

IPA Marc Egli - Puzzle ITC

IPA-Daten und beteiligte Personen	
Firma, Abteilung	Puzzle ITC, /dev/ruby
Berufsschule	GIBB
Valid Experte	Lawson Mike
Hauptexpertin	Müller Lorenz
Nebenexperte	Moser Michael
Verantwortliche Fachkraft	Illi Daniel
Zusätzliche Verantwortliche Fachkraft	Steiner Robin
Berufsbildner	Steiner Robin
Fachrichtung	Applikationsentwicklung
Projektvorgehensmodell	SCRUM
Jahrgang der IPA-Durchführung und Kanton	IPA 2025, Kanton Bern
Abgabedatum	

Tabelle 1: IPA Daten

Teil I

Ablauf, Organisation und Umfeld

Inhaltsverzeichnis

I	Ablauf, Organisation und Umfeld	1
1	Aufgabenstellung	6
1.1	Titel der Arbeit	6
1.2	Ausgangslage	6
1.3	Detaillierte Aufgabenstellung	7
1.3.1	Mittel und Methoden	9
1.3.2	Vorkenntnisse	10
1.3.3	Vorarbeiten	10
1.3.4	Neue Lerninhalte	10
1.3.5	Arbeiten in den letzten 6 Monaten	10
2	Firmenstandards	11
2.1	Code conventions	11
2.1.1	Mehrsprachigkeit	11
2.1.2	Lizenz	11
2.2	Git Commit Message Conventions	12
3	IPA-Schutzbedarfsanalyse	13
3.1	Datensicherheit	13
3.2	Applikationssicherheit	13
4	Organisation der IPA-Ergebnisse	14
4.1	Datensicherung	14
4.1.1	Dokumentation	14
4.1.2	Code	15
4.1.3	Wiederherstellung des Codes	15
4.2	Änderungskontrolle	15
5	Projektmethode	16
5.1	Einsatz von Scrum	16
5.1.1	Sprints	16
5.1.2	Verwaltungstool	16
5.1.3	Meetings	17
5.1.4	Abweichungen	18
5.2	Definition of Done	19
5.2.1	DoD Code	19
5.2.2	DoD Dokumentation	19

5.2.3	Akzeptanzkriterien	20
5.3	Verwendungsgrund	20
6	Projektaufbauorganisation	21
6.1	Projektrollen Scrum	21
6.2	Projektrollen IPA	22
6.3	Projektrollen Scrum in der IPA	23
7	Zeitplan	24
7.1	Erläuterung zum Zeitplan	24
7.2	Sprints	24
8	Arbeitsjournale	25
8.1	Tag 1: 04.03.2025	25
8.2	Tag 2: 05.03.2025	28
8.3	Tag 3: Datum	31
8.4	Tag 4: Datum	32
8.5	Tag 5: Datum	33
8.6	Tag 6: Datum	34
8.7	Tag 7: Datum	35
8.8	Tag 8: Datum	36
8.9	Tag 9: Datum	37
8.10	Tag 10: Datum	38
8.11	Tag 11: Datum	39
8.12	Tag 12: Datum	40
9	Persönliches Fazit	41
9.1	Was lief weniger gut	41
9.2	Was lief gut	41
9.3	Schlussreflexion	41
II	Projektdokumentation	42
10	Einführung	43
11	Analyse	44
11.1	Ist-Zustand	44
11.1.1	Personenlisten	44
11.1.2	Abonnemente	44
11.2	Soll-Zustand	44
11.3	Bedürfniserhebung	44
11.4	Anforderungen	44
11.4.1	Nicht funktionale Anforderungen	44
11.4.2	Funktionale Anforderungen	44
11.5	Abgrenzung	44
11.6	Benötigter Rahmen	44

11.6.1 Fehlende Informationen	44
11.7 Persönliche Vorgehensziele	44
12 Risikoanalyse und Sicherheitsmassnahmen	45
12.1 Schnittstellen	45
12.2 Benutzer und Datenzugriffe	46
12.2.1 Datenstruktur	47
12.2.2 Beispiel Zugriff Heinz	48
12.2.3 Beispiel Zugriff Tim	49
12.2.4 Beispiel Zugriff Rudolf	50
12.2.5 Bedeutung für die Schnittstellen	50
12.2.6 Risikoanalyse	51
12.3 Risikomatrix	52
12.4 Auswertung	52
13 Entwurf	53
13.1 Anwendungskonzept	54
13.1.1 Anwendungsdiagramm	54
13.1.2 Anwendungsfälle	54
13.2 Systemkonzept	54
13.2.1 Betroffene Services	54
13.2.2 Status quo	54
13.2.3 Lösungsvarianten	54
13.2.4 Variantenentscheid	54
13.3 Sicherheitskonzept	54
13.3.1 SQL-Injection	54
13.3.2 Cross-Site Scripting	54
13.3.3 URL Interpretation	54
13.3.4 Kommunikation HTTP/S	54
13.4 Fehlerbehandlungskonzept	54
13.4.1 Nutzereingabe	54
13.4.2 Laufzeitfehler	54
13.5 Testsetup	54
13.6 Testkonzept	54
13.6.1 Testinfrastruktur	54
13.6.2 Fehlerklassen	54
13.6.3 Manuelle Tests	54
14 Ausführung	55
14.1 Einsatz von KI-Modellen	55
14.2 Gems	55
14.2.1 can-can-can	55
14.2.2 dry-crud	55
15 Einführung	56
15.1 Instruktion	56

15.2	Unvorhergesehene Änderungen	56
15.2.1	application.rb	56
15.2.2	_list.html.haml	56
16	Sprintabschlüsse	57
16.1	Abschluss Sprint Initialisierung	57
16.1.1	Backlog	57
16.2	Abschluss Sprint Umsetzung	57
16.2.1	Backlog	57
16.3	Abschluss Sprint Finalisierung	57
16.3.1	Backlog	57
III	Anhang und Verzeichnisse	58
17	Verzeichnisse	59
17.1	Code	59
17.2	Tabellenverzeichnis	59
17.3	Abbildungsverzeichnis	59
	Quellenverzeichnis	61
18	Verwendete Abkürzungen	62
19	Glossar	63
20	Anhänge	64
20.1	Git Commit Message Convention	64
20.2	Daily-Protokolle	64
20.3	Sitzungsprotokolle	64
20.4	Git commit convention	64
20.5	Security conventions	64

1 Aufgabenstellung

1.1 Titel der Arbeit

Hitobito: Neue Generation von Personen-Filtern

1.2 Ausgangslage

Hitobito ist eine Open Source Webapplikation zum Verwalten von Mitgliedern, Events und vielem mehr. Die Ruby on Rails Applikation wurde 2012 von Puzzle ITC initiiert und wird stets weiterentwickelt.

Die Basis für die Software bildet das Webframework Ruby on Rails. Für das User Interface wird neben statischer Technologie wie HTML und CSS auch JavaScript oder Hotwire verwendet. Der komplette Source-Code steht auf Github zur Verfügung: [Hitobito](#)

Eine Kernfunktionalität von Hitobito ist das Filtern von Personenlisten und von Mailinglistenempfängern mit konfigurierbaren Filtern. Diese werden über das Webinterface konfiguriert. Das Webinterface wurde mit statischen Webtechnologien entwickelt und ist inzwischen ziemlich in die Jahre gekommen.

Eine Erneuerung dieser Komponente ist ein Wunsch vieler Kunden.

1.3 Detaillierte Aufgabenstellung

Mit dieser IPA soll ein neues UI mit Hotwire für die Persistierung von Filter-Parametern im Hitobito Generic-Wagon erstellt werden (rein Frontend).

- Die Ansichten zur Konfiguration für Filter der Personenlisten und Abonnemente werden mit dem neuen UI ersetzt.
- Die neuen Ansichten werden nach einem gegebenen Mockup umgesetzt. Dieses Mockup wurde vom Kandidaten in Zusammenarbeit mit einem UX Experten erarbeitet und muss als Grundlage für die Ausarbeitung des Interfaces verwendet werden. Des weiteren muss das Interface auf das visuelle Design der existierenden Applikation abgestimmt sein.
- Das Backend darf nicht angepasst werden, das heisst das neue Interface verwendet die bestehenden Endpunkte und schickt die Daten im selben Format wie das alte Interface. Dies muss mit automatisierten Tests sichergestellt werden.
- Formular zur Konfiguration von Personen-Listen Filter: Das bestehende Formular muss ersetzt werden durch eine neue Implementation mit den in Mittel und Methoden definierten Web Technologien. Diese neue Umsetzung muss es erlauben, dynamisch weitere Filterkriterien hinzuzufügen im Gegensatz zur alten Implementation welche mit einem statischen Formular implementiert ist.
- Formular zur Konfiguration von Abo-Empfänger Filter: Das bestehende Formular besteht aus mehreren Teilen, wovon im Rahmen der IPA nur der Teil für die Globalen Filterbedingungen angepasst werden muss. Wie bei den Personen-Listen Filter muss das Formular nun dynamisch implementiert werden. Die Formulare für die weiteren Filterbedingungen werden im Rahmen der IPA nicht angepasst.
- Code der während dieser IPA entsteht soll auf ein privates Github Repo gepushed werden. Die VFs haben dabei stets Lese-Rechte.
- Die Konventionen des Ruby Style Guide, des Rails Style Guide und für Git Commit Messages müssen eingehalten werden (siehe Mittel und Methoden).

Out of Scope - wird erst nach der IPA umgesetzt:

- Filterung für Rollen, Gruppen, Events, People bei Abonnements.
- Anpassungen der Ansicht in den anderen Wagons.
- Anpassungen der bisher bestehenden Tests in Hitobito welche die zu erweiternden Ansichten betreffen.

Weitere Anforderungen zu spezifischen Bewertungskriterien:

- G1: Dokumentation fachlicher und technischer Anforderungen: Die fachlichen und technischen Anforderungen müssen dokumentiert werden.
- G10: Konforme Implementierung und Versionierung: Applikationen und Schnittstellen müssen konform implementiert und versioniert werden.
- A13: Erhebung und Dokumentation der Bedürfnisse und Umfeld: Die Bedürfnisse und das Umfeld werden adäquat erhoben und dokumentiert.
- A15: Instruktion: Es wird für den Projektowner eine Instruktion durchgeführt. Diese muss dem Projektowner die relevanten Änderungen aufzeigen.
- C11: Einsatz von KI-Modellen: Wir setzen bei Puzzle KI in Form von Kopiloten und Chatbots als Hilfsmittel ein. Die Lernenden werden im sinnvollen Einsatz von solcher KI geschult. Dies umfasst z.B. den Umgang in Bezug auf Output Validierung, Transparenz und Sicherheit. Die IPA soll möglichst repräsentativ für unseren Alltag als Entwickler sein, dementsprechen darf KI ein Teil davon sein.
- G5: Risikoanalyse und Sicherheitsmassnahme: Sicherheitsrisiken von Applikationen und Schnittstellen müssen identifiziert und adressiert werden.
- G6: Entwicklung und Anpassung des Anforderungskatalogs: Ein Anforderungskatalog für Sicherheitsmassnahmen von Applikationen und/oder Schnittstellen muss erstellt oder angepasst werden.
- User Experience und visuelles Design: Das Feature muss visuell gut gestaltet sein um die Usability und Nutzerfreundlichkeit des Features sicherzustellen.

- Versionsverwaltung mit Git (Source Code): Die Versionsverwaltung mit Git muss gemäss den Best Practices erfolgen. Es müssen sprechende und einheitliche Commit-Messages geschrieben werden und commit-spezifische Inhalte müssen passend zur Message sein und unter der Einhaltung der Firmenguidelines erfolgen.
- Bewertung von Aussagen: Aussagen in der Arbeit müssen klar zwischen persönlichen Meinungen und auf Quellen basierenden Informationen differenziert werden.

1.3.1 Mittel und Methoden

Technologie und Plattform:

- Ruby, Ruby on Rails, Active Record
- HTML, CSS, Javascript, Hotwire
- PostgreSQL
- Git

Entwicklungsumgebungen:

- IntelliJ
- Visual Studio Code
- Github
- Rake
- Rubocop

Textverarbeitung und Diagramme

- Latex
- draw.io

1.3.2 Vorkenntnisse

Marc arbeitet bereits seit einigen Monaten an Features von Hitobito. Ausserdem hat er bereits seit dem 2. Lehrjahr Erfahrung auch in anderen Ruby on Rails Projekten gesammelt.

1.3.3 Vorarbeiten

- Vorbereitung Dokumentvorlage
- Probe-IPA: Vereinheitlichung der Personenlisten- und Abonnementsfilterlogik im Backend
- Entwurf eines Mockups

1.3.4 Neue Lerninhalte

- Eigenständiges Umsetzen eines Designs nach gegebenem Mockup
- Eigenständiges Projektmanagement während der IPA

1.3.5 Arbeiten in den letzten 6 Monaten

- Umsetzung diverser Features und Bugfixes für Hitobito (Ruby on Rails)
- Probe-IPA: Vereinheitlichung der Personenlisten- und Abonnementsfilterlogik
- PostgreSQL Migration Hitobito
- Ruby on Rails Major Upgrade Hitobito

2 Firmenstandards

2.1 Code conventions

Als Code convention werden die Ruby [Style Guides](#) verwendet. Die Überprüfung dieser Style Guidelines wird mit Rubocop (Formatter) sichergestellt. Die Konfiguration dieses Formatters ist unter [rubocop.yml](#) ersichtlich.

2.1.1 Mehrsprachigkeit

Hitobito ist eine mehrsprachige Applikation. Alle Erweiterungen oder Anpassungen müssen in Deutsch übersetzt werden. Übersetzungen werden in einer Übersetzungsdatei gespeichert oder können vom Kunden in einem Tool namens Transifex verwaltet werden.

2.1.2 Lizenz

Hitobito ist ein Open Source Projekt. In jedem File in Hitobito wird das Copyright für den jeweiligen Kunden in Kommentarform beschrieben. Diese Lizenz- und Kundeninformationen können über folgenden Befehl eingefügt werden:

```
rake license:insert
```

Die daraus entstehende Lizenz sieht wie folgt aus:

```
1  # Copyright (c) 2012 -2021 , hitobito AG . This file is part of
2  # hitobito and licensed under the Affero General Public License version 3
3  # or later . See the COPYING file at the top - level directory or at
4  # https :// github . com / hitobito / hitobito .
```

Alternativ dazu können diese Informationen mit

```
rake license:remove
```

entfernt oder mit

```
rake license:update
```

aktualisiert werden.

2.2 Git Commit Message Conventions

Die Git Commit Messages werden nach den Regeln von Puzzle ITC formuliert. Im Anhang unter Git Commit Message Conention finden sie eine Kopie der Firmenkonventionen. Diese wurden basierend auf folgendem Tutorial definiert: [Tutorial](#)

- Sprache: Englisch
- Kurze und prägnante Message, idealerweise unter 50 Zeichen
- Mit Grossbuchstaben beginnen
- Kein Punkt am Schluss
- Den *imperative mood* (Befehlsform) verwenden, also «Fix bug with X» statt «Fixed bug with X» oder «More fixes for broken stuff»
- Wenn vorhanden Ticket referenzieren:
 - Bei Open Project Work Packages: «Add X, refs #12345»
 - Bei Gitlab/Github Issues: «Add X #12345»

3 IPA-Schutzbedarfanalyse

3.1 Datensicherheit

3.2 Applikationssicherheit

4 Organisation der IPA-Ergebnisse

4.1 Datensicherung

In dieser IPA unterteilen wir die Datensicherung in:

- Dokumentation
- Code

4.1.1 Dokumentation

Dokumentation	
Tools	Git und USB
Versioniert	Ja
Interval	Mind. 2x täglich
Beschreibung	Die Dokumentation ist im ipa-puzzle-template Repository unter dem Branch probe-ipa angelegt. Sobald ein Dokumentationsticket abgeschlossen wurde, werden die Änderungen auf den Github Server in das private Repository gepushed. Dies geschieht mind. 2x täglich. Zusätzlich, wird pro Tag ein Ordner auf einem USB-Stick erstellt. Am Ende des Tages wird eine Kopie der Dokumentation in diesen Ordner geladen.

Tabelle 4.1: Sicherung Dokumentation

4.1.2 Code

Code	
Tools	Git und USB
Versioniert	Ja
Interval	Mind. 2x täglich
Beschreibung	Für die Entwicklung wurden die Repositories hitobito und hitobitogeneric geforked. Auf diesen Repositories wird an Tagen an welchen entwickelt wird, mind. 2x täglich committed. An diesen Tagen wird zur doppelten Sicherung zusätzlich eine Kopie des Projektes auf den USB Stick gespeichert, unter dem Ordner des jeweiligen Tages.

Tabelle 4.2: Sicherung Code

4.1.3 Wiederherstellung des Codes

Im Falle eines Datenverlusts, können die Daten entweder über das Github Repository oder den USB-Stick wiederhergestellt werden. Bei der Wiederherstellung mit Git wird der SSH-Key für die Klonung des Repositories benötigt. Ist dieser SSH-Key nicht verfügbar, wird die Wiederherstellung über den USB-Stick vorgenommen und das Projekt des letzten Speicherstandes kopiert. Im Falle des USB-Sticks ist mit mehr Datenverlust zu rechnen, falls der Datenverlust gegen Mittag oder Nachmittag auftritt, da die Speicherung erst am Ende des Tages erfolgt. Aus diesem Grund ist die Datenwiederherstellung mit Git zu bevorzugen.

Die Nachweise für die jeweiligen Datensicherungen finden sie im Anhang.

4.2 Änderungskontrolle

5 Projektmethode

Die verwendete Projektmethode dieser IPA ist Scrum. Im folgenden Abschnitt wird der Einsatz, Abweichungen, Werkzeuge und Begründung der Wahl dieser Projektmethode beschrieben. Des weiteren beschreibt dieser Abschnitt die Definition of Done (DoD).

5.1 Einsatz von Scrum

5.1.1 Sprints

Die IPA wird insgesamt in drei Sprints unterteilt. Jedem Sprint wird eine Phase der Arbeit zugewiesen. Die Aufteilung ist wie folgt:

- Sprint 1: Initialisierung
- Sprint 2: Umsetzung
- Sprint 3: Finalisierung

5.1.2 Verwaltungstool

Als Verwaltungstool wird Github Projects eingesetzt. Das Board hierzu kann unter [Github Board](#) aufgerufen werden. Das Board ist in sechs Spalten unterteilt:

- Backlog: User-Stories werden grob erfasst, keine Details nötig.
- Refinement: User-Stories werden genauer Beschrieben und Akzeptanzkriterien werden definiert.
- Ready: User-Story wurde refined und geschätzt. Sie kann jetzt bearbeitet werden.
- In-Progress: User-Story wird momentan bearbeitet.

- In-Review: User-Story wurde abgeschlossen, alle Akzeptanzkriterien sind erfüllt.
- Done: User-Story erfüllt DoD (Definition of Done).

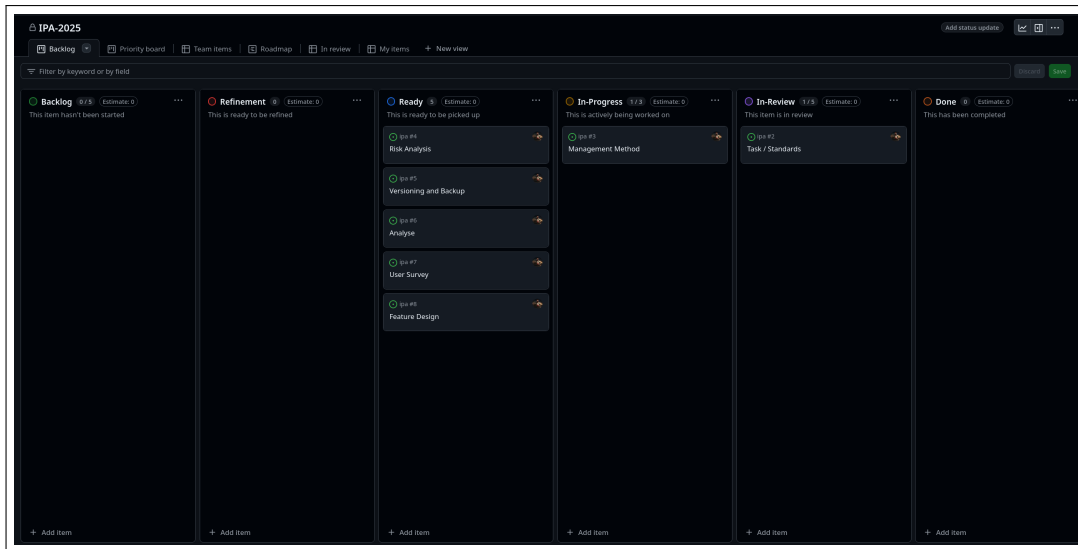


Abbildung 5.1: Github Projects Board

5.1.3 Meetings

Sprint Planning

Zu Beginn eines Sprints werden werden alle Aufgaben in Form von User-Stories im Backlog erfasst. Die Stories werden anschliessend refined und danach geschätzt. Das Sprint Planning umfasst den Prozess der Erfassung von User-Stories, deren Refinement und Schätzung. Konnten im letzten Sprint die geplanten User-Stories nicht alle abgeschlossen werden, umfasst das Planning zusätzlich das Neurefinement und die Neuschätzung dieser User-Stories. Anwesend beim Sprint Planning ist ausschliesslich der Kandidat.

Dailies

Während eines Sprints wird jeden Tag um 09:00 Uhr ein Daily durchgeführt. Das Daily findet bei Puzzle ITC im Sitzungszimmer SSudoßstatt. Anwesend sind dabei der Kandidat, die verantwortliche Fachkraft und die zusätzliche verantwortliche Fachkraft. Ausgenommen von dieser Regel ist der erste Tag der IPA (04.03.2025) an welchem kein Daily durchgeführt wird. Grund dafür ist, dass zu diesem Zeitpunkt noch keine Organisation und

Projektvorgehensweise definiert wurde und die ersten Prozesse von Scrum erst ab dem 2. Tag eintreffen können.

Im Daily ist es dem Kandidat möglich, Fragen an seine verantwortlichen Fachkräfte zu stellen. Jedes Daily wird protokolliert. Die Protokolle der Dailies können unter Daily-Protokolle eingesehen werden.

Sprintabschlüsse

Nach jedem Sprint findet ein einstündiges Meeting für den Sprintabschluss statt. Darin werden die abgeschlossenen User-Stories in der In-Review-Spalte verifiziert. Erfüllt die hinterlegte Arbeit alle Akzeptanzkriterien wird die User-Story auf Done geschoben. Sind die Akzeptanzkriterien nicht erfüllt, wird die User-Story auf Refinement geschoben. Anwesend beim Sprintabschluss ist ausschliesslich der Kandidat. In Folge des Sprintabschlusses wird das Sprint Planning durchgeführt.

5.1.4 Abweichungen

Trotz der Verwendung von Scrum, wurden Änderungen an der Definition dieser Projektvorgehensmethode vorgenommen. Grund dafür ist, dass Scrum durch die Änderungen besser auf die IPA zugeschnitten ist.

Schätzung

Scrum verzichtet auf Schätzungen in Personenstunden und verwendet deswegen eine Währung namens SStory Points". Story Points werden der [Fibonacci-Zahlenreihe](#) folgend vergeben. Der Sinn dabei ist, der Schätzung einer User-Story nach Personenstunden auszuweichen.

Dieses Konzept wird in dieser IPA verworfen, um in der Lage zu sein einen Zeitplan mit genauen Angaben in Personenstunden zu erstellen. Dies macht es dem Kandidaten möglich besser einzuschätzen, wie gut er in der Zeit liegt.

Abnahme Akzeptanzkriterien

Nach Scrum werden User-Stories vom Product Owner abgenommen. Um ständige Meetings mit dem Product Owner von Hitobito und den mithergehenden Zeitverlust zu vermeiden, werden die User-Stories vom Kandidaten selbst abgenommen. Den Prozess dazu finden ist unter Sprintabschlüsse ersichtlich.

Sprint Retro

Das Sprint Retro bietet dem Product Owner eine Möglichkeit einen Überblick über die Stimmung im Entwicklerteam zu erhalten. Sprint Retros finden im Geschäftsalltag Monatsweise statt. Auf das Sprint-Retro wird in dieser Arbeit verzichtet. Grund ist der kleine Zeitrahmen der IPA, welcher es unnötig macht ein solches Meeting durchzuführen.

5.2 Definition of Done

Die Definition of Done definiert wann eine User-Story abgeschlossen werden kann. Eine User-Story kann erst abgeschlossen werden, wenn sie alle Kriterien der Definition of Done erfüllt. Im Rahmen der IPA werden zwei Definition of Done's verwendet. Eine für User-Stories welche den Code betreffen, eine zweite für User-Stories welche die Dokumentation betreffen.

5.2.1 DoD Code

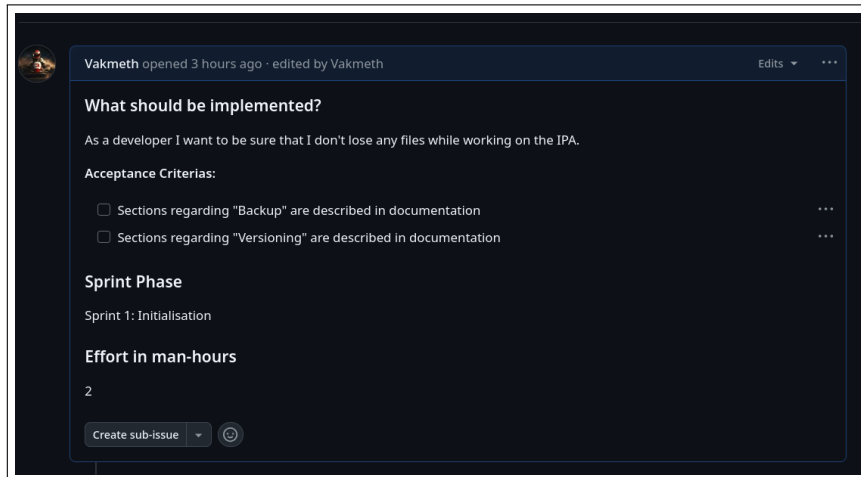
- Nur notwendige Konsolenausgaben vorhanden
- Feature relevante Tests vorhanden
- Sprechender Code implementiert
- Nicht verwendete Methoden gelöscht
- Feature manuell getestet
- Alle Akzeptanzkriterien erfüllt

5.2.2 DoD Dokumentation

- Definierte Sektion beschreiben
- Kriterien aus Kriterienkatalog erfüllt
- Kriterien gemäss Dokumentenvorlage erfüllt
- Keine Grammatik- / Rechtschreibfehler vorhanden
- Quellen angegeben

5.2.3 Akzeptanzkriterien

Die Akzeptanzkriterien einer User-Story werden im dazugehörigen Ticket verwaltet. Jede User-Story wurde nach einem definierten Template erstellt, welches in Github hinterlegt wurde. Eine User-Story kann folgendermassen aufgebaut sein:



The screenshot shows a GitHub issue template for a User Story. The template is titled 'Vakmeth opened 3 hours ago · edited by Vakmeth'. It includes the following sections:

- What should be implemented?**
As a developer I want to be sure that I don't lose any files while working on the IPA.
- Acceptance Criteria:**
 - ☐ Sections regarding "Backup" are described in documentation
 - ☐ Sections regarding "Versioning" are described in documentation
- Sprint Phase**
Sprint 1: Initialisation
- Effort in man-hours**
2

At the bottom, there is a 'Create sub-issue' button and a 'Close' button.

Abbildung 5.2: Example of User Story

5.3 Verwendungsgrund

Die Projektvorgehensmethode wurde so gewählt, da sie für die IPA mehrere Vorteile bringt:

- **Sprint Ende:** SCRUM zwingt den Entwickler dazu am Ende des Sprints ein vorzeigbares Produkt zu haben
- **Agilität:** Wenn eine Story nicht erreicht wurde, kann sie in den nächsten Sprint gezogen werden
- **Daily:** Durch die Dailies wird ein täglicher Austausch zwischen Fachkraft und Kandidat sichergestellt
- **Akzeptanzkriterien:** Mit den Kriterien verhindern wir das abschliessen von halbfertigen Features oder fehlerhafter Software
- **Board:** Durch das Github Projects Board ermöglichen wir eine schnelle Übersicht über den Stand der IPA

6 Projektaufbauorganisation

6.1 Projektrollen Scrum

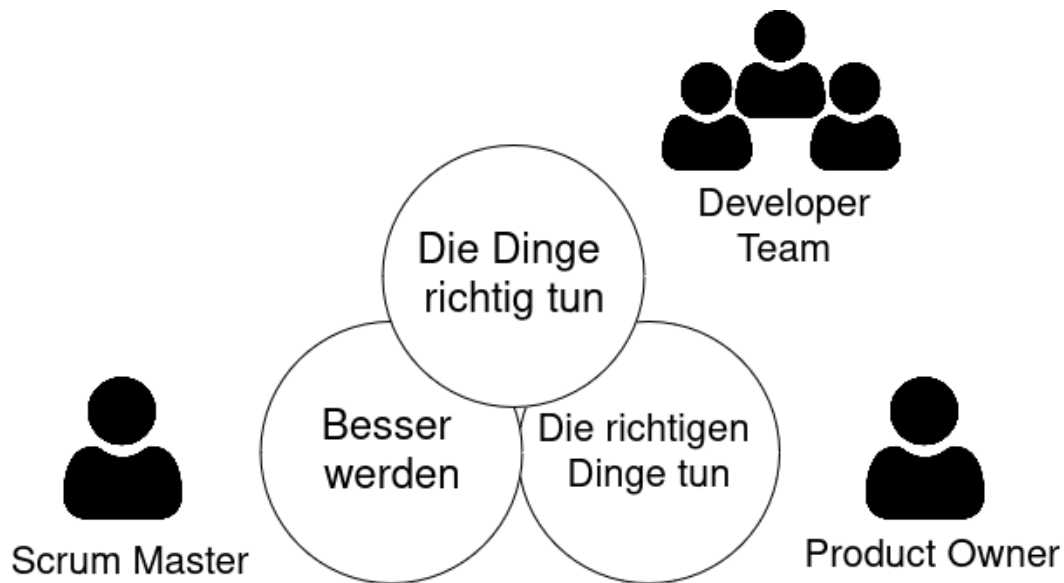


Abbildung 6.1: Rollen in Scrum, selbstgezeichnet mit Draw.io

Rollenbeschreibung	
Product Owner	Der Product Owner vertritt die Interessen des Kunden. Er priorisiert die Aufgaben im Product Backlog
Scrum Master	Der Scrum Master unterstützt die Entwickler und beseitigt Hindernisse. Er sorgt für eine kontinuierliche Verbesserung in der Arbeit.
Entwicklerteam	Das Entwicklerteam arbeitet selbstorganisiert den Sprint Backlog ab. Durch Dailies wird ein laufender Informationsaustausch sichergestellt.

Tabelle 6.1: Rollenbeschreibung

6.2 Projektrollen IPA

Rollenbeschreibung	
Verantwortliche Fachkraft	Unterstützt den Kandidaten von seiten des Lehrbetriebes. Erste Anlaufstelle bei Problemen.
Zusätzliche verantwortliche Fachkraft	Unterstützung für die verantwortliche Fachkraft
Experten	Validierungsexperte: Validiert die IPA-Aufgabenstellung. Hauptexperte: Verantwortlich für die Bewertung der IPA. Nebenexperte: Unterstützung für den Hauptexperten.

Tabelle 6.2: Rollenbeschreibung

6.3 Projektrollen Scrum in der IPA

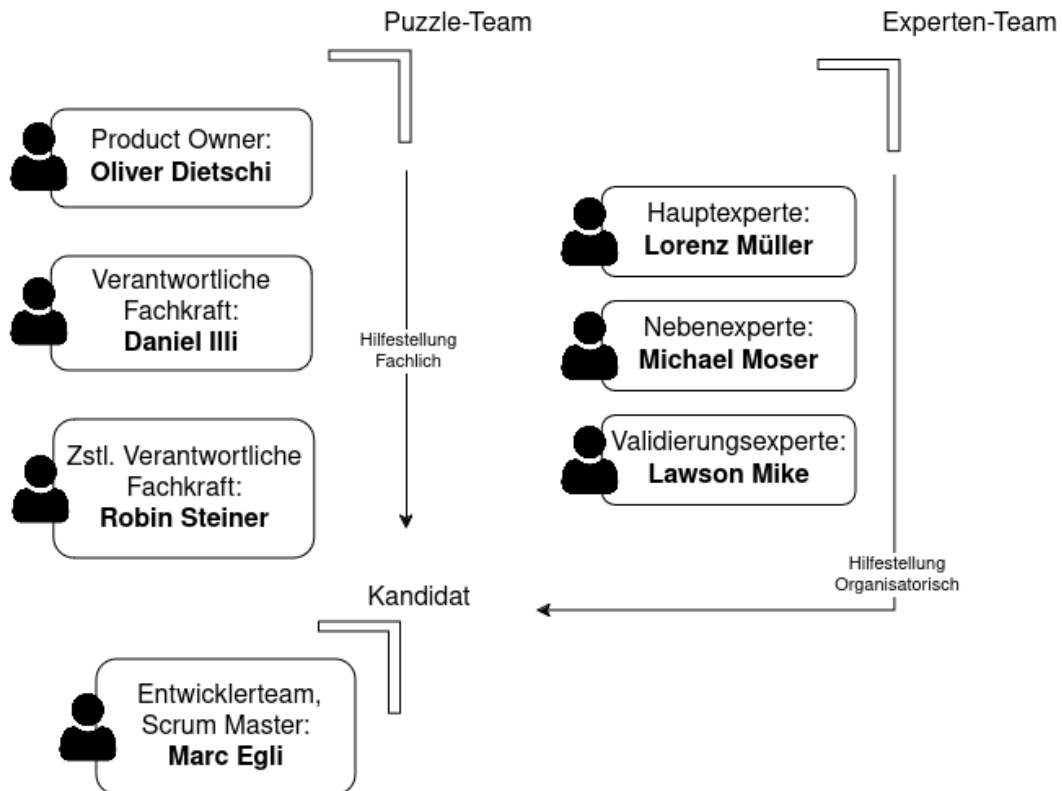


Abbildung 6.2: Rollenverteilung in der IPA, selbstgezeichnet mit Draw.io

Rollenbeschreibung IPA	
Verantwortliche Fachkraft	Daniel Illi
Zusätzliche verantwortliche Fachkraft	Robin Steiner
Validierungsexperte	Lawson Mike
Hauptexperte	Lorenz Hess
Nebenexperte	Michael Moser
Scrum Master	Marc Egli
Development Team	Marc Egli
Kandidat	Marc Egli

Tabelle 6.3: Rollenbeschreibung IPA

7 Zeitplan

7.1 Erläuterung zum Zeitplan

7.2 Sprints

8 Arbeitsjournale

8.1 Tag 1: 04.03.2025

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Raum einrichten, Kriterien aufhängen	Marc Egli	1	1
Zeitplan erstellen	Marc Egli	1	1
Sprint Planning	Marc Egli	1	1.5
Task / Standards beschreiben	Marc Egli	1	1
Management Method beschreiben	Marc Egli	2	1.5
Risikoanalyse beschreiben	Marc Egli	2	2.75
Arbeitsjournal schreiben	Marc Egli	1	0.5
Total		8.25	9.25

Tabelle 8.1: Tätigkeiten Tag 1

Tagesablauf

Ich startet heute morgen um 07.45 Uhr mit der IPA. Als erstes begann ich damit, den Raum einzurichten, was bedeutet: Dockingstation anschliessen, Wasser bereitlegen und alle Kriterien meiner IPA aufhängen. Danach habe ich alle Kriterien mit verschiedenen Farben unterteilt: Blau steht für Kriterien, welche über die gesamte IPA hinweg zählen, Rosa für Kriterien welche in der Umsetzung zu beachten sind und Gelb für Kriterien welche die Dokumentation betreffen. Als ich mit der Zimmereinrichtung fertig war, startete ich direkt mit dem Zeitplan. Ich passte das Template, welches ich vorbereitet habe, auf die Dauer der IPA an und machte alles bereit um die ersen User-Stories einzutragen. Nachdem der Zeitplan fertig war, startet ich das Sprint Planning. Darin organisierte ich als erstes das Daily mit meiner verantwortlichen Fachkraft und meiner zusätzlichen verantwortlichen Fachkraft. Das Daily setzte ich auf 09:00 Uhr morgens an.

Später im Planning habe ich alle nötigen User Stories für den kommenden Sprint definiert und diese anschliessend in den Zeitplan mit der

dazugehörigen Schätzung übertragen. Auf der Uhr war nun schon 11:15 Uhr. Ich startete den ersten Teil des Beschriebes der Aufgabenstellung und der Firmenstandards und ging danach in den Mittag.

Nach dem Mittag beendete ich den Beschrieb der Aufgabenstellung und der Firmenstandards und begann mit der Projektvorgehensmethode. Hier kam ich überraschend schnell durch und konnte so nach 1.5 Stunden die Risikoanalyse beginnen an welcher ich bis kurz vor dem Schluss des Tages, 17:30 gearbeitet habe. Beim erstellen der Risikoanalyse, bemerkte ich, dass ich noch Fragen zum Berechtigungskonzept in Hitobito hatte. Dementsprechend ging ich zu Niklas Jäggi, da er gerade zu gegen war, welcher mir dann das Konzept erklärte. Ganz am Ende schrieb ich dann noch das Arbeitsjournal.

Hilfestellungen

- Niklas Jäggi: Erklärung des Berechtigungsaufbaus in Hitobito

Reflexion

Was lief gut

Der Einstieg lief meiner Meinung nach sehr gut. Ich kam schnell voran und konnte die ersten paar Teile der Dokumentation beschreiben. Sogar das erste Kriterium, A11 (Projektaufbauorganisation) konnte ich scho abschliessen, was mich sehr motiviert hat.

Was lief weniger gut

Obwohl ich schnell voran kam, habe ich heute dennoch den geplanten Aufwand um 1/4-Stunde überschossen. Hier muss ich aufpassen, dass ich unbedingt früher anfangen das Arbeitsjournal zu schreiben. Zusätzlich hatte ich beim Sprint Planning ein Problem mit dem Erstellen eines Issue-Templates. Ich hatte mich spontan dazu entschieden, dass es sehr hilfreich wäre, ein Template zu haben, in welchem man neue Issues während der IPA erfassen kann und so nicht alles immer neu machen muss. Allerdings hatte ich noch nie ein solches Template erstellt, wesegen das Planning dann auch eine 1/2-Stunde mehr Zeit in Anspruch nahm als geplant.

Meine Erkenntnisse von heute

Nicht zu viel Zeit mit Themen verlieren, in welchen ich wenig Erfahrung habe. Besser wäre es gewesen mit dem Issue-Template und dann in einem Daily danach zu fragen. Dennoch kann ich nun das Wissen um die Erstellung dieses Templates schon als ersten Erfolg in dieser IPA verbuchen.

Nächste Schritte

Morgen werde ich eine Zusammenfassung der Risikoanalyse verfassen um das Kriterium G5 (Risikoanalyse und Sicherheitsmassnahmen) abzuschliessen. Danach werde ich weiter am Board arbeiten, dass heisst als nächstes die Sektionen Versionierung und Backup in der Dokumentation beschreiben. Zusätzlich findet am Morgen noch der erste Expertenbesuch statt, welcher mir perfekt dient, um meinen vorbereiteten Fragenkatalog abzuarbeiten. Hier werde ich sicher Fragen zu organisatorischen Bereichen der IPA stellen, wie dem Zeitplan, Diagrammen oder dem Code-Anhang.

8.2 Tag 2: 05.03.2025

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Risikoanalyse abschliessen	Marc Egli	0	0.75
Daily Meeting	Marc Egli, Robin Steiner, Daniel Illi	0.25	0.5
Expertenbesuch	Marc Egli, Robin Steiner, Daniel Illi, Lorenz Müller	1.5	1.5
Versionierung und Backup	Marc Egli	2	0.75
Arbeitsjournal	Marc Egli	0.25	0.5
Total		4	4

Tabelle 8.2: Tätigkeiten Tag 2

Tagesablauf

Am morgen startete ich mit dem Abschluss der Risikoanalyse. Ich mit dem Abschluss der Risikoanalyse. Danach fand unmittelbar das Daily statt. Im Daily präsentierte ich den verantwortlichen Fachkräften den Stand der IPA. Danach stellte ich eine Rückfrage an Daniel Illi bezüglich des Berechtigungskonzeptes, da ich 100% sicher sein wollte, das die Informationen welche ich von Niklas Jäggi bezogen haben stimmen. Die Nachfrage ergab, dass das Berechtigungskonzept stimme, jedoch ein Diagramm dies noch falsch abbildete. Ich notierte mir somit die Änderung welche ich an diesem Diagramm noch machen muss und schloss das Daily ab. Nebst der Nachfrage zum Berechtigungskonzept, fragte ich ob es in Ordnung sei, wenn ich reale Personen-Namen in einem Diagramm verwende. Z.B. Heinz statt User 1. Hierzu bekam ich das OK meiner verantwortlichen Fachkräfte.

Nach dem Daily fand dann der Expertenbesuch statt. Das Sitzungsprotokoll hierzu habe ich im Anhang hinterlegt. Der Besuch lief gut und ich konnte vieles daraus mitnehmen unter anderem das ich eine Person für das Gegenlesen auswählen darf (ohne diese angeben zu müssen). Ausserdem bekam ich weitere Inputs betreffend dem Zeitplan und meiner Kriterien.

Nach dem Expertenbesuch begann ich mit der Sektion zur Versionierung und der Backup-Strategie meiner IPA. Diese konnte ich zeitig abschliessen und danach das Arbeitsjournal verfassen.

Hilfestellungen

- Daniel Illi: Nachfrage bezüglich des Berechtigungskonzeptes
- Robin Steiner und Daniel Illi: Nachfrage der Verwendung von Echtnamen in Diagrammen

Reflexion

Was lief gut

Der Tag heute war vor allem dem Expertenbesuch gewidmet, welcher ich sehr positiv fand. Obwohl es noch ein paar Anpassungen zu machen gibt, so denke ich das durch die Hinweise meines Hauptexperten diese IPA gut herauskommen wird. Wichtig ist jetzt, dass ich das Protokoll für diesen Besuch verfasse und alle gewünschten Änderungen umsetze.

Was lief weniger gut

Heute hatte ich den Eindruck das nichts negativ gelaufen ist. Obwohl es ein paar Fehleinschätzungen im Zeitplan gab, bin ich dennoch immer noch auf Kurs.

Meine Erkenntnisse von heute

Alle Erkenntnisse welche ich im Sitzungsprotokoll vermerkt habe. Ausserdem nehme ich noch einen Satz meines Hauptexperten mit: "Nachvollziehbarkeit ist wichtig". Für mich heisst das, alles so klar wie möglich in der Dokumentation zu beschreiben und stets einen Blick auf die Kriterien zu werfen.

Nächste Schritte

Morgen werde ich damit verbringen die Analyse und die Bedürfniserhebung vorzubereiten. Zusätzlich werde ich die gesammelten Änderungsvorschläge meines Hauptexperten in einem Sitzungsprotokoll aufführen und im Anhang hinterlegen. Die gewünschten Änderungen werde ich dann ebenso direkt umsetzen.

8.3 Tag 3: Datum

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Tätigkeit	Personen	Stunden soll	Stunden ist
Total		Gesamt-stunden soll	Gesamt-stunden ist

Tabelle 8.3: Tätigkeiten Tag 1

Tagesablauf

Hilfestellungen

- Person: Hilfestellung

Reflexion

Was lief gut

Was lief weniger gut

Meine Erkenntnisse von heute

Nächste Schritte

8.4 Tag 4: Datum

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Tätigkeit	Personen	Stunden soll	Stunden ist
Total		Gesamt-stunden soll	Gesamt-stunden ist

Tabelle 8.4: Tätigkeiten Tag 1

Tagesablauf

Hilfestellungen

- Person: Hilfestellung

Reflexion

Was lief gut

Was lief weniger gut

Meine Erkenntnisse von heute

Nächste Schritte

8.5 Tag 5: Datum

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Tätigkeit	Personen	Stunden soll	Stunden ist
Total		Gesamt-stunden soll	Gesamt-stunden ist

Tabelle 8.5: Tätigkeiten Tag 1

Tagesablauf

Hilfestellungen

- Person: Hilfestellung

Reflexion

Was lief gut

Was lief weniger gut

Meine Erkenntnisse von heute

Nächste Schritte

8.6 Tag 6: Datum

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Tätigkeit	Personen	Stunden soll	Stunden ist
Total		Gesamt-stunden soll	Gesamt-stunden ist

Tabelle 8.6: Tätigkeiten Tag 1

Tagesablauf

Hilfestellungen

- Person: Hilfestellung

Reflexion

Was lief gut

Was lief weniger gut

Meine Erkenntnisse von heute

Nächste Schritte

8.7 Tag 7: Datum

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Tätigkeit	Personen	Stunden soll	Stunden ist
Total		Gesamt-stunden soll	Gesamt-stunden ist

Tabelle 8.7: Tätigkeiten Tag 1

Tagesablauf

Hilfestellungen

- Person: Hilfestellung

Reflexion

Was lief gut

Was lief weniger gut

Meine Erkenntnisse von heute

Nächste Schritte

8.8 Tag 8: Datum

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Tätigkeit	Personen	Stunden soll	Stunden ist
Total		Gesamtstunden soll	Gesamtstunden ist

Tabelle 8.8: Tätigkeiten Tag 1

Tagesablauf

Hilfestellungen

- Person: Hilfestellung

Reflexion

Was lief gut

Was lief weniger gut

Meine Erkenntnisse von heute

Nächste Schritte

8.9 Tag 9: Datum

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Tätigkeit	Personen	Stunden soll	Stunden ist
Total		Gesamt-stunden soll	Gesamt-stunden ist

Tabelle 8.9: Tätigkeiten Tag 1

Tagesablauf

Hilfestellungen

- Person: Hilfestellung

Reflexion

Was lief gut

Was lief weniger gut

Meine Erkenntnisse von heute

Nächste Schritte

8.10 Tag 10: Datum

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Tätigkeit	Personen	Stunden soll	Stunden ist
Total		Gesamt-stunden soll	Gesamt-stunden ist

Tabelle 8.10: Tätigkeiten Tag 1

Tagesablauf

Hilfestellungen

- Person: Hilfestellung

Reflexion

Was lief gut

Was lief weniger gut

Meine Erkenntnisse von heute

Nächste Schritte

8.11 Tag 11: Datum

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Tätigkeit	Personen	Stunden soll	Stunden ist
Total		Gesamt-stunden soll	Gesamt-stunden ist

Tabelle 8.11: Tätigkeiten Tag 1

Tagesablauf

Hilfestellungen

- Person: Hilfestellung

Reflexion

Was lief gut

Was lief weniger gut

Meine Erkenntnisse von heute

Nächste Schritte

8.12 Tag 12: Datum

Tätigkeiten	Beteiligte Personen	Aufwand Geplant (std)	Aufwand Effektiv (std)
Tätigkeit	Personen	Stunden soll	Stunden ist
Total		Gesamt-stunden soll	Gesamt-stunden ist

Tabelle 8.12: Tätigkeiten Tag 1

Tagesablauf

Hilfestellungen

- Person: Hilfestellung

Reflexion

Was lief gut

Was lief weniger gut

Meine Erkenntnisse von heute

Nächste Schritte

9 Persönliches Fazit

9.1 Was lief weniger gut

9.2 Was lief gut

9.3 Schlussreflexion

Teil II

Projektdokumentation

Hitobito: Neue Generation von Personen-Filtern
Autor: Marc Egli

10 Einführung

11 Analyse

11.1 Ist-Zustand

11.1.1 Personenlisten

11.1.2 Abonnemente

11.2 Soll-Zustand

11.3 Bedürfniserhebung

11.4 Anforderungen

11.4.1 Nicht funktionale Anforderungen

11.4.2 Funktionale Anforderungen

11.5 Abgrenzung

11.6 Benötigter Rahmen

11.6.1 Fehlende Informationen

11.7 Persönliche Vorgehensziele

12 Risikoanalyse und Sicherheitsmassnahmen

12.1 Schnittstellen

Action	Controller	Funktion
index	PeopleController	Diese Schnittstelle liefert alle Personen zurück, wobei diese durch den gegebenen Filter gefiltert werden. Der Filter kann entweder durch die Angabe einer Filter-ID oder dem Mitgeben von Parametern im Request definiert werden.
index	SubscriptionController	Diese Schnittstelle liefert alle Abonnemente zurück, wobei diese durch die definierten Filter gefiltert werden. Die Filter können über diverse Attribute bestimmt werden, im Rahmen dieser IPA sind allerdings ausschliesslich die globalen Bedingungen zu beachten, welche auf Maillinglisten gespeichert werden, welche wiederum mehrere Abonnenmente verwalten.

Tabelle 12.1: Schnittstellen

12.2 Benutzer und Datenzugriffe

Benutzer im Hitobito besitzen immer eine Rolle. Die Rolle des Benutzers bestimmt seine Berechtigungen. Die Berechtigungen welche ein User haben kann sind:

Name	Berechtigung
Group_Full	Hat Schreib- und Leserechte auf seiner Gruppe
Group_Read	Hat Leserechte auf seiner Gruppe
Layer_Full	Hat Schreib- und Leserechte auf seiner Gruppe und den Gruppen welche der Ebene dieser Gruppe unterliegen.
Layer_Read	Hat Leserechte auf seiner Gruppe und den Gruppen welche der Ebene dieser Gruppe unterliegen.
Layer_And_Below_Full	Hat Schreib- und Leserechte auf seiner Gruppe, allen Gruppen der Ebene dieser Gruppe und allen unterliegenden Ebenen.
Layer_And_Below_Read	Hat Leserechte auf seiner Gruppe, allen Gruppen der Ebene dieser Gruppe und allen unterliegenden Ebenen.

Tabelle 12.2: Berechtigungen

Um die Berechtigungen besser verständlich zu machen, dienen folgende Diagramme:

12.2.1 Datenstruktur

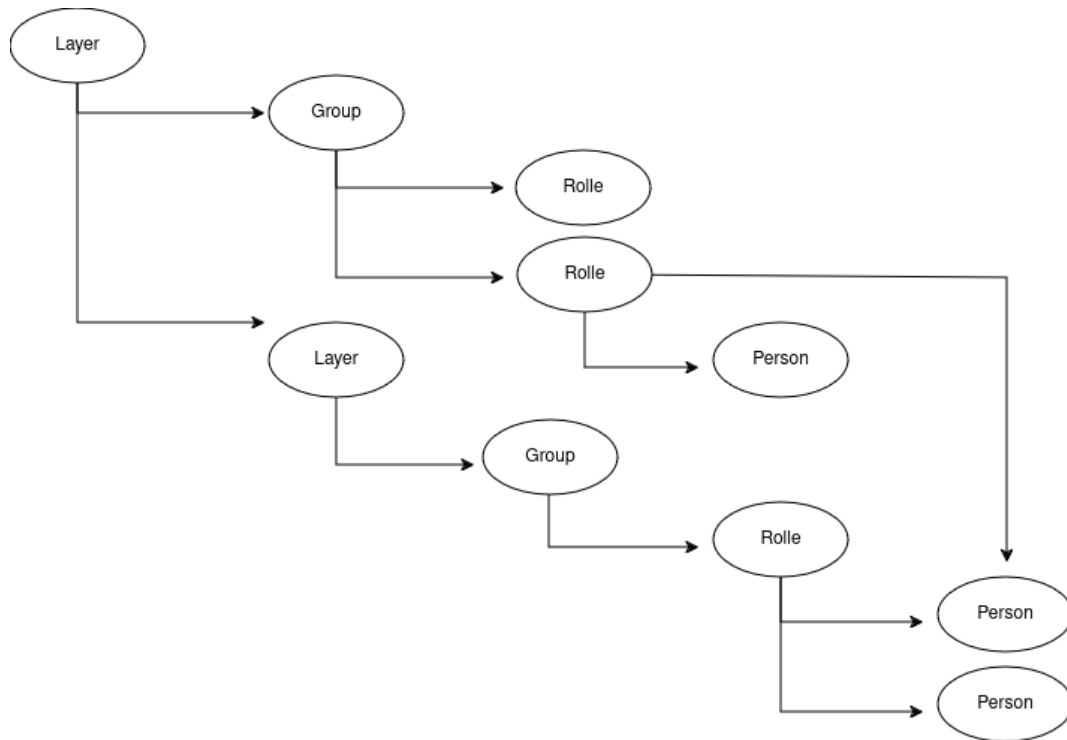


Abbildung 12.1: Gruppen und Ebenen, selbstgezeichnet mit Draw.io

Die Berechtigungen verwalten den Zugriff auf Layer und Gruppen. Ein Layer kann mehrere Gruppen haben, eine Gruppe besitzt mehrere Rollen und eine Rolle kann wiederum mehrere Personen besitzen. Personen können mehrere Rollen und somit eine Vielzahl von Berechtigungen besitzen.

12.2.2 Beispiel Zugriff Heinz

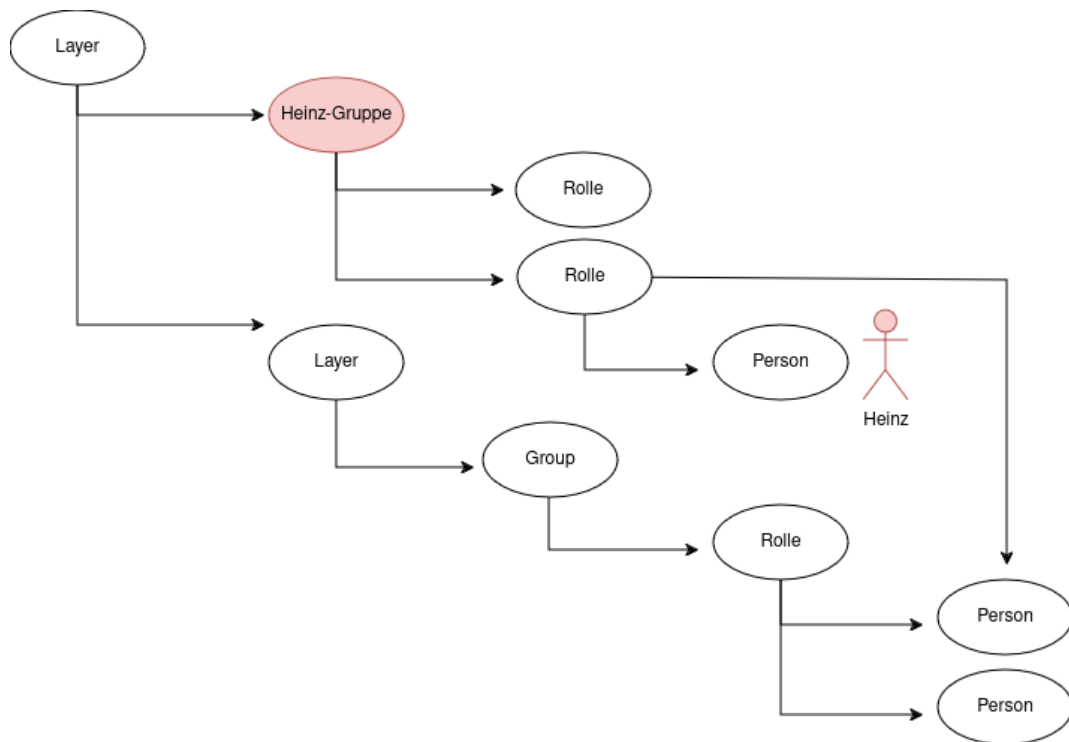


Abbildung 12.2: Beispiel Berechtigungen von Heinz, selbstgezeichnet mit Draw.io

Dieses Diagram erklärt das Beispiel der Berechtigung "Group_Full". Wir haben einen User namens Heinz in unserem System. Heinz besitzt eine Rolle welche mit der Heinz-Gruppe verknüpft ist. Die Rolle besitzt die Berechtigung "Group_Full".

Dank dieser Verknüpfung besitzt Heinz Schreib- und Leserechte auf die Heinz-Gruppe.

12.2.3 Beispiel Zugriff Tim

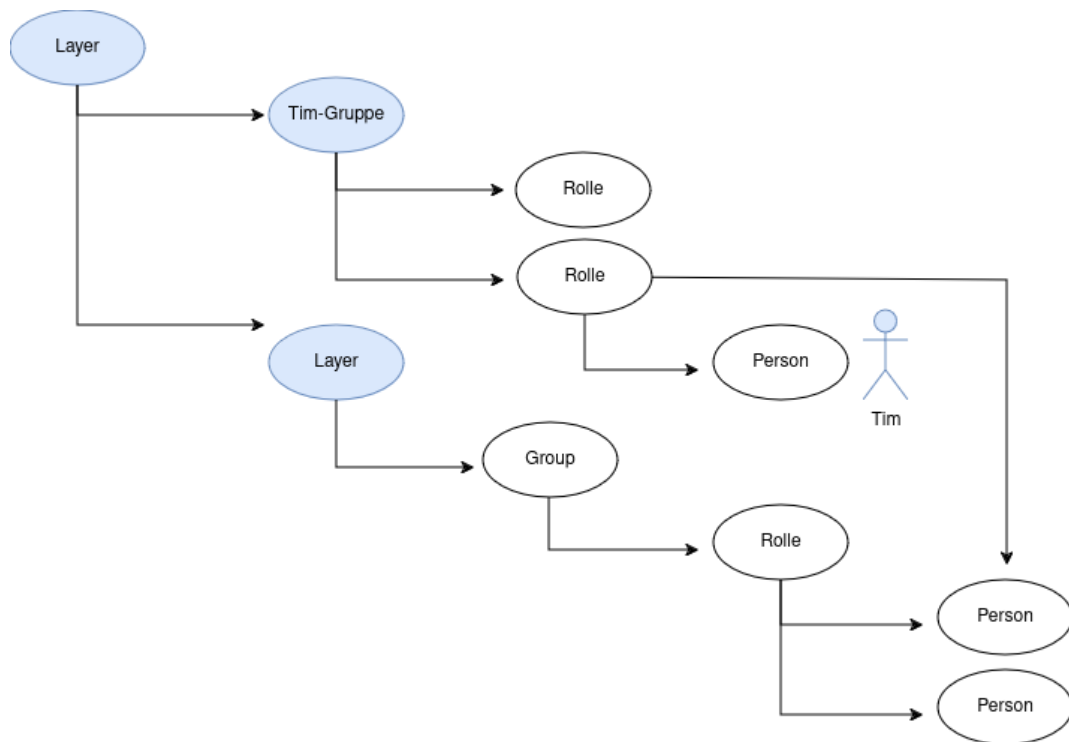


Abbildung 12.3: Beispiel Berechtigungen von Tim, selbstgezeichnet mit Draw.io

Dieses Diagramm erklärt das Beispiel der Berechtigung "Layer_Full". Wir haben einen User namens Tim in unserem System. Tim besitzt eine Rolle welche mit der Tim-Gruppe verknüpft ist. Die Rolle besitzt die Berechtigung "Layer_Full".

Durch diese Verknüpfung hat Tim Schreib- und Leserechte auf alle Gruppen, welche dem Layer seiner Gruppe unterliegen.

12.2.6 Risikoanalyse

Nr	Risikobeschreibung	Auswirkung	Vor Massnahme				Massnahmen	Nach Massnahme			
			W	S	Risiko	Handlungsweise		W	S	Risiko	Handlungsweise
1	Daten ausserhalb der Berechtigung eines Benutzers werden angezeigt	Benutzer kann verbotene Informationen einsehen	W2	S2	Niedrig	Risikominderung	Daten werden vor dem Anzeigen im Filter anhand der Berechtigungen des Benutzers gefiltert	W1	S1	Niedrig	Risikoakzeptanz
2	Benutzer kann einen Filter auf einer Ebene speichern, auf welcher er keinen Zugriff hat	Verwirrte Benutzer durch den neuen Filter	W2	S2	Niedrig	Risikominderung	Sicherstellen das der Benutzer nur Filter seiner Berechtigung entsprechend speichern kann.	W1	S1	Niedrig	Risikoakzeptanz
3	SQL-Injection in ein Filter Eingabefeld (XSS)	Datenbank kann ausgelesen oder verändert werden	W4	S4	Hoch	Risikominderung	Alle Eingaben des Benutzers escapen	W2	S1	Niedrig	Risikoakzeptanz
4	Bash-Injection in ein Filter Eingabefeld (XSS)	Schädliche Befehle werden serverseitig ausgeführt	W3	S4	Hoch	Risikominderung	Alle Eingaben des Benutzers escapen	W2	S1	Niedrig	Risikoakzeptanz
5	Falsche Verwendung einer Library	Schwachstelle der Library kann von Angreifern ausgenutzt werden	W2	S3	Mttel	Risikominderung	Dokumentation der Libraries gut durchgehen, diese auf Schwachstellen überprüfen	W2	S2	Niedrig	Risikoakzeptanz

Tabelle 12.3: Risikoanalyse

Schadensausmass:
S1 = führt zu keinem Schaden am Projekt
S2 = führt zu geringem Schaden
S3 = hoher Schaden
S4 = führt zu schwerem Schaden am Projekt
Eintrittswahrscheinlichkeit:
W1 = unvorstellbar
W2 = unwahrscheinlich
W3 = eher vorstellbar
W4 = vorstellbar
W5 = Eintreffen hoch

12.3 Risikomatrix

W5				
W4				③
W3				④
W2	③ ⑤ ④			⑤
W1	① ②			
	S1	S2	S3	S4

Tabelle 12.4: Risikomatrix

Legende:

- Risiko ohne Massnahme
- Risiko nach Massnahme
- Geringes Risiko
- Mittleres Risiko
- Hohes Risiko

12.4 Auswertung

Die aufgeführten Risiken sowie die entsprechenden Massnahmen wurden mit den Stakeholdern besprochen und von ihnen abgesegnet. Durch die Bestätigung der Stakeholder, werden die Massnahmen zur Risikominderung in der Anforderungskatalog überführt.

13 Entwurf

13.1 Anwendungskonzept

13.1.1 Anwendungsdiagramm

13.1.2 Anwendungsfälle

13.2 Systemkonzept

13.2.1 Betroffene Services

13.2.2 Status quo

13.2.3 Lösungsvarianten

13.2.4 Variantenentscheid

13.3 Sicherheitskonzept

13.3.1 SQL-Injection

13.3.2 Cross-Site Scripting

13.3.3 URL Interpretation

13.3.4 Kommunikation HTTP/S

13.4 Fehlerbehandlungskonzept

Version 1.0

5. März 2025

Seite 54 von 64

13.4.1 Nutzereingabe

13.4.2 Laufzeitfehler

14 Ausführung

14.1 Einsatz von KI-Modellen

14.2 Gems

14.2.1 can-can-can

14.2.2 dry-crud

15 Einführung

15.1 Instruktion

15.2 Unvorhergesehene Änderungen

15.2.1 application.rb

15.2.2 _list.html.haml

16 Sprintabschlüsse

16.1 Abschluss Sprint Initialisierung

16.1.1 Backlog

16.2 Abschluss Sprint Umsetzung

16.2.1 Backlog

16.3 Abschluss Sprint Finalisierung

16.3.1 Backlog

Teil III

Anhänge und Verzeichnisse

Hitobito: Neue Generation von Personen-Filtern
Autor: Marc Egli

17 Verzeichnisse

17.1 Code

17.2 Tabellenverzeichnis

1	IPA Daten	1
4.1	Sicherung Dokumentation	14
4.2	Sicherung Code	15
6.1	Rollenbeschreibung	21
6.2	Rollenbeschreibung	22
6.3	Rollenbeschreibung IPA	23
8.1	Tätigkeiten Tag 1	25
8.2	Tätigkeiten Tag 2	28
8.3	Tätigkeiten Tag 1	31
8.4	Tätigkeiten Tag 1	32
8.5	Tätigkeiten Tag 1	33
8.6	Tätigkeiten Tag 1	34
8.7	Tätigkeiten Tag 1	35
8.8	Tätigkeiten Tag 1	36
8.9	Tätigkeiten Tag 1	37
8.10	Tätigkeiten Tag 1	38
8.11	Tätigkeiten Tag 1	39
8.12	Tätigkeiten Tag 1	40
12.1	Schnittstellen	45
12.2	Berechtigungen	46
12.3	Risikoanalyse	51
12.4	Risikomatrix	52
18.1	Verwendete Abkürzungen	62
19.1	Glossar	63

17.3 Abbildungsverzeichnis

5.1	Github Projects Board	17
5.2	Example of User Story	20

6.1	Rollen in Scrum, selbstgezeichnet mit Draw.io	21
6.2	Rollenverteilung in der IPA, selbstgezeichnet mit Draw.io	23
12.1	Gruppen und Ebenen, selbstgezeichnet mit Draw.io	47
12.2	Beispiel Berechtigungen von Heinz, selbstgezeichnet mit Draw.io	48
12.3	Beispiel Berechtigungen von Tim, selbstgezeichnet mit Draw.io .	49
12.4	Beispiel Berechtigungen von Tim, selbstgezeichnet mit Draw.io .	50
20.1	Puzzle ITC Git commit conventions	64

Quellenverzeichnis

[Github Docs - Understanding connections between repositories]

<https://docs.github.com/en/repositories/viewing-activity-and-data-for-your-repository/understanding-connections-between-repositories>, (04.03.2025)

[Github Docs - Configuring issue templates] <https://docs.github.com/en/communities/using-templates-to-encourage-useful-issues-and-pull-requests/configuring-issue-templates-for-your-repository>, (04.03.2025)

[Leo - Translating] <https://dict.leo.org/german-english>, (04.03.2025)

[Icon made by Freeplk from <http://www.flaticon.com/>] https://www.flaticon.com/free-icon/user_1077114?term=person&page=1&position=1&origin=search&related_id=1077114, (04.03.2025)

[Agile Scrum Group - Product Owner] <https://agilescrumgroup.de/product-owner-aufgaben/>, (04.03.2025)

[Agile Scrum Group - Scrum Master] <https://agilescrumgroup.de/scrum-master-aufgaben/>, (04.03.2025)

[Agile Scrum Group - Entwickler] <https://scrumguide.de/entwickler/>, (04.03.2025)

18 Verwendete Abkürzungen

Abkürzung	Bedeutung
UML	Unified Modeling Language

Tabelle 18.1: Verwendete Abkürzungen

19 Glossar

Bezeichnung	Bedeutung
Hitobito	Community Management Tool

Tabelle 19.1: Glossar

20 Anhänge

20.1 Git Commit Message Convention

Konvention Commit Message

Falls keine besonderen Vorgaben durch den Kunden vorhanden, empfehlen wir – angelehnt an den Artikel [How to Write a Git Commit Message](#) – folgende Konvention zu verwenden:

- Sprache: Englisch
- Kurze und prägnante Message, idealerweise unter 50 Zeichen ([Details](#))
- Mit Grossbuchstaben beginnen ([Details](#))
- Kein Punkt am Schluss ([Details](#))
- Den *imperative mood* (Befehlsform) verwenden, also «Fix bug with X» statt «Fixed bug with X» oder «More fixes for broken stuff» ([Details](#))
- Wenn vorhanden das Ticket referenzieren:
 - Bei Open Project Work Packages: «Add X, refs #12345»
 - Bei Gitlab/Github Issues: «Add X #12345»

Dies entspricht grundsätzlich auch dem Stil wie ihn viele Open Source Projekte wie z.B. der [Linux Kernel](#), [Spring Boot](#), [Rails](#) oder auch [Git](#) selber anwenden.

Für grössere Projekte, bei welchen auch das Changelog automatisiert generiert wird, kann die [Conventional Commits](#) Spezifikation sinnvoll sein.

Abbildung 20.1: Puzzle ITC Git commit conventions

20.2 Daily-Protokolle

20.3 Sitzungsprotokolle

20.4 Git commit convention

20.5 Security conventions