten:

1. **Frontend**
   Eine responsive Webanwendung (möglich z. B. mit React, Vue oder Angular), die wichtige Funktionen wie Registrierung, Anmelden, Dokumentenübersicht und Suchfunktionen bereitstellt.
2. **Backend / API**
   Ein zentrales Framework (z. B. Node.js/Express, Django, Laravel) zur Bereitstellung von Endpunkten für Authentifizierung, Dokumentenmanagement und Sicherheitsprüfungen.
3. **Datenbank**
   Eine relationale Datenbank (MySQL, PostgreSQL) oder NoSQL (MongoDB), abhängig von den benötigten Metadatenstrukturen und Suchanforderungen.
4. **Cloud-Infrastruktur**
   Ein Hosting-Provider (z. B. AWS, Azure, GCP) zur Nutzung von IaaS- oder PaaS-Diensten. Containervirtualisierung mittels Docker. Eine CI/CD-Pipeline soll automatisierte Deployments ermöglichen.
5. **Sicherheitsmassnahmen**
   Einsatz von Verschlüsselung, Multifaktor-Authentifizierung sowie umfassende Audit-Funktionen.

Lösungsvarianten

Variantenübersicht

Variante	Bezeichnung
V1	*Komplett in der Cloud (IaaS + PaaS, Managed Services)*
V2	*Hybrider Ansatz (On-Premise + Cloudspeicher)*
V3	*Vollständig On-Premise*

Tabelle 10: Variantenübersicht

Variante V1

Kurzbeschreibung

Bei dieser Variante wird die gesamte Infrastruktur in der Cloud betrieben. Datenbank, Container-Orchestrierung und Storage werden als verwaltete Dienste genutzt. Dadurch reduziert sich der Aufwand für Hardwaremanagement, und Skalierbarkeit lässt sich leichter umsetzen.

Systemkontext (Soll)

Ein API Gateway in der Cloud empfängt Anfragen von extern und leitet sie an Microservices (Backend) weiter.

Eine verwaltete Datenbank kümmert sich um Datenspeicherung und Backup.

Kontextdiagramm (Soll)

- Nutzer (Web-Frontend),
- Backend (API-Services),
- Cloud-Datenbank und
- Externe Services (E-Mail, SMS, Monitoring).

Geschäftsorganisation

- Konsolidierte Prozesse für alle Abteilungen an einem Ort.
- Zentrales Rechte- und Rollenkonzept (Kategorien, Freigaben).

Produkt oder IT-System

- Architektur: Containerisierte Microservices (z. B. Docker, Docker Compose für Entwicklung und AWS ECS/EKS o. ä. in Produktion).
- Schnittstellen: Integration externer Dienste (Benachrichtigungen, Logging).

Informationssicherheit und Datenschutz

- TLS-Verschlüsselung für alle Anfragen
- Verschlüsselung der gespeicherten Daten (z. B. über KMS)
- Backups werden automatisiert in der Cloud erstellt und versioniert.

Voraussetzungen, Abhängigkeiten

- Vertrag mit Cloud-Provider
- Schulung des Projektteams hinsichtlich Cloud- und Sicherheitsthemen
- Anpassung interner Richtlinien, falls Daten ausserhalb des eigenen Rechenzentrums liegen

Variante V2: Hybrider Ansatz

Kurzbeschreibung
Teile der Infrastruktur bleiben On-Premise (z. B. Datenbank), während Services wie Applikationsserver oder Storage in der Cloud betrieben werden. Dies kann einerseits den Umstieg erleichtern, erfordert aber eine robuste VPN- oder Direct-Connect-Verbindung.

Systemkontext (Soll)

- Backend teilweise lokal gehostet, ggf. als VM oder Container-Cluster
- Frontend über Cloud-Services
- Datenbank On-Premise oder in der Cloud (je nach Sicherheitsvorgaben)
- Schnittstellen: VPN für Verbindung zwischen On-Premise-Netzwerk und Cloud

Geschäftsorganisation

- Mischbetrieb: Abteilungen, die kritische Daten nur lokal behalten wollen, profitieren von On-Premise
- Andere Abteilungen können Cloud-Dienste direkt nutzen

Produkt oder IT-System

- Architektur: Lokale Container + Cloud Container (z. B. Docker Swarm/ Kubernetes Hybrid)
- Wartungsaufwand: gesteigert, da mehrere Umgebungen integriert werden müssen

Informationssicherheit und Datenschutz

- Achtung auf Datenströme, die On-Premise-Datenschutzanforderungen vs. Cloud-Anforderungen erfüllen müssen
- Komplexere Audits, da Hybrid-Set-up

Voraussetzungen, Abhängigkeiten

- Sichere Verbindung (VPN oder Direct Connect)
- Zusätzliche Koordination zwischen Cloud-Team und On-Prem-Team

Variante V3: Vollständig On-Premise

Kurzbeschreibung
Alle relevanten Systeme (Server, Datenbanken, Container-Orchestrierung etc.) bleiben im firmeneigenen Rechenzentrum. Ggf. hoher Hardware- und Personalaufwand, aber volle Kontrolle über Hardware und Daten.

Systemkontext (Soll)

- Backend: On-Prem-Cluster / eigene VM-Infrastruktur
- Datenbank: Nur lokal, Backup-Strategie rein intern
- Anbindung externer Services wie E-Mail-/SMS-Dienste etc. über definierte Schnittstellen

Geschäftsorganisation

- Abteilungen greifen intern via Intranet oder per VPN zu
- Hoher interner IT-Betriebsaufwand

Produkt oder IT-System

- Container-/VM-Management durch das eigene IT-Team
- Vollständige Dokumentation der lokalen Infrastruktur

Informationssicherheit und Datenschutz

- Volle physische Kontrolle, jedoch auch volle Verantwortung (Zertifizierungen, Strom, Brandschutz, etc.)
- Regelmässige Audits, ggfs. ISO 27001 intern aufwendig umzusetzen

Voraussetzungen, Abhängigkeiten

- Sichergestellte Fachkompetenz im Bereich Serverinfrastruktur, Netzwerk, Security
- Höherer Kapazitätsbedarf (Strom, Kühlung, Platz)

Analyse und Bewertung der Varianten

Zielerreichung

Nr.	Beschreibung	Gewicht *	V1	V2	V3
1	*Zentrale Dokumentenverwaltung*	*Muss*	*Ja*	*Teilw.*	*Ja*
2	*Schnelle Volltextsuche*	*Soll*	*Ja*	*Ja*	*Ja*
3	*Geringe Anfangsinvestitionen*	*Kann*	*Mittel*	*Niedrig*	*Hoch*
**** Gewicht: M = Muss / S = Soll / K = Kann***					

Tabelle 11: Zielerreichungsgrad

Anforderungsabdeckung

ID	Anforderungsbeschreibung	Wichtigkeit *	V1	V2	V3
A1	*Registrierung*	*5*	*Ja*	*Ja*	*Ja*
A2	*Login (E-Mail/SMS)*	*5*	*Ja*	*Ja*	*Ja*
A7	*Volltextsuche*	*5*	*Ja*	*Teilw.*	*Ja*
A10	*Audit/Logging*	*5*	*Ja*	*Eingeschr.*	*Höherer Aufwand on-prem*
**** Wichtigkeit: 5 = muss zwingend umgesetzt werden; 4 = sehr wichtig, 3 = wichtig, 2 = normal, 1 = nicht wichtig***					

Tabelle 12: Anforderungsabdeckung

Weitere Kriterien

Kriterium	V1 (100 % Cloud)	V2 (Hybrid)	V3 (On-Premise)
Kosten	Mittlere Einstiegs- und laufende Kosten (verwalte Dienste)	Niedrige Investition initially, ggf. höhere Kosten später	Hohe Anfangsinvestition, laufende Wartung der Hardware
Skalierbarkeit	Automatisch, sehr hoch	Teilweise eingeschränkt durch lokale Komponenten	Begrenzt, muss Hardware erweitert werden
Abhängigkeit vom Anbieter	Hoch (Cloud-Provider)	Teilweise, auf Cloud- und eigene Systeme verteilt	Gering (alles in eigener Hand)
Risiken	Weniger Kontrolle über physische Infrastruktur	Komplexer Betrieb (Cloud + On-Prem)	Höheres Risiko bei Hardware-Ausfällen und Ressourcenengpässen
Nachhaltigkeit	Effiziente Ressourcennutzung beim Anbieter (Sharing)	Gemischt	Energiebedarf und Lebenszyklus des eigenen Rechenzentrums
Sicherheit	Anbieter-Zertifizierungen (ISO 27001 usw.)	Mögliche Lücken bei der Schnittstelle	Vollständig selbstverwaltet, hoher interner Aufwand

Gesetzliche Vorgaben	Je nach Standort des Rechenzentrums, GDPR/DSG-Klarheit nötig	Zusätzliche Abstimmungen für Hybridlösungen	Eigene Audits und Zertifizierungen erforderlich

Tabelle 13: Weitere Kriterien

Variantenwahl

Auf Basis der Bewertung wird Variante V1 (vollständiger Cloud-Einsatz) empfohlen. Die Vorteile liegen in der einfachen Skalierbarkeit, dem geringeren Wartungsaufwand für eigene Hardware und den meist integrierten Sicherheits- und Verwaltungsfunktionen. Zwar entsteht eine Abhängigkeit vom Cloud-Provider, doch die Kostenersparnisse sowie die schnelle Inbetriebnahme überwiegen.

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
eCH Standard	Eidgenössischer E-Government Standard
HERMES	Vorgehensmethodik für Projekte und Programme HERMES 5 ist ein eCH Standard
CI/CD	Kontinuierliche Integration und Auslieferung (Continuous Integration / Deployment)
Docker	Plattform zum Erstellen und Ausführen containerisierter Anwendungen
IaaS / PaaS	Infrastructure / Platform as a Service (Cloud-Dienstmodelle)
MFA	Mehrfaktor-Authentifizierung (z. B. Passwort + SMS)
GDPR/DSGVO	Europäische Datenschutz-Grundverordnung
HERMES	Vorgehensmethodik für Projekte und Programme in der Schweiz
ISO 27001	Internationaler Standard für Informationssicherheits-Management

Tabelle 14: Abkürzungen und Glossar

Programmmanagementplan

Projektname

Klassifizierung	VERTRAULICH
Status	in Arbeit
Programmnummer	
Programmleiter	Programmleiter
Version	0.1
Datum	19. Dezember 2024
Auftraggeber	Noah Lezama
Autor/Autoren	Noah Lezama
Verteiler	

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	19.12.2024	Start	Noah Lezama

Tabelle 15: Änderungskontrolle

Beschreibung

Der Programmmanagementplan beinhaltet die Programmstrategie, die Gesamtplanung des Programms und die Programmorganisation. Er zeigt die Querschnittsleistungen des Programms und die wesentlichen Regelungen zu Methoden, Techniken, Rollen und Hilfsmitteln, die programmspezifisch festgelegt werden. Der Programmmanagementplan dient als einheitliche Handlungsgrundlage für alle Programmbeteiligten. Er wird im Programmverlauf nach dem Prinzip der rollenden Planung und Steuerung kontinuierlich konkretisiert und nachgeführt. Bei Phasenabschluss eines Projekts wird der Programmmanagementplan den veränderten Bedingungen angepasst.

Programmbeschreibung

Kurzbeschreibung

Im Rahmen dieses Programms wird eine bereits existierende Webanwendung zur digitalen Dokumentenverwaltung konzeptionell analysiert und nach der Projektmanagementmethode HERMES dokumentiert. Zudem wird eine technische Vorlage (Template) erstellt, die als Ausgangspunkt für künftige (Weiter-)Entwicklungen dienen soll. Die eigentliche Implementierung (Neuentwicklung) findet in diesem Programmscope nicht statt; stattdessen konzentriert sich das Programm auf Planung, Architekturkonzeption, Test- und Risikomanagement sowie die Bereitstellung einer einsatzfähigen Projektvorlage (Frontend, Backend, Docker, CI/CD usw.).

Ausgangslage

Die Organisation nutzt bereits eine Webapplikation zur Dokumentenverwaltung, die allerdings funktionell evaluiert und dokumentiert werden muss.

Bisher existieren Insellösungen in verschiedenen Fachabteilungen mit unterschiedlichen Cloud- oder On-Premise-Ansätzen.

Es fehlt eine harmonisierte Architektur inklusive Testkonzept, Backup-Strategie und DevOps-Ansatz (z.B. CI/CD).

Das Programm knüpft an erste Vorarbeiten (z.B. grobe Analyse der bestehenden Anwendung, erste Cloud-Evaluierungen) an.

Vorarbeiten und bisher erbrachte Ergebnisse

Programminitialisierungsauftrag (Version 0.1) liegt vor, in dem die Ziele und Rahmenbedingungen für die Programminitialisierung definiert sind.

Programmrechtsgrundlagenanalyse: Erste Prüfung von DSGVO/DSG-Konformität und E-Government-Standards (eCH).

Programmstudie: Enthält eine erste Variantenanalyse (100 % Cloud, Hybrid, On-Premise) und empfiehlt einen Full-Cloud-Ansatz.

Gesamtplan

Phasen und Meilensteine

Das Programm wird nach HERMES-Methodik in Phasen unterteilt. Folgende Meilensteine wurden festgelegt:

1. Programminitialisierung
    - Start: 11.12.2024
    - Abschluss Programminitialisierung: 27.12.2024
2. Programmfreigabe
    - Ziel: Offizielle Freigabe und Budgetzusage für die nächste Phase
    - Datum: 12.01.2025 (geplant)
3. Durchführungsphase
    - Meilenstein: Erstellung der vollständigen Projektdokumentation, Infrastruktur- und Testkonzepte
    - Datum: 05.05.2025 (Zwischenpräsentation)
4. Zwischenevaluation
    - Datum: 29.07.2025 (Überprüfung der entstandenen technischen Vorlage)
5. Programmabschluss
    - Datum: 01.09.2025 (Übergabe der Dokumentation und der Vorlage)

Abb. 1 - Programm-Planung

Übersicht der Projekte

Zur Zielerreichung werden einzelne Themen in separaten Projekten/Arbeitspaketen organisiert.

Liste der Projekte

Projekt	Aktuelle Phase	Projektfreigabe	Projektabschluss	Szenario
Projekt A: Dokumentationsaufbau (Analyse & HERMES-Doku)	*Konzept / Realisierung*	*15.12.2024*	*17.02.2025*	*Wasserfall/Klassisch*
Projekt B: Infrastruktur & Cloud-Auswahl	*Konzept*	*24.01.2025*	*28.04.2025*	*HERMES (klassisch)*
Projekt C: Technische Projektvorlage	*Umsetzung*	*17.02.2025*	*29.07.2025*	*Agil (Scrum/Kanban)*

Tabelle 16: Liste der Projekte

Beschreibung der Projekte

Projekt A

Ziel: Vollständige HERMES-Dokumentation der bestehenden Webanwendung

Umfang: Analyse der Funktionen (z.B. Upload, Suche, Teilen), Erstellung von Projekt- & Testkonzept

Ergebnisse: Projektmanagement-Dokumente (Zeitplan, Statusberichte, Hermes-Phasenabschlussdokumente)

Projekt B

Ziel: Evaluierung, Auswahl und Kostenschätzung eines geeigneten Cloud-Providers (z.B. AWS, Azure oder GCP)

Umfang: Definition der Services (z.B. IaaS vs. PaaS), Preisvergleich und Sicherheitsanforderungen

Ergebnisse: Infrastrukturkonzept inkl. Betriebsmodell, Backup- und Wiederherstellungsstrategie

Projekt C (agil)

Ziel: Entwicklung einer Template-Lösung (Frontend- & Backend-Grundstruktur, Docker-Compose, CI/CD, Unit Testing)

Umfang: Implementierung eines minimalen, aber funktionsfähigen Prototyps, Demonstration der Verbindung Frontend–Backend

Ergebnisse: GitHub-Repository mit lauffähigem Docker-Setup und CI/CD-Pipeline

Programmorganisation

Rollenbesetzung

Rolle in der Stammorganisation	Name	Kürzel	Funktion / Vertretene Organisationseinheit
Leitung (Direktion)	*Patrick Michel*	*micp*	*Direktion IT*
Interne Revision	*Paul Hauser*	*hap*	
Kompetenzzentrum Hermes	*(n.n.)*	*kzh*	*Beratung & Coaching in Projektmanagement nach Hermes*

Tabelle 17: Rollenbesetzung Stammorganisation

Rolle in der Programmorganisation	Name	Kürzel	Funktion / Vertretene Organisationseinheit
Programmauftraggeber	*Noah Lezama*	*leno*	*Gesamtverantwortung, Freigaben*
Programmausschuss	*Patrick Michel*		
(evtl. andere Stakeholder)		*Strategische Leitung, Eskalation, Budgetkontrolle*	
Programmleiter	*Noah Lezama*	*lez*	*Operative Leitung, Koordination aller Projekte*
Programmkoordinator			*Unterstützung des Programmleiters, Termin- & Kostencontrolling*
Qualitäts- und Risikomanager			*Qualitätssicherung, Risikomanagement*
Programmunterstützung			*Administrative Arbeiten*

Tabelle 18: Rollenbesetzung Programm

Programmorganigramm

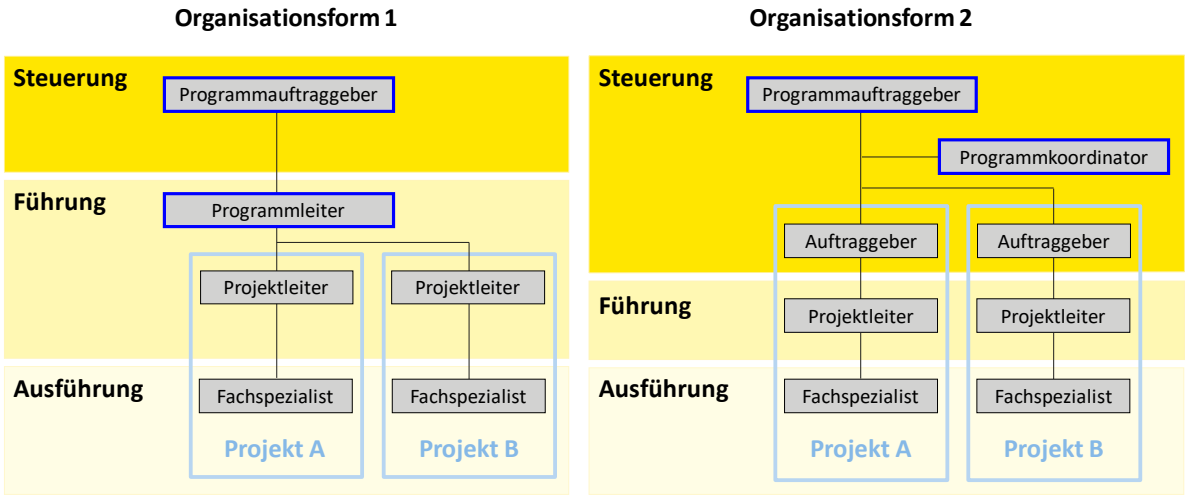

Abb. 2 - Programmorganisation

Eskalation

Wird ein Problem in einem der Projekte nicht innerhalb von 5 Tagen gelöst, eskaliert es an die nächste Ebene: zunächst an den Programmleiter, ggf. weiter an den Programmausschuss.

Bei Konflikten, die auf Programmebene nicht gelöst werden können, entscheidet letztlich der Programmauftraggeber.

Querschnittsleistungen des Programms

Leistungen die projektübergreifend durch das Programm erbracht werden.

Leistung / Aufgabe	Verantwortlich Programm	Prozess
Testmanagement (global)	*Programmkoordinator*	*Die Teilprojekte bestellen Testsupport je nach Bedarf*
Risikomanagement	*Qualitäts-/Risikomanager*	*Regelmässige Kontrolle und Berichterstattung an Programmleiter*
CI/CD-Standards	*Fachausschuss*	*Übergreifende Richtlinien, zentrale Build-Pipeline*

Tabelle 19: Leistungen des Programms

Vorgaben des Programms an die Projekte

Vorgabe	Verantwortlich Projekt	Prozess
Budgetprozess	*Programmkoordinator*	*Projekte melden Budgetbedarf -> Freigabe durch Programmausschuss*
Infrastruktur-Blueprint	*IT-Architekt (Fachausschuss)*	*Cloud-Services, Docker-Setup, Sicherheitsanforderungen*
Dokumentationsstandard	*Programmleiter*	*HERMES-konforme Dokumentation, Testkonzept, Statusberichte*

Tabelle 20: Vorgaben des Programms

Rollen und Kompetenzen

Entscheidungsprozesse Programm

Hierarchieebene Programm	Entscheid	Rolle
Steuerung	*Programminitialisierungsauftrag*	*Programmauftraggeber + Leitung*
Steuerung	*Programmfreigabe*	*Programmauftraggeber + Leitung*
Steuerung	*Strategieprüfung*	*Programmauftraggeber*
Steuerung	*Phasenfreigabe / Programmabschluss*	*Programmauftraggeber*
Steuerung	*Budgetgenehmigung über 10.000 CHF*	*Programmausschuss*

Tabelle 21: Entscheidungsprozesse auf Programmebene

Entscheidungsprozesse je Projekt

Hierarchieebene Pro Projekt	Entscheid	Rolle
Steuerung	*Projektinitialisierungsauftrag*	*Projektauftraggeber gemeinsam mit Programmauftraggeber*
Steuerung	*Projektfreigabe*	*Projektauftraggeber*
Steuerung	*Phasenfreigabe (Konzept, Realisierung ...)*	*Projektauftraggeber (ggf. Programmausschuss)*
Steuerung	*Betriebsaufnahme*	*Programmleiter / Fachbereich*
Steuerung	*Projektabschluss*	*Projektauftraggeber*

Tabelle 22: Entscheidungsprozesse je Projekt

Programm- und Projektergebnisstrukturplan

Eine mögliche hierarchische Gliederung (Beispiel in Mindmap-Form):

1. Programmleitung und -steuerung
    - Programminitialisierung
    - Budgetbewilligung
    - Gesamtabschluss
2. Teilprojekt A: Dokumentationsaufbau
    - Analyse der bestehenden App
    - Zeitplan, Arbeitsjournal, Statusberichte
    - Testkonzept & Testfällen
3. Teilprojekt B: Infrastruktur & Cloud
    - Cloud-Provider-Auswahl
    - Kosten- & Betriebsmodell (IaaS vs. PaaS)
4. Teilprojekt C: Technische Vorlage
    - Frontend/Backend-Grundstruktur
    - Docker-Setup (Docker Compose, Dockerfile)
    - CI/CD-Pipeline
5. Qualität und Risikomanagement (übergreifend)
6. Kommunikation & Reporting (übergreifend)

Abb. 3 - Beispiel eines Programmergebnisstrukturplans

Abhängigkeiten

Abhängigkeiten Programm

Vorhaben	Abhängigkeit	Termin	Auswirkungen auf das Programm	Ansprechperson
Cloud-Provider-Vertrag	Muss vor Aufbau der Infrastruktur unterschrieben sein	24.01.2025	Verzögerung gefährdet Programmfortschritt	Programmleiter
Beschaffung Lizenzen (z.B. Tools)	Muss rechtzeitig über Einkauf laufen	10.02.2025	Verzögert CI/CD-Aufbau	Programmkoordinator

Tabelle 23: Abhängigkeiten der Programmvorhaben

Abhängigkeiten der Projekte

PROJEKT	ABHÄNGIGKEIT	TERMIN	AUSWIRKUNGEN AUF DAS PROGRAMM	ANSPRECHPERSON
PROJEKT A	Hermes-Schulung für Team	14.12.2024	Verlangsamt Dokumentationsstart	Projektleiter (A)
PROJEKT B	Projekt A liefert Anforderungskatalog	15.01.2025	Ohne Anforderungskatalog keine Cloud-Wahl	Projektleiter (B)
PROJEKT C	Finaler Cloud-Anbieter aus Projekt B	24.01.2025	Docker-Setup abhängig von gewählter Infrastruktur	Projektleiter (C)

Tabelle 24: Abhängigkeiten der Projekte im Programm

Prüfplan

Prüfplan für Ergebnisse des Programms

Phase / Ergebnis	Prüfmethode	Verantwortlich	Prüfer	Termin	Status
Initialisierung	Walk-Through (WT)	Programmleiter	Fachausschuss	27.12.2024	in Prüfung
Programminitialisierungsauftrag	*Review (SR + MR)*	*Auftraggeber*	*Patrick Michel*	*12.01.2025*	*ausstehend*
Programmmanagementplan (dieses Dok.)	*Schriftl. Review*	*Programmleiter*	*Fachausschuss*	*12.01.2025*	*Entwurf*
Durchführung					
Zwischenevaluationsbericht	Review (MR)	Programmleiter	Programmausschuss	29.07.2025	geplant
QS- und Risikobericht	*Schriftl. Review*	*Risikomanager*	*Programmleiter*	*laufend*	*wiederkehrend*
Abschluss					
Programmabschlussbericht	*Review (SR)*	*Auftraggeber*	*Programmausschuss*	*01.09.2025*	*geplant*

Tabelle 25: Prüfplan

Vorgaben zu Prüfplänen der Projekte

Phase / Ergebnis	Prüfmethode	Verantwortlich	Prüfer
Projektinitialisierungsauftrag	Schriftl. Review	Projektleiter	Programmleiter
Projektauftrag	*Review (MR)*	*Projektleiter*	*Programmleiter*
Konzeptphasenbericht	*Schriftl. Review*	*Projektleiter*	*Programmausschuss*
Realisierungsphasenbericht	Walk-Through (WT)	Projektleiter	Programmleiter
Projektabschlussbericht	*Review (SR)*	*Projektleiter*	*Programmauftraggeber*

Tabelle 26: Diese Projektergebnisse müssen abgenommen werden.

Terminplan

Meilensteine und Termine Programm

Meilenstein	Datum Plan	Datum ist
Programminitialisierungsauftrag	*11.12.2024*	*12.12.2024*
Programmfreigabe	*12.01.2025*	
Zwischenevaluation	*29.07.2025*	
Phasenfreigabe Programmabschluss	*01.09.2025*	
Programmabschluss	*01.09.2025*	

Tabelle 27: Meilensteine und Termine Programm

Meilensteine und Termine Projekt xx

Meilenstein	Datum Plan	Datum ist
Projektinitialisierungsauftrag	*15.12.2024*	
Variantenwahl / Konzept	*30.12.2024*	
Phasenfreigabe Realisierung	*17.02.2025*	
Projektabschluss	*01.03.2025*	

Tabelle 28: Meilensteine und Termine Projekt xx

Kostenplan

Die Kosten werden im Rahmen der Programmdokumentation in einem separaten Budgetplan geführt.

Phase	Plan-Budget (CHF)
Programminitialisierung	10’000
Durchführung	30’000
Abschluss	5’000
Summe	45’000

Ressourcenplan

Ressourcenplan Programm

Personalressourcen Programmsteuerung und Programmführung

Gemäss den Vorgaben der Stammorganisation.

Rolle / Person	Monat 1	Monat 2	Monat 3	Monat 4	Monat 5	Total	Bestätigung Vorgesetzter
Programmleiter	*100%*	*100%*	*100%*	*100%*	*100%*	*5 Monate*	*Leitung (micp)*
Programmkoordinator		*50%*	*50%*	*50%*	*50%*	*4 Monate*	
Risikomanager		*20%*	*20%*	*20%*	*20%*	*4 Monate*	

Tabelle 29: Personalressourcen Programm

Sachmittel Programmsteuerung und Programmführung

Räume: 1 Projektraum für reguläre Teammeetings (inkl. Whiteboard)

IT-Infrastruktur: Zugriff auf PM-Tools, Cloud-Testaccounts, Lizenzen für Dokumentationssoftware

Ressourcenplan Projekt A

Personalressourcen Projekt A

Rolle / Person	Monat 1	Monat 2	Monat 3	Total	Bestätigung Vorgesetzter
Projektleiter (A)	*50%*	*100%*	*50%*	*2,5 Monate*	
Technischer Analyst	*80%*	*80%*		*1,5 Monate*	
Tester (QA)		*40%*	*40%*	*1 Monat*	

Tabelle 30: Personalressourcen Projekt xx

Sachmittel Sachmittel Projekt

Testumgebung: Dedizierte Umgebung für Dokumentations- und Funktionstests

Tools: (z.B. JIRA, Confluence)

Beschaffungsplan

Beschaffung für Programm

Bedarf / Bezeichnung	Menge	Wert CHF	Zeitpunkt	Beschaffungsart
Lizenzen für PM-Software	*10*	*2'000*	*01.2025*	*Freihändige Vergabe nach Offerten*
Cloud Service (Test- & PoC-Phase)		*3'000*	*01–03.2025*	*Monatsweise, Pay-as-you-go*

Tabelle 31: Beschaffung für Programm

Koordinierte Beschaffungen für Projekte

Folgende Beschaffungen werden durch die Programmführung koordiniert:

Bedarf / Bezeichnung	Menge	Wert CHF	Zeitpunkt	Beschaffungsart
Entwicklerkapazität (externe)	*100 PT*	*80'000*	*02–07.2025*	*Ausschreibung (kurzes Verfahren)*
CI/CD-Pipeline-Setupsupport	*20 PT*	*16'000*	*Nach Projekt B-Freigabe*	*Ggf. Abruf auf Rahmenvertrag*

Tabelle 32: Beschaffungsplan

Kommunikationsplan

Stakeholder orientierte Kommunikation

Adressat der Information	Kommunikationsverantw.	Inhalt	Ziel	Mittel / Medium	Termin
Abteilungsleiter (alle)	Programmleiter (Noah Lezama)	Projektstatus, Meilensteine, Risiken	*Transparenz, Commitment*	*Regel-Meeting*	*monatlich*
Fachabteilungen	*Programmleiter / Koordinator*	*Ergebnisse Cloud-Auswahl, Testkonzept*	*Wissenstand, Feedback einholen*	*E-Mail/Portal*	*ad hoc*
IT-Security-Team	*Risikomanager*	*Sicherheits- und Datenschutzthemen*	*Abstimmung, rasche Gegenmassnahmen*	*Meeting/ Report*	*bei Bedarf*

Tabelle 33: Kommunikationsplan
Tabelle 34: Kommunikation mit Stakeholdern

Kommunikation im Programm (Meetings)

Meeting	Häufigkeit	Protokoll
Programmausschusssitzung	*vierteljährlich*	*SharePoint/PM-Tool*
Arbeitsmeeting Teilprojekt A	*wöchentlich*	*Protokoll in Confluence*
Statusmeeting Programmleitung	*zweiwöchentlich*	*MS Teams / Notizen*

Tabelle 35: Kommunikation im Programm (Meetings)

Reporting

Ergebnis	Periodizität	Verantwortlich	Empfänger	Termin
Programmstatusbericht	*monatlich*	*Programmleiter*	*Programmauftraggeber*	*jeweils zum Monatsende*
QS- und Risikobericht	*quartalsweise*	*Risikomanager*	*Programmauftraggeber, Programmausschuss*	*jeweils Quartalsende*
Projektstatusberichte (A,B,C)	*monatlich*	*Projektleiter (A,B,C)*	*Programmleiter*	*jeweils 5. Werktag / Monat*
Phasenberichte	*nach Bedarf*	*Projektleiter (je Teilprojekt)*	*Programmleiter, Programmausschuss*	*nach Phasenabschluss*

Tabelle 36: Berichte

Vorgaben, Methoden und Werkzeuge

Titel	Vorgaben, Methoden und Werkzeuge	Version
Programmmanagement	*HERMES*	*5.2*
Projektmanagement	*HERMES, interne PM-Richtlinien*	*2024*
Beschaffung	*BöB, VöB*	
Software-Entwicklung	*Git, Linter, Formatter, Docker, CI/CD*	*jeweils aktuell*
Testmanagement	*Testkonzept nach ISTQB-Grundlagen, Hermes-Tests*	*2024*

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
DSG/DSGVO	Datenschutzgesetz (CH) / Datenschutz-Grundverordnung (EU)
eCH-Standard	Eidgenössischer E-Government-Standard
HERMES	Vorgehensmethodik für Projekte/Programme in der Schweiz
CI/CD	Continuous Integration / Continuous Deployment (bzw. Delivery)
IaaS / PaaS	Infrastructure / Platform as a Service
Docker	Plattform zum Erstellen und Ausführen containerisierter Anwendungen
QA	Quality Assurance (Qualitätssicherung)
PT	Personentage

Tabelle 37: Abkürzungen und Glossar

Anforderungsanalyse

Klassifizierung	intern
Status	in Arbeit
Programmnummer	1
Programmleiter	Programmleiter
Version	0.1
Datum	8. Januar 2025
Auftraggeber	Noah Lezama
Autor/Autoren	Noah Lezama
Verteiler	

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	08.01.2025	Start	Noah Lezama

Tabelle 38: Änderungskontrolle

Beschreibung

Dieses Dokument beschreibt die bestehenden Funktionen sowie die funktionalen und nicht-funktionalen Anforderungen der digitalen Dokumentenverwaltungs-Applikation. Darüber hinaus werden die relevanten Stakeholder (Interessengruppen) und deren Erwartungen skizziert.

1. Dokumentation der bestehenden Funktionen

Die derzeit verfügbare Applikation ermöglicht es Anwendern, Dokumente digital zu verwalten. Die wichtigsten Beobachtungen und Funktionen sind:

Benutzerregistrierung

- Neue Benutzer können sich registrieren, indem sie grundlegende Daten (z. B. Name, E-Mail, evtl. Telefonnummer) angeben.
- Aktuelle Einschränkung: Unklar ist, ob eine Verifizierung (z. B. Double Opt-in) per E-Mail oder SMS stattfindet.

Login mit E-Mail/SMS

- Die Anmeldung erfolgt über Benutzername/E-Mail und Passwort; optional kann ein Code per SMS angefordert werden.
- Einschränkung: Unklar, ob hier ein echtes Mehr-Faktor-Authentifizierungsverfahren (MFA) obligatorisch ist.

Dokument-Upload

- Nutzer können Dateien (z. B. PDF, Bilder etc.) hochladen.
- Einschränkungen:
  - Keine klaren Informationen zur maximalen Dateigrösse.
  - Metadaten-Unterstützung scheint eher rudimentär zu sein (keine komplexen Tagging- oder Kategorisierungsfunktionen ersichtlich).

Bearbeitung von Metadaten / Dokumentinformationen

- Nutzer haben die Möglichkeit, Titel, Beschreibung, Kategorie usw. anzupassen.
- Einschränkung: Keine Versionshistorie für Metadaten-Änderungen oder Dokumentation dieser Änderungen im System.

„Inbox“-Funktion und Aufgabenverwaltung

- Die Benutzeroberfläche zeigt Dokumente, die auf eine bestimmte Aktion warten (z. B. Freigabe, Bearbeitung).
- Einschränkung: Unklar ist, ob es einen echten Workflow (z. B. mehrstufige Genehmigung) gibt.

Archivierung und Löschung

- Dokumente können aus der aktiven Ansicht entfernt (archiviert) oder endgültig gelöscht werden.
- Einschränkung: Es ist nicht ersichtlich, ob ein Aufbewahrungszeitraum eingehalten werden muss bzw. ob ein „Papierkorb“ zur Wiederherstellung existiert.

Dokumenten-Suche (Volltextsuche)

- Suchfunktion sowohl nach Metadaten als auch (vermutlich) nach Volltext-Inhalten.
- Einschränkung: Unklar, wie performant die Suche bei sehr grossen Dokumentenmengen ist. Kein Hinweis auf fortgeschrittene Filteroptionen (z. B. nach Datum, Eigentümer, Status).

Teilen von Dokumenten

- Dokumente können via Link oder per Einladung mit anderen Nutzern geteilt werden.
- Einschränkung: Fehlende Optionen für detaillierte Berechtigungen (z. B. nur Lesen, Lesen/Schreiben).

Stammdaten / Kategorien

- Nutzer können Kategorien erstellen und Dokumente zuordnen.
- Einschränkung: Es ist nicht klar, ob eine Kategorisierung mit Hierarchien oder Verschachtelungen möglich ist.

Insgesamt bietet die bestehende Anwendung ein Grundgerüst für ein Dokumentenmanagement, weist aber Lücken bei Auditing, Sicherheit (z. B. MFA), Workflows (Freigabeprozesse) und Versionierung auf.

2. Funktionale Anforderungen

Im Folgenden sind die funktionalen Anforderungen aufgeführt, die aus der Analyse und den übergeordneten Zielen (gemäss HERMES-Methodik) hervorgehen. Sie dienen als Grundlage für eine (Neu-)Konzeption oder Weiterentwicklung der Anwendung.

Benutzerverwaltung

- Das System muss eine Registrierung mit Bestätigungsmechanismus (z. B. Double Opt-in via E-Mail oder SMS) ermöglichen.

- Das System muss eine Rollenverwaltung (z. B. Standardnutzer, Administratoren) unterstützen, um unterschiedliche Zugriffsrechte zu vergeben.

Sichere Authentifizierung und Autorisierung

- Das System muss einen Login-Prozess mit E-Mail/Benutzernamen und Passwort ermöglichen; idealerweise wird ein zweiter Faktor (MFA) unterstützt.
- Das System muss Zugriffsrechte anhand von Rollen und Berechtigungen verwalten (z. B. Dokumentzugriff nur für bestimmte Gruppen).

Dokument-Upload und -Verwaltung

- Das System muss das Hochladen verschiedener Dateiformate (PDF, DOCX, Bilder etc.) bis zu einer festgelegten Maximalgrösse erlauben.
- Das System muss Metadaten verwalten und aktualisieren (Titel, Beschreibung, Kategorie, Eigentümer, Datum).
- Eine einfache Versionierung oder zumindest eine Änderungsprotokollierung soll für Dokumente vorhanden sein.

Workflows und „Inbox“-Funktion

- Das System muss Dokumente mit einem definierten Status (z. B. „zu prüfen“, „in Bearbeitung“) kennzeichnen und in einer Übersichtsseite („Inbox“) anzeigen.
- Das System kann einen mehrstufigen Freigabeprozess (z. B. 2-stufige Genehmigung) unterstützen, sofern dies projektspezifisch erforderlich ist.

Such- und Filterfunktionen

- Das System muss eine Volltextsuche im Dokumenteninhalt sowie in den Metadaten anbieten.
- Das System muss Filter nach Kategorien, Datum, Eigentümer und Status ermöglichen.

Archivierung und Löschung

- Das System muss Dokumente wahlweise archivieren (aus dem aktiven Bestand entfernen) oder endgültig löschen können.
- Ein optionaler Zwischenschritt (Papierkorb), in die Dokumente erst nach einer definierten Zeit endgültig gelöscht werden, kann implementiert werden.

Teilen von Dokumenten

- Das System muss die Möglichkeit bieten, Dokumente über Links oder Einladungen an andere Nutzer/Externe freizugeben.
- Die vergebenen Berechtigungen (Lesen/Schreiben) müssen klar definier- und nachverfolgbar sein.

Kategorien / Stammdaten

- Das System muss die Anlage, Bearbeitung und Löschung von Kategorien ermöglichen.
- Eine Hierarchisierung (Unterkategorien) kann vorgesehen werden.

Auditing und Versionierung

- Das System muss alle kritischen Aktionen (z. B. Hochladen, Ändern, Löschen, Teilen) protokollieren und in einem Auditing-Log festhalten.
- Das System muss Dokumentversionen verwalten oder zumindest Änderungen an Metadaten nachvollziehbar machen.

Vier-Augen-Prinzip (4-Augen-Prinzip)

- Bestimmte kritische Aktionen (z. B. endgültiges Löschen von Dokumenten, Änderungen an globalen Einstellungen) müssen eine zweite Freigabe (durch einen berechtigten Nutzer) erfordern.

3. Nicht-funktionale Anforderungen

Diese Anforderungen betreffen Aspekte wie Leistung, Sicherheit, Wartbarkeit und Usability. Sie ergänzen die funktionalen Ziele und müssen bei der Umsetzung berücksichtigt werden.

Leistung und Skalierbarkeit

- Das System soll bei mindestens 100 gleichzeitigen Zugriffen noch Reaktionszeiten unter 3 Sekunden für Standardfunktionen (z. B. Suchen, Upload) bieten.
- Es muss eine Architektur vorgesehen werden, die bei Bedarf horizontal skaliert werden kann (z. B. Cloud-Container, Load Balancing).

Sicherheit

- Alle Kommunikation muss über verschlüsselte Verbindungen (HTTPS/TLS) erfolgen.
- Die Speicherung von Dokumenten soll verschlüsselt erfolgen (Encryption at rest), ebenso das Anlegen von Backups.

- Das System muss die gesetzlichen Datenschutzbestimmungen (z. B. DSG/DSGVO) einhalten.
- Ein Audit-Log muss existieren, um sicherheitsrelevante Aktionen (z. B. Login, Löschungen, Änderungen) rückverfolgbar zu machen.
- Kritische Eingriffe erfordern das Vier-Augen-Prinzip, um Fehlbedienung oder Missbrauch zu minimieren.

Verfügbarkeit und Ausfallsicherheit

- Angestrebte Verfügbarkeit: mindestens 99,5 % während der regulären Betriebszeiten.
- Das System soll über ein Backup-und-Wiederherstellungs-Konzept verfügen, das regelmässig getestet wird (z. B. monatliches Restore-Testen).

Wartbarkeit und Erweiterbarkeit

- Der Code soll modular aufgebaut sein (Trennung von Frontend, Backend, Datenbank).
- Versionierung und CI/CD-Pipelines müssen vorhanden sein, um regelmässige, stabile Releases zu ermöglichen.
- Die Dokumentation der Architektur und Installationsprozesse soll zentral gepflegt werden (z. B. Wiki, Git-Repository).

Benutzerfreundlichkeit und Barrierefreiheit

- Das User Interface soll intuitiv sein, gängige Browser (Chrome, Firefox, Safari, Edge) und verschiedene Bildschirmgrössen (responsive Design) unterstützen.
- Wenn möglich, soll eine grundlegende Barrierefreiheit (z. B. WCAG-Standards) berücksichtigt werden.

Überwachung und Protokollierung

- Das System muss relevante Metriken und Logs (CPU-Auslastung, Speichernutzung, Fehlerlogs etc.) erfassen, um bei Problemen schnell reagieren zu können.
- Ein Monitoring-Tool (z. B. Prometheus, ELK-Stack) kann integriert werden, um Warnmeldungen (Alerts) für sicherheits- oder performancekritische Ereignisse auszugeben.

Risikoanalyse

Klassifizierung	VERTRAULICH
Status	in Arbeit
Programmnummer	1
Programmleiter	Programmleiter
Version	0.1
Datum	8. Januar 2025
Auftraggeber	Noah Lezama
Autor/Autoren	Noah Lezama
Verteiler	

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	08.01.2025	Start	Noah Lezama

Tabelle 39: Änderungskontrolle

Beschreibung

Dieses Dokument beschreibt die relevanten Projektrisiken im Rahmen der konzeptionellen Neuentwicklung bzw. Weiterentwicklung einer Webapplikation zur digitalen Dokumentenverwaltung. Für jedes Risiko werden Wahrscheinlichkeits- und Auswirkungsstufen definiert (hoch, mittel, gering) sowie entsprechende Massnahmen zur Minderung (Mitigation) aufgezeigt.

1 Identifikation der Risiken

Zeitmangel / Personelle Engpässe

- Beschreibung: Es besteht das Risiko, dass nicht genügend Ressourcen (Personal, Zeit) zur Verfügung stehen, um alle Phasen (Analyse, Architektur, Tests) termingerecht abzuschliessen.

Technische Kompatibilitätsprobleme

- Beschreibung: Unterschiedliche Betriebssysteme, Browser oder Framework-Versionen könnten zu Integrationsproblemen führen (z. B. Backend/Frontend nicht kompatibel, Cloud-Dienste nicht unterstützt).

Unsicherheit bezüglich Cloud-Anbieter

- Beschreibung: Eine falsche Wahl des Cloud-Providers (z. B. unzureichende Zertifizierungen, Kostenfalle bei Skalierung, fehlende Datenschutzkonformität) könnte den Betrieb und die Sicherheit des Systems gefährden.

Mängel bei Sicherheit und Datenschutz

- Beschreibung: Unzureichende Umsetzung von Verschlüsselung, 4-Augen-Prinzip oder DSG/DSGVO-Anforderungen kann zu Datenschutzverletzungen führen, mit rechtlichen und reputativen Folgen.

Unklare Anforderungen / Scope Creep

- Beschreibung: Fehlende oder unvollständige Anforderungen können zu Scope Creep führen (ständige Erweiterung des Projektumfangs), was Zeit- und Budgetüberschreitungen nach sich zieht.

2 Bewertung nach Wahrscheinlichkeit und Auswirkung

In der folgenden Tabelle werden die Risiken hinsichtlich ihrer Wahrscheinlichkeit (W) und Auswirkung (A) in hoch (H), mittel (M) oder gering (G) eingestuft.

Risiko	Wahrscheinlichkeit (W)	Auswirkung (A)	Kurzbegründung
1. Zeitmangel / Personelle Engpässe	M	H (hoch)	Engpässe in kritischen Phasen verzögern Meilensteine erheblich; schwerwiegender Effekt auf Termin.
2. Technische Kompatibilitätsprobleme	M	M	Diverse Technologien (Cloud, On-Prem) und Browser können Integrationen erschweren, aber meist lösbar.
3. Unsicherheit Cloud-Anbieter	M	H	Falsche Provider-Wahl kann erhebliche Kosten/Nachbesserungen nach sich ziehen (SLA, Compliance).
4. Mängel bei Sicherheit/Datenschutz	G	H	Eher geringe Eintrittswahrscheinlichkeit, aber sehr hoher Schaden (Reputation, rechtliche Folgen).
5. Unklare Anforderungen / Scope Creep	H	M	Häufige Anforderungsänderungen können Projektbudgets und Timelines stark beeinflussen.

3 Massnahmenplan (Mitigation / Vermeidung)

Risiko 1: Zeitmangel / Personelle Engpässe

- Massnahme:
    - Frühzeitige Planung und detaillierter Projektplan mit Pufferzeiten.
    - Eventuell externe Ressourcen für Spitzenlasten einplanen (Freelancer, zusätzliche Entwickler).
    - Klare Priorisierung der Tasks (Minimum Viable Product).

Risiko 2: Technische Kompatibilitätsprobleme

- Massnahme:
    - Durchführung von Proof-of-Concepts (PoC) frühzeitig im Projekt, um kritische Technologien zu evaluieren.
    - Einsatz standardisierter Frameworks bzw. Versionen (z. B. Docker-Container, CI/CD), die Abhängigkeiten minimieren.
    - Kontinuierliche Integrationstests (automatisiert), die relevante Systemumgebungen abdecken.

Risiko 3: Unsicherheit bzgl. Cloud-Anbieter

- Massnahme:
    - Erstellung einer klaren Evaluationsmatrix (Kosten, Datenschutz, Zertifizierungen).
    - Testphase (Pilot) mit einer kleineren Instanz, bevor produktive Daten migriert werden.
    - Vertragliche Flexibilität (Optionen für Exit-Strategie / Multi-Cloud).

Risiko 4: Mängel bei Sicherheit und Datenschutz

- Massnahme:
    - Implementierung eines sicheren Konzepts (TLS, Verschlüsselung, 4-Augen-Prinzip bei Löschungen/Änderungen).
    - Regelmässige Audits und Penetrationstests.
    - Schulung aller Beteiligten in Sachen Sicherheit und Datenschutz (Sensibilisierung).

-

Risiko 5: Unklare Anforderungen / Scope Creep

- Massnahme:
  - Klare Dokumentation der Anforderungen (Lasten-/Pflichtenheft) und Freigabe durch Stakeholder.
  - Einführung eines formalen Änderungsmanagement-Prozesses (Change-Requests nur mit Genehmigung).
  - Regelmässiger Abgleich zwischen Projektleitung und Fachabteilungen (z. B. in Review-Meetings).

Evaluation

Klassifizierung	VERTRAULICH
Status	in Arbeit
Programmnummer	1
Programmleiter	Programmleiter
Version	0.1
Datum	10. Januar 2025
Auftraggeber	Noah Lezama
Autor/Autoren	Noah Lezama
Verteiler	

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	10.01.2025	Start	Noah Lezama

Tabelle 40: Änderungskontrolle

Beschreibung

In diesem Kapitel werden die bisherigen Ergebnisse und Dokumente des Projekts bewertet. Dabei wird geprüft, ob die definierten Ziele erreicht, ob wesentliche Anforderungen erfüllt und ob die Risiken angemessen behandelt worden sind. Zusätzlich werden allfällige Verbesserungspotenziale und Folgeaktivitäten aufgezeigt.

1. Zielsetzung der Evaluation

Die Evaluation dient dazu:

1. Den aktuellen Stand im Vergleich zu den ursprünglichen Zielen und Meilensteinen zu überprüfen.
2. Abweichungen zu identifizieren und Massnahmen zur Korrektur oder Optimierung abzuleiten.
3. Die Wirksamkeit der bisherigen Vorgehensweise (HERMES-konform) und der getroffenen Entscheidungen zu bewerten.
4. Grundlagen für weitere Entscheide (z. B. Programmfreigabe, Änderungsmanagement) zu schaffen.

2. Überblick über den Dokumentationsstand

Im Projekt wurden bislang folgende Dokumente erstellt bzw. aktualisiert:

- Programminitialisierungsauftrag (Version 0.1)
- Programmrechtsgrundlagenanalyse
- Programmstudie (Variantenvergleich, Empfehlung für Full-Cloud-Ansatz)
- Programmmanagementplan
- Anforderungsanalyse (inkl. funktionaler und nicht-funktionaler Anforderungen)
- Risikoanalyse

2.1 Erfüllungsgrad der bisherigen Ziele

Die wichtigsten Ziele der Programminitialisierung laut Dokumentation:

Ziel	Status / Erfüllungsgrad
Vollständige HERMES-Dokumentation	In Arbeit – Diverse HERMES-Dokumente liegen vor, einzelne Teile (z. B. Phasenabschlussberichte) noch ausstehend.
Infrastruktur-/Architekturkonzept	In Arbeit – Variantenvergleich abgeschlossen, Tendenz zu Full-Cloud-Lösung. Detailliertes Architektur-Dokument muss folgen.
Testkonzept (erstellt / geplant)	Grundlagen definiert – Globale Teststrategie skizziert, Detailkonzept noch zu erstellen.
Technische Vorlage (Prototyp)	Noch nicht umgesetzt – Konzeption der Vorlage ist auf Projekt C (Umsetzung) verlagert.

3. Bewertung der Ergebnisse und Dokumente

1. Programminitialisierungsauftrag
    - Positiv: Klare Zielformulierung und Prioritäten (Muss/Soll/Kann).
    - Verbesserung: Noch stärkerer Zeitplan-Abgleich (Soll-/Ist-Zeiten für jeden Meilenstein).
2. Programmrechtsgrundlagenanalyse
    - Positiv: Übersicht über DSGVO/DSG und eCH-Standards, erste Identifikation von Compliance-Lücken.
    - Verbesserung: Detailtiefe hinsichtlich konkreter Umsetzung (Konsequenzen für Datenhaltung, Cloud-Standort) könnte ausgebaut werden.
3. Programmstudie
    - Positiv: Detaillierte Variantenanalyse (Cloud vs. Hybrid vs. On-Prem), fundierte Empfehlung für Full-Cloud.
    - Verbesserung: Kosten-/Risikoaspekte müssten im weiteren Verlauf noch konkreter verifiziert werden (Proof of Concept).
4. Programmmanagementplan
    - Positiv: Guter Überblick über Rollen, Meilensteine, Abhängigkeiten, Kommunikations- und Prüfplan.

- Verbesserung: Detailplanung für Projekt B (Cloud-Auswahl) und Projekt C (Vorlagenentwicklung) sollte genauer terminiert werden (Ressourcen, Deadlines).

5. Anforderungsanalyse
    - Positiv: Klare Strukturierung in funktionale und nicht-funktionale Anforderungen, Einbindung des 4-Augen-Prinzips, Sicherheitsaspekte.
    - Verbesserung: Priorisierung könnte noch weiter verfeinert werden (z. B. genaue Deadlines / Milestones pro Anforderung).
6. Risikoanalyse
    - Positiv: Relevante Projektrisiken (Zeit, Cloud-Anbieter, Sicherheit etc.) sind identifiziert und bewertet.
    - Verbesserung: Laufendes Monitoring bzw. Aktualisierung des Risikostatus und der Risikomatrix in Statusmeetings sind sicherzustellen.

4. Abweichungen und Verbesserungsansätze

4.1 Zeitliche Abweichungen

- Geplante vs. tatsächliche Termine: Die meisten erstellten Dokumente wurden im Zeitrahmen fertiggestellt. Kleinere Verzögerungen sind beim Testkonzept und bei der detaillierten Infrastrukturplanung zu verzeichnen.
- Empfehlung: Eine Nachsteuerung mit aktualisiertem Terminplan (inkl. Puffer) wird vorgeschlagen.

4.2 Inhaltliche Abweichungen

- Konkretisierung des Cloud-Betriebs: Die Empfehlung für einen Full-Cloud-Ansatz liegt vor, doch detaillierte Serviceauswahl (AWS, Azure, GCP o. Ä.) und technische Proof-of-Concepts fehlen noch.
- Empfehlung: Vorgezogene Machbarkeitsprüfung (PoC) als eigene Aktivität in Projekt B definieren.

4.3 Qualität der Dokumente

- Die vorhandenen Dokumente entsprechen mehrheitlich den HERMES-Vorgaben. Teilweise könnten jedoch Checklisten (z. B. für Compliance) und Prüfpunkte (z. B. Testkonzept-Details) noch vertieft werden.

5. Stakeholder-Zufriedenheit

- Fachabteilungen: Haben laut Projektstatusbericht erste Rückmeldungen zur Anforderungsanalyse gegeben, sind mehrheitlich zufrieden, wünschen sich aber klare Roadmap zur Umsetzung.
- IT-Architekt / Entwicklungsteam: Positive Rückmeldung zur klaren Struktur (Docker, CI/CD), allerdings mehr Detailbedarf zur Cloud-Konfiguration.
- Management: Erwartet weitere Kostenschätzung und Timing, besonders für die Realisierungsphase (Projekt C).

6. Ausblick und nächste Schritte

1. Vertiefung des Testkonzepts
    - Detailausarbeitung von Teststufen (Unit, Integration, E2E), Testdaten-Strategie und konkreter Testfallkatalog (mind. 10 Fälle).
2. Detaillierte Infrastruktur- und Cloud-Planung
    - Evaluationsmatrix: Anbieter, Zertifizierungen, Kostenmodell, Migrationsaufwand.
    - Erarbeiten eines PoC-Ansatzes (kleiner Testlauf).
3. Vorlage (Prototyp) vorbereiten
    - Strukturprojekt aufsetzen (Docker-Compose, Boilerplate-Frontend/Backend).
    - Dokumentation in Git-Repository, inkl. Linter/Formatter/Testing-Framework.
4. Fortlaufende Risikoüberwachung
    - Aktualisierung der Risikomatrix pro Statusmeeting.
    - Mögliche neue Risiken: Change Requests (Scope-Erweiterungen) oder Personalausfall.

Programmarbeitsauftrag

Klassifizierung	VERTRAULICH
Status	in Arbeit
Programmnummer	1
Programmleiter	Programmleiter
Version	0.1
Datum	10. Januar 2025
Auftraggeber	Auftraggeber
Autor/Autoren	Noah Lezama
Verteiler	

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	10.01.2025	Start	Noah Lezama

Tabelle 41: Änderungskontrolle

Beschreibung

Der Programmarbeitsauftrag enthält alle relevanten Informationen zur Erledigung einer gestellten Aufgabe. Der Programmleiter erteilt mit ihm die Arbeiten an die Programmmitarbeiter. Zusammen mit dem Fertigstellungsgrad der Ergebnisse wird auf der Basis der Arbeitsaufträge die Fortschrittskontrolle in Bezug auf das Programm durchgeführt. Arbeitsaufträge können intern oder extern vergeben werden. Für jedes Arbeitspaket wird ein Programmarbeitsauftrag erteilt.

Arbeitsziele

Nr.	Ziel
01	*Erstellung einer Projekt-/System-Dokumentation (HERMES-konform) für die bestehende Webanwendung*
02	*Entwicklung einer technischen Vorlage (inkl. Docker, CI/CD, grundlegende Architektur)*
03	*Vorbereitung eines Testkonzepts mit mindestens 10 Testfällen*

Tabelle 42: Arbeitsziele

Ergebnisse

Nr.	Ergebnis	Beschreibung
01	*Detaillierte Projektdokumentation*	*HERMES-konforme Dokumentation (Projektstruktur, Rollen, Phasen, Risikomanagement etc.)*
02	*Technische Vorlage (Prototyp)*	*Repository mit Basis-Setup: Frontend-Framework, Backend-API, Docker-Compose, CI/CD-Pipeline*
03	*Testkonzept (Entwurf)*	*Beschreibung der Teststrategie, Teststufen, Testdaten und mindestens 10 konkrete Testfälle*

Tabelle 43: Ergebnisse

Abgrenzung

In diesem Arbeitspaket wird kein fertiges Produkt für den Produktivbetrieb entwickelt, sondern eine technische Vorlage (Proof-of-Concept/Prototyp).

Keine finale Cloud-Migration: Die Evaluierung (Marktabklärung) zu Cloud-Providern ist Teilprojekt B, doch grundlegende Architekturvorgaben (Docker, CI/CD) werden hier konzeptionell festgelegt.

Keine Angeboteinholung: Die Beschaffung von externen Services bzw. Lizenzen ist in Projekt B/Programmleitung integriert.

Voraussetzungen und Abhängigkeiten

Voraussetzungen für den Arbeitsbeginn

Verfügbarkeit des Projektleiters und IT-Architekten (50% bzw. 100% in den ersten Monaten).

Abgeschlossene Grundlagendokumente (Programmstudie, grobe Architekturentscheidung) liegen bereits vor.

Entwicklungssysteme (z. B. lokale Docker-Umgebung, PM-Tools) sind eingerichtet und nutzbar

Abhängigkeiten von anderen / zu anderen

Nr	Abhängigkeit von	Ergebnis	Termin	Verantwortlich
A-01	*Projekt B (Cloud-Auswahl)*	*Entscheidung über Cloud-Provider*	*24.01.2025*	*Programmleiter (B)*
A-02	*Schulung Hermes (intern)*	*Einführung Team in HERMES*	*14.12.2024*	*Kompetenzzentrum Hermes*

Tabelle 44: Abhängigkeiten von anderen

Nr	Abhängigkeit zu	Ergebnis	Termin	Verantwortlich
B-01	*Projekt C (Umsetzung)*	*Technische Vorlage (Beta)*	*29.07.2025*	*Projektleiter (C)*
B-02	*Programmabschluss*	*Vollständige HERMES-Dokumentation, Testkonzept*	*01.09.2025*	*Programmleiter*

Tabelle 45: Abhängigkeiten zu anderen

Aktivitäten

Nr.	Ergebnis	Aktivität*	Verantwortlich, Mitwirkend	Plan Std.	Termin	Status
1	*Detaillierte Projektdoku*	*Überführen der vorhandenen Analyse (Ausgangslage, Ziele, Risiken) in HERMES-Struktur, inkl. Review-Runden*	*Programmleiter, IT-Architekt*	*16*	*15.01.2025*	*offen*
2	*Detaillierte Projektdoku*	*Walk-Through der Dokumentation (Qualitätssicherung)*	*Programmleiter, Team*	*4*	*20.01.2025*	*offen*
3	*Technische Vorlage (Prototyp)*	*Einrichten des Git-Repos, Grundstruktur (Frontend-Framework, Backend-API, Dockerfiles), Konfiguration CI/CD*	*IT-Architekt, Dev-Team*	*32*	*15.02.2025*	*offen*
4	*Testkonzept (Entwurf)*	*Erstellung einer Teststrategie (Unit, Integration, End-to-End), Erfassung von Testfällen (≥10)*	*Tester (QA), Projektleiter*	*12*	*28.04.2025*	*offen*
5		*Präsentation der Zwischenergebnisse an Programmleitung*	*Projektleiter*	*4*	*05.05.2025*	*offen*
Total				*68*		
**** inklusive Qualitätssicherungsmassnahmen wie Walk-Throughs, Reviews (gemäss Prüfplan im Projektmanagementplan)umgesetzt***						

Tabelle 46: Liste der Aktivitäten

Ressourcenbedarf

Personalressourcen (Stunden)

Rolle / Person	Monat 1	Monat 2	Monat 3	Monat 4	Monat 5	Total
Programmleiter	*40 h*	*40 h*	*40 h*	*40 h*	*40 h*	*200 h*
IT-Architekt	*20 h*	*20 h*	*40 h*	*20 h*	*20 h*	*120 h*
Tester (QA)	*–*	*10 h*	*10 h*	*20 h*	*20 h*	*60 h*

Tabelle 47: Personalressourcen in Stunden

Sachmittel

IT-Infrastruktur: Zugriff auf Cloud-Testkonten, Docker-Registry, CI/CD-Umgebung.

Software: Lizenzen für PM-Tool, Code-Repository, Ticketsystem (z. B. JIRA), Testing-Frameworks.

Räume: Teamraum für Meeting (2x pro Woche) + Konferenzraum für Reviews.

Ergebnisdarstellung

Dokumentation

- Struktur im Word-/PDF-Format bzw. in Confluence/wiki.
- Kapitelaufteilung gemäss HERMES (Projektstruktur, Phasen, Risiko, Testkonzept).

Technische Vorlage

- GitHub-/GitLab-Repository mit einer lauffähigen Docker-Compose-Konfiguration (Frontend + Backend + optional DB).
- Skripte für CI/CD (z. B. GitLab-CI, GitHub-Actions).

Testkonzept

- Getrenntes Dokument oder eigenes Kapitel, inklusive Testfallliste.

Qualitätssicherung

Walk-Throughs für Dokumentation und Testkonzept (siehe Aktivitäten Nr. 2).

Regelmässige Code-Reviews und Linter-Einsatz im Git-Repository.

Reviews gemäss Prüfplan (z. B. vom Programmleiter, Fachausschuss).

Testbuilds in CI/CD: automatisierte Überprüfung der Grundfunktionen (Unit- und Integrations-Tests).

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
HERMES	Vorgehensmethodik für Projekte/Programme in der Schweiz
CI/CD	Continuous Integration / Continuous Deployment (bzw. Delivery)
Docker	Plattform zum Erstellen und Ausführen containerisierter Anwendungen
QA	Quality Assurance (Qualitätssicherung)
PoC	Proof of Concept (Machbarkeitsnachweis)
MVP	Minimum Viable Product (minimale, nutzbare Produktversion)

Tabelle 48: Abkürzungen und Glossar

Testkonzept

Klassifizierung	VERTRAULICH
Status	in Arbeit
Programmname	Workshop
Projektnummer	1
Projektleiter	Noah Lezama
Version	0.1
Datum	15. Januar 2025
Auftraggeber	Auftraggeber
Autor/Autoren	Noah Lezama
Verteiler	

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	15.01.2025	Start	Noah Lezama

Tabelle 49: Änderungskontrolle

Beschreibung

Das Testkonzept beschreibt die Testziele, Testobjekte, Testarten, Testinfrastruktur sowie die Testorganisation. Es umfasst ebenfalls die Testplanung und die Testfallbeschreibungen. Für jeden Testfall wird eine detaillierte Testfallbeschreibung erstellt. Diese stellt die Spezifikation des Tests dar. Die Testplanung legt den logischen und zeitlichen Ablauf der Tests fest. Das Testkonzept bildet die Grundlage, auf der die Testorganisation und die Testinfrastruktur bereitgestellt und die Tests durchgeführt werden. Es wird bei neuen Erkenntnissen stets nachgeführt.

Testziele

Globale messbare Testziele über alle Testfälle hinweg:

Nr.	Beschreibung	Messgrösse	Priorität*
1	*Funktionalität: Alle Kernfunktionen (z.B. Registrierung, Login, Dokumentenupload) funktionieren fehlerfrei.*	*Fehlerquote in Testfällen < 2 %*	*M*
2	*Sicherheit: Vertraulichkeit (TLS, Verschlüsselung), 4-Augen-Prinzip und Rollen-/Rechte-System sind wirksam.*	*Pen-Test/Bug-Report keine kritischen Findings*	*M*
3	– Performance: Unter Last (mind. 100 gleichzeitige User) bleiben Antwortzeiten < 3 Sekunden.	*Loadtest-Resultate*	*M*
4	*Benutzerfreundlichkeit: UI ist intuitiv; Nutzer finden Kernfunktionen (Suche, Upload etc.) leicht.*	*Befragung / Feedback*	*M*
5	*Wiederherstellungsfähigkeit: Backups und Restore-Prozesse funktionieren korrekt.*	*Erfolgreiche Restore-Tests*	*2*
6	*Protokollierung und Monitoring: System generiert ausreichende Logs und kann überwacht werden.*	*Monitoring-/Logging-Test (keine Lücken)*	*2*
**** Priorität: M = Muss / 1 = hoch, 2 = mittel, 3 = tief***			

Tabelle 50: Übergeordnete Testziele

Teststrategie und Teststufen

Teststrategie

- Die Entwicklung wird begleitet durch eine CI/CD-Pipeline, in der automatisierte Tests (Unit-, Integrationstests) bereits beim Erstellen von Pull Requests ausgeführt werden.
- Nach jedem Build erfolgt ein Deployment in eine Testumgebung (Staging), um manuelle Tests (z. B. System- und Benutzertests) durchzuführen.
- Sicherheit und Performance werden mittels separater Tools (z. B. Penetrationstests, Lasttests) in definierten Milestones geprüft.
- Änderungen am Code, insbesondere sicherheitskritische Funktionen, unterliegen dem 4-Augen-Prinzip (Code-Reviews + Abnahme durch Prüfer).

Teststufen

- Unit-Tests: Prüfen einzelne Funktionen/Methoden (z. B. Upload-Funktion, Validierung). Werden automatisiert bei jedem Commit ausgeführt.
- Integrations- und API-Tests: Verifizieren das Zusammenspiel zwischen Backend, Frontend und Datenbank sowie Schnittstellen (z. B. E-Mail-Versand, SMS-Verifikation).
- Systemtests: Umfassen End-to-End-Szenarien in einer Testumgebung (Simulieren realen Nutzerfluss).
- Abnahmetests: Durch Endanwender und Fachabteilungen, prüfen Qualität, Benutzerfreundlichkeit und korrekte Umsetzung der Anforderungen (z. B. Four-Eyes bei Löschungen).

- Last-/Performance-Tests: Messen Reaktionszeiten und Stabilität bei 100+ parallelen Nutzern.
- Sicherheits-Tests (Penetrationstests): Erkennen potenzielle Lücken in Authentifizierung, Verschlüsselung etc.
- Backup- und Wiederherstellungstests: Verifizieren, dass Daten nach fehlerhafter Löschung oder Ausfall wiederhergestellt werden können.

Testobjekte

Nr.	Objekt	Beschreibung
1	*Benutzerverwaltung*	*Registrierung, Login (E-Mail/SMS), Rollen- und Rechtesystem*
2	*Dokumentenmanagement*	*Upload, Bearbeiten von Metadaten, Archivieren/Löschen, Teilen, Suche*
3	*Inbox / Workflow*	*Offene Dokumente, Freigabeprozesse*
4	*Systemkomponenten (Backend, Frontend etc.)*	*Technische Integrationen, CI/CD, Logging, Monitoring, Backup/Restore*

Tabelle 51: Testobjekte

Testarten

Nr.	Testart	Beschreibung
1	*Unit-Tests*	*Prüfen einzelne Funktionen (z. B. Validierungsfunktionen, Datenbank-Queries). Werden automatisiert in der CI/CD-Pipeline ausgeführt.*
2	*Integrations- / API-Tests*	*Sicherstellen, dass Schnittstellen zwischen Microservices (Backend, Datenbank) oder zu externen Diensten (E-Mail/SMS) korrekt funktionieren.*
3	*Systemtests / End-to-End*	*Durchgängige Szenarien (User-Login, Dokument hochladen, Freigeben, Archivieren). Simulieren reale Benutzerflüsse.*
4	*Performance- / Lasttests*	*Überprüfen Antwortzeiten bei steigender Nutzerzahl (100–500 parallele Sessions). Messen Systemverhalten (z. B. CPU, RAM).*
5	*Sicherheitstests*	*Fokus auf Penetrationstests (OWASP Top 10), Prüfung von Verschlüsselung, Konfiguration und 4-Augen-Prinzip.*
6	*Backup- und Wiederherstellung*	*Regelmässige Tests (z. B. einmal pro Sprint), ob Daten gesichert und im Notfall korrekt wiederhergestellt werden können.*
7	*Benutzerfreundlichkeitstests (Usability)*	*Bewertung UI/UX (z. B. via Testgruppen, Feedback-Formulare).*
8	*Monitoring- und Protokollierungstests*	*Validiert, ob Log-Dateien komplett und aussagekräftig sind und ob Monitoring-Alerts korrekt ausgelöst werden.*

Tabelle 52: Testarten

Testabdeckung

Übersicht Testfälle

Nr.	Testobjekt	Testfälle
1	*Benutzerverwaltung*	*1) Registrierung (Privatperson)* *2) Login mit E-Mail/SMS* *3) Rollen-/Rechteprüfung (z. B. Admin vs. User)*
2	*Dokumentenmanagement*	*4) Dokument hochladen* *5) Dokument bearbeiten (Titel, Beschreibung)* *6) Dokument archivieren/löschen* *7) Dokument suchen (Volltextsuche)* *8) Dokument teilen*
3	*Inbox/Workflow*	*9) Dokument im Posteingang erledigen (z. B. genehmigen, Workflow)*
4	*Stammdaten*	*10) Kategorienverwaltung: Erfassung neuer Kategorie & Zuordnung*

Tabelle 53: Testabdeckung

Beurteilung Testziele und Testabdeckung

Alle Muss-Anforderungen (z. B. Login, Upload, 4-Augen-Prinzip) sind in entsprechenden Testfällen berücksichtigt.

Performance und Sicherheit werden in separaten Tests (Last-, Pen-Tests) erfasst.

Backup/Wiederherstellung und Monitoring-Tests erfolgen ergänzend, sind nicht an einzelne Funktionsobjekte, sondern an Systemkomponenten geknüpft.

Testrahmen

Testvoraussetzungen

Tester: Mindestens 1 QA-Engineer + 1 Fachperson (z. B. Enduser-Vertreter) für Benutzertests.

Vorkenntnisse: Grundsätzliches Verständnis der Webapplikation (Bedienung, Rollen). Für Sicherheitstests: Security-Experte.

Testumgebung: Staging-System mit möglichst realen Daten (Dummy-Daten), identische Konfiguration wie Produktionsumgebung (Server OS, Docker-Setup etc.).

Mängelklassifizierung

Nr.	Mängelklassen	Beschreibung
0	mängelfrei	Einwandfrei und anforderungsgerecht
1	belangloser Mangel	Verwendung möglich, Brauchbarkeit ist vorhanden, Mängel sollte dennoch nicht vorkommen
2	leichter Mangel	Verwendung möglich, Brauchbarkeit ist nur wenig beeinträchtigt
3	schwerer Mangel	Verwendung ist noch möglich, Brauchbarkeit ist stark verringert
4	kritischer Mangel	Unbrauchbar; Wesentliche Funktionalität ist nicht gegeben; Betrieb ist nicht verantwortbar (z.B. sicherheitsspezifisch)

Tabelle 54: Mängelklassen

Die Klassifizierung spiegelt die Folgenschwere und den Aufwand zur Behebung der möglich feststellbaren Mängel. Die Zuordnung der festgestellten Mängel zu einer Mängelklasse gibt grob auch die Priorität vor, in welcher Reihenfolge die Behebung der Mängel angegangen werden soll.

Wird eine Mängelklasse zwischen 1-3 erreicht, kann das System/Produkt unter Vorbehalt abgenommen werden. Zur Behebung der Mängel sind jedoch Massnahmen zu definieren. Eine Nachprüfung ist zwingend.

Werden hingegen Mängel der Klasse 4 festgestellt, kann das System/Produkt nicht abgenommen werden und der Auftragnehmer muss umgehend Massnahmen treffen, um diese Mängel zu beheben. Der Auftragnehmer hat zudem die erneute Abnahme zu veranlassen.

Start- und Abbruchbedingungen

Start:

Testumgebung ist installiert und lauffähig (identische Version wie angestrebtes Release).

Testdaten stehen bereit (Dummy-Benutzer, Testdokumente).

Tester wurden informiert und haben Zugriff auf Dokumentation.

Abbruch:

Kritischer Mangel (Klasse 4) aufgetreten Teststopp, Ticket an Entwicklung/Architektur.

Keine Reproduzierbarkeit mehr möglich (Umgebung instabil) Muss stabilisiert werden.

Testumgebung

Staging-System auf Cloud-Plattform (z. B. AWS oder Azure), Docker-Container (Backend, Datenbank, ggf. Redis).

Zugriff via Test-Domain (z. B. test-umgebung.domain.tld) mit SSL-Zertifikaten.

Monitoring: Tools wie Prometheus oder Datadog zur Messung von Auslastung, Logging in ELK-Stack (Elasticsearch, Logstash, Kibana).

Testinfrastruktur

Testsystem

Docker-Compose-Setup spiegelbildlich zum Entwicklungsstand.

CI/CD-Pipeline (GitHub Actions oder GitLab CI) führt Unit-/Integrationstests automatisch aus.

Testdaten

Anonymisierte Musterdaten (Dokumente, Benutzeraccounts).

Musterdokumente in unterschiedlichen Formaten (PDF, DOCX, PNG).

Synthetische Nutzlast für Performance-/Lasttests.

Testhilfsmittel

Testmanagement: z. B. JIRA/XRay, Zephyr oder ein anderes Ticket-/Testfall-Verwaltungstool.

Reporting: Automatisierte Reports aus CI/CD, Performance-Reports (z. B. JMeter), Sicherheitsberichte (z. B. OWASP ZAP).

Testorganisation

Verantwortung:

Testmanager (QA) koordiniert alle Tests, prüft Ergebnisse, erstellt Berichte.

Fachabteilung validiert fachliche Anforderungen (Workflows, Freigaben).

Entwicklungsteam behebt Mängel (Tickets) nach Priorität.

Ablauf:

Tägliche Abstimmung bei laufenden Testphasen (Daily Stand-up).

Mängel werden im Issue-Tracker erfasst, priorisiert und gesprintet.

Regelmässige Statusberichte an den Projektleiter.

Testfallbeschreibungen

ID / Bezeichnung	*T-001*	*Registrierung als Privatperson*	
Beschreibung	*Testet, ob ein neuer User sich erfolgreich registrieren kann und eine Bestätigungsnachricht (E-Mail/SMS) erhält.*		
Testvoraussetzung	*- Testumgebung lauffähig* *- Keine bestehende Test-E-Mail (oder Telefon-Nr.)*		
Testschritte	*1) Aufruf Registrierungsseite* *2) Eingabe User-Daten (Name, E-Mail, Passwort, etc.)* *3) Absenden* *4) Erhalt von Bestätigungs-E-Mail / SMS* *5) Klick auf Bestätigungslink oder Eingabe Code*		
Erwartetes Ergebnis	*- Benutzerkonto wird angelegt* *- E-Mail/SMS kommt korrekt an* *- Der User kann sich nach Bestätigung einloggen*		

Tabelle 55: Testfallbeschreibung

Testplan

Nr.	Aktivität	Verantwortlich	Mitarbeit	Termin
1	*Einrichten Testumgebung (Staging)*	*DevOps-Team*	*QA, IT-Architekt*	*ab 15.02.2025*
2	*Ausführung Unit-Tests (fortlaufend)*	*Entwicklung*		*bei jedem Commit*
3	*Integrations-/API-Tests*	*QA-Engineer*	*Backend/Frontend-Entwickler*	*ab 01.03.2025*
4	*End-to-End / Systemtests*	*QA-Engineer*	*Fachabteilung*	*ab 10.03.2025*
5	*Security-Test (Pen-Test)*	*Security-Team*	*QA, IT-Architekt*	*ab 20.03.2025*
6	*Performance-Test (Lasttest)*	*QA + DevOps*		*ab 25.03.2025*
7	*Backup/Restore-Test*	*QA + DevOps*		*ab 01.04.2025*
8	*Usability-Test (Probanden)*	*Fachabteilung*	*QA, UI/UX-Designer*	*ab 05.04.2025*
9	*Abnahmetests mit Fachseite*	*QA-Engineer*	*Fachabteilungen, Projektleiter*	*ab 25.04.2025*

Tabelle 56: Testplan

Abkürzungen und Glossar

Abkürzung / Fachwort	Erläuterung
HERMES	Vorgehensmethodik für Projekte und Programme HERMES 5 ist ein eCH Standard
CI/CD	Continuous Integration / Continuous Deployment (bzw. Delivery)
QA	Quality Assurance (Qualitätssicherung)
PoC	Proof of Concept
E2E	End-to-End (Test)
Pen-Test	Penetrationstest: geprüft werden Sicherheitslücken, v. a. nach OWASP-Top-10
4-Augen-Prinzip	Kritische Aktionen müssen von zwei berechtigten Personen bestätigt werden

Tabelle 57: Abkürzungen und Glossar

Testprotokoll

Klassifizierung	VERTRAULICH
Status	in Arbeit
Programmname	Workshop
Projektnummer	1
Projektleiter	Projektleiter
Version	0.1
Datum	16. Januar 2025
Auftraggeber	Noah Lezama
Autor/Autoren	Noah Lezama
Verteiler	

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	16.01.2025	Start	Noah Lezama

Tabelle 58: Änderungskontrolle

Beschreibung

Das Testprotokoll hält die Testergebnisse fest. Die Testergebnisse sind gemäss den im Testkonzept definierten Mängelklassen bewertet.

Übersicht der Testfälle / Testdurchführungen

Hinweis auf das Testkonzept

ID	Bezeichnung	Testdatum	Tester	FK*
T-001	*Registration als Privatperson*			
T-002	*Login mit E-Mail/SMS*			
T-003	*Dokument hochladen*			
T-004	*Dokument bearbeiten*			
T-005	*Inbox-Dokument erledigen*			
T-006	*Dokument archivieren / löschen*			
T-007	*Dokument suchen (Volltextsuche)*			
T-008	*Dokument teilen*			

ID	Bezeichnung	Testdatum	Tester	FK*
T-009	*Stammdaten (Kategorie erfassen & zuordnen)*			
T-010	*Vier-Augen-Prinzip (kritische Aktion)*			
Legende: FK = Mängelklasse (Testergebnis)				

Testfall 1

Testfallbeschreibung

ID / Bezeichnung	*T-001*	*Registration als Privatperson*
Beschreibung	*Prüft, ob ein neuer User erfolgreich angelegt werden kann und ob die Bestätigungsnachricht (E-Mail oder SMS) korrekt eintrifft.*	
Testvoraussetzung	*Lauffähige Testumgebung* *Keine vorregistrierte Test-E-Mail-Adresse oder Telefonnummer*	
Testschritte	1. *Aufruf der Registrierungsseite.* 2. *Eingabe der Nutzerangaben (Name, E-Mail, Passwort, optional Telefonnummer).* 3. *Bestätigen des Formulars.* 4. *Prüfung, ob E-Mail/SMS zur Verifizierung eintrifft.* 5. *Verifizierungslink anklicken oder Code eingeben.*	
Erwartetes Ergebnis	*Neues Benutzerkonto wird erstellt.* *E-Mail/SMS trifft zeitnah ein.* *Nach Bestätigung ist ein Login möglich.*	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	*s*
Tester	
Mängelklasse*	
Mangelbeschreibung	
Bemerkungen	

**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*

Testfall 2

Testfallbeschreibung

ID / Bezeichnung	*T-002*	*Login mit E-Mail/SMS*
Beschreibung	*Validiert den Anmeldeprozess mit E-Mail/Passwort und optionalem SMS-Code.*	
Testvoraussetzung	*Benutzerkonto existiert (verifiziert).* *Handy-Nummer im Profil hinterlegt (falls SMS-Code relevant).*	
Testschritte	1. *Öffnen der Login-Seite.* 2. *Eingabe von E-Mail + Passwort.* 3. *(Optional) Eingabe des SMS-Codes, sofern konfiguriert.* 4. *Absenden und Warten auf Rückmeldung.*	
Erwartetes Ergebnis	*Benutzer wird fehlerfrei angemeldet.* *Optionaler SMS-Code (MFA) wird geprüft.*	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	
Tester	
Mängelklasse	
Mangelbeschreibung	
Bemerkungen	

**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*

Testfall 3

Testfallbeschreibung

ID / Bezeichnung	*T-003*	*Dokument hochladen*
Beschreibung	*Überprüfung, ob ein Anwender verschiedene Dokumente (PDF, Bilder, etc.) hochladen kann.*	
Testvoraussetzung	*Gültiger Benutzer ist eingeloggt.* *Dokumente (Testdateien) in verschiedenen Formaten verfügbar.*	
Testschritte	1. *Klick auf „Dokument hochladen“.* 2. *Auswahl einer Datei (z. B. PDF).* 3. *Bestätigung des Uploads.*	
Erwartetes Ergebnis	*Dokument wird im System gespeichert.* *Bestätigung / Erfolgsmeldung wird angezeigt.* *Metadaten sind korrekt sichtbar.*	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	
Tester	
Mängelklasse	
Mangelbeschreibung	
Bemerkungen	
**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*	

Testfall 4

Testfallbeschreibung

ID / Bezeichnung	*T-004*	*Dokument bearbeiten*
Beschreibung	*Überprüfung, ob Titel/Beschreibung eines Dokuments geändert werden können.*	
Testvoraussetzung	*Erfolgreich hochgeladenes Dokument liegt vor.* *Benutzer ist eingeloggt und hat Bearbeitungsrechte.*	
Testschritte	1. *Dokumentdetails aufrufen (z. B. Klick auf Dokument in der Liste).* 2. *Titel/Beschreibung abändern und speichern.*	
Erwartetes Ergebnis	*Änderungen sind sofort oder nach Aktualisierung ersichtlich.* *Änderung wird ggf. in der Versions-/Änderungshistorie protokolliert.*	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	
Tester	
Mängelklasse*	
Mangelbeschreibung	
Bemerkungen	

**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*

Testfall 5

Testfallbeschreibung

ID / Bezeichnung	*T-005*	*Inbox-Dokument erledigen*
Beschreibung	*Testet das Abarbeiten eines Dokuments, das im Posteingang (Inbox) liegt.*	
Testvoraussetzung	*Ein hochgeladenes Dokument mit Status „zu erledigen“ existiert.* *User hat Berechtigung, das Dokument zu bearbeiten/freigeben.*	
Testschritte	1. *Öffnen der Inbox.* 2. *Auswahl des zu erledigenden Dokuments.* 3. *Aktion durchführen (z. B. Freigabe, Bearbeitung, Kommentar).* 4. *Speichern/Abschliessen.*	
Erwartetes Ergebnis	*Status des Dokuments wechselt auf „erledigt“ oder „freigegeben“.* *Dokument verschwindet aus der Inbox.*	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	
Tester	
Mängelklasse*	
Mangelbeschreibung	
Bemerkungen	

**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*

Testfall 6

Testfallbeschreibung

ID / Bezeichnung	*T-006*	*Dokument archivieren / löschen*
Beschreibung	*Prüft, ob ein Dokument korrekt archiviert bzw. endgültig gelöscht werden kann.*	
Testvoraussetzung	*Dokument ist im System vorhanden und dem User zugänglich. Archivierungs-/Löschfunktion ist aktiv und freigegeben.*	
Testschritte	*1. Dokumentdetails aufrufen.* *2. Aktion „Archivieren“ wählen bzw. „Löschen“.* *3. (Bei endgültigem Löschen) ggf. 4-Augen-Bestätigung einholen.*	
Erwartetes Ergebnis	*Dokumentstatus wechselt auf „archiviert“ oder wird entfernt.* *Im Fall Löschung: Dokument ist nicht mehr in der Suche ersichtlich; ggf. Audit-Eintrag vorhanden.*	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	
Tester	
Mängelklasse*	
Mangelbeschreibung	
Bemerkungen	

**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*

Testfall 7

Testfallbeschreibung

ID / Bezeichnung	*T-007*	*Dokument suchen*
Beschreibung	*Prüft die Funktion der Volltextsuche und deren Trefferquote.*	
Testvoraussetzung	*Mehrere Dokumente (unterschiedliche Inhalte, Titel) existieren.* *Benutzer ist eingeloggt.*	
Testschritte	1. *Eingabe eines Suchbegriffs (z. B. Stichwort aus Dokumenttitel).* 2. *Optional: Filter (Kategorie, Datum).* 3. *Start der Suche.*	
Erwartetes Ergebnis	*Relevante Dokumente werden gelistet.* *Volltextindizes greifen korrekt; ggf. Hervorhebung der Suchwörter.*	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	
Tester	
Mängelklasse*	
Mangelbeschreibung	
Bemerkungen	

**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*

Testfall 8

Testfallbeschreibung

ID / Bezeichnung	*T-008*	*Dokument teilen*
Beschreibung	*Testet das Freigeben von Dokumenten via Link oder gezielter Einladung eines externen Nutzers.*	
Testvoraussetzung	*Dokument liegt vor; User hat Rechte zum Teilen.* *Externe E-Mail-Adresse / Empfänger ist bekannt.*	
Testschritte	1. *Aufruf der Teilen-Funktion beim Dokument.* 2. *Eingabe der E-Mail (oder Erzeugung eines Freigabe-Links).* 3. *Optionale Rechtezuweisung (Nur Lesen vs. Lesen/Schreiben).* 4. *Senden des Links / Einladungs-Mail.*	
Erwartetes Ergebnis	*Externer Link/E-Mail wird generiert und versendet.* *Zugriff des externen Empfängers funktioniert entsprechend der erteilten Berechtigung.*	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	
Tester	
Mängelklasse*	
Mangelbeschreibung	
Bemerkungen	

**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*

Testfall 9

Testfallbeschreibung

ID / Bezeichnung	*T-009*	*Stammdaten (Kategorie erfassen & zuordnen)*
Beschreibung	*Überprüfung der Anlage neuer Kategorien und Zuweisung dieser Kategorien an vorhandene Dokumente.*	
Testvoraussetzung	*User besitzt Berechtigung, Stammdaten zu verwalten.* *Mindestens ein Dokument ist bereits hochgeladen.*	
Testschritte	1. *Aufruf „Kategorie verwalten“.* 2. *Neue Kategorie anlegen (Name, ggf. Beschreibung).* 3. *Vorhandenes Dokument auswählen und dieser neuen Kategorie zuordnen.*	
Erwartetes Ergebnis	*Kategorie wird gespeichert und in Liste angezeigt.* *Dokument verweist sichtbar auf die zugehörige Kategorie.*	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	
Tester	
Mängelklasse*	
Mangelbeschreibung	
Bemerkungen	

**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*

Testfall 10

Testfallbeschreibung

ID / Bezeichnung	*T-010*	*Vier-Augen-Prinzip*
Beschreibung	*Prüft, ob kritische Funktionen (z. B. endgültiges Löschen, globale Systemeinstellungen) eine zweite Freigabe erfordern.*	
Testvoraussetzung	*System verfügt über 4-Augen-Mechanismus.* *Zwei unterschiedliche Benutzer mit entsprechendem Berechtigungslevel existieren.*	
Testschritte	1. *Benutzer A versucht, ein Dokument endgültig zu löschen.* 2. *System fordert Bestätigung durch Benutzer B (zweiter freigabeberechtigter User).* 3. *Benutzer B loggt sich ein und bestätigt / lehnt die Aktion.*	
Erwartetes Ergebnis	*Ohne Freigabe durch B wird die Aktion abgebrochen.* *Mit Freigabe durch B wird Dokument tatsächlich gelöscht.* *Audit-Log verzeichnet beide Freigaben.*	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	
Tester	
Mängelklasse*	
Mangelbeschreibung	
Bemerkungen	

**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*

Table of Contents