

Google Chrome Erweiterung (Extension) entwickeln

PA-Dokumentation

Lukas Bucher (BUL)

Version 1.0

Projektinformationen

Projektname	Google Chrome Erweiterung (Extension) entwickeln
Verantwortliche Fachkraft	Boris Basic (BBI)
Autor	Lukas Bucher (BUL)
Lehrbetrieb	Suva
Projektdauer	21.02.2022 – 04.03.2022

Versionsliste

Datum	Version	Kommentar
21.02.2022	0.1	Initiale Erstellung des Dokuments
21.02.2022	0.2	Grobplanung und allgemeine Infos
22.02.2022	0.3	User Stories und Big Picture
23.02.2022	0.4	Finalisierte User Stories, Detailplanung und Bedienkonzept
02.03.2022	0.5	Testing
03.03.2022	0.6	Kurzfassung, Einleitung, Installationsanleitung, GUI, Erscheinungsbild ändern
04.03.2022	1.0	Lösungsvariante, Endsystem, Schlusswort, Verzeichnisse, Anhang, Finalisierung

Inhalt

1	Kurzfassung	5
1.1	Ausgangssituation.....	5
1.2	Umsetzung.....	5
1.3	Ergebnis.....	5
2	PA-Bericht Teil 1	6
2.1	Einleitung.....	6
2.1.1	Zum Dokument.....	6
2.1.2	Die Suva	6
2.1.3	Rahmenbedingungen	6
2.1.4	Versionierung	6
2.1.5	Projektaufbauorganisation.....	7
2.2	Aufgabenstellung gemäss PkOrg.....	8
2.2.1	Ausgangslage.....	8
2.2.2	Detaillierte Aufgabenstellung.....	8
2.2.3	Mittel und Methoden.....	9
2.2.4	Vorkenntnisse	10
2.2.5	Vorarbeiten	10
2.2.6	Neue Lerninhalte.....	10
2.2.7	Arbeiten in den letzten 6 Monaten	10
2.2.8	Individuelle Beurteilungskriterien.....	10
2.3	Planung	11
2.3.1	Projektmanagement.....	11
2.3.2	Meilensteine	11
2.3.3	Grobplanung	12
2.3.4	Detailplanung.....	13
2.4	Arbeitsjournal	16
2.4.1	Tag 1 21.02.2022	16
2.4.2	Tag 2 22.02.2022	16
2.4.3	Tag 3 23.02.2022	17
2.4.4	Tag 4 24.02.2022	18
2.4.5	Tag 5 25.02.2022	18
2.4.6	Tag 6 28.02.2022	19
2.4.7	Tag 7 01.03.2022	19
2.4.8	Tag 8 02.03.2022	20

2.4.9	Tag 9 03.03.2022	21
3	PA-Bericht Teil 2	22
3.1	Anforderungen	22
3.1.1	User Stories	22
3.1.2	User Story 1 Verwaltung der Extension	22
3.1.3	User Story 2 Anzeige der Hauptseite	24
3.1.4	User Story 3 Anzeige der Suva-SAFe-Daten	24
3.2	Testing	27
3.2.1	Testkonzept	27
3.2.2	Testfälle	27
3.2.3	Testprotokoll	34
3.3	Lösungsvariante	37
3.3.1	Big Picture	37
3.3.2	Bedienkonzept der Info-App «Suva SAFe»	37
3.3.3	Datenbezug	38
3.4	Endsystem	39
3.4.1	Installationsanleitung	39
3.4.2	GUI	40
3.4.3	Erscheinungsbild ändern	42
3.4.4	Mitarbeiter-Listen abrufen	45
3.4.5	Safe-Struktur erstellen	46
3.4.6	Navigation bauen	48
3.4.7	Detail-Informationen anzeigen	49
4	Schlusswort	51
5	Glossar	52
6	Quellenverzeichnis	53
7	Abbildungsverzeichnis	54
8	Tabellenverzeichnis	54
9	Anhang	56
9.1	Sourcecode	56
9.1.1	manifest.json	56
9.1.2	background.js	56
9.1.3	app.js	56
9.1.4	safe.js	57
9.1.5	options.js	69
9.1.6	index.html	71

9.1.7	safe.html.....	72
9.1.8	options.html.....	73
9.1.9	style.css	74
9.1.10	safe.css.....	77
9.1.11	options.css.....	80
9.1.12	dark.css.....	81
9.1.13	safe-dark.css	83

1 Kurzfassung

1.1 Ausgangssituation

Momentan wird in der Suva ein Programm namens «SuvaTel» als internes Telefonbuch verwendet. Es beinhaltet diverse Funktionen wie die Mitarbeiter-Suche, Auflistung wichtiger Telefonnummern sowie Abteilungs- und Gebäudeadressen. Außerdem kann man eine Darstellung der SAFe-Organisation der Suva einsehen.

Mit meiner Übungs-PA ermittelte ich mithilfe einer Entscheidungsmatrix und anhand der Erstellung eines PoCs die Möglichkeiten und Potenziale, welche der Einsatz einer Chrome Extension für die Suva haben könnte. Dabei kam ich zum Entschluss, dass es eine Sinnvolle Alternative zu Desktop- und Web-Applikationen darstellt und somit gut für eine interne Applikation verwendet werden könnte. Also befasse ich mich in meiner PA erneut mit einer Google Chrome Extension.

1.2 Umsetzung

Der Ablauf der Arbeit basiert auf der Projektmanagementmethode IPERKA. So werden in der Planungsphase erst Meilensteine, später konkrete Arbeitspakete geplant. Zudem werden die Anforderungen mittels User Stories festgelegt.

In der darauffolgenden Umsetzungsphase werden die definierten User Stories nach erstelltem Plan implementiert. Im anschliessenden Testing werden Testfälle geschrieben, welche auf die Akzeptanzkriterien der User Stories basieren und mithilfe dieser das Testing durchgeführt

1.3 Ergebnis

Das Ergebnis meiner praktischen Arbeit stellt die funktionsfähige Google Chrome Extension dar. Dabei ist die Grundlage für persönliche Einstellungen geschaffen. Das Ändern des Erscheinungsbildes ist dabei bereits als Feature entwickelt.

Die Extension beinhaltet eine Hauptseite, welche später beliebig mit weiteren Modulen erweitert werden kann. Das erste Modul ist implementiert. Darin wird die SAFe-Organisation der Suva in Form eines hierarchischen Baums dargestellt. Zu ARTs, Products, Teams und Mitarbeitern lassen sich Detail-Informationen einsehen.

2 PA-Bericht Teil 1

2.1 Einleitung

2.1.1 Zum Dokument

Zweck dieses Dokuments ist die Dokumentation meiner QV PA (Qualifikationsverfahren Praktische Arbeit). Dabei werden mein Vorgehen und das Endprodukt der Arbeit beschrieben. Der PA-Bericht umfasst zwei Teile. Im ersten wird die Organisation des Projekts beleuchtet. Dazu zählen die Voraussetzungen sowie Planungs- und Journal-Dokumentation. Der zweite Teil fokussiert sich auf den eigentlichen Inhalt der Arbeit und beinhaltet Spezifikationen zu Anforderungen und dem Endprodukt.

2.1.2 Die Suva

Die Schweizerische Unfallversicherung Suva ist ein wichtiger Teil des schweizerischen Sozialversicherungssystems. Als selbstständiges Unternehmen des öffentlichen Rechts versichert die Suva Menschen im Beruf und in der Freizeit.

Die Suva hilft nicht nur nach einem Unfall oder bei einer Berufskrankheit. Ihre Leistungen gehen über den obligatorischen Versicherungsschutz nach einem Unfall hinaus.

Die Suva ist die grösste Unfallversicherung der Schweiz. Sie beschäftigt am Hauptsitz in Luzern, in den schweizweit 18 Agenturen und in den zwei Rehakliniken Bellikon und Sion rund 4200 Mitarbeitende.

Quelle: suva.ch, 21.02.2022

2.1.3 Rahmenbedingungen

Die Arbeit wird im Rahmen von 80 Stunden umgesetzt, aufgeteilt auf 10 Arbeitstage à 8 Stunden. Die Präsentation ist in den 80 Stunden nicht inbegriffen. Ein Unterbruch ist lediglich durch Unterricht an der Berufsfachschule, Feiertag oder Krankheit/Unfall gestattet. Sie wird als Einzelarbeit durchgeführt, Hilfestellungen sind erlaubt und massvoll erwünscht, sind jedoch zu dokumentieren. Gleiches gilt für Vorbereitungen.

Da die Corona-Massnahmen gelockert wurden, ist es für mich Möglich, die Umsetzung der Arbeit vor Ort in der Rösslimatt durchzuführen. Dafür entschied ich mich auch, da ich wahrscheinlich so eine höhere Produktivität erreichen kann.

2.1.4 Versionierung

Die Dokumentation sowie der Sourcecode wird zur Datensicherung täglich in einem Bitbucket-Repository eingecheckt. Die Daten werden im VCS von IntelliJ mit Git committed und gepusht. Somit ist der Arbeitsfortschritt jederzeit online in dem Versionsverwaltungssystem sowie lokal auf meinem Computer gesichert.

2.1.5 Projektaufbauorganisation

Kandidat

Lukas Bucher (BUL)
Beckenriederstrasse 62
6374 Buochs
079 951 63 28
041 419 63 98 (G)
lukas.bucherr@gmail.com (P)
Lukas.bucher@suva.ch (G)

Betrieb (Durchföhrungsort)

Suva
Rösslimattstrasse 39
6005 Luzern
041 419 51 11
www.suva.ch

Berufsbildner

Urs Grubenmann (GUA)
077 475 67 67
041 419 64 36 (G)
urs.grubenmann@suva.ch (G)

Verantwortliche Fachkraft

Boris Basic
079 575 17 52
041 419 65 18 (G)
boris.basic@suva.ch (G)

Hauptexperte

Frezza Roberto
CSS AG
Tribschenstrasse 21
6002 Luzern
078 768 65 88
058 277 12 68 (G)
roberto.frezza@gmail.com
robertofrezza@bluewin.ch

Nebenexperte

Weike Dennis
Frey+Cie Telecom AG
Stationsstrasse 89
6023 Rothenburg
079 192 82 04
dennis.weike@freytelecom.ch

2.2 Aufgabenstellung gemäss PkOrg

Quelle: 2022.pkorg.ch, 21.02.2022

2.2.1 Ausgangslage

Erweiterungen (Extensions) auf der Plattform Google Chrome bieten eine interessante Möglichkeit, einfache Applikationen (wie z.B. das Anzeigen von Team-Strukturen oder andere Organisations-Informationen) direkt im Browser zur Verfügung zu stellen.

Es soll eine Google Chrome Extension erstellt werden, welche die SAFe Team-Strukturen der IT von der Suva abbildet. Die Hauptseite soll so gestaltet werden, dass eine Übersicht der implementierten Informations-Module (in dieser Arbeit vorerst das Modul "SAFe Teams") angezeigt wird.

2.2.2 Detaillierte Aufgabenstellung

2.2.2.1 Erwartete Resultate

- Entwicklung eines Applikations-Rahmens als Google Chrome Extension, welche organisatorische Informationen als thematische Module anzeigt.
- Einstiegs-/Hauptseite, welche die verfügbaren Module präsentiert.
- Implementieren des ersten Moduls, um SAFe Team-Strukturen anzuzeigen.
- Team-Strukturen als hierarchischen Baum.
- Team-Member als Detail-Info.
- Basis für persönliche Einstellungen (Optionen) legen, z.B. Theme Dark/Light.

2.2.2.2 Zu erreichende Ziele

- Die Extension steht als Ordner-Struktur für eine klassische Extension-Installation für Google Chrome zur Verfügung.
- Die Hauptseite steht zur Verfügung.
- Das Modul "SAFe Teams" ist auf der Hauptseite anwähl-/startbar.
- Das Modul "SAFe Teams" besteht aus einem hierarchischen Navigationsteil (links) und einem Detail/Content-Teil (rechts).
- Auf der Hauptseite wird der Zugang zu den Optionen zur Verfügung gestellt (Header-Bereich).
- Die präsentierten Einstellungsmöglichkeiten sind funktionsfähig.
- Vorgeschlagene User Stories werden mit dem Auftraggeber überprüft/besprochen und abgenommen.

2.2.2.3 Erwartete Tests

- Extension-Installations-Prozess.
- Alle Akzeptanz-Kriterien der definierten User Stories.

2.2.2.4 Erwartete Dokumentation

- Big-Picture der interagierenden Komponenten.
- User-Story nach Suva.
- Minimale Installations-Anleitung für die Extension-Installation.

2.2.3 Mittel und Methoden

- Hardware:
 - Suva-Standard-Laptop/pVDI
- Software:
 - Windows 10
 - IntelliJ
 - Google Chrome
 - Confluence
 - Bitbucket
- Programmiersprachen:
 - JavaScript
 - HTML
 - CSS
- Firmenvorgaben:
 - Source-Code-Ablage in Bitbucket
 - User-Story-Format

2.2.4 Vorkenntnisse

- Tätigkeiten:
 - Interne und externe Web-Inhalte zur Verfügung stellen mit CMS-Systemen wie Wordpress
 - Einfachere WebApp mit Node.js erstellt. Mitarbeit/Unterstützung einer WebApp-Eigenentwicklung mit dem GUI-Framework Webix
 - Gehversuche im Bereich Google Chrome Extension
- Produkte:
 - Node.js
 - Webix
- Techniken:
 - Web-Inhalte bereitstellen
 - Plugin-Entwicklung für Wordpress
 - Node.js

2.2.5 Vorarbeiten

- Erörterung der Möglichkeiten mit Extensions auf der Plattform Google Chrome etwas zu machen.

2.2.6 Neue Lerninhalte

- Vertiefter Umgang mit der Google Chrome Extension API.
- Breitere Nutzung einer REST-Schnittstelle welche Organisations-Daten liefert.
- Extrahieren und zusammensetzen einer hierarchischen Team-Strukturen im Kontext SAFe.

2.2.7 Arbeiten in den letzten 6 Monaten

- Entwicklung eines einfachen Elementor Widgets mit Datenbank-Zugriffen.
- Umsetzungen an der Web-App VMS (node.js und Webix).
- Kennenlernen Extensions auf der Plattform Google Chrome.

2.2.8 Individuelle Beurteilungskriterien

- I1 - Auftrag, Forderungen (Workflow)
- I2 - Bestehender Workflow, Einbettung und Schnittstellen
- I3 - Erfüllung der Funktionen (Applikation)
- I4 - Dokumentation (Workflow-Applikation)
- I5 - Benutzerfreundlichkeit: GUI, Bedienung
- I6 - Implementierung von Lösungen (Programmieren)
- I7 - Brauchbarkeit (Applikation)

2.3 Planung

2.3.1 Projektmanagement

Für diese Arbeit habe ich mich dafür entschieden, IPERKA als Projektmanagementmethode zu wählen, da für ein Projekt dieses Umfangs eine iterative Methode schwer umsetzbar wäre. Zudem besteht diese Arbeit aus mehreren Teilaufgaben, welche durch IPERKA eingrenzbarer und übersichtlicher gemacht werden können.

IPERKA besteht aus den Schritten Informieren, Planen, Entscheiden, Realisieren, Kontrollieren, Auswerten. Übertragen auf mein Projekt würden diese Punkte folgendermassen aussehen: «Informieren» stellt die Informationsbeschaffung zu Chrome-Extensions und deren Entwicklung dar, ein Grossteil davon wurde bereits vor Beginn des Projekts erledigt. In der Phase «Planung» wird erst eine Grobplanung und später eine Detailplanung zum Ablauf des Projekts erstellt. Anschliessend wird beim Punkt «Entscheiden» evaluiert, welche Anforderung die zu realisierende Applikation erreichen soll. Die Entwicklung der App wird schliesslich im folgenden Schritt «Realisieren» durchgeführt. «Kontrollieren» beinhaltet das Testing und «Auswertung» die Dokumentation des Endprodukts.

2.3.2 Meilensteine

Ich werde mich für die Planung auf diese vier primären Meilensteine beziehen und Arbeiten, welche zur Erreichung eines jeweiligen Meilensteins beitragen, mit der entsprechenden Farbe kennzeichnen.

Deadline	Abkürzung	Beschreibung	Soll [h]	Ist [h]
22.02.2022	M1	Vorbereitung & Planung abgeschlossen	12	10.5
02.03.2022	M2	Umsetzung Applikation abgeschlossen	34	33
02.03.2022	M3	Tests durchgeführt & dokumentiert	4.5	5
04.03.2022	M4	Dokumentation finalisiert & abgegeben	29.5	31
			Total	80
				79.5

Tabelle 1 | Meilensteine

2.3.3 Grobplanung

21.02.2022	22.02.2022	23.02.2022	24.02.2022
Arbeitsumgebung bereitstellen Allgemeine Angaben zur Arbeit beschreiben Grobplanung Dokumentation Einleitung, Aufgabenstellung, Planung Beginn Detailplanung	Besprechung & Definition Umfang Extension Erster Besuchstag Hauptexperte User Stories definieren User Stories schreiben	Beginn Umsetzung Extension	Umsetzung Extension
Überlegungen zur Extension Dokumentation	Besprechung User Stories Finalisierung Detailplanung Dokumentation	Umsetzung Extension Dokumentation	Umsetzung Extension Dokumentation

25.02.2022	28.02.2022	01.03.2022	02.03.2022
Umsetzung Extension	Umsetzung Extension	Umsetzung Extension	Umsetzung Extension abschliessen Testfälle schreiben Tests durchführen
Umsetzung Extension Dokumentation	Umsetzung Extension Dokumentation	Umsetzung Extension Dokumentation	Tests dokumentieren Dokumentation

03.03.2022	04.03.2022
Dokumentation Extension	Abschluss Dokumentation Überprüfung Dokumentation
Dokumentation Extension Dokumentation	Reserve Abgabe Arbeit

Tabelle 2 | Grobplanung

2.3.4 Detailplanung

Tag	MS	Tätigkeit	Soll [h]	Ist [h]
21.02.2022	M1	Gerüst Chrome Extension erstellen	1	1
	M4	Dokumentation Einleitung & Allgemeines	2	2
	M1	Meilensteine bestimmen	0.5	0.5
	M1	Grobplanung erstellen	1	1
	M1	Bitbucket einrichten	0.5	0.5
	M1	Detailplanung beginnen	1	1
	M1	Überlegungen zur Extension	1	0.5
	M4	Dokumentation ergänzen	1	1
			8	7.5
22.02.2022	M1	Besprechung API SAFe-Daten	1	0.5
	M1	Erster Besuchstag Hauptexperte	1	0.5
	M1	User Stories definieren	1	1
	M1	User Stories schreiben	2	2
	M1	User Stories besprechen	1	0
	M1	Detailplanung finalisieren	1	0
	M4	Dokumentation ergänzen	1	3
			8	7

	M1	User Stories besprechen	0	1
	M4	Detailplanung finalisieren	0	1
	M4	Bedienkonzept erstellen	0.5	1
23.02.2022	M2	Umsetzung User Story 2 Szenario 1	5	4
		Umsetzung User Story 3 Szenario 1		
	M4	Dokumentation ergänzen	2.5	2
			8	9
24.02.2022	M2	Umsetzung User Story 2 Szenario 1	4	3
		Umsetzung User Story 3 Szenario 1		
	M2	Umsetzung User Story 3 Szenarios 2-5 exkl. Detail-Infos	2	4
	M4	Dokumentation ergänzen	2	1
			8	8
25.02.2022	M2	Umsetzung User Story 3 Szenarios 2-5 exkl. Detail-Infos	6	6
	M4	Dokumentation ergänzen	2	2
			8	8
28.02.2022	M2	Umsetzung User Story 3 Szenarios 2-5 exkl. Detail-Infos	2	2
	M2	Umsetzung User Story 3 Szenarios 2-5 Detail-Infos	5	5
		Umsetzung User Story 3 Szenario 6		
	M4	Dokumentation ergänzen	1	1
			8	8
01.03.2022	M2	Umsetzung User Story 3 Szenarios 2-5 Detail-Infos	4	2
		Umsetzung User Story 3 Szenario 6		
	M2	Umsetzung User Story 1 Szenario 3	3	5
	M4	Dokumentation ergänzen	1	1
			8	8

	M2	Umsetzung abschliessen	3	2
	M3	Testkonzept erstellen	1	1
	M3	Testfälle schreiben	2	1.5
02.03.2022		Tests ausführen	0.5	0.5
	M3	Tests dokumentieren	1	2
	M4	Dokumentation ergänzen	0.5	1
			8	8
03.03.2022	M4	Dokumentation Endsystem	5	3
03.03.2022	M4	Dokumentation ergänzen	3	5
			8	8
04.03.2022	M4	Dokumentation finalisieren	2	4
		Dokumentation kontrollieren	3	2
		Reserve für letzte Anpassungen	2	1.5
		Arbeit abgeben	1	0.5
			8	8
Total			80	79.5

Tabelle 3 | Detailplanung

2.4 Arbeitsjournal

2.4.1 Tag 1 | 21.02.2022

M1	Gerüst Chrome Extension erstellen	1	1
M4	Dokumentation Einleitung & Allgemeines	2	2
	Meilensteine bestimmen	0.5	0.5
	Grobplanung erstellen	1	1
M1	Bitbucket einrichten	0.5	0.5
	Detailplanung beginnen	1	1
	Überlegungen zur Extension	1	0.5
M4	Dokumentation ergänzen	1	1
		8	7.5

Tabelle 4 | Journal Tag 1

Dank eines Dokumentgerüsts, welches ich vor Beginn der Arbeit anhand der Rückmeldungen meiner Übungs-PAs erstellt hatte, verlief der Start ziemlich reibungslos. Ich konnte alle bereits bekannten grundlegenden Informationen Dokumentieren und die Grobplanung erstellen. Auch ein Gerüst für eine Extension zu erstellen war ohne Probleme möglich, da ich mich bereits zuvor mit der Entwicklung einer Extension befasst hatte und mir somit daran ein Beispiel nehmen konnte.

Morgen werde ich mit Boris die Möglichkeiten und Funktionalität der API besprechen, auf welche ich zugreifen werde. Zudem wird der erste Besuchstag des Hauptexperten durchgeführt.

2.4.2 Tag 2 | 22.02.2022

22.02.2022	M1	Besprechung API SAFe-Daten	1	0.5
		Erster Besuchstag Hauptexperte	1	0.5
	M1	User Stories definieren	1	1
		User Stories schreiben	2	2
		User Stories besprechen	1	0
		Detailplanung finalisieren	1	0
	M5	Dokumentation ergänzen	1	3
			8	7

Tabelle 5 | Journal Tag 2

Die heutige Besprechung der API wurde eher kurz abgehalten. Boris zeigte mir die URL, mit welcher der Zugriff auf die SAFe-Daten möglich ist. Auch der erste Besuchstag des Hauptexperten konnte erfolgreich abgehalten werden. Der Austausch fand über Microsoft Teams statt. Neben dem Hauptexperten Roberto Frezza nahmen Boris als verantwortliche Fachkraft, Urs als Berufsbildner und Dennis Weike als Nebenexperte am Meeting Teil. Da bei allen Parteien kaum Unklarheiten vorhanden waren, konnte es innerhalb einer halben Stunde abgehalten werden.

Ich konnte alle User Stories definieren und beschreiben. Dazu beachtete ich die hierarchische SAFe-Struktur, wie sie im SuvaTel anzutreffen ist, und informierte mich über die Rollen und Stufen einer SAFe-Struktur. Da Boris heute wenig Zeit hatte, um meine Fortschritte zu besprechen, werde ich die User Stories erst morgen überarbeiten, finalisieren und in der Detailplanung einplanen können. Die dadurch entstandene Zeit benutzte ich teilweise, um einige Anpassungen und Ergänzungen an der Dokumentation vorzunehmen. Ich werde ausserdem an einigen Tagen etwas länger arbeiten können, um das Zeitdefizit aufzuholen.

2.4.3 Tag 3 | 23.02.2022

23.02.2022	M1	User Stories besprechen	0	1
	M4	Detailplanung finalisieren	0	1
	M4	Bedienkonzept erstellen	0.5	1
	M2	Umsetzung User Story 2 Szenario 1	5	4
		Umsetzung User Story 3 Szenario 1		
	M4	Dokumentation ergänzen	2.5	2
			8	9

Tabelle 6 | Journal Tag 3

Die ausstehenden Arbeiten von gestern konnte ich heute erledigen. Ich habe die User Stories mit Boris besprochen und sie entsprechend seinem Feedback angepasst. So konnte ich schliesslich auch die Detailplanung anhand der Szenarios ergänzen.

Das Bedienkonzept kostete mich etwas mehr Zeit als geplant, da ich einige Punkte des GUIs mit Boris besprechen musste und das Erstellen eines Mockups etwas zeitaufwendig war.

Schliesslich konnte ich mit der Umsetzung beginnen. Da Szenario 1 von User Story 2 und Szenario 1 von User Story 3 grösstenteils voneinander abhängig sind, werden sie zusammen aufgeführt. Die Entwicklung der Hauptseite lief ohne weitere Probleme ab. Somit erhielt ich als Ergebnis eine erste Variante der Hauptseite mit der Info-App «Suva SAFe», die mit einem Klick aufgerufen werden kann. Es wurden dabei auch einige Styling-Grundlagen geschaffen, welche sich durch die Extension ziehen werden. Die Icons und monochromatische Farbpalette richtet sich nach der von mir als Übungs-PA erstellten «Suva-Mitarbeiter»-Extension.

2.4.4 Tag 4 | 24.02.2022

24.02.2022	M2	Umsetzung User Story 2 Szenario 1	4	3
		Umsetzung User Story 3 Szenario 1		
		Umsetzung User Story 3 Szenarios 2-5 exkl. Detail-Infos	2	4
	M4	Dokumentation ergänzen	2	1
			8	8

Tabelle 7 | Journal Tag 4

Heute ging es vorwiegend darum, das Layout der Hauptseite fertigzustellen und mit dem Gerüst der SAFe-Seite zu beginnen, sodass ich anschliessend daran arbeiten kann, die Daten der API einzufügen. Den Plan konnte ich einhalten, die Finalisierung der Hauptseite konnte ich sogar in weniger als der geplanten Zeit abschliessen, da ich gestern bereits weit gekommen war. So nahm ich mir mehr Zeit dafür, am Layout der SAFe-Seite zu arbeiten. Mit dem höheren Zeitaufwand war es mir möglich, das Layout fast komplett zu erstellen. Ich achtete bereits auf das Design der Seite, jedoch nur im Ausmass, mit welchem ich trotzdem ein konstantes Arbeitstempo aufrechterhalten konnte. Dabei habe ich Platzhalter für Navigationspunkte und Detail-Infos eingefügt, welche es später gilt, durch die von der API erhaltenen Daten zu ersetzen. Morgen werde ich mit Boris die Umsetzung diesbezüglich besprechen, da ich mir noch etwas unsicher bin, wie der Prozess der Erstellung der Navigationspunkte ablaufen soll, sodass beim Klick die korrekten Informationen im Content-Bereich angezeigt werden.

2.4.5 Tag 5 | 25.02.2022

25.02.2022	M2	Umsetzung User Story 3 Szenarios 2-5 exkl. Detail-Infos	6	6
	M4	Dokumentation ergänzen		
			8	8

Tabelle 8 | Journal Tag 5

Ich arbeitete heute an der dynamischen Anzeige der Daten weiter. Ich merkte, dass für dies nicht nur die API mit den Suva-SAFe-Daten benötigt wird, denn dort sind einige Mitarbeiter-Infos nicht auffindbar. Deshalb müssen diese beiden Datenquellen vereint verwendet werden. Ich erstellte eine Funktion, welche alle Daten in Form einer hierarchischen Struktur gemäss meinem Navigationsbaum sortiert. Mit dieser Datenstruktur konnte ich schliesslich eine Variante zum Aufbauen eines Navigationsbaumes entwickeln, welche gut funktionierte. Deshalb war die Besprechung mit Boris auch eher kurz, denn viele meiner Unklarheiten erschlossen sich mir während des Arbeitens oder ich konnte sie durch Recherche beseitigen. Er gab mir jedoch einige Tipps zu Design und Code, welche ich morgen umsetzen werde.

Ich denke, dass ich momentan noch gut im Zeitplan liege, ich mache stetige spürbare Fortschritte, was für mich motivierend ist. Einzig die Dokumentation ist noch etwas im Verzug, was aber vor Allem daran liegt, dass ich den Grossteil davon für das Ende der PA eingeplant habe. Dies auch deshalb, weil es bei vielen Punkten am meisten Sinn macht, diese nach Abschluss der Umsetzung zu beschreiben.

2.4.6 Tag 6 | 28.02.2022

28.02.2022	M2	Umsetzung User Story 3 Szenarios 2-5 exkl. Detail-Infos	2	2
	M2	Umsetzung User Story 3 Szenarios 2-5 Detail-Infos	5	5
	M4	Umsetzung User Story 3 Szenario 6		
	M4	Dokumentation ergänzen	1	1
			8	8

Tabelle 9 | Journal Tag 6

Heute verlief erneut alles nach Plan. Die Anzeige der Daten ist fast finalisiert, so bin ich der Planung sogar etwas voraus. Dies gibt mir etwas mehr Spielraum, um Features wie beispielsweise einen voll funktionsfähigen Dark Mode noch einzubauen.

Ich traf bei der heutigen Umsetzung nur auf wenige kleinere Probleme, welche ziemlich schnell wieder gelöst werden konnten. Beispielsweise hatten einige PMs nicht «PM» als Rolle hinterlegt und waren noch an einem anderen Ort in der Struktur vorhanden. Deshalb werde ich bei diesen Fällen die Daten noch einzeln nachreichen müssen.

Zudem werde ich mir morgen noch Daten beschaffen, welche bei den ARTs, Products und Teams angezeigt werden könnten. Dann wären alle Akzeptanzkriterien erfüllt und ich könnte mich etwas früher als geplant an den Feinschliff und vielleicht den Einbau weiterer Funktionen machen.

2.4.7 Tag 7 | 01.03.2022

01.03.2022	M2	Umsetzung User Story 3 Szenarios 2-5 Detail-Infos	4	2
	M2	Umsetzung User Story 3 Szenario 6		
	M4	Umsetzung User Story 1 Szenario 3	3	5
	M4	Dokumentation ergänzen	1	1
			8	8

Tabelle 10 | Journal Tag 7

Auch heute kam ich schnell und ohne grosse Probleme voran. Die letzten Unvollständigkeiten konnten behoben werden, sodass die grundlegende Funktionalität der Extension gewährleistet ist. Anschliessend nahm ich mir etwas länger dafür Zeit, die Option für ein dunkles Erscheinungsbild zu implementieren. Ich konnte dies trotzdem heute noch beenden, sodass die Umsetzung nun grösstenteils abgeschlossen ist. Kleine Details werde ich vielleicht noch überarbeiten, wenn noch Zeit bleibt. Daten für ARTs, Products und Teams werde ich beispielsweise mit Link oder Beschreibung ergänzen, falls vorhanden. Ich habe vorläufig die jeweiligen E-Mail-Adressen von ARTs und Teams und die ausgeschriebene Abkürzung der ARTs als Beschreibung verwendet.

Morgen werde ich mich grösstenteils mit dem Testing und Teilen der Dokumentation beschäftigen. Nach meiner Erfahrung ist das Testing oft nicht so zeitaufwendig wie geplant. Deshalb erhoffe ich mir etwas mehr Zeit für die Dokumentation aufwenden zu können.

2.4.8 Tag 8 | 02.03.2022

02.03.2022	M2	Umsetzung abschliessen	3	1.5
	M3	Testkonzept erstellen	1	1
	M3	Testfälle schreiben	2	1.5
	M3	Tests ausführen	0.5	0.5
	M3	Tests dokumentieren	1	2
	M4	Dokumentation ergänzen	0.5	1.5
			8	8

Tabelle 11 | Journal Tag 8

Das heute eingeplante Testing lief reibungslos ab. Alle Testfälle konnten erfolgreich geschrieben, ausgeführt und dokumentiert werden, was heisst, dass alle Akzeptanzkriterien erfüllt sind. Da ich nur noch wenig an der Vollendung der Extension arbeiten musste, konnte ich bereits mit Teilen der Dokumentation des Endsystems beginnen. Doch es bleibt immer noch viel zu dokumentieren, was ich jedoch vor Allem morgen eingeplant habe. Ich erhoffe mir daher morgen den Grossteil davon abschliessen zu können, sodass ich am letzten Tag genügend Zeit für die Schlusskontrolle inklusive Reserve übrig habe. Nach meinem Gefühl sollte dies auch möglich sein, wenn ich mein bisheriges konstantes Arbeitstempo aufrechterhalten kann.

2.4.9 Tag 9 | 03.03.2022

03.03.2022	M4	Dokumentation Endsystem	5	3
		Dokumentation ergänzen	3	5
			8	8

Tabelle 12 | Journal Tag 9

Ich arbeitete heute fast ausschliesslich an der Dokumentation. Dabei wendete ich weniger Zeit als geplant für das Endsystem auf, konnte mich deshalb aber einigen anderen Teilen stärker widmen. Ich konnte beispielsweise Kurzfassung und Einleitung ergänzen. Bezuglich Endsystem dokumentierte ich bereits die Installationsanleitung, das GUI und das Ändern des Erscheinungsbildes.

Ich werde morgen die Dokumentation des Endsystems also noch finalisieren müssen. Ich denke trotzdem, dass ich genügend Zeit dafür haben werde, da ich eine gewisse Reserve eingeplant habe.

2.4.10 Tag 10 | 04.03.2022

04.03.2022	M4	Dokumentation finalisieren	2	4
		Dokumentation kontrollieren	3	2
		Reserve für letzte Anpassungen	2	1.5
		Arbeit abgeben	1	0.5
			8	8

Tabelle 13 | Journal Tag 10

Boris gab mir heute noch letzte Hinweise und Verbesserungsvorschläge für Extension-Design und Dokumentation. Ansonsten wendete ich die gesamte Zeit für den Abschluss des schriftlichen Teils der Arbeit auf.

Die Finalisierung der Dokumentation verlief gut. Ich musste zwar noch etwas mehr Zeit als geplant aufwenden, doch mit der Reserve, welche eingeplant worden war, war dies kein grosses Problem. Ich konnte die Zeit dann für den Feinschliff des Dokuments verwenden.

3 PA-Bericht Teil 2

3.1 Anforderungen

Die Anforderungen werden mittels User Stories nach Suva-Standard definiert und beschrieben.

Nach Absprache mit Boris wird bei den in der Aufgabenstellung als «Module» bezeichneten Teilen der Extension in den Anforderungen von «Info-Apps» gesprochen, da diese Bezeichnung weniger allgemein ist.

3.1.1 User Stories

US1	Verwaltung der Extension	S1	Extension hinzufügen
		S2	Extension entfernen
		S3	Extension-Optionen verwalten
US2	Anzeige der Hauptseite	S1	Hauptseite aufrufen
US3	Anzeige der Suva-SAFe-Daten	S1	SAFe-Info-App anzeigen
		S2	ARTs anzeigen
		S3	ART-Inhalt anzeigen
		S4	Product-Inhalt anzeigen
		S5	Team-Inhalt anzeigen
		S6	Mitarbeiter-Infos anzeigen

Tabelle 14 | User Stories

3.1.2 User Story 1 | Verwaltung der Extension

Als	Browser-Benutzer mit Admin-Rechten
möchte ich	die Extension im Chrome Extension Manager verwalten können
damit	die Extension im Browser zur Verfügung gestellt werden kann

Tabelle 15 | User Story 1

Szenario	S1: Extension hinzufügen
Vorausgesetzt	<ul style="list-style-type: none"> - Der Chrome Browser wird nicht von meiner Organisation verwaltet (Chromium, Testumgebung) - Die Extension-Files sind entpackt auf meinem Computer gespeichert
Wenn	<ul style="list-style-type: none"> - Ich den Chrome Browser öffne - Ich auf die Seite «chrome://extensions/» navigiere - Ich den Schalter «Entwicklermodus» aktiviere - Ich auf den Button «Entpackte Erweiterung laden» klicke - Ich den Extension-Ordner auswähle
Dann	<ul style="list-style-type: none"> - Wird die Extension im Extension-Dashboard angezeigt - Wird die Extension dem Chrome Browser hinzugefügt und sie erscheint oben rechts in der Toolbar

Tabelle 16 / US1 S1

Szenario	S2: Extension entfernen
Vorausgesetzt	<ul style="list-style-type: none"> - Der Chrome Browser wird nicht von meiner Organisation verwaltet (Chromium) - Die Extension ist dem Chrome Browser hinzugefügt - Der Entwicklermodus ist im Extension-Dashboard aktiviert
Wenn	<ul style="list-style-type: none"> - Ich den Chrome Browser öffne - Ich auf die Seite «chrome://extensions/» navigiere - Ich bei der Extension auf den Button «Entfernen» klicke
Dann	<ul style="list-style-type: none"> - Wird die Extension aus dem Chrome Browser entfernt

Tabelle 17 / US1 S2

Szenario	S3: Extension-Optionen verwalten
Vorausgesetzt	<ul style="list-style-type: none"> - Der Chrome Browser wird nicht von meiner Organisation verwaltet (Chromium) - Die Extension ist dem Chrome Browser hinzugefügt - Der Entwicklermodus ist im Extension-Dashboard aktiviert
Wenn	<ul style="list-style-type: none"> - Ich den Chrome Browser öffne - Ich auf die Seite «chrome://extensions/» navigiere - Ich bei der Extension auf den Button «Details» klicke - Ich auf den Link «Optionen» klicke
Dann	<ul style="list-style-type: none"> - Werden die Optionen der Extension angezeigt und sie können geändert werden

Tabelle 18 / US1 S3

3.1.3 User Story 2 | Anzeige der Hauptseite

Als	Browser-Benutzer
möchte ich	bei Aufrufen der Extension in einem neu eröffneten Tab eine Hauptseite sehen, welche verfügbare Info-Apps anzeigt
damit	ich einen Überblick über die mir zur Verfügung stehenden Info-Apps erhalte und diese verwenden kann

Tabelle 19 | User Story 2

Szenario	S1: Hauptseite aufrufen
Vorausgesetzt	<ul style="list-style-type: none">- Die Extension ist im dem Chrome Browser hinzugefügt
Wenn	<ul style="list-style-type: none">- Ich den Chrome Browser öffne- Ich oben rechts in der Toolbar auf das Extension-Icon klicke
Dann	<ul style="list-style-type: none">- Wird mir eine Hauptseite mit allen verfügbaren Info-Apps angezeigt

Tabelle 20 | US2 S1

3.1.4 User Story 3 | Anzeige der Suva-SAFE-Daten

Als	Browser-Benutzer
möchte ich	die SAFe Team-Strukturen der Suva in der Extension in Form eines hierarchischen Baums anzeigen lassen können
damit	ich die SAFe-Strukturen nachvollziehen kann ich gezielt nach Informationen bezüglich deren suchen kann

Tabelle 21 | User Story 2

Szenario	S1: SAFe-Info-App anzeigen
Vorausgesetzt	<ul style="list-style-type: none">- Die Extension ist im dem Chrome Browser hinzugefügt
Wenn	<ul style="list-style-type: none">- Ich den Chrome Browser öffne- Ich oben rechts in der Toolbar auf das Extension-Icon klicke
Dann	<ul style="list-style-type: none">- Wird mir auf der Hauptseite die Info-App «Suva SAFe» angezeigt

Tabelle 22 | US3 S1

Szenario	S2: ARTs anzeigen
Vorausgesetzt	<ul style="list-style-type: none">- Die Extension ist im dem Chrome Browser hinzugefügt
Wenn	<ul style="list-style-type: none">- Ich den Chrome Browser öffne- Ich oben rechts in der Toolbar auf das Extension-Icon klicke- Ich auf das SAFe-Info-App klicke
Dann	<ul style="list-style-type: none">- Werden mir im Navigations-Bereich alle ARTs als aufklappbares Element eines hierarchischen Baums angezeigt

Tabelle 23 | US3 S2

Szenario	S3: ART-Inhalt anzeigen
Vorausgesetzt	<ul style="list-style-type: none">- Die Extension ist im dem Chrome Browser hinzugefügt
Wenn	<ul style="list-style-type: none">- Ich den Chrome Browser öffne- Ich oben rechts in der Toolbar auf das Extension-Icon klicke- Ich auf das SAFe-Info-App klicke- Ich auf ein ART klicke und es somit aufklappe
Dann	<ul style="list-style-type: none">- Werden mir alle BOs dieses ARTs als Unterelement angezeigt- Wird mir der RTE dieses ARTs als Unterelement angezeigt- Werden mir alle SAs dieses ARTs als Unterelement angezeigt- Werden mir alle Products dieses ARTs als aufklappbares Element angezeigt- Werden mir die Detail-Infos dieses ARTs im Content-Bereich angezeigt

Tabelle 24 | US3 S3

Szenario	S4: Product-Inhalt anzeigen
Vorausgesetzt	<ul style="list-style-type: none"> - Die Extension ist im dem Chrome Browser hinzugefügt
Wenn	<ul style="list-style-type: none"> - Ich den Chrome Browser öffne - Ich oben rechts in der Toolbar auf das Extension-Icon klicke - Ich auf das SAFe-Info-App klicke - Ich auf ein ART klicke und es somit aufklappe - Ich auf ein Product klicke und es somit aufklappe
Dann	<ul style="list-style-type: none"> - Wird mir der PM dieses Products als Unterelement angezeigt - Werden mir alle Teams dieses Products als aufklappbares Element angezeigt - Werden mir die Detail-Infos dieses Products im Content-Bereich angezeigt

Tabelle 25 / US3 S4

Szenario	S5: Team-Inhalt anzeigen
Vorausgesetzt	<ul style="list-style-type: none"> - Die Extension ist im dem Chrome Browser hinzugefügt
Wenn	<ul style="list-style-type: none"> - Ich den Chrome Browser öffne - Ich oben rechts in der Toolbar auf das Extension-Icon klicke - Ich auf das SAFe-Info-App klicke - Ich auf ein ART klicke und es somit aufklappe - Ich auf ein Product klicke und es somit aufklappe - Ich auf ein Team klicke und es somit aufklappe
Dann	<ul style="list-style-type: none"> - Werden mir alle Mitglieder dieses Teams als Unterelement angezeigt, POs und SM sind dabei als solche markiert - Werden mir die Detail-Infos dieses Teams im Content-Bereich angezeigt

Tabelle 26 / US3 S5

Szenario	S6: Mitarbeiter-Infos anzeigen
Vorausgesetzt	<ul style="list-style-type: none"> - Die Extension ist im dem Chrome Browser hinzugefügt
Wenn	<ul style="list-style-type: none"> - Ich den Chrome Browser öffne - Ich oben rechts in der Toolbar auf das Extension-Icon klicke - Ich auf das SAFe-Info-App klicke - Ich im hierarchischen Baum zu einem Mitarbeiter navigiere - Ich auf diesen Mitarbeiter klicke
Dann	<ul style="list-style-type: none"> - Werden mir die Detail-Infos dieses Mitarbeiters im Content-Bereich angezeigt

Tabelle 27 / US3 S6

3.2 Testing

3.2.1 Testkonzept

Das Testing soll die Qualität und Funktionalität der Applikation sicherstellen. Ich werde lediglich manuelle Tests durchführen, da bei einem Projekt dieses Umfangs eine Automatisierung oder Anwendung weiterer Testing-Tools nicht sinnvoll wäre.

Es werden Testfälle anhand der Definierten Anforderungen erstellt. Dabei sollen alle Grundlegenden Funktionen abgedeckt werden. Die durchgeföhrten Tests werden sinngemäss protokolliert und ausgewertet, schliesslich werden allfällige Fehler behoben und der Test wird falls nötig wiederholt.

Die Testumgebung besteht aus dem Chromium Browser, mit welchem ich die Berechtigung zur Extension-Verwaltung besitze. Der Extension-Ordner kann dann unverändert im Extension-Dashboard hinzugefügt werden. Als Schnittstellen gelten die zwei REST APIs, welche ich für den Datenbezug benötige. Da sie nur lesende sind werden die produktiven Schnittstellen verwendet.

3.2.2 Testfälle

ID	1
Bezeichnung	Extension hinzufügen
Beschreibung	Es wird getestet, ob die Extension reibungslos im Extension-Dashboard hinzugefügt werden kann.
Getestete Anforderung	User Story 1 Akzeptanzkriterium S1
Testvoraussetzung	<ul style="list-style-type: none">- Mein Chrome Browser wird nicht von meiner Organisation verwaltet (Chromium)- Die Extension-Files sind entpackt auf meinem Computer gespeichert
Testschritte	<ol style="list-style-type: none">1. Chrome Browser öffnen2. Auf die Seite «chrome://extensions/» navigieren3. Schalter «Entwicklermodus» aktivieren4. Button «Entpackte Erweiterung laden» klicken5. Extension-Ordner auswählen
Erwartetes Ergebnis	Die Extension wird fehlerfrei zum Browser hinzugefügt und erscheint oben rechts in der Toolbar. Die Extension wird im Dashboard als Kachel dargestellt. In der Toolbar und dem Dashboard wird das korrekte Extension-Icon angezeigt.
Resultat	Test erfolgreich durchgeführt 02.03.2022 10:05 BUL

Tabelle 28 | Testfall 1

ID	2
Bezeichnung	Extension entfernen
Beschreibung	Es wird getestet, ob die Extension reibungslos im Extension-Dashboard entfernt werden kann.
Getestete Anforderung	User Story 1 Akzeptanzkriterium S2
Testvoraussetzung	<ul style="list-style-type: none"> - Der Chrome Browser wird nicht von meiner Organisation verwaltet (Chromium) - Die Extension ist dem Chrome Browser hinzugefügt - Der Entwicklermodus ist im Extension-Dashboard aktiviert
Testschritte	<ol style="list-style-type: none"> 1. Chrome Browser öffnen 2. Auf die Seite «chrome://extensions/» navigieren 3. Button «Entfernen» der Extension-Kachel anklicken
Erwartetes Ergebnis	Die Extension wird fehlerfrei vom Browser entfernt und erscheint nicht mehr oben rechts in der Toolbar oder im Dashboard.
Resultat	Test erfolgreich durchgeführt 02.03.2022 10:15 BUL

Tabelle 29 | Testfall 2

ID	3
Bezeichnung	Extension-Option «Erscheinungsbild» auf «dunkel» setzen
Beschreibung	Es wird getestet, ob die Extension-Optionen korrekt angezeigt werden und erwartungsgemäss funktionieren. Sie sind dabei auf der Haupt-Seite als Info-App auffindbar. Beim Aktivieren des dunklen Erscheinungsbildes soll das Extension-GUI in dunklen Farben angezeigt werden.
Getestete Anforderung	User Story 1 Akzeptanzkriterium S3
Testvoraussetzung	<ul style="list-style-type: none"> - Die Extension ist im dem Chrome Browser hinzugefügt - Das Erscheinungsbild ist auf «hell» oder «Systemstandard» gesetzt, im letzteren Fall ist im System das helle Farbschema eingestellt (Windows: Einstellungen – Farben – Farbe auswählen – Hell/Benutzerdefiniert – Standard-App-Modus auswählen – Hell)

Testschritte	<ol style="list-style-type: none">1. Chrome Browser öffnen2. Oben rechts in der Toolbar das Extension-Icon anklicken3. Info-App «Optionen» anklicken4. Button «dunkel» in Abschnitt «Erscheinungsbild» anklicken
Erwartetes Ergebnis	Die Optionen werden sofort im dunklen Erscheinungsbild angezeigt. Das Icon des Abschnitts «Erscheinungsbild» ändert sich dabei. Beim nächsten neuen Laden bzw. Aufrufen der Seite ein.
Resultat	Test erfolgreich durchgeführt 02.03.2022 10:30 BUL

Tabelle 30 / Testfall 3

ID	4
Bezeichnung	Extension-Option «Erscheinungsbild» auf «Systemstandard» setzen
Beschreibung	Es wird getestet, ob beim Aktivieren des dunklen Erscheinungsbildes das Extension-GUI gemäss den Systemeinstellungen in dunklen oder hellen Farben angezeigt wird.
Getestete Anforderung	User Story 1 Akzeptanzkriterium S3
Testvoraussetzung	<ul style="list-style-type: none">- Die Extension ist im dem Chrome Browser hinzugefügt- Das Erscheinungsbild ist auf «hell» oder «dunkel» gesetzt
Testschritte	<ol style="list-style-type: none">1. Chrome Browser öffnen2. Oben rechts in der Toolbar das Extension-Icon anklicken3. Info-App «Optionen» anklicken4. Button «Systemstandard» in Abschnitt «Erscheinungsbild» anklicken

Erwartetes Ergebnis	Die Optionen werden sofort im in den Systemeinstellungen (Windows: Einstellungen – Farben – Farbe auswählen – Hell/Dunkel/Benutzerdefiniert – Standard-App-Modus auswählen – Hell/Dunkel) ausgewählten Erscheinungsbild angezeigt. Das Icon des Abschnitts «Erscheinungsbild» ändert sich dabei gegebenenfalls. Beim nächsten neuen Laden bzw. Aufrufen der Seite ein.
Resultat	Test erfolgreich durchgeführt 02.03.2022 10:45 BUL

Tabelle 31 | Testfall 4

ID	5
Bezeichnung	Hauptseite aufrufen
Beschreibung	Es wird getestet, ob beim Öffnen der Extension eine Hauptseite mit allen verfügbaren Info-Apps angezeigt wird.
Getestete Anforderungen	User Story 2 Akzeptanzkriterium S1 User Story 3 Akzeptanzkriterium S1
Testvoraussetzung	- Die Extension ist im dem Chrome Browser hinzugefügt
Testschritte	1. Chrome Browser öffnen 2. Oben rechts in der Toolbar das Extension-Icon anklicken
Erwartetes Ergebnis	Die Hauptseite mit den zurzeit verfügbaren Info-Apps wird angezeigt. Mit einem Klick auf eine Info-App gelangt man auf die entsprechende Seite deren.
Resultat	Test erfolgreich durchgeführt 02.03.2022 11:00 BUL

Tabelle 32 | Testfall 5

ID	6
Bezeichnung	ARTs anzeigen
Beschreibung	Es wird getestet, ob beim Öffnen der Info-App «Suva SAFe» alle ARTs angezeigt werden.
Getestete Anforderung	User Story 3 Akzeptanzkriterium S2
Testvoraussetzung	- Die Extension ist im dem Chrome Browser hinzugefügt

Testschritte	<ol style="list-style-type: none">1. Chrome Browser öffnen2. Oben rechts in der Toolbar das Extension-Icon anklicken3. Info-App «Suva SAFe» anklicken
Erwartetes Ergebnis	<p>Die Seite «Suva SAFe» wird mit einem Navigationsbereich angezeigt, in welchem alle vorhandenen ARTs als aufklappbare Elemente eines hierarchischen Baums aufgelistet sind.</p> <p>Der Content-Bereich ist leer.</p>
Resultat	Test erfolgreich durchgeführt 02.03.2022 11:10 BUL

Tabelle 33 | Testfall 6

ID	7
Bezeichnung	ART-Inhalt anzeigen
Beschreibung	Es wird getestet, ob beim Anklicken eines ARTs die dazugehörigen Unterelemente und Informationen angezeigt werden.
Getestete Anforderung	User Story 3 Akzeptanzkriterium S3
Testvoraussetzung	<ul style="list-style-type: none">- Die Extension ist im dem Chrome Browser hinzugefügt
Testschritte	<ol style="list-style-type: none">1. Chrome Browser öffnen2. Oben rechts in der Toolbar das Extension-Icon anklicken3. Info-App «Suva SAFe» anklicken4. Einen ART anklicken
Erwartetes Ergebnis	<p>Der ART wird markiert. Alle vorhandenen BOs und SAs des ARTs werden als Unterelemente dessen aufgelistet. Der RTE des ARTs wird als Unterelement dessen aufgelistet. Alle vorhandenen Products des ARTs werden als aufklappbare Unterelemente dessen aufgelistet.</p> <p>Im Content-Bereich werden das ART-Icon und die korrekten Informationen (E-Mail, Beschreibung) des ARTs angezeigt.</p>
Resultat	Test erfolgreich durchgeführt 02.03.2022 11:25 BUL

Tabelle 34 | Testfall 7

ID	8
Bezeichnung	Product-Inhalt anzeigen
Beschreibung	Es wird getestet, ob beim Anklicken eines Products die dazugehörigen Unterelemente und Informationen angezeigt werden.
Getestete Anforderung	User Story 3 Akzeptanzkriterium S4
Testvoraussetzung	<ul style="list-style-type: none"> - Die Extension ist im dem Chrome Browser hinzugefügt
Testschritte	<ol style="list-style-type: none"> 1. Chrome Browser öffnen 2. Oben rechts in der Toolbar das Extension-Icon anklicken 3. Info-App «Suva SAFe» anklicken 4. Einen ART anklicken 5. Ein Product anklicken
Erwartetes Ergebnis	<p>Das Product wird markiert. Der PM des Products wird als Unterelement dessen aufgelistet. Alle vorhandenen Teams des Products werden als aufklappbare Unterelemente dessen aufgelistet.</p> <p>Im Content-Bereich werden das Product-Icon und die korrekten Informationen (Beschreibung) des Products angezeigt.</p>
Resultat	Test erfolgreich durchgeführt 02.03.2022 11:35 BUL

Tabelle 35 | Testfall 8

ID	9
Bezeichnung	Team-Inhalt anzeigen
Beschreibung	Es wird getestet, ob beim Anklicken eines Teams die dazugehörigen Unterelemente und Informationen angezeigt werden.
Getestete Anforderung	User Story 3 Akzeptanzkriterium S5
Testvoraussetzung	<ul style="list-style-type: none"> - Die Extension ist im dem Chrome Browser hinzugefügt

Testschritte	<ol style="list-style-type: none">1. Chrome Browser öffnen2. Oben rechts in der Toolbar das Extension-Icon anklicken3. Info-App «Suva SAFe» anklicken4. Einen ART anklicken5. Ein Product anklicken6. Ein Team anklicken
Erwartetes Ergebnis	<p>Das Team wird markiert. Alle vorhandenen Mitglieder des Teams werden als Unterelemente dessen aufgelistet. POs und SM werden dabei mithilfe des Icons als solche markiert.</p> <p>Im Content-Bereich werden das Team-Icon und die korrekten Informationen (E-Mail, Beschreibung) des Teams angezeigt.</p>
Resultat	Test erfolgreich durchgeführt 02.03.2022 11:45 BUL

Tabelle 36 | Testfall 9

ID	10
Bezeichnung	Mitarbeiter-Inhalt anzeigen
Beschreibung	Es wird getestet, ob beim Anklicken eines Mitarbeiters die dazugehörigen Unterelemente und Informationen angezeigt werden.
Getestete Anforderung	User Story 3 Akzeptanzkriterium S6
Testvoraussetzung	<ul style="list-style-type: none">- Die Extension ist im dem Chrome Browser hinzugefügt
Testschritte	<ol style="list-style-type: none">1. Chrome Browser öffnen2. Oben rechts in der Toolbar das Extension-Icon anklicken3. Info-App «Suva SAFe» anklicken4. Im Navigationsbaum zu einem Mitarbeiter navigieren und ihn anklicken
Erwartetes Ergebnis	<p>Der Mitarbeiter wird markiert.</p> <p>Im Content-Bereich werden entsprechendes Icon gemäss SAFe-Rolle und korrekte Informationen (Funktion, SAFe-Infos, E-Mail, Telefon, Mobil, Abteilung, Gebäude & Büro, Sprache) des Mitarbeiters angezeigt.</p>
Resultat	Test erfolgreich durchgeführt 02.03.2022 11:55 BUL

Tabelle 37 | Testfall 10

3.2.3 Testprotokoll

Testfall	Durchführung	Werte	Resultat
1	02.03.2022 10:05 BUL		Die Extension wurde fehlerfrei zum Browser hinzugefügt und erscheint oben rechts in der Toolbar. Die Extension wird im Dashboard als Kachel dargestellt. In der Toolbar und dem Dashboard wird das korrekte Extension-Icon angezeigt.
2	02.03.2022 10:15 BUL		Die Extension wurde fehlerfrei vom Browser entfernt und erscheint nicht mehr oben rechts in der Toolbar oder im Dashboard.
3	02.03.2022 10:30 BUL		Die Optionen wurden sofort im dunklen Erscheinungsbild angezeigt. Das Icon des Abschnitts «Erscheinungsbild» änderte sich dabei. Beim restlichen Extension-GUI trat der Effekt beim nächsten Aufrufen der Seite ein.
4	02.03.2022 10:45 BUL	Windows: Einstellungen – Farben – Farbe auswählen – Dunkel	Die Optionen wurden sofort im in den Systemeinstellungen ausgewählten Erscheinungsbild (<i>Dunkel</i>) angezeigt. Das Icon des Abschnitts «Erscheinungsbild» änderte sich dabei. Beim restlichen Extension-GUI tritt der Effekt beim nächsten neuen Laden bzw. Aufrufen der Seite ein.

5	02.03.2022 11:00 BUL		Die Hauptseite mit den zurzeit verfügbaren Info-Apps (<i>Suva SAFe, Optionen</i>) wurde angezeigt. Mit einem Klick auf eine Info-App gelangte man auf die entsprechende Seite deren.
6	02.03.2022 11:10 BUL		Die Seite « <i>Suva SAFe</i> » wurde mit einem Navigationsbereich angezeigt, in welchem alle vorhandenen ARTs (<i>ARTMP, ARTVS, ARTWI</i>) als aufklappbare Elemente eines hierarchischen Baums aufgelistet sind. Der Content-Bereich war leer.
7	02.03.2022 11:25 BUL	ART: <i>ARTWI</i>	Der ART (<i>ARTWI</i>) wurde markiert. Alle vorhandenen BOs (<i>Stefan Scherrer, Oliver Eugster</i>) und SAs (<i>Brian Mathis</i>) des ARTs wurden als Unterelemente dessen aufgelistet. Der RTE (<i>Daniel Portmann</i>) des ARTs wurde als Unterelement dessen aufgelistet. Alle vorhandenen Products (<i>Workplace, Security Continuity und ITSM, Infrastruktur, IT Management Services</i>) des ARTs wurden als aufklappbare Unterelemente dessen aufgelistet. Im Content-Bereich wurden das ART-Icon und die korrekten Informationen (<i>\$GOM-SAFe-ARTWI@suvanet.ch, Workplace & Infrastrukturen</i>) des ARTs angezeigt.

8	02.03.2022 11:35 BUL	ART: ARTMP Product : <i>Prävention</i>	<p>Das Product (<i>Prävention</i>) wurde markiert. Der PM (<i>Jason Schenk</i>) des Products wurde als Unterelement dessen aufgelistet. Alle vorhandenen Teams (<i>Pilatus, Kraftwerk</i>) des Products wurden als aufklappbare Unterelemente dessen aufgelistet.</p> <p>Im Content-Bereich wurden das Product-Icon und die korrekten Informationen (<i>Beschreibung</i>) des Products angezeigt.</p>
9	02.03.2022 11:45 BUL	ART: ARTVS Product: <i>Backend-Solutions & Data-Science</i> <i>Schadenmanagement</i> Team: <i>Monty</i>	<p>Das Team (<i>Monty</i>) wurde markiert. Alle vorhandenen Mitglieder (<i>Mauro Blaser, Thomas Bauer, Dieter Spinnler, Simon Figura</i>) des Teams wurden als Unterelemente dessen aufgelistet. POs (<i>Thomas Bauer</i>) und SM (<i>Mauro Blaser</i>) wurden dabei mithilfe des Icons als solche markiert.</p> <p>Im Content-Bereich wurden das Team-Icon und die korrekten Informationen (<i>\$GOM-SAFe-ARTVS-Monty@suvanet.ch, Beschreibung</i>) des Teams angezeigt.</p>
10	02.03.2022 11:55 BUL	ART: ARTWI Product: <i>Security Continuity und ITSM</i> Team: <i>Joker</i> Mitarbeiter: <i>Lukas Bucher</i>	<p>Der Mitarbeiter (<i>Lukas Bucher</i>) wurde markiert.</p> <p>Im Content-Bereich wurden das Personen-Icon und die korrekten Informationen (<i>Lernender, ARTWI - Security Continuity und ITSM – Joker, lukas.bucher@suvanet.ch, +41414196398, +41799516328, IF -IFO - IFOTA, RM Z – OG05.5124, DE</i>) des Mitarbeiters angezeigt.</p>

Tabelle 38 | Testprotokoll

3.3 Lösungsvariante

Die Lösungsvariante meiner Arbeit steht bereits fest, denn durch meine formulierten User Stories wurde die Formulierung der Aufgabenstellung genügend konkretisiert.

3.3.1 Big Picture

In dieser Grafik sind die Schnittstellen zwischen den einzelnen Komponenten und Akteuren sowie deren Zusammenhänge ersichtlich. Links sind Benutzer, rechts die Datenquellen. In der Mitte findet man die Komponenten, welche im Google Chrome Browser für die Funktionsfähigkeit der Extension grundlegend sind.

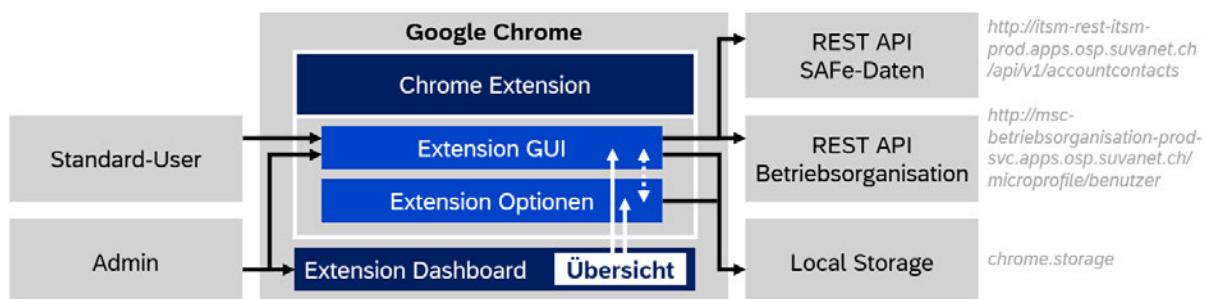


Abbildung 1 | Big Picture

Ein Standard-User stellt einen normalen Benutzer der Extension dar. In diesem Fall wäre dies der Suva-Mitarbeiter, welcher damit Informationen finden möchte, beispielsweise zur Suva-SAFe-Organisation. Er hat, wenn die Extension dem Browser hinzugefügt worden ist, mittels des Extension-Icon in der Toolbar Zugriff auf das Extension-GUI. In diesem besteht die Verbindung zu den Optionen in Form einer Info-App. So kann der Standard-User diese ebenfalls ändern. In den Optionen steht der Home-Button als Verknüpfung zurück zum GUI zur Verfügung.

Der Administrator kann die Extension im Extension Dashboard verwalten. Er kann sie also hinzufügen, entfernen oder aktualisieren. In der Extension-Übersicht finden sich Links zum GUI sowie den Optionen. Der Admin kann wie der Standard-User ebenfalls über das Extension-Icon in der Toolbar zum GUI gelangen.

Für die Erzeugung der SAFe-Struktur im GUI werden auf zwei REST APIs zugegriffen, eine API für SAFe-Daten und die Betriebsorganisation-API. Beide liefern Suva-Mitarbeiter-Daten in Form von JSON als Response. Das Extension GUI und die Optionen machen von dem Chrome Local Storage Gebrauch. Dieser ist nötig, um die Einstellungen des Benutzers zu speichern und sie bei erneutem Laden der Seite wieder aufrufen zu können. Das in den Optionen gewählte Erscheinungsbild bleibt so zum Beispiel bestehen.

3.3.2 Bedienkonzept der Info-App «Suva SAFe»

Um die SAFe-Organisation der Suva nachvollziehen zu können orientierte ich mich an der Struktur, wie sie im SuvaTel zu finden ist. Als Modell für die Umsetzung dieser Hierarchie erstellte ich eine Grafik. Das folgende Mockup zeigt meine Vorstellungen der SAFe-Seite. Mit dieser Vorlage wird mir der Aufbau der Seiten-Struktur und die Visualisierung des Layouts erleichtert. Die Extension werde ich auf Grundlage dieser Grafik entwickeln.

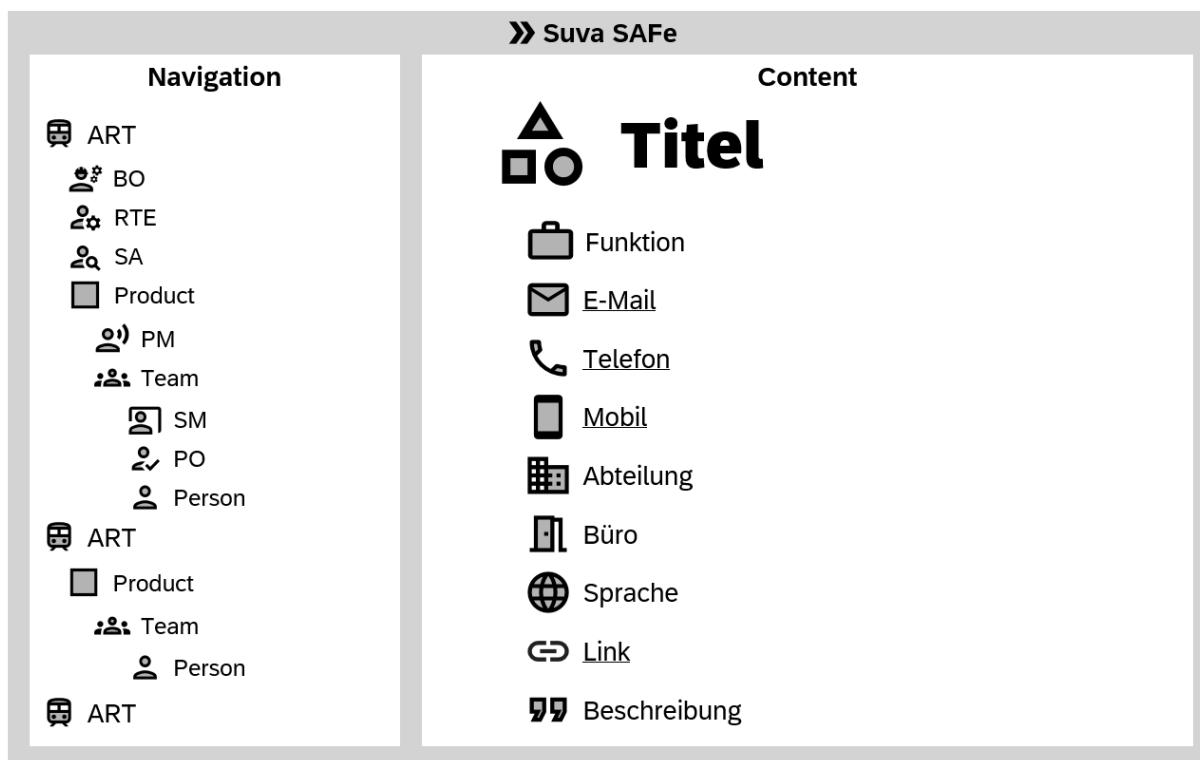


Abbildung 2 | Mockup

Die Seite wird in zwei Bereiche unterteilt. Links findet sich die Navigation, wo die ARTs, Products, Teams und Mitarbeiter anhand der SAFe-Organisation als Elemente eines hierarchischen Baums aufgelistet werden. Beim Klicken auf ein aufklappbares Element sollen dabei alle Unterelemente dessen erscheinen. Im rechten Teil, dem Content-Bereich, werden die Detail-Informationen des eines Elements der Navigation ersichtlich, sobald dies durch Mausklick angewählt wird. Bei Mitarbeitern bestehen die Infos aus Funktion, E-Mail, Telefon, Mobil, Abteilung, Büro und Sprache. Zu ARTs, Products und Teams werden Beschreibung und Link gezeigt, falls diese vorhanden sind, zu ARTs und Teams zudem die E-Mail-Adresse. E-Mail-Adressen, Telefon- und Mobilnummern sowie Links sind anwählbar und leiten den Benutzer entsprechend weiter.

3.3.3 Datenbezug

Die Quelle der Daten besteht aus zwei internen REST-Schnittstellen. Diese müssen beide verwendet werden, da beide Unterschiedliche Mitarbeiter-Daten liefern, welche ich für die Umsetzung benötige. Die API mit der URL «<http://itsm-rest-itself-prod.apps.osp.suvanet.ch/api/v1/accountcontacts>» benutze ich vorwiegend für den Bezug der SAFe-Daten. Persönliche Informationen, welche dann noch fehlen, beispielsweise Gebäude und Büro, werden von der REST API Betriebsorganisation mit der URL «<http://msc-betriebsorganisation-prod-svc.apps.osp.suvanet.ch/microprofile/benutzer>» entnommen. Beide Schnittstellen liefern eine JSON-Struktur. Diese beinhaltet ein Array aller Mitarbeiter als Objekte mit Attributen, welche die Mitarbeiter-Informationen darstellen.

3.4 Endsystem

In diesem Kapitel werden die implementierten Funktionen der Google Chrome Extension aufgezeigt. Dabei wird die Art der Umsetzung deren erläutert. Es werden vereinzelt Code-Fragmente gezeigt und erklärt. Der komplette Source Code ist im Anhang zu finden.

3.4.1 Installationsanleitung

Voraussetzung für die Installation der Google Chrome Extension ist, dass der Browser nicht von der Organisation verwaltet ist. Deshalb verwende ich dazu Chromium.

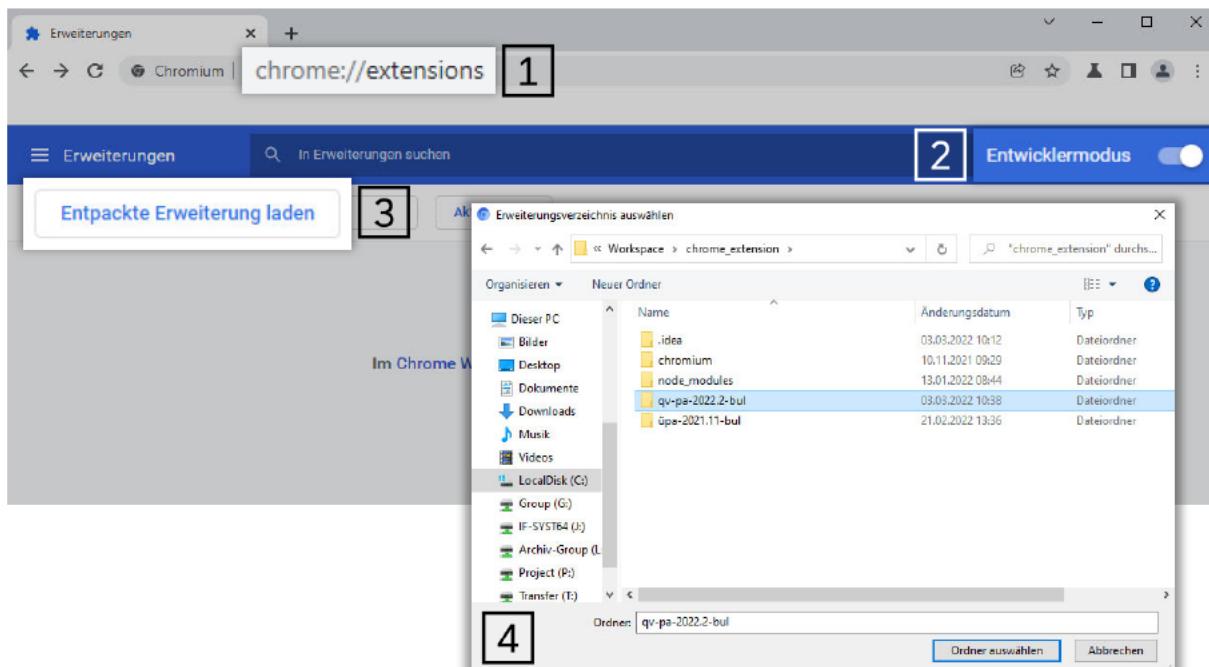


Abbildung 3 | Installationsanleitung

1. URL «chrome://extensions/» (Dashboard Erweiterungen) aufrufen
2. Schalter «Entwicklermodus» aktivieren
3. Button «Entpackte Erweiterung laden» klicken
4. Im erscheinenden Explorer zu Erweiterungspfad Navigieren und Erweiterungs-Ordner auswählen

Nach Hinzufügen der Erweiterung sieht das Dashboard dann folgendermassen aus:

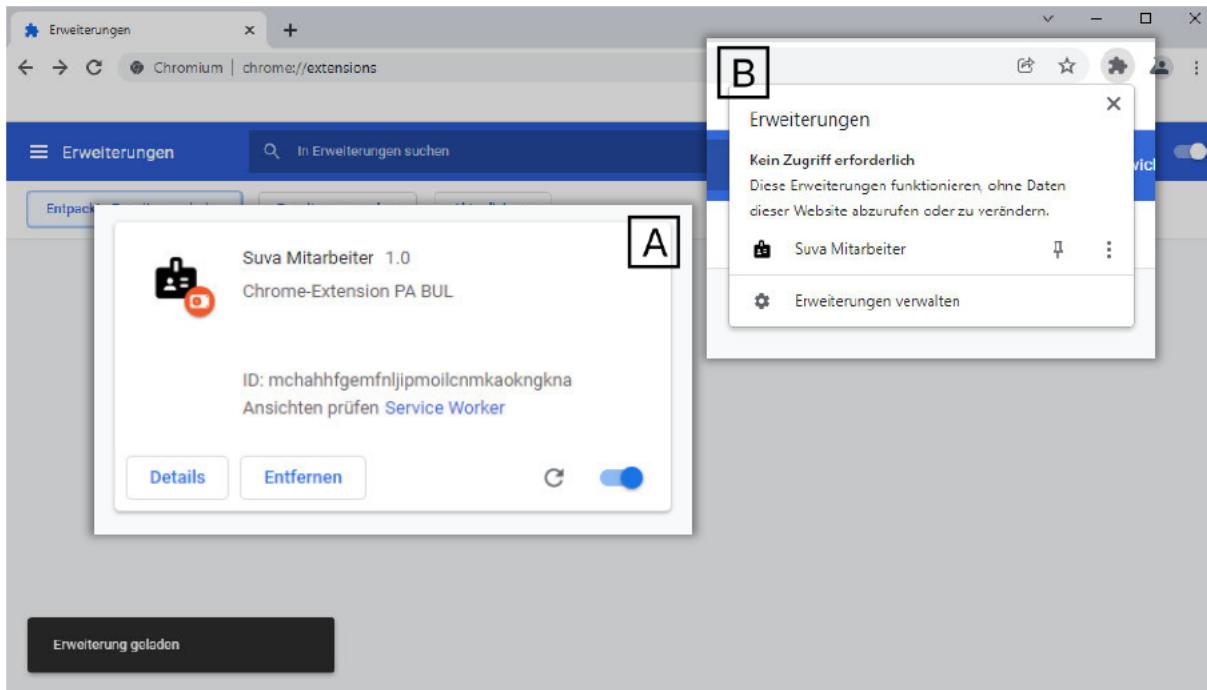


Abbildung 4 | Erklärung Extension Dashboard

- A. Übersichts-Kachel, zeigt Informationen der Erweiterung an, Erweiterung kann hier entfernt, neu geladen oder deaktiviert werden
- B. Erweiterungen in der Toolbar, zeigt alle aktivierte Erweiterungen an, Erweiterung kann hier ausgeführt, an die Toolbar gepinnt oder verwaltet werden

3.4.2 GUI

Das GUI-Design der Extension baut auf das Design meiner in der letzten Übungs-PA erstellten Extension auf. Das Farbschema besteht aus kühlen Grautönen und beinhaltet dieses Mal vereinzelt blaue Akzente. Alle verwendeten Icons stammen aus der kostenlosen Google Fonts Library.

Auf der Hauptseite findet man den Titel «Suva Mitarbeiter» mit dem Extension-Icon, einem Badge. Darunter finden sich die verfügbaren Info-Apps in Kachelform mit den jeweiligen Icons. Momentan sind die Info-Apps «Suva SAFe» und «Optionen» verfügbar.

Im hellen Erscheinungsbild sieht die Hauptseite folgendermassen aus:

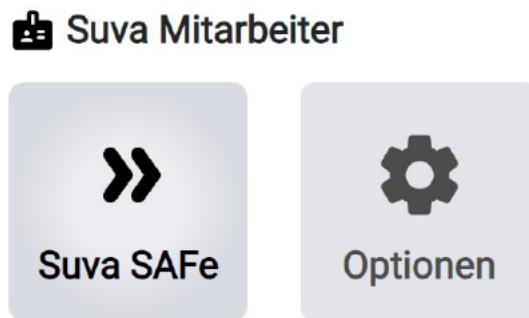


Abbildung 5 | GUI Hauptseite

Auf der Seite «Suva SAFe» findet man den Titel «Suva SAFe» mit dem SAFe-Icon. Darunter sieht man das Interface mit den zwei Bereichen, links Navigation, rechts Content. In der Navigation ist der hierarchische Baum zu sehen, welcher die SAFe-Struktur der Suva darstellt. Mit einem Klick auf ein ART, Product oder Team öffnet sich dieses und dessen Unterelemente erscheinen. Klickt man auf ein Element der Navigation, so wird dieses markiert und dessen Detail-Informationen erscheinen rechts im Content-Bereich mit dem entsprechenden Icon. Mit einem Klick auf eine E-Mail-Adresse öffnet sich ein neues Outlook-Mail mit der Adresse als Empfänger, bei einer Telefon-Nummer öffnet sich nach Bestätigen des Benutzers der Skype-Chat des Mitarbeiters und er kann angerufen werden.

Mit dem Home-Button oben links gelangt man zurück zur Hauptseite, der Info-Button oben rechts führt einen auf die Webseite von SAFe, mit der Erklärung des Scaled Agile Framework.

Im hellen Erscheinungsbild sieht die Seite «Suva SAFe» folgendermassen aus:

The screenshot shows the main interface of the Suva SAFe extension. On the left, there is a navigation tree with the following structure:

- ARTMP
 - BO - Stefan Lüthy
 - BO - Armin Zimmermann
 - BO - Birgit Lang
 - RTE - Adrian Baumann
 - SA - Leif Jakob
 - SA - Andreas Imlig
- Online Services
 - Hyperlink
 - SM - Celine Hunkeler (highlighted)
 - PO - Sandro Pigoni
 - Peter Vögeli
 - Patrick Oswald

On the right, there is a detailed view for "Celine Hunkeler (C6H)". It includes her profile picture, name, and a sidebar with various icons for communication and management.

Abbildung 6 | GUI SAFe

Auf der Optionen-Seite findet man den Titel «Optionen» mit dem Optionen-Icon. Darunter sieht man alle Optionen aufgelistet. Aktive Einstellungen sind markiert. Mit dem Home-Button oben links gelangt man zurück zur Hauptseite.

Im hellen Erscheinungsbild sieht die Seite «Optionen» folgendermassen aus:

⌂ ⚙ Optionen



Abbildung 7 | GUI Optionen

3.4.3 Erscheinungsbild ändern

```
//app.js
darkMode();
[...]
function darkMode() {
    chrome.storage.sync.get("mode", (data) => {
        let currentMode = data.mode;
        if(currentMode == "dark" || (currentMode == "default" &&
            window.matchMedia('prefers-color-scheme: dark').matches)){
            for(icon of document.getElementsByClassName("icon")){
                let source = icon.src;
                if(!source.includes("-dark") &&
                    source.includes(".svg")){
                    icon.src = source.slice(0,source.length-4) +
                    "-dark.svg";
                }
            }
        } else {
            document.getElementById("css-dark").disabled = true;
        }
        if(window.matchMedia(
            '(prefers-color-scheme: dark)').matches){
            document.getElementById("icon").href =
            "images/icon-dark.svg";
        }
    });
}
[...]
```

Für das Speichern der Optionen wird der Chrome Storage `chrome.storage.sync` verwendet. Um das Erscheinungsbild auf hell zu ändern wird das CSS-File im HTML-Head auf `disabled` gesetzt. Ansonsten wird es so gelassen und die Seite wird dunkel erscheinen. Durch alle Icons werden mithilfe einer `for of` Schleife iteriert und sie werden durch das weisse Icon ersetzt.

Beim Klick auf einen Button in den Optionen wird die Seite direkt neu geladen und sofort im entsprechenden Erscheinungsbild angezeigt. Beim restlichen Extension-GUI tritt der Effekt beim nächsten Aktualisieren oder Aufrufen der Seite ein.

Bei der Einstellung «Systemstandard» wird jeweils mithilfe von `window.matchMedia('prefers-color-scheme: dark')`.`matches` überprüft, ob der Farbmodus des Betriebssystems (Windows: Einstellungen – Farben – Farbe auswählen – Hell/Dunkel/Benutzerdefiniert – Standard-App-Modus auswählen – Hell/Dunkel) auf dunkel gesetzt ist und ändert anhand dessen das Erscheinungsbild der Extension.

Die Extension-Seiten sehen im dunklen Erscheinungsbild folgendermassen aus:

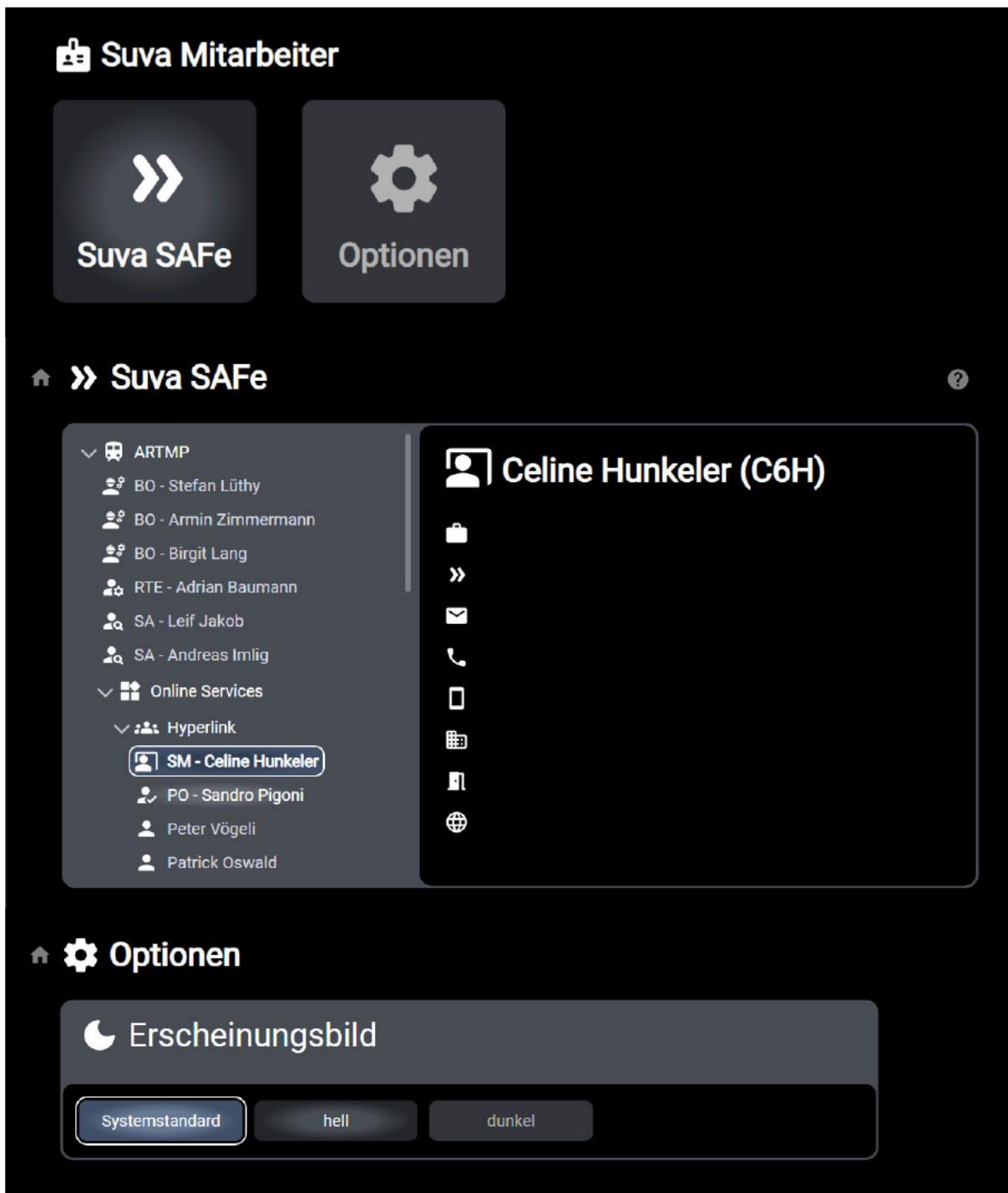


Abbildung 8 | GUI dunkles Erscheinungsbild

3.4.4 Mitarbeiter-Listen abrufen

```
//safe.js
[...]
window.addEventListener('load', async () => {
    document.getElementById("home").addEventListener("click",
        ()=> {
            chrome.tabs.update({url: chrome.runtime.getURL("index.html")});
    });
    document.getElementById("info").addEventListener("click", ()=> {
        chrome.tabs.create(
            {url:"https://www.scaledagileframework.com/"}
        );
    });
    let safeUsers = await fetchSafeUsers();
    let suvaUsers = await fetchSuvaUsers();
    let safeStructure =
        createSafeStructure(safeUsers.data.rows, suvaUsers);
    buildNavigation(safeStructure);
});
[...]
async function fetchSafeUsers(){
    const response = await fetch("http://itsm-rest-itsm-
prod.apps.osp.suvanet.ch/api/v1/accountcontacts",{
        method: "get",
        headers: {
            'Accept': 'application/json',
            'Content-Type': 'application/json'
        }
    })
    return await response.json();
}
async function fetchSuvaUsers(){
    const response = await fetch("http://msc-betriebsorganisation-
prod-svc.apps.osp.suvanet.ch/microprofile/benutzer", {
        method: "get",
        headers: {
            'Accept': 'application/json',
            'Content-Type': 'application/json'
        }
    })
    return await response.json();
}
[...]
```

Die Mitarbeiterdaten der Suva werden mithilfe der Funktionen `fetchSuvaUsers()` und `fetchSafeUsers()` abgerufen. In diesen wird jeweils ein Fetch auf die entsprechende API mit der Methode «Get» ausgeführt. Die Resultate werden zurückgegeben und schliesslich der Funktion `createSafeStructure()` mitgegeben, um die Safe-Struktur mit den Daten zu erstellen. Diese wird schliesslich für den Aufbau der Navigation verwendet, da sie dieselbe Struktur besitzt, anstelle von der simplen Auflistung von Personen, wie sie in den Rohdaten der APIs anzutreffen ist.

3.4.5 Safe-Struktur erstellen

```
//safe.js
[...]
function createSafeStructure(safeUsers, suvaUsers){
    let arts = [];
    for(let safeu of safeUsers){
        if(safeu.safe_art !== null){
            let suvau = suvaUsers.find(x => x.visum == safeu.visum);
            if(arts.find(x => x.name == safeu.safe_art) ==
            undefined){
                let art = {
                    name: safeu.safe_art,
                    bos: [],
                    rte: {},
                    sas: [],
                    products: []
                };
                arts.push(art);
            }
            if(arts.find(x => x.name == safeu.safe_art)
            .products.find(x => x.name == safeu.safe_product) ==
            undefined){
                let product = {
                    name: safeu.safe_product,
                    pm: {},
                    teams: []
                };
                arts.find(x => x.name ==
                    safeu.safe_art).products.push(product);
            }
            if(arts.find(x => x.name ==
                safeu.safe_art).products.find(x => x.name ==
                safeu.safe_product).teams.find(x => x.name ==
                safeu.safe_scrum_team) == undefined){
                let team = {
                    name: safeu.safe_scrum_team,
                    sm: {},
                    pos: [],
                    members: [],
                    art: safeu.safe_art
                };
            }
        }
    }
}
```

```
arts.find(x => x.name ==
safeu.safe_art).products.find(x => x.name ==
safeu.safe_product).teams.push(team);
}
[...]
let person = {
  name: suvau.vorname + " " + suvau.nachname,
  visum: suvau.visum,
  function: suvau.funktion.de,
  email: suvau.email,
  phone: suvau.telefon,
  mobile: suvau.mobile,
  department: department,
  office: office,
  language: suvau.sprache,
  safe_role: safeu.safe_rolle,
  safe_team: safeu.safe_scrum_team,
  safe_product: safeu.safe_product,
  safe_art: safeu.safe_art
}
switch (person.safe_role){
  case "BO":
    arts.find(x => x.name ==
safeu.safe_art).bos.push(person);
    break;
  [...]
  case "":
  default:
    arts.find(x => x.name ==
safeu.safe_art).products.find(x => x.name ==
safeu.safe_product).teams.find(x => x.name ==
safeu.safe_scrum_team).members.push(person);
    break;
}
}
}
return arts;
}
[...]
```

In der Funktion `createSafeStructure(safeUsers, suvaUsers)` wird die SAFe-Organisation anhand der Mitarbeiter-Daten als JSON-Struktur dargestellt, sodass sie im Aufbau gleich ist, wie die Navigation aussehen soll. Dafür wird durch alle Benutzer iteriert und geprüft, ob dessen Team, Product und ART bereits in der Struktur vorhanden ist. Ist dies nicht der Fall wird das entsprechende Element hinzugefügt. Die Person wird schliesslich je nach SAFe-Rolle am richtigen Ort in die Organisation eingefügt und mit ihren Mitarbeiterdaten befüllt.

3.4.6 Navigation bauen

```
//safe.js
[...]
function buildNavigation(safeStructure) {
    document.getElementById("nav").innerHTML = "";
    for(let artNr in safeStructure){
        let art = safeStructure[artNr];
        buildART(art, artNr)
        for(let bo of art.bos){
            buildBO(bo, artNr);
        }
        buildRTE(art.rte, artNr);
        for(let sa of art.sas){
            buildSA(sa, artNr);
        }
        for(let productNr in art.products){
            let product = art.products[productNr];
            buildProduct(product, productNr, artNr);
            if(product.pm.name){
                buildPM(product.pm, productNr, artNr);
            }else{
                manualBuildPM(product.name, productNr, artNr);
            }
            for(let teamNr in product.teams){
                let team = product.teams[teamNr]
                buildTeam(team, teamNr, productNr, artNr);
                buildSM(team.sm, teamNr, productNr, artNr);
                for(let poNr in team.pos){
                    let po = team.pos[poNr];
                    buildPO(po, teamNr, productNr, artNr);
                }
                for(let member of team.members){
                    buildMember(member, teamNr, productNr, artNr);
                }
            }
        }
    }
    darkMode();
}

function buildART(art, artNr) {
    let container = document.getElementById("nav");
    let art_li = document.createElement('li');
    art_li.innerHTML = `<button class="btn btn-toggle"
        data-bs-toggle="collapse"
        data-bs-target="#art-${artNr}"
        aria-expanded="false">
            
            ${art.name}
    `;
    container.appendChild(art_li);
}
```

```
</button>
    <div class="collapse"
        id="art-${artNr}">
        <ul class="list-unstyled"
            id="art-${artNr}-ul">
        </ul>
    </div>;
art_li.childNodes[1].addEventListener("focus",
()=>{showGroup(art, "ART")});
container.appendChild(art_li);
}
function buildBO(bo, artNr) {
    let bo_li = document.createElement('li');
    bo_li.innerHTML = `<a href="#" class="link-dark">
        
        BO - ${bo.name}
    </a>`;
    bo_li.firstChild.addEventListener("focus",
()=>{showPerson(bo)});
    document.getElementById("art-"+artNr+"-ul").appendChild(bo_li);
}
[...]
```

Mit der erstellten Struktur lässt sich nun die Navigation im GUI aufbauen. Die Funktion `buildNavigation(safeStructure)` ist dafür zuständig. Darin befindet sich ein Konstrukt aus `for of` Schleifen, welches Stufenweise durch alle ARTs, Products, Teams und Mitarbeiter iteriert. Jedes Element wird dann in seiner spezifischen `build`-Funktion erstellt. Damit beim anklicken des Elements später die Detail-Informationen im Detail-Bereich angezeigt werden können wird ihm mittels `.addEventListener("focus", [...])`; ein Event Listener hinzugefügt, welcher bei Fokus auf dem Element die Funktion `showPerson([...])`, beziehungsweise `showGroup([...])` auruft.

3.4.7 Detail-Informationen anzeigen

```
//safe.js
[...]
function showPerson(person) {
    let image;
    switch (person.safe_role) {
        case "BO":
            image = "engineer";
            break;
        [...]
        default:
            image = "person";
            break;
    }
}
```

```
document.getElementById("icon-content").src =
`images/${image}.svg`;
document.getElementById("h1-content").innerText = person.name +
" (" + person.visum + ")";
let container = document.getElementById("info-content");
container.innerHTML = "";
let info_div = document.createElement('div');
info_div.innerHTML = getPersonInfo(person);
container.appendChild(info_div);
darkMode();
}
[...]
function getPersonInfo(person) {
let funktion = "";
[...]
if(person.function != ""){
    funktion = `
        <p class="txt-info">${person.function}</p>`
}
[...]
let info = funktion + safe + email + telefon + mobile +
department + office + language;
return info;
}
[...]
```

Für die Anzeige der Detail-Informationen werden aus dem SAFe-Struktur-Objekt der Person oder der Gruppe die Daten entnommen und schliesslich im Content-Bereich der Seite angehängt. Dabei wird auf Rolle oder Art ermittelt, sodass das richtige Icon erscheint und es wird für jedes Attribut geprüft, ob dieses leer ist. Ist dies der Fall, so wird dieser Abschnitt nicht hinzugefügt.

4 Schlusswort

Der Abschluss meiner praktischen Arbeit des Qualifikationsverfahrens sehe ich alles in allem als Erfolg an. Rückblickend denke ich, dass meine Leistung während den zehn Tagen ziemlich konstant auf einem hohen Niveau lag.

Zu Beginn der Arbeit hatte ich Gegenüber den früheren Übungs-PAs grössere Kontrolle und konnte so sicherer in die Arbeit starten. Die Erfahrungen, welche ich gesammelt hatte, halfen mir in der Einstiegsphase besonders, da ich dabei zuvor manchmal auf Schwierigkeiten getroffen war. Das Dokumentgerüst, welches vor Beginn der Arbeit angefertigt wurde, ersparte mir einige wertvolle Zeit. Denn ich musste mir nicht mehr viele Gedanken darüber machen, womit ich beginnen soll oder was alles Teil der Dokumentation sein muss. Die Planungsphase fiel mir nicht sehr schwer, da ich die Umsetzung mit ähnlichen vergangenen Arbeiten vergleichen und so den Zeitaufwand gut einschätzen konnte. Im Nachhinein erkenne ich, dass ich durchaus meist eine gute Prognose traf, wenn man von kleineren zu erwartenden Abweichungen absieht. Auch die Formulierung der Anforderungen verlief dank gewisser Routine in dem Bereich ohne weiter Probleme.

Durch den schnellen erfolgreichen Abschluss der Planung konnte ich bereits früh in die Umsetzungsphase starten. Ich begann das Programmieren mit gewisser Motivation, denn meine bisherigen Erfahrungen mit der Extension-Entwicklung waren grösstenteils positiv und es offenbarten sich anhand meiner Anforderungsbeschreibungen keine Unklarheiten bezüglich des Aufbaus der Extension. Während der Entwicklung fand ich, meiner Meinung nach, einen guten Ausgleich von Produktivität und Ausarbeitung von Details. Manchmal konzentrierte ich mich zwar etwas zu sehr auf bestimmte Aspekte des Designs, da dieser Teil der Umsetzung mir besonders gefiel. Dies geschah jedoch in einem vertretbaren Rahmen, sodass es keine grosse Auswirkung auf meinen sonst konstanten Fortschritt hatte. Ich bin stolz, dass ich bei der Entwicklung selbstständig vorankam, und nur selten Boris für Hilfe konsultieren musste. Viele Probleme konnte ich mit eigenen Methoden lösen, was mir weitere Zuversicht auf meinen Erfolg gab.

Auch Testing und Dokumentation konnte ich reibungslos abschliessen. Wenn ich jedoch etwas anders machen könnte, so wäre dies wohl die Einplanung der Dokumentation. Genügend Zeit war zwar da, doch das meiste war für die letzten Tage geplant. Das hatte zur Folge, dass ich nicht mehr in die Materie vertieft war, wie es beispielsweise während der Entwicklung der Fall war. So musste ich zur Erläuterung von Aufbau und Code-Fragmenten erneut die geleistete Arbeit analysieren, um volles Verständnis davon zu erlangen. Ausserdem nahm meine Motivation etwas ab, da das Schreiben ziemlich repetitiv werden konnte. Etwas mehr Abwechslung hätte da sicherlich nicht geschadet.

Als ganzes gesehen bin ich mit meiner Arbeit zufrieden. Die Extension ist voll funktionsfähig und ich sehe darin auch eine Möglichkeit, in der Zukunft eine erweiterte Version davon in der Suva einzusetzen. Ich könnte mir gut vorstellen, sie auszubauen, sodass sie die Funktionen des jetzigen SuvaTel beinhaltet und vielleicht sogar irgendwann ersetzen kann.

5 Glossar

(REST) API	(Representational State Transfer) Application Programming Interface, Schnittstelle, welche Datenaustausch ermöglicht
Array	Datentyp, Liste von Elementen
ART	Agile Release Train, SAFe-Arbeitsgruppe
Attribut	Eigenschaft eines (JSON-)Objektes
Bitbucket	Online-Versionsverwaltungstool, welches in der Suva verwendet wird
BO	Business Owner, SAFe-Rolle, trägt Verantwortung über ART
Chrome	Webbrowser von Google, welcher in der Suva verwendet wird
Chromium	Open-Source-Browser, welches einen Grossteil des Quelltextes von Chrome beinhaltet
(Extension-) Dashboard	Unter der URL chrome://extensions/ auffindbares Interface, in welchem Chrome Extensions verwaltet werden können
(Chrome) Extension / Erweiterung	Kleine Software-Erweiterung des Chrome-Browsers, welche Zusatz-Features enthält
Git	Versionsverwaltungssoftware für Dateien
GUI	Graphical User Interface, Grafische Benutzeroberfläche
IntelliJ	Entwicklungsumgebung von JetBrains, welche in der Suva verwendet wird
JSON	JavaScript Object Notation, Datenformat für Datenaustausch
Local Storage	Lokaler Speicher von Google Chrome
Objekt	Datentyp, enthält Attribut, Grundlage für das JSON-Datenformat
PM	Product Manager, SAFe-Rolle, trägt Verantwortung über Product
PO	Product Owner, SAFe-Rolle, arbeitet gemeinsam mit Team am Product und überprüft es

Product	SAFe-Begriff, Ergebnis der Arbeit
Response	Antwort der API (bei fetch())
RTE	Release Train Engineer, SAFe-Rolle, Manager eines ARTs
SA	Safe Agilist, SAFe-Rolle, Leiter eines ARTs
SAFe	Scaled Agile Framework, agiles Projektmanagement-Framework für Unternehmen
SM	Scrum Master, SAFe-Rolle, Coach eines Teams
SuvaTel	Desktop-Applikation zur Suche von Suva-Mitarbeiter-Daten, welche in der Suva verwendet wird
Toolbar	Balken oben im Chrome-Browser, wo sich installierte Extensions befinden
VCS	Version Control System, Versionsverwaltungssystem (Git)
VM	Virtual Machine, ein virtueller Rechnersystem innerhalb eines physischen Rechners

Tabelle 39 | Glossar

6 Quellenverzeichnis

1. Refsnes Data. W3Schools. 2022. Available at: <https://www.w3schools.com/>. Accessed March 04, 2022.
2. MDN Web Docs. 2022. Available at: <https://developer.mozilla.org/de/>. Accessed March 04, 2022.
3. Stack Exchange Inc. Stack Overflow. 2022. Available at: <https://stackoverflow.com/>. Accessed March 04, 2022.
4. Bootstrap. Bootstrap. Available at: <https://getbootstrap.com/docs/>. Accessed March 04, 2022.
5. Coyier C. CSS-Tricks. Available at: <https://css-tricks.com/>. Accessed March 04, 2022.
6. Suva. Suva. 2022. Available at: <https://www.suva.ch/>. Accessed March 04, 2022.
7. Ivaris AG. PkOrg. 2017. Available at: <https://2022.pkorg.ch/>. Accessed March 04, 2022.
8. Google Developers. Chrome Developers. Available at: <https://developer.chrome.com/docs/extensions/>. Accessed April 03, 2022.

7 Tabellenverzeichnis

Tabelle 1 Meilensteine	11
Tabelle 2 Grobplanung	13
Tabelle 3 Detailplanung	15
Tabelle 4 Journal Tag 1	16
Tabelle 5 Journal Tag 2	16
Tabelle 6 Journal Tag 3	17
Tabelle 7 Journal Tag 4	18
Tabelle 8 Journal Tag 5	18
Tabelle 9 Journal Tag 6	19
Tabelle 10 Journal Tag 7	19
Tabelle 11 Journal Tag 8	20
Tabelle 12 Journal Tag 9	21
Tabelle 13 Journal Tag 10	21
Tabelle 14 User Stories	22
Tabelle 15 User Story 1	22
Tabelle 16 US1 S1	23
Tabelle 17 US1 S2	23
Tabelle 18 US1 S3	23
Tabelle 19 User Story 2	24
Tabelle 20 US2 S1	24
Tabelle 21 User Story 2	24
Tabelle 22 US3 S1	24
Tabelle 23 US3 S2	25
Tabelle 24 US3 S3	25
Tabelle 25 US3 S4	26
Tabelle 26 US3 S5	26
Tabelle 27 US3 S6	26
Tabelle 28 Testfall 1	27
Tabelle 29 Testfall 2	28
Tabelle 30 Testfall 3	29
Tabelle 31 Testfall 4	30
Tabelle 32 Testfall 5	30
Tabelle 33 Testfall 6	31
Tabelle 34 Testfall 7	31
Tabelle 35 Testfall 8	32
Tabelle 36 Testfall 9	33
Tabelle 37 Testfall 10	33
Tabelle 38 Testprotokoll	36
Tabelle 39 Glossar	53

8 Abbildungsverzeichnis

Abbildung 1 Big Picture.....	37
Abbildung 2 Mockup.....	38
Abbildung 3 Installationsanleitung	39
Abbildung 4 Erklärung Extension Dashboard.....	40
Abbildung 5 GUI Hauptseite	41
Abbildung 6 GUI SAFe	41
Abbildung 7 GUI Optionen.....	42
Abbildung 8 GUI dunkles Erscheinungsbild.....	44

9 Anhang

9.1 Sourcecode

9.1.1 manifest.json

```
{  
  "name": "Suva Mitarbeiter",  
  "description": "Chrome-Extension PA BUL",  
  "version": "1.0",  
  "manifest_version": 3,  
  "background": {  
    "service_worker": "background.js"  
  },  
  "action": {  
    "default_title": "Suva Mitarbeiter",  
    "default_icon": "images/icon.png"  
  },  
  "icons": {  
    "16": "images/icon.png",  
    "32": "images/icon.png",  
    "48": "images/icon.png",  
    "128": "images/icon.png"  
  },  
  "options_page": "options.html",  
  "permissions": [  
    "storage",  
    "tabs"  
  ],  
  "host_permissions": [  
    "http://msc-betriebsorganisation-prod-svc.apps.osp.suvanet.ch/*"  
  ],  
  "homepage_url": "http://msc-betriebsorganisation-prod-  
svc.apps.osp.suvanet.ch"  
}
```

9.1.2 background.js

```
chrome.action.onClicked.addListener(tab => {  
  chrome.tabs.create({url:chrome.runtime.getURL("index.html")});  
});
```

9.1.3 app.js

```
darkMode();  
window.addEventListener('load', async () => {  
  document.getElementById("safe").addEventListener("click",  
  loadSafe);  
  document.getElementById("options").addEventListener("click",  
  loadOptions);  
  //document.getElementById("more").addEventListener("click",  
  //()=>{alert("Weitere Funktionen folgen...")});
```

```
});

function loadSafe(){
    chrome.tabs.update({url: chrome.runtime.getURL("safe.html")});
}
function loadOptions(){
    chrome.tabs.update({url:
chrome.runtime.getURL("options.html")});
}

function darkMode() {
    chrome.storage.sync.get("mode", (data) => {
        let currentMode = data.mode;
        if(currentMode == "dark" || (currentMode == "default" &&
window.matchMedia('prefers-color-scheme: dark').matches)){
            for(icon of document.getElementsByClassName("icon")){
                let source = icon.src;
                if(!source.includes("-dark") &&
source.includes(".svg")){
                    icon.src = source.slice(0,source.length-4) +
"-dark.svg";
                }
            }
        } else {
            document.getElementById("css-dark").disabled = true;
        }
        if(window.matchMedia(
            '(prefers-color-scheme: dark)').matches){
            document.getElementById("icon").href =
"images/icon-dark.svg";
        }
    });
}
```

9.1.4 safe.js

```
darkMode();
window.addEventListener('load', async () => {
    document.getElementById("home").addEventListener("click",
()=> {
    chrome.tabs.update({url: chrome.runtime.getURL("index.html")
});
}); 
document.getElementById("info").addEventListener("click", ()=> {
    chrome.tabs.create(
        {url:"https://www.scaledagileframework.com/"}
));
let safeUsers = await fetchSafeUsers();
let suvaUsers = await fetchSuvaUsers();
let safeStructure =
createSafeStructure(safeUsers.data.rows, suvaUsers);
```

```
        buildNavigation(safeStructure);
    });

function darkMode() {
    chrome.storage.sync.get("mode", (data) => {
        let currentMode = data.mode;
        if(currentMode == "dark" || (currentMode == "default" &&
            window.matchMedia('prefers-color-scheme: dark').matches)){
            for(icon of document.getElementsByClassName("icon")){
                let source = icon.src;
                if(!source.includes("-dark") &&
                    source.includes(".svg")){
                    icon.src = source.slice(0,source.length-4) +
                    "-dark.svg";
                }
            }
        } else {
            document.getElementById("css-dark").disabled = true;
            document.getElementById("css-safe-dark").disabled =
            true;
        }
        if(window.matchMedia(
            '(prefers-color-scheme: dark)').matches){
            document.getElementById("icon").href =
            "images/safe-dark.svg";
        }
    });
}

async function fetchSafeUsers(){
    const response = await fetch("http://itsm-rest-itsm-
prod.apps.osp.suvanet.ch/api/v1/accountcontacts",{
        method: "get",
        headers: {
            'Accept': 'application/json',
            'Content-Type': 'application/json'
        }
    })
    return await response.json();
}
async function fetchSuvaUsers(){
    const response = await fetch("http://msc-betriebsorganisation-
prod-svc.apps.osp.suvanet.ch/microprofile/benutzer", {
        method: "get",
        headers: {
            'Accept': 'application/json',
            'Content-Type': 'application/json'
        }
    })
    return await response.json();
}
```

}

```
function createSafeStructure(safeUsers, suvaUsers){
    let arts = [];
    for(let safeu of safeUsers){
        if(safeu.safe_art !== null){
            let suvau = suvaUsers.find(x => x.visum == safeu.visum);
            if(arts.find(x => x.name == safeu.safe_art) ==
            undefined){
                let art = {
                    name: safeu.safe_art,
                    bos: [],
                    rte: {},
                    sas: [],
                    products: []
                };
                arts.push(art);
            }
            if(arts.find(x => x.name == safeu.safe_art)
            .products.find(x => x.name == safeu.safe_product) ==
            undefined){
                let product = {
                    name: safeu.safe_product,
                    pm: {},
                    teams: []
                };
                arts.find(x => x.name ==
                safeu.safe_art).products.push(product);
            }
            if(arts.find(x => x.name ==
                safeu.safe_art).products.find(x => x.name ==
                safeu.safe_product).teams.find(x => x.name ==
                safeu.safe_srum_team) == undefined){
                let team = {
                    name: safeu.safe_srum_team,
                    sm: {},
                    pos: [],
                    members: [],
                    art: safeu.safe_art
                };
                arts.find(x => x.name ==
                safeu.safe_art).products.find(x => x.name ==
                safeu.safe_product).teams.push(team);
            }
            let department = safeu.abteilung;
            if(safeu.bereich != null){
                department += " - " + safeu.bereich;
            }
            if(safeu.team != ""){
                department += " - " + safeu.team;
            }
        }
    }
}
```

```
        }
    let office = "";
    if(suvau.gebaeude != ""){
        office += suvau.gebaeude;
    }
    if(suvau.gebaeude != "" && suvau.buero != ""){
        office += " - ";
    }
    if(suvau.buero != ""){
        office += suvau.buero;
    }
    let person = {
        name: suvau.vorname + " " + suvau.nachname,
        visum: suvau.visum,
        function: suvau.funktion.de,
        email: suvau.email,
        phone: suvau.telefon,
        mobile: suvau.mobile,
        department: department,
        office: office,
        language: suvau.sprache,
        safe_role: safeu.safe_rolle,
        safe_team: safeu.safe_scrum_team,
        safe_product: safeu.safe_product,
        safe_art: safeu.safe_art
    }
    switch (person.safe_role){
        case "BO":
            arts.find(x => x.name ==
                safeu.safe_art).bos.push(person);
            break;
        case "RTE":
            arts.find(x => x.name == safeu.safe_art).rte =
                person;
            break;
        case "SA":
            arts.find(x => x.name ==
                safeu.safe_art).sas.push(person);
            break;
        case "PM":
            arts.find(x => x.name ==
                safeu.safe_art).products.find(x => x.name ==
                    safeu.safe_product).pm = person;
            break;
        case "SM":
            arts.find(x => x.name ==
                safeu.safe_art).products.find(x => x.name ==
                    safeu.safe_product).teams.find(x => x.name ==
                        safeu.safe_scrum_team).sm = person;
            break;
    }
}
```

```
        case "PO":  
            arts.find(x => x.name ==  
safeu.safe_art).products.find(x => x.name ==  
safeu.safe_product).teams.find(x => x.name ==  
safeu.safe_scrum_team).pos.push(person);  
            break;  
        case "":  
        default:  
            arts.find(x => x.name ==  
safeu.safe_art).products.find(x => x.name ==  
safeu.safe_product).teams.find(x => x.name ==  
safeu.safe_scrum_team).members.push(person);  
            break;  
        }  
    }  
    return arts;  
}  
  
function buildNavigation(safeStructure) {  
    document.getElementById("nav").innerHTML = "";  
    for(let artNr in safeStructure){  
        let art = safeStructure[artNr];  
        buildART(art, artNr)  
        for(let bo of art.bos){  
            buildBO(bo, artNr);  
        }  
        buildRTE(art.rte, artNr);  
        for(let sa of art.sas){  
            buildSA(sa, artNr);  
        }  
        for(let productNr in art.products){  
            let product = art.products[productNr];  
            buildProduct(product, productNr, artNr);  
            if(product.pm.name){  
                buildPM(product.pm, productNr, artNr);  
            }else{  
                manualBuildPM(product.name, productNr, artNr);  
            }  
            for(let teamNr in product.teams){  
                let team = product.teams[teamNr]  
                buildTeam(team, teamNr, productNr, artNr);  
                buildSM(team.sm, teamNr, productNr, artNr);  
                for(let poNr in team.pos){  
                    let po = team.pos[poNr];  
                    buildPO(po, teamNr, productNr, artNr);  
                }  
                for(let member of team.members){  
                    buildMember(member, teamNr, productNr, artNr);  
                }  
            }  
        }  
    }  
}
```

```
        }
    }
}

darkMode();

}

function buildART(art, artNr) {
    let container = document.getElementById("nav");
    let art_li = document.createElement('li');
    art_li.innerHTML = `<button class="btn btn-toggle"
        data-bs-toggle="collapse"
        data-bs-target="#art-${artNr}"
        aria-expanded="false">
        
        ${art.name}
    </button>
    <div class="collapse"
        id="art-${artNr}">
        <ul class="list-unstyled"
            id="art-${artNr}-ul">
        </ul>
    </div>`;
    art_li.childNodes[1].addEventListener("focus",
        ()=>{showGroup(art, "ART")});
    container.appendChild(art_li);
}

function buildBO(bo, artNr) {
    let bo_li = document.createElement('li');
    bo_li.innerHTML = `<a href="#" class="link-dark">
        
        BO - ${bo.name}
    </a>`;
    bo_li.firstChild.addEventListener("focus",
        ()=>{showPerson(bo)});
    document.getElementById("art-"+artNr+"-ul").appendChild(bo_li);
}

function buildRTE(rte, artNr) {
    if(rte.name != undefined){
        let rte_li = document.createElement('li');
        rte_li.innerHTML = `<a href="#" class="link-dark">
            
            RTE - ${rte.name}
        </a>`;
        rte_li.firstChild.addEventListener("focus",
            ()=>{showPerson(rte)}));
    }
}
```

```
        document.getElementById("art-" + artNr +
            "-ul").appendChild(rte_li);
    }
}

function buildSA(sa, artNr) {
    let sa_li = document.createElement('li');
    sa_li.innerHTML = `<a href="#" class="link-dark">
        
        SA - ${sa.name}
    </a>`;
    sa_li.firstChild.addEventListener("focus",
        ()=>{showPerson(sa)});
    document.getElementById("art-"+artNr+"-ul").appendChild(sa_li);
}

function buildProduct(product, productNr, artNr) {
    if(product.name != ""){
        let product_li = document.createElement('li');
        product_li.classList.add("lvl");
        product_li.innerHTML = `<button class="btn btn-toggle"
            data-bs-toggle="collapse"
            data-bs-target=
                "#product-${artNr}-${productNr}"
            aria-expanded="false">
            
            ${product.name}
        </button>
        <div class="collapse"
            id="product-${artNr}-${productNr}">
            <ul class="list-unstyled"
                id="product-${artNr}-
                    ${productNr}-ul">
            </ul>
        </div>`;
        product_li.firstChild.addEventListener("focus",
            ()=>{showGroup(product, "Product")});
        document.getElementById("art-" + artNr +
            "-ul").appendChild(product_li);
    }
}

function buildPM(pm, productNr, artNr) {
    if(pm != null){
        let pm_li = document.createElement('li');
        pm_li.innerHTML = `<a href="#" class="link-dark">
            
            PM - ${pm.name}
        </a>`;
    }
}
```

```
        </a>`  
    pm_li.firstChild.addEventListener("focus",  
    ()=>{showPerson(pm)});  
    document.getElementById("product-" + artNr + "-" +  
    productNr + "-ul").appendChild(pm_li);  
}  
}  
function manualBuildPM(product, productNr, artNr) {  
    let pm = null;  
    switch (product){  
        case "Kundenbetreuung":  
            pm = {  
                department: "IF - IFM",  
                email: "stavros.antifakos@suva.ch",  
                function: "Bereichsleiter",  
                language: "DE",  
                mobile: "+41795992166",  
                name: "Stavros Antifakos",  
                office: "RM Z - OG06.6114",  
                phone: "+41414195546",  
                safe_art: "ARTWI",  
                safe_product: "IT Management Services",  
                safe_role: "PO",  
                safe_team: "Koalas",  
                visum: "ASV"  
            }  
            break;  
        //hier werden alle nicht als solche markierten PMs auf die  
        gleiche Art erstellt  
        case "":  
        default:  
            break;  
    }  
    buildPM(pm, productNr, artNr);  
}  
function buildTeam(team, teamNr, productNr, artNr) {  
    if(team.name != ""){  
        let team_li = document.createElement('li');  
        team_li.classList.add("lvl");  
        team_li.innerHTML = `<button class="btn btn-toggle" data-bs-  
        toggle="collapse" data-bs-target=  
        "#product-${artNr}-${productNr}-${teamNr}" aria-expanded="false">  
              
            ${team.name}  
        </button>  
        <div class="collapse"  
        id="product-${artNr}-${productNr}-  
        ${teamNr}">
```

```
        <ul class="list-unstyled"  
            id="team-${artNr}-${productNr}-  
            ${teamNr}-ul">  
            </ul>  
        </div>  
        team_li.firstChild.addEventListener("focus",  
            ()=>{showGroup(team, "Team")});  
        document.getElementById("product-" + artNr + "-" +  
            productNr + "-ul").appendChild(team_li);  
    }  
}  
  
function buildSM(sm, teamNr, productNr, artNr) {  
    if(sm.name !== undefined){  
        let sm_li = document.createElement('li');  
        sm_li.innerHTML = `<a href="#" class="link-dark">  
              
            SM - ${sm.name}  
        </a>`  
        sm_li.firstChild.addEventListener("focus",  
            ()=>{showPerson(sm)});  
        document.getElementById("team-" + artNr + "-" + productNr +  
            "-" + teamNr + "-ul").appendChild(sm_li);  
    }  
}  
  
function buildPO(po, teamNr, productNr, artNr) {  
    if(po.name !== undefined){  
        let po_li = document.createElement('li');  
        po_li.innerHTML = `<a href="#" class="link-dark">  
              
            PO - ${po.name}  
        </a>`  
        po_li.firstChild.addEventListener("focus",  
            ()=>{showPerson(po)});  
        document.getElementById("team-" + artNr + "-" + productNr +  
            "-" + teamNr + "-ul").appendChild(po_li);  
    }  
}  
  
function buildMember(member, teamNr, productNr, artNr) {  
    if(member.safe_team != "" && member.safe_product != ""){  
        let member_li = document.createElement('li');  
        member_li.innerHTML = `<a href="#" class="link-dark">  
              
            ${member.name}  
        </a>`  
    }  
}
```

```
        member_li.firstChild.addEventListener("focus",
        ()=>{showPerson(member)});
        document.getElementById("team-" + artNr + "-" + productNr +
        "-" + teamNr + "-ul").appendChild(member_li);
    }
}

function showPerson(person) {
    let image;
    switch (person.safe_role) {
        case "BO":
            image = "engineer";
            break;
        case "RTE":
            image = "cog";
            break;
        case "SA":
            image = "magnifying";
            break;
        case "PM":
            image = "talk";
            break;
        case "SM":
            image = "board";
            break;
        case "PO":
            image = "check";
            break;
        default:
            image = "person";
            break;
    }
    document.getElementById("icon-content").src =
`images/${image}.svg`;
    document.getElementById("h1-content").innerText = person.name +
" (" + person.visum + ")";
    let container = document.getElementById("info-content");
    container.innerHTML = "";
    let info_div = document.createElement('div');
    info_div.innerHTML = getPersonInfo(person);
    container.appendChild(info_div);
    darkMode();
}
function showGroup(group, type) {
    let image;
    switch (type) {
        case "ART":
            image = "train";
            break;
        case "Product":
```

```
        image = "product";
        break;
    case "Team":
    default:
        image = "people";
        break;
}
document.getElementById("icon-content").src =
`images/${image}.svg`;
document.getElementById("h1-content").innerText = group.name;
let container = document.getElementById("info-content");
container.innerHTML = "";
let info_div = document.createElement('div');
info_div.innerHTML = getGroupInfo(group, type);
container.appendChild(info_div);
darkMode();
}

function getPersonInfo(person) {
let funktion = "";
let safe = "";
let safeART = "";
let safeProduct = "";
let safeTeam = "";
let safeRole = "";
let email = "";
let telefon = "";
let mobile = "";
let department = "";
let office = "";
let language = "";
if(person.function != ""){
    funktion = `
        <p class="txt-info">${person.function}</p>`;
}
if(person.safe_art != ""){
    safeART = person.safe_art;
    if(person.safe_product != ""){
        safeProduct = " - " + person.safe_product;
        if(person.safe_team != ""){
            safeTeam = " - " + person.safe_team;
        }
    }
    if(person.safe_role != ""){
        safeRole = " - " + person.safe_role;
    }
    safe = `
        <p class="txt-info">${safeART + safeProduct +
        safeTeam + safeRole}</p>`;
}
}
```

```
        }
        if(person.email != ""){
            email = `
                <p class="txt-info">
                    <a href=mailto:${person.email}
                        class="link">${person.email}</a></p>
            `
        }
        if(person.phone != ""){
            telefon = `
                <p class="txt-info">
                    <a href=tel:${person.phone}
                        class="link">${person.phone}</a></p>
            `
        }
        if(person.mobile != ""){
            mobile = `
                <p class="txt-info">
                    <a href="tel:${person.mobile}"
                        class="link">${person.mobile}</a></p>
            `
        }
        if(person.department != ""){
            department = `
                <p class="txt-info">${person.department}</p>
            `
        }
        if(person.office != ""){
            office = `
                <p class="txt-info">${person.office}</p>
            `
        }
        if(person.language != ""){
            language = `
                <p class="txt-info">${person.language}</p>
            `
        }
        let info = funktion + safe + email + telefon + mobile +
        department + office + language;
        return info;
    }
    function getGroupInfo(group, type) {
        let email = "";
        let link = "";
        let beschreibung = "";

        switch (type){
            case "ART":
                email = "$GOM-SAFE-" + group.name + "@suvanet.ch";
                switch (group.name){
                    case "ARTWI":
                        beschreibung = "Workplace & Infrastrukturen";
                }
        }
    }
}
```

```
        break;
    case "ARTVS":
        beschreibung = "Versichern & Schaden-
Management";
        break;
    case "ARTMP":
        beschreibung = "Marktbearbeitung & Prävention";
        break;
    default:
        break;
    }
    break;
case "Product":
//beschreibung = "Product " + group.name + "
Beschreibung";
    break;
case "Team":
    email = "$GOM-SAFE-" + group.art + "-" + group.name +
"@suvanet.ch";
//beschreibung = "Team " + group.name + " Beschreibung";
    break;
default:
    break;
}

if(email != ""){
    email = `
        <p class="txt-info"><a href="mailto:${email}"`  

class="link" target="_blank">${email}</a></p>`  

}
if(link != ""){
    link = `
        <p class="txt-info"><a href="${link}"`  

class="link" target="_blank">${link}</a></p>`  

}
if(beschreibung != ""){
    beschreibung = `
        <p class="txt-info">${beschreibung}</p>`  

}
let info = beschreibung + email + link;
return info;
}
```

9.1.5 options.js

```
darkMode();
window.addEventListener('load', async () => {
    document.getElementById("home").addEventListener("click",
loadHome);
    constructOptions();
```

```
});

function loadHome(){
    chrome.tabs.update({url: chrome.runtime.getURL("index.html")});
}

function handleButtonClick(event) {
    let current = event.target.parentElement.querySelector(
        `.${current}`);
    );
    if (current && current !== event.target) {
        current.classList.remove("current");
    }

    let mode = event.target.dataset.mode;
    event.target.classList.add("current");
    chrome.storage.sync.set({ mode });

    chrome.tabs.reload();
}

function darkMode(){
    chrome.storage.sync.get("mode", (data) => {
        let currentMode = data.mode;
        if (currentMode == "dark" || (currentMode == "default" &&
            window.matchMedia('prefers-color-scheme: dark').matches)) {
            for (icon of document.getElementsByClassName("icon")) {
                let source = icon.src;
                if (!source.includes("-dark") && source.includes(".svg")) {
                    console.log(source);
                    icon.src = source.slice(0, source.length - 4) +
                    "-dark.svg";
                }
            }
            document.getElementById("l-light").hidden = true;
            document.getElementById("l-dark").hidden = false;
        } else {
            document.getElementById("l-light").hidden = false;
            document.getElementById("l-dark").hidden = true;
            document.getElementById("css-dark").disabled = true;
        }
        if(window.matchMedia('prefers-color-scheme: dark').matches){
            document.getElementById("icon").href = "images/options-
dark.svg";
        }
    });
}

function constructOptions() {
    chrome.storage.sync.get("mode", (data) => {
```

```
let currentMode = data.mode;
let defaultButton = document.getElementById("defaultButton");
let lightButton = document.getElementById("lightButton");
let darkButton = document.getElementById("darkButton");
defaultButton.dataset.mode = "default";
lightButton.dataset.mode = "light";
darkButton.dataset.mode = "dark";
if (currentMode == "default" || currentMode == undefined) {
    defaultButton.classList.add("current");
} else if (currentMode == "light") {
    lightButton.classList.add("current");
} else if (currentMode == "dark") {
    darkButton.classList.add("current");
}
defaultButton.addEventListener("click", handleClick);
lightButton.addEventListener("click", handleClick);
darkButton.addEventListener("click", handleClick);
});
```

9.1.6 index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Suva Mitarbeiter</title>
    <link rel="icon" type="image/x-icon" href="images/icon.svg"
    id="icon">

    <link rel="stylesheet" type="text/css"
    href="libs/bootstrap-5.1.3-dist/css/bootstrap.css">
    <link rel="stylesheet" type="text/css" href="css/style.css">
    <link rel="stylesheet" type="text/css" href="css/dark.css"
    id="css-dark">
    <script src="libs/bootstrap-5.1.3-dist/js/bootstrap.bundle.js">
    </script>

    <script src="app.js"></script>
</head>
<body>
<div id="container">
    <div id="header">
        
        <h1>Suva Mitarbeiter</h1>
    </div>
    <div id="interface">
        <button class="btn-tile" id="safe">
            
            <br>
            <h1>Suva SAFe</h1>
        </button>
    </div>
</div>
```

```
</button>
<!--
<button class="btn-tile" id="more">
    
    <br>
    <h1></h1>
</button>
-->
<button class="btn-tile" id="options">
    
    <br>
    <h1>Optionen</h1>
</button>
</div>
</div>
</body>
</html>
```

9.1.7 safe.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Suva SAFe</title>
    <link rel="icon" type="image/x-icon" href="images/safe.svg"
    id="icon">

    <link rel="stylesheet" type="text/css"
    href="libs/bootstrap-5.1.3-dist/css/bootstrap.css">
    <link rel="stylesheet" type="text/css" href="css/style.css">
    <link rel="stylesheet" type="text/css" href="css/safe.css">
    <link rel="stylesheet" type="text/css" href="css/dark.css"
    id="css-dark">
    <link rel="stylesheet" type="text/css" href="css/safe-dark.css"
    id="css-safe-dark">
    <script src="libs/bootstrap-5.1.3-dist/js/bootstrap.bundle.js">
    </script>

    <script src="safe.js"></script>
</head>
<body>
    <button id="home" title="Home">
        
    </button>
    <div id="container">
        <div id="header">
            
            <h1>
                Suva SAFe
                <button id="info" title="Was ist SAFe?">
```

```

</button>
</h1>
</div>
<div id="interface">
  <div id="navigation">
    <ul class="list-unstyled" id="nav">
      
    </ul>
  </div>
  <div id="fade"></div>
  <div id="content">
    <div id="title-content">
      <img src="" class="icon" id="icon-content">
      <h1 id="h1-content"></h1>
    </div>
    <div id="info-content">
    </div>
  </div>
  </div>
</div>
</body>
</html>
```

9.1.8 options.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Optionen</title>
    <link rel="icon" type="image/x-icon" href="/images/options.svg"
          id="icon">

    <link rel="stylesheet" type="text/css"
          href="libs/bootstrap-5.1.3-dist/css/bootstrap.css">
    <link rel="stylesheet" type="text/css" href="css/style.css">
    <link rel="stylesheet" href="css/options.css">
    <link rel="stylesheet" type="text/css" href="css/dark.css"
          id="css-dark">
    <script src="libs/bootstrap-5.1.3-dist/js/bootstrap.bundle.js">
    </script>

    <script src="options.js"></script>
  </head>
  <body>
    <button id="home" title="Home">
      
    </button>
    <div id="container">
```

```
<div id="header">
  
  <h1>Optionen</h1>
</div>
<div id="interface">
  <div class="option">
    
    
    <p class="txt-option">Erscheinungsbild</p>
    <div class="content-option">
      <button class="btn-option"
        id="defaultButton">Systemstandard</button>
      <button class="btn-option" id="lightButton">hell</button>
      <button class="btn-option" id="darkButton">dunkel</button>
    </div>
  </div>
  <!--
  <div class="option">
    
    <p class="txt-option">Weitere Optionen folgen...</p>
  </div>
  -->
  </div>
</div>
</body>
</html>
```

9.1.9 style.css

```
@import
url('https://fonts.googleapis.com/css2?family=Roboto&display=swap');

body {
  min-height: 400px;
}
* {
  font-family: 'Roboto', sans-serif;
}

#container{
  margin: 5vh 5%;
  height: 90vh;
  min-height: 300px;
  display: flex;
  flex-direction: column;

}
#header {
  height: 10%;
```

```
min-height: 80px;
}

#interface{
    height: 85%;
    display: flex;
    flex-wrap: wrap;
    justify-content: flex-start;
    gap: 5%;
}

#home, #info {
    background: none;
    width: 50px;
    height: 50px;
    padding: 0;
    margin-left: 2%;
    border: none;
    float: left;
    opacity: 0.5;
}
#info {
    float: right;
    vertical-align: center;
}
#btn-info {
    margin: 10px;
}
#btn-home, #btn-info {
    height: 30px;
}
#home:hover, #info:hover {
    opacity: 0.7;
    background: radial-gradient(#e3eaf1, #e3eaf1, transparent,
    transparent);
}
#home:active, #info:active {
    opacity: 1;
    background: radial-gradient(#c6d1df, #c6d1df, #e3eaf1,
    transparent, transparent, transparent);
}

h1 {
    font-size: 40px;
    font-weight: bold;
}
#logo, .logo{
    width: 50px;
    float: left;
    margin-right: 10px;
}
```

```
.btn-tile {  
    background: #d9dbe2;  
    width: 250px;  
    height: 250px;  
    outline: none;  
    border: none;  
    border-radius: 15px;  
    margin-bottom: 10px;  
    opacity: 0.7;  
    font-size: 40px;  
}  
.btn-tile:hover{  
    background: radial-gradient(#ecedf1, #ecedf1, #d9dbe2, #d9dbe2);  
    opacity: 1;  
}  
.btn-tile:active {  
    background: radial-gradient(white, white, white, #ecedf1,  
    #d9dbe2, #d9dbe2);  
    box-shadow: 0 0 0 2px white,  
    0 0 0 4px black;  
}  
.icon-tile {  
    width: 100px;  
    margin: 20px;  
}  
  
::-webkit-scrollbar {  
    width: 6px;  
    height: 6px;  
}  
::-webkit-scrollbar-track {  
    background: transparent;  
}  
::-webkit-scrollbar-thumb {  
    background: #aaaeb7;  
    border-radius: 10px;  
}  
::-webkit-scrollbar-thumb:hover {  
    background: #7b818b;  
}  
::-webkit-scrollbar-thumb:active {  
    background: #626770;  
}  
::-webkit-scrollbar-track-piece:end {  
    background: transparent;  
    margin-bottom: 10px;  
}  
  
::-webkit-scrollbar-track-piece:start {
```

```
background: transparent;
margin-top: 10px;
}

::selection {
    background-color: #c6d1df;
}
::-moz-selection {
    background-color: #c6d1df;
}

.loader {
    height: 100px;
    opacity: 0.7;
    margin-left: 5%;
    margin-top: 3%;
    background: radial-gradient(white, white, transparent,
        transparent);
    -webkit-animation: spin 2s linear infinite;
    animation: spin 2s linear infinite;
}
@-webkit-keyframes spin {
    0% { -webkit-transform: rotate(0deg); }
    100% { -webkit-transform: rotate(360deg); }
}
@keyframes spin {
    0% { transform: rotate(0deg); }
    100% { transform: rotate(360deg); }
}

@media only screen and (max-width: 1200px) {
    #interface {
        justify-content: space-evenly;
    }
}
```

9.1.10 safe.css

```
body {
    min-width: 800px;
}
#container {
    min-height: 300px;
}
#interface {
    border: solid #d9dbe2 3px;
    border-radius: 17px;
    background-color: #d9dbe2;
    justify-content: space-evenly;
    gap: 0;
    position: relative;
```

```
}

#navigation {
    width: 38%;
    height: 100%;
    border-radius: 15px 0 0 15px;
    padding: 10px;
    overflow-y: scroll;
    overflow-x: hidden;
}

#fade {
    position: absolute;
    right: calc(62% + 6px);
    overflow-y: hidden;
    height: 100%;
    width: 20px;
    background: linear-gradient(90deg, transparent, #d9dbe2);
}

#content {
    display: flex;
    flex-direction: column;
    justify-content: start;
    gap: 30px;
    width: 61%;
    height: 100%;
    margin-left: 1%;
    background-color: white;
    border-radius: 15px;
}

#icon-content {
    width: 60px;
    float: left;
}

#h1-content {
    padding: 10px 30px 0 70px;
}

#title-content {
    padding-left: 30px;
    margin-top: 20px;
}

#info-content {
    margin-right: 1%;
    margin-bottom: 10px;
    overflow-y: scroll;
    overflow-x: hidden;
    padding: 0 20px 0 30px;
}

.icon-info {
    width: 30px;
    float: left;
    margin-right: 10px;
```

```
}

.txt-info {
    margin: 0 20px 20px 50px;
    font-size: 20px;
}
.a.link {
    color: #404f67;
    opacity: 0.7;
}
.a.link:hover {
    opacity: 1;
    background: radial-gradient(#e3eaf1, #e3eaf1, transparent,
transparent);
}
.a.link:active {
    color: black;
    background: radial-gradient(#c6d1df, #c6d1df, #e3eaf1,
transparent, transparent, transparent);
}

.icon-btn {
    height: 30px;
    min-width: 30px;
    margin-right: 10px;
    vertical-align: 100%;
}
.lvl {
    margin-left: 20px;
}
.btn-toggle {
    display: inline-flex;
    align-items: center;
    text-align: left;
    white-space: nowrap;
    padding: 2px 5px;
    margin: 5px;
    font-weight: 400;
    font-size: 20px;
    color: #484c54;
    border: 0;
    border-radius: 10px;
}
.btn-toggle::before {
    content: url(..../images/arrow.svg);
    width: 1.25em;
    line-height: 0;
    transition: transform .35s ease;
    transform-origin: .5em 50%;
    opacity: 0.7;
}
```

```
.btn-toggle[aria-expanded="true"] {  
    color: black;  
}  
.btn-toggle[aria-expanded="true"]::before {  
    transform: rotate(90deg);  
}  
a.link-dark {  
    color: #484c54;  
    display: inline-flex;  
    padding: 2px 5px;  
    margin-top: .125rem;  
    margin-left: 30px;  
    font-size: 20px;  
    text-decoration: none;  
    border-radius: 10px;  
    overflow:hidden;  
    white-space:nowrap;  
}  
.btn-toggle:hover, a.link-dark:hover {  
    color: black;  
    background: radial-gradient(#ecedf1, #ecedf1, #d9dbe2, #d9dbe2);  
}  
.btn-toggle:focus, a.link-dark:focus {  
    color: black;  
    background: radial-gradient(#e3eaf1, #e3eaf1, #c6d1df, #c6d1df);  
}  
.btn-toggle:focus, a.link-dark:focus {  
    box-shadow: 0 0 0 0 transparent,  
                0 0 0 2px black;  
    font-weight: bold;  
}
```

9.1.11 options.css

```
#interface {  
    flex-direction: column;  
    justify-content: start;  
}  
  
.option {  
    max-width: 1000px;  
    background-color: #d9dbe2;  
    border: solid #d9dbe2 3px;  
    border-radius: 15px;  
}  
.content-option {  
    background-color: white;  
    width: 100%;  
    border-radius: 13px;  
    padding: 20px;
```

```
}

.icon-option {
    width: 50px;
    float: left;
    margin: 15px 10px 0 20px;
}

.txt-option {
    font-size: 40px;
    margin: 10px 30px 30px 0;
    padding-left: 80px;
}

.btn-option {
    background-color: #d9dbe2;
    height: 50px;
    width: 200px;
    font-size: 20px;
    outline: none;
    border: none;
    border-radius: 10px;
    margin-right: 10px;
    opacity: 0.7;
}

.btn-option.current {
    background: radial-gradient(#e3eaf1, #e3eaf1, #c6d1df, #c6d1df);
    box-shadow: 0 0 0 2px white,
                0 0 0 4px black;
    opacity: 1;
}

.btn-option:hover {
    background: radial-gradient(#ecedf1, #ecedf1, #d9dbe2, #d9dbe2);
    opacity: 1;
}

.btn-option:active {
    background: radial-gradient(white, white, white, #ecedf1, #d9dbe2,
#d9dbe2);
    box-shadow: 0 0 0 2px white,
                0 0 0 4px black;
}

.btn-option.current:hover {
    background: radial-gradient(#e3eaf1, #e3eaf1, #c6d1df, #c6d1df);
}

.btn-option.current:active {
    background: radial-gradient(white, white, white, #e3eaf1, #c6d1df,
#c6d1df);
}
```

9.1.12 dark.css

```
body {
```

```
background-color: black;
}
* {
    color: white;
}

#home:hover, #info:hover {
    background: radial-gradient(#677993, #404f67, transparent,
transparent);
}
#home:active, #info:active {
    background: radial-gradient(#677993, #677993, #404f67,
transparent, transparent, transparent);
}

.btn-tile, .btn-option {
    background: #484c54;
}
.btn-option:hover, .btn-tile:hover {
    background: radial-gradient(#484c54, #484c54, #24262a, #24262a);
}
.btn-option:active, .btn-tile:active {
    background: radial-gradient(#626770, #626770, #626770, #484c54,
#24262a, #24262a);
    box-shadow: 0 0 0 3px black,
                0 0 0 5px white;
}
.btn-option.current, .btn-option.current:hover {
    background: radial-gradient(#677993, #677993, #404f67, #404f67);
    box-shadow: 0 0 0 3px black,
                0 0 0 5px white;
}
.btn-option.current:active {
    background: radial-gradient(#97a5b9, #97a5b9, #677993, #404f67,
#404f67);
}

.option {
    background-color: #484c54;
    border-color: #484c54;
}
.content-option {
    background-color: black;
}

::-webkit-scrollbar-thumb {
    background: #7b818b;
}
::-webkit-scrollbar-thumb:hover {
```

```
background: #aaaeb7;
}

::-webkit-scrollbar-thumb:active {
    background: #d9dbe2;
}

::selection {
    background-color: #677993;
}
::-moz-selection {
    background-color: #677993;
}

.loader {
    background: radial-gradient(#626770, #626770, transparent,
    transparent);
}
```

9.1.13 safe-dark.css

```
#interface {
    border-color: #484c54;
    background-color: #484c54;
}
#content {
    background-color: black;
}
a.link {
    color: #e3eaf1;
}
a.link:hover {
    background: radial-gradient(#404f67, #404f67, transparent,
    transparent);
}
a.link:active {
    color: white;
    background: radial-gradient(#677993, #677993, #404f67,
    transparent, transparent, transparent);
}
#fade {
    background: linear-gradient(90deg, transparent, #484c54);
}
.btn-toggle {
    color: #d9dbe2;
}
.btn-toggle[aria-expanded="true"] {
    color: white;
}
a.link-dark {
    color: #d9dbe2;
```

```
}

.btn-toggle::before {
    content: url(..../images/arrow-dark.svg);
}

.btn-toggle:hover, a.link-dark:hover {
    color: white;
    background: radial-gradient(#626770, #626770, transparent,
    transparent);
}

.btn-toggle:focus, a.link-dark:focus {
    box-shadow: 0 0 0 0 transparent,
                0 0 0 2px white;
    color: white;
    background: radial-gradient(#313e54, #313e54, #404f67, #404f67);
}
```