



Diplomarbeit

TEAM RESSOURCEN MANAGEMENT

Projektteam:
Studiengang:
Klasse
Abgabe Datum:
Diplomlehrer:
Fachexperte:

Marco Studer
Technischer Informatiker HF
OTIN-18-T-a TEKÖ Olten
25.10.2021
Benjamin Bani
Marc Mangold



Inhaltsverzeichnis

1	Management Summary	5
1.1	Ausgangslage	5
1.2	Vorgehen.....	5
1.3	Ergebnisse	5
2	Einleitung.....	6
3	Qualifikationsprofil.....	7
4	Kurzer beruflicher Werdegang von Marco Studer.....	9
5	Projektinitialisierung.....	10
5.1	Projektauftrag.....	10
5.1.1	Projektbeschreibung.....	10
	Systemziele	11
	Vorgehensziele.....	12
5.1.2	Projekttyp, Organisation und Genehmigung	12
5.1.3	Projektplanung	13
6	Projektplanung	13
6.1	Projektorganisation- und Leitung	13
6.1.1	Projektrollen	14
6.2	Projektstrukturplan	15
6.3	Projektablaufplan	16
6.4	Ressourceneinteilung.....	17
6.4.1	Übersicht Projektdauer.....	17
6.4.2	Übersicht pro Woche.....	17
6.5	Kommunikation	18
7	Realisierung Projekt «TRESMENT».....	19



7.1	Grobbeschrieb Varianten	19
7.1.1	Webanwendung	19
7.1.2	Desktopanwendung.....	20
7.2	Nutzwertanalyse.....	21
7.2.1	Kriterien	21
7.2.2	Präferenzmatrix	22
7.2.3	Nutzwertanalyse.....	23
7.2.4	Sensitivitätsanalyse.....	24
7.2.5	Sensitivitätsanalyse 2.....	25
7.2.6	Sensitivitätsanalyse 3.....	26
7.2.7	Logische Argumentation	26
7.3	Entwicklung	27
7.3.1	Programmiersprache.....	27
7.3.2	Infrastruktur	28
7.3.3	Backend API.....	28
7.3.3.1	FastAPI.....	29
7.3.3.2	Zeep	30
7.3.3.3	Datenbank	31
7.3.3.4	Entity Relationship Model.....	32
7.3.3.5	Physisches Datenmodell.....	33
7.3.3.6	Threading	37
7.3.4	Frontend	38
7.3.4.1	Entwürfe	39
7.3.4.2	Entwicklung der Anwendung.....	41
7.3.4.2.1	Entwicklung Oberfläche	43
7.3.4.2.2	Entwicklung Datenbank.....	51



7.4	Testing.....	52
7.4.1	Update 24.09.2021	52
7.4.2	Update 29.09.2021	52
7.4.3	Update 01.10.2021	53
7.4.4	Update 06.10.2021	53
7.4.5	Update 14.10.2021	53
7.5	Dokumentation	54
7.5.1	Anwender	54
7.5.2	Administrator	54
7.5.3	Entwickler	54
7.6	GO – Live	55
7.7	Wirtschaftlichkeit	56
8	Projektabschluss.....	57
8.1	Reflexion der Projektziele	57
8.2	Reflexion und Erkenntnisse	59
8.2.1	Zusammenfassung.....	59
8.2.2	Weg zum Ziel	59
8.2.3	Methoden und Hilfsmittel.....	59
8.2.4	Planung	59
8.2.5	Einzelarbeit.....	59
9	Quellenverzeichnis.....	60
9.1	Webquellen	60
10	Verzeichnisse	60
10.1	Abbildungsverzeichnis	60
10.2	Tabellenverzeichnis.....	61
11	Glossar.....	62



12 Redlichkeitserklärung	63
13 Danksagung	64
14 Anhang	65
14.1 Pflichtenheft	65
14.2 Handbücher	90
14.3 Projektstatusbericht	129



1 Management Summary

Im Auftrag der SSI Schäfer AG erstelle ich eine neue Anwendung für das Planen und Einsehen der Absenzen jedes Mitarbeiters.

1.1 Ausgangslage

Bei der Planung der Absenzen benutzt jedes Team seine Tools oder Anwendungen, um sicher zu stellen, dass nicht alle zur gleichen Zeit abwesend sind. Meist verwenden die Teams eine eigens gestaltete Excel Liste oder versuchen via Outlook die An- bzw. Abwesenheiten im Griff zu haben. Mit der kommenden Umstrukturierung in die agile Softwareentwicklung führt die bisherige Umgangsweise mit den Abwesenheiten zu noch mehr Komplikationen. Es besteht die Möglichkeit, dass sich Mitarbeiter während eines Projekts aus zum Beispiel vier verschiedenen Teams zu einem Scrumteam zusammensetzen. Für die Planung muss der Projektleiter mit dem Teamleiter Rücksprache halten, um einen Überblick über die Ressourcen zu haben.

Dieser ungenaue und aufwändige Prozess soll mit einer Softwareanwendung erleichtert werden. Die Software umfasst eine Schnittstelle zum Microsoft Dynamics Navision ¹ und einer Visualisierung, die die beantragten und freigegebenen Urlaubstage jedes Mitarbeitenden der SSI Schäfer Schweiz anzeigt.

Im nachfolgenden Dokument wird jeweils von Urlaubstagen gesprochen, hierbei ist immer die Rede von beantragten und freigegebenen Urlaubstagen.

1.2 Vorgehen

Das Projekt ist im 4-Phasenmodell durchgeführt worden:

- Projektinitialisierung
- Projektplanung
- Projektrealisierung
- Projektabschluss

Die Projektdetails sind mittels Auftragsklärung ermittelt worden, aus der ein gültiges Pflichtenheft erstellt worden ist.

1.3 Ergebnisse

Als Ergebnis wird das Konzept und die Anwendung «TRESMENT» inklusive vollständiger Dokumentation vorgelegt.

Zu Beginn wurde bereits klar, dass die Durchführung des Projekts mit einem grossen Risikofaktor belegt ist. Das Projekt ist ein Potenzial- und Innovationsprojekt, welches in diesem Umfang noch nicht besteht.

¹ Microsoft Dynamics NAV ist eine Standardsoftware für ERP(Enterprise-Resource-Planning)-Systeme. Wenn im Dokument die Rede von Navision ist, ist Microsoft Dynamics NAV gemeint.



2 Einleitung

Als Abschlussarbeit an der TEKO Olten im Studiengang „Technischer Informatiker Applikationsentwicklung“ wurde der Auftrag erteilt ein Projekt Schritt für Schritt umzusetzen und dies zu dokumentieren. Im Anschluss wird eine Präsentation abgehalten.

Anhand der Analyse der Ausgangslage, Pflichtenhefterstellung und der aktuell nicht vorhandenen Planungsmittel wurde dies durch den Auftraggeber mitgegeben, um die genaue Auftragserklärung dieses Projektes zu bestimmen.

Ziel war es eine funktionierende Anwendung zu konzipieren, zu entwickeln und im vorgegebenen Zeitraum von 6 Wochen umzusetzen.

Das Projekt wurde anhand des 4-Phasen-Modells erstellt. Das Modell wird unter [1.2 Vorgehen](#) erläutert.

Im [fünften Kapitel](#) wird die Projektinitialisierung dargestellt, welche das Fundament des eigentlichen Projektes erklärt. Es werden auch der Projektauftrag und die verschiedenen Projektrollen beschrieben.

Im folgenden Kapitel wird die Projektplanung genauer erläutert. Darin enthalten sind die Meilensteine und die Arbeitspakete über das ganze Projekt. Die Projektrealisierung ([Kapitel 7](#)) bildet der Kern dieser Arbeit.

Im letzten Kapitel des 4-Phasen-Modells wird der Projektabschluss mit den Reflexionen der erreichten Ziele dargelegt. In den letzten drei Kapitel sind sämtliche Quellenverzeichnisse, ein Glossar und Anhänge (wie Statusberichte im Verlaufe des Projektes usw.) aufgelistet.

Das vorliegende Konzept beschreibt die Ausarbeitung und Entwicklung einer Anwendung für die Team Ressourcen Planung für die Firma SSI Schäfer Schweiz.



3 Qualifikationsprofil

Marco Studer
Dipl. Techniker HF, Informatik
Meisenweg 25
4528 Zuchwil

Qualifikationsprofil

Dipl. Techniker HF, Informatik

Menschen führen

Prozess 1

Als Berufsbildner die Aufgabe wahrgenommen Lehrlinge geführt, gefördert, gefordert und ausgebildet zu haben. Ziele definiert und die Ergebnisse auf Qualität geprüft und diese mit dem Auszubildenden reflektiert.

Entscheidungen fällen

Prozess 2

Gut durchdachte, geprüfte und sorgfältige Entscheidungen zu fällen und entstehende Konsequenzen gewissenhaft zu meistern in Bezug auf Evaluationen von Software Erweiterungen für die Kunden der SSI Schäfer Schweiz.

Projekte planen und leiten

Prozess 3

Komplexe Kundenanforderungen (Erweiterung der Software, Modellierung der Lagerfläche, Planung neuer Hallen) wurden mittels optimal zusammengestellten Teams oder selbstständig entwickelt und erfolgreich implementiert.

Unternehmensprozesse verstehen und mitgestalten

Prozess 6

Neue Prozesse im Unternehmen eingeführt, welche dazu dienen die Qualität zu verbessern und die geschäftlichen Prozesse zu optimieren. Die Erreichung von mehr Effizienz und mehr Qualität im Verhältnis zu den Kosten sind essenzielle Bestandteile im Prozess-Controlling.

Geschäftsziele erreichen

Prozess 7

Um die festgelegten Geschäftsziele zu erreichen sind eine gute Planung sowie die bestmögliche Umsetzung im Verhältnis der festgelegten Ressourcen zu koordinieren.

Probleme analysieren und lösen

Prozess 9

Als 2nd Level Supporter im Bereich Lagerverwaltung Software wurden komplexe Probleme analysiert, Lösungen ausgearbeitet und diese umgesetzt.



Sich persönlich weiter entwickeln
Prozess 10

Durch den aktuellen Arbeitsbereich aber auch im Rahmen der Weiterbildung konnten viele neue Kompetenzen erlangt, aber auch bestehende weiterentwickelt werden. Durch aktives Feedback einholen konnten viele Eigenschaften verbessert werden.

Applikationen entwickeln, Programme erstellen und testen
Prozess 16

Im Rahmen der Diplomarbeit eine neue Applikation designet, entwickelt und releast, um die Planung der Absenzen zu verbessern. Als Entwickler bei SSI Schäfer die Eigenschaft angeeignet, Applikationen weiterzuentwickeln und neue Module an bestehende Module anzuknüpfen.

Konzepte und Dienste entwickeln
Prozess 18

Diverse Kundenanforderungen wurden aufgenommen und die Bedürfnisse mittels Workshops tiefgründiger analysiert und ein Konzept für die Erweiterung erstellt.

Konzepte und Dienste umsetzen
Prozess 19

Aufgrund von technischen Spezifikationen oder ausgearbeiteten Pflichtenheften, Anforderungen implementiert und Testkonzepte dazu erstellt.



4 Kurzer beruflicher Werdegang von Marco Studer

2021 bis Heute	SSI Schäfer AG Muhen <ul style="list-style-type: none">▪ Verantwortlicher Berufsbildner▪ 2nd Level Supporter LVS▪ Projektmitarbeiter▪ Quality Level LVS
2019 bis 2021	SSI Schäfer AG Muhen <ul style="list-style-type: none">▪ 2nd Level Support LVS▪ Projektmitarbeiter▪ Quality Level LVS▪ Teilverantwortlicher Berufsbildner
2015 bis 2019	SSI Schäfer AG Muhen <ul style="list-style-type: none">▪ 1st Level Support LVS▪ 1st Level Support MFR▪ 1st Level Support SPS▪ Teilverantwortlicher Berufsbildner
2014 bis 2015	Planzer Härkingen <ul style="list-style-type: none">▪ Verantwortlicher Serverlandschaft▪ Verantwortlicher Automationsbereich
2011 bis 2014 Leitstand TKL	Migros Verteilbetrieb Neuendorf AG TKL Tiefkühl-Lager <ul style="list-style-type: none">▪ Datenaufbereitung und Verarbeitung für die Auftragsabwicklung▪ Materialfluss-Steuerung zur Sicherstellung einer optimalen Anlagenleistung▪ Systempflege zur Sicherstellung einer fristgerechten Filialbelieferung▪ First Level Support vom LVS WAMAS® (Warehouse Management Systeme von SSI SCHÄFER. Steuerung und Optimierung von Logistikprozessen innerhalb der Supply Chain Kette)▪ Mitarbeitende in der Anwendung des LVS schulen▪ Mitarbeit in verschiedenen Logistik-Projekten
2008 bis 2011 Lehre zum Logistiker EFZ in allen Non-Food Bereichen	Migros Verteilbetrieb Neuendorf AG <ul style="list-style-type: none">▪ Wareneingang, Kontrollen, Verbuchungen im Computer, Einlagerungen mit verschiedenen Fördersystemen und -Techniken▪ Warenausgang, Verpacken, Verdichten, Beschriftungen, Beladen von Bahn- und Strassentransporten▪ Zwischenlagerungen, Kommissionierungen mit und ohne Pick by Voice▪ Koordination, Prüfung der Bestände▪ 3 Monate externer Einsatz Distribution (Poststelle Solothurn)

Tabelle 1: Beruflicher Werdegang



5 Projektinitialisierung

5.1 Projektauftrag

Projekttitel	TRESMENT – Team Ressourcen Management		
Projektauftraggeber	SSI Schäfer AG		
Projektdaten			
Start:	13.09.2021	Ende:	25 Oktober 2021

Tabelle 2: Projektauftrag

5.1.1 Projektbeschreibung

Punkte	Beschreibung
Ausgangslage / Projektbegründung:	SSI Schäfer Schweiz wünscht eine Übersicht über die An- bzw. Abwesenheiten der Mitarbeiter, um die Ressourcen optimal einzuteilen und den Aufwand für die jeweiligen Teamleiter zu minimieren.
Sinn und Zweck / Nutzen:	Um die Planung der Abwesenheiten und Ferien für die Scrumteams zu vereinfachen, soll eine Anwendung entwickelt werden, welche die Planung der Mitarbeiter in den Teams bestmöglich unterstützt.
Projektrichtziel:	Eine neue Applikation für das «Team Ressourcen Management» ist erstellt.

Tabelle 3: Projektbeschreibung

**Systemziele**

Nr.	Kategorie	Beschreibung	Messgrösse	Priorität*
1	Funktionalität	Die Anwendung ist im internen Firmennetzwerk für alle zugänglich.	Die Anwendung kann nicht geladen werden, sobald der Anwender nicht mit dem internen Firmennetzwerk verbunden ist.	1
2	Funktionalität	Der Zugriff auf die Anwendung kann von jedem Nutzer ausgeführt werden.	Anwender müssen sich nicht Authentifizieren, um die Anwendung zu öffnen.	1
3	Funktionalität	Die Konfiguration kann nur von eingetragenen Administratoren administriert werden.	Die Administratoren werden in einer anwendungsinternen Liste eingetragen und können dort all-fällige Mutationen vornehmen. Der Entwickler ist der Super-User und hat Zugriff auf sämtliche Daten.	1
4	Funktionalität	Jeder Mitarbeiter kann eigene Teams erstellen, um die beantragten und freigegebenen Abwesenheiten anzuzeigen	Jeder Anwender kann in der Anwendung ein Team erstellen	1
5	Funktionalität	Jeder Mitarbeiter kann seine eigenen Teams erstellen, verwalten und löschen.	Jeder Anwender kann innerhalb der Anwendung ein Team verwalten und auch löschen.	1
6	Funktionalität	Jeder Mitarbeiter kann nur innerhalb des internen Firmennetzwerkes die Anwendung aufrufen.	Wird die Anwendung ausserhalb des internen Firmennetzwerkes aufgerufen, erscheint eine Fehlermeldung.	1
7	Wartung	Alle Funktionen, sowie der Aufbau der Anwendung werden im Confluence dokumentiert.	Ein neuer Bereich für die Dokumentation im Confluence wird für das Projekt erstellt.	2
8	Handhabung	Es wird eine Benutzeranleitung für die Anwender erstellt.	Die Anleitung wird im Confluence abgelegt und ist für alle Anwender zugänglich.	2

* **Priorität: 1 = hoch, 2 = mittel, 3 = tief**

Tabelle 4: Systemziele / Endergebnisse und Erfolgskriterien

**Vorgehensziele**

Nr.	Kategorie	Beschreibung	Messgrösse	Priorität*
1	Stakeholder	Das Design und Anforderungen der Anwendung werden mit den ausgewählten Anwendern abgesprochen.	Bedürfnisse der ausgesuchten Test Anwender werden vor der Realisierung ermittelt und die Applikation wird darauf aufgebaut.	1
2	Stakeholder	Es werden wöchentliche Meetings mit den Anwendern abgehalten, um diese über den Fortschritt zu informieren.	Funktions- und Designanpassungen werden den Anwendern laufend präsentiert, um allfällige Änderungswünsche vorzunehmen. Meetings werden fortlaufen terminiert	1
3	Stakeholder	Meeting alle 2 Wochen mit Fachexperte und Auftraggeber	Der Zwischenstand wird dem Fachexperten und Auftraggeber in wöchentlichen Meetings präsentiert	1

* **Priorität: 1 = hoch, 2 = mittel, 3 = tief**

Tabelle 5: Vorgehensziele / Endergebnisse und Erfolgskriterien

5.1.2 Projekttyp, Organisation und Genehmigung

Projekttyp	Begründung
<input type="checkbox"/> Routineprojekt <input type="checkbox"/> Komplexes Standardprojekt <input checked="" type="checkbox"/> Potenzial- / Innovationsprojekt <input type="checkbox"/> Pionierprojekt	Es wird zur Verbesserung und Innovation / bzw. Abklärung der bisherigen Zeiterfassungsübersicht dienen.
Organisationstyp	SQUAD nach Safe Scrum
Projektmitarbeiter & Stellenprozent:	50% Marco Studer Stellenprozente werden unter Berücksichtigung der Hauptberufstätigkeit berechnet.
Steering Comitee:	Keines
Beteiligte:	SSI Schäfer AG, Fachexperte: Marc Mangold TEKO Diplomlehrer: Benjamin Bäni Interne SSI Schäfer Testpersonen: Michael Collavo, Martin Fehlmann, Christian Burkhart
Zielgenehmigung	Die Ziele wurden im Rahmen des Pflichtenheftes bestätigt und von dem Fachexperten Marc Mangold und dem CFO Luca Cathomen unterzeichnet.

Tabelle 6: Projekttyp, Organisation und Genehmigung



5.1.3 Projektplanung

**Projektphasen/
Meilensteine (MS)**

- 1. Projektinitialisierung**
MS1: Kickoff
- 2. Projektplanung**
MS2: Projektplanung erstellt
- 3. Projektrealisierung**
MS3: Nutzwertanalyse
MS4: Vorbereitung Infrastruktur
MS5: Frontend fertiggestellt
MS6: Backend fertiggestellt
MS7: Erfolgreicher GO - Live
- 4. Projektabschluss Dokumentation**
MS5: Abgabe Druck und Versand
- 5. Abschluss- Präsentation**
Präsentation halten

Projektentscheid

- Das Projekt wird bewilligt.
 - Das Projekt wird abgelehnt.
- Begründung bei Ablehnung:
Datum: 11.08.2021
Bewilligt ohne Signatur durch den Auftraggeber

Tabelle 7: Projektplanung und Entscheid

6 Projektplanung

Die Projektplanung dient dazu eine Übersicht über die offenen Pendenzen und die Struktur zu schaffen.

6.1 Projektorganisation- und Leitung

Es handelt sich hierbei um eine reine Linien - Projektorganisation mit den folgend abgebildeten Informationen.

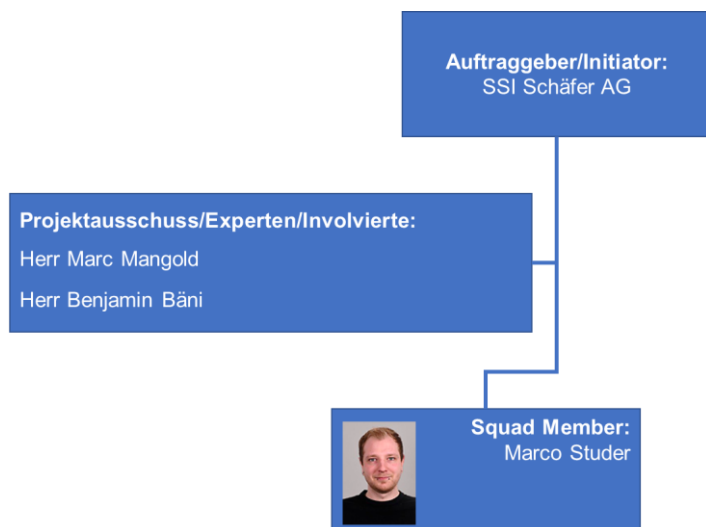


Abbildung 1: Projektorganisation



6.1.1 Projektrollen

Projektrolle	Beschreibung
Auftraggeber:	Der Auftraggeber hat die Aufgabenstellung ausgearbeitet und stellt zugleich den «Kunden» dar
Projektausschuss:	Der Projektausschuss besteht aus einem Experten/Dozenten der Hochschule TEKO Olten und einem Experten der Firma SSI Schäfer AG.
Squad Member mit 50%:	Der Squad Member erarbeitet zusammen mit dem Auftraggeber und dokumentiert mit dem Ziel, das Projekt nach Zeitplan und dem Interesse des Projektausschuss, wie dessen des Auftraggebers umzusetzen.

Tabelle 8: Projektrollen

Hinweis: Die Projektorganisation besteht aus der Konstellation von drei Rollen (Abbildung 1). In diesem Projekt gibt aufgrund der Zusammensetzung keinen Projektleiter.



6.2 Projektstrukturplan

Der Projektstrukturplan zeigt die Abarbeitungsreihenfolge der einzelnen Projektpunkte auf.

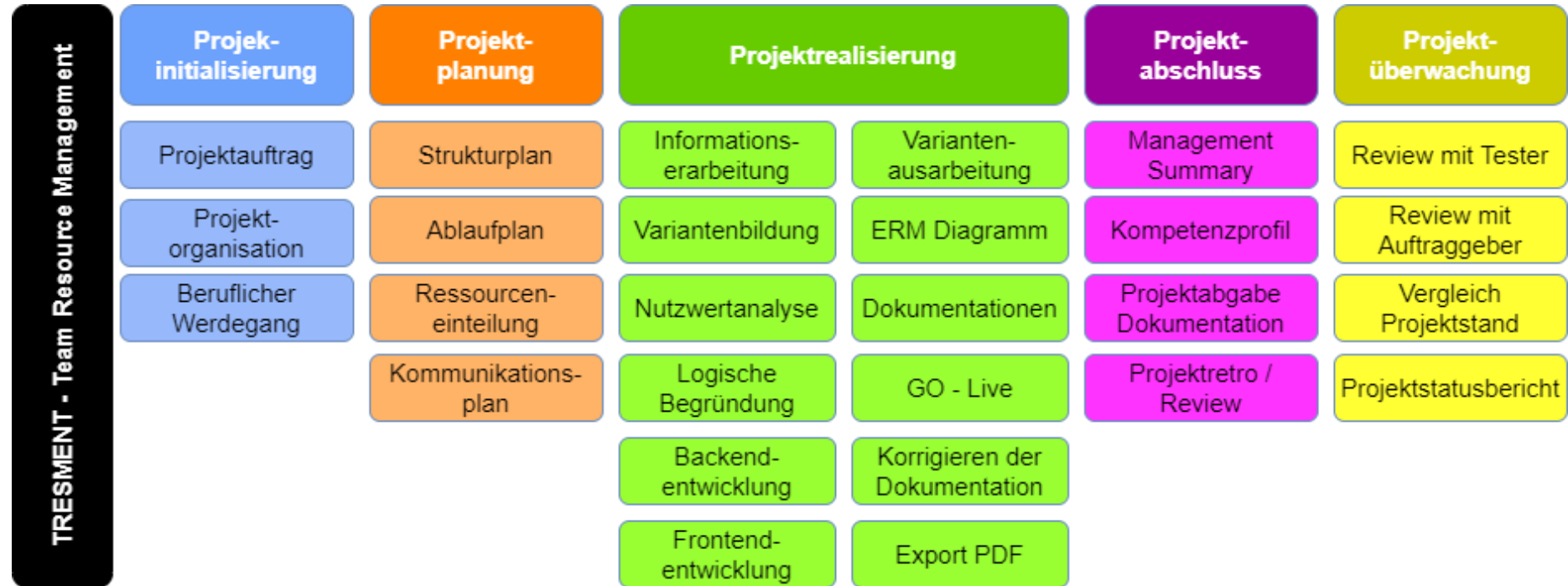


Abbildung 2: Projektstrukturplan



6.3 Projektablaufplan

PROJEKTPLAN			September															Oktober																														
			10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
			Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo
SSi Schäfer - Team Resource Management																																																
OTIN-18-T-a (Marco Studer)																																																
	h	Verantw.																																														
Kickoff																																																
Abgabe Arbeit																																																
Präsentation																																																
Projektinitialisierung																																																
	Projekttauftrag	2 Marco Studer																																														
	Projektorganisation	2 Marco Studer																																														
	Zielscheibe	1 Marco Studer																																														
	Dokumentation	3 Marco Studer																																														
Projektplanung																																																
	Projektstrukturplan	2 Marco Studer																																														
	Projektablaufplan	8 Marco Studer																																														
	Meilenstein Projektplanung																																															
	Ressourcenplan	1 Marco Studer																																														
	Kommunikationsplan	1 Marco Studer																																														
Projektrealisierung																																																
	Informationsbeschaffung	4 Marco Studer																																														
	Ideen- und Lösungsfindung	1 Marco Studer																																														
	Grobbeschreibung	3 Marco Studer																																														
	Nutzwertanalyse	4 Marco Studer																																														
	Meilenstein Nutzwertanalyse Begründung																																															
	Ausarbeitung Variante	7 Marco Studer																																														
	Wirtschaftlichkeitsrechnung	1 Marco Studer																																														
	Meilenstein Vorbereitung Infrastruktur																																															
	Realisierung Frontend	96 Marco Studer																																														
	Meilenstein Frontend fertiggestellt																																															
	Realisierung Backend	32 Marco Studer																																														
	Meilenstein Backend fertiggestellt																																															
	Testing mit Anwendern	4 Marco Studer																																														
	Anwenderdokumentation	8 Marco Studer																																														
	Entwicklerdokumentation	8 Marco Studer																																														
	Administratordokumentation	8 Marco Studer																																														
Projekt GO-Live																																																
	GO - Live	1 Marco Studer																																														
	Bugfixing	16 Marco Studer																																														
	Meilenstein Erfolgreich in betrieb genommen																																															
Projektabschluss																																																
	Management Summary	2 Marco Studer																																														
	Projektrückblick	2 Marco Studer																																														
	Zielprüfung	2 Marco Studer																																														
	Korrekturlesen / Formatieren	20 Marco Studer																																														
	Total	239	3	8	8	9	9	8	8	8	8	8	8	9	8	8	8	8	8	4	5	4	4	4	4	4	4	0	0	4	4	4	4	4	4	0	0	1	8	8	6	5	10	5	0			
Abwesenheiten & Aufwand																																																
	Marco Studer		3	8	8	8	6	8	8	8	8	8	9	8	9	9	9	9	9	2	3	6	4	4	5	5	0	4	6	6	6	0	6	3	4	4	0	0	0	4	2	8	8	8	5	10	5	1
	Total	245	3	8	8	8	6	8	8	8	8	8	9	8	9	9	9	9	9	2	3	6	4	4	5	5	0	4	6	6	6	0	6	3	4	4	0	0	0	4	2	8	8	8	5	10	5	1

Abbildung 3: Projektablaufplan



6.4 Ressourceneinteilung

6.4.1 Übersicht Projektdauer

RES / KW	KW37	KW38	KW39	KW40	KW41	KW42
MAU	F	F			G	

Tabelle 9: Übersicht Projektdauer Ressourcen

Verwendete Abkürzungen

KW	Kalenderwoche
MAU	Studer Marco
F	Ferien
G	Geschäftliche Abwesenheit
U	Unfall/Krank

6.4.2 Übersicht pro Woche

Member/Tag	MO	DI	MI	DO	FR	SA/SO/FT
SM	V	V	S+V	V	V	V

Tabelle 10: Übersicht Woche Ressourcen

Verwendete Abkürzungen

V	Verfügbar (gem. definiertem Arbeitspensum)
S	Schule
+	Kombinierte Verfügbarkeit
MO-SO	Montag bis Sonntag
FT	Feiertag

Hinweis: Dieses Projekt findet im Rahmen einer Modularbeit statt, dass keine reservierten Blockzeiten nebst der ordentlichen Unterrichtszeit vorsehen. Aus diesem Grund ist ein Ressourcenplan unabdingbar.



6.5 Kommunikation

Der Kommunikation wird mittels einer Tabelle abgebildet. Es wird festgehalten wie vorzugehen und was zu besprechen ist. In der folgenden Tabelle sieht man, welche Kommunikationskanäle benutzt wurden und was bei den Meetings oder Berichten gemacht wurde.

Bezeichnung	Teilnehmer	Ziele / Inhalte	Wiederholung
Kick-off Meeting Microsoft Teams	Squad Member Tester Auftraggeber	<ul style="list-style-type: none">- Ausgangssituation des Projektes- Projektziele- Projektorganisation und -rollen- Gezielte Hervorhebung von Erwartungen, Befürchtungen, Hoffnungen und Wünsche- Vorstellung der projektinternen Kommunikationswege und Verteiler	Einmalig
Projektvergleich Microsoft Teams	Squad Member Auftraggeber	<ul style="list-style-type: none">- Ist- und Sollzustand abgleichen mit dem Auftraggeber.- Probleme aufzeigen, wenn vorhanden.	Alle zwei Wochen
Testen Microsoft Teams	Squad Member Tester	<ul style="list-style-type: none">- Funktionen mit den Testern prüfen und erklären.	1x pro Woche
Statusbericht E-Mail	Squad Member	<ul style="list-style-type: none">- Kommunikation an den Projektausschuss der strategisch getroffenen Entscheidung, welche das Projekt betreffen- Bericht über den aktuellen Fortschritt in der Abarbeitung der Arbeitspakete- Erhebung laufender Aktivitäten und Prüfung offener Arbeitspakete- Erkennen kritischer Planabweichungen und Evaluation von Ideen zur Behebung- Ausblick und Klärung nächster Schritte	1x pro Woche
Projektübergabe E-Mail	Squad Member	<ul style="list-style-type: none">- Übergabe des Projekts an den Auftraggeber	Einmalig am 25.10.2021
Vorzeigetermin 1 Microsoft Teams	Squad Member Experte	<ul style="list-style-type: none">- Zielscheibe mit konkreten Endergebnissen und Erfolgskriterien- Projektstruktur- sowie Projektablaufplanung inkl. Meilensteine- Aufzeigen allfälliger aufgetretenen Probleme	23.09.2021
Vorzeigetermin 2 Microsoft Teams	Squad Member Experte	<ul style="list-style-type: none">- Standortbestimmung in Bezug auf die Ziele- Aufbau und Struktur der Dokumentation	04.10.2021

Tabelle 11: Kommunikationsplan



7 Realisierung Projekt «TRESMENT»

Die Realisierung der Anwendung beruht auf den Anforderungen des Pflichtenheftes. Die Art der Anwendung und auch die Programmiersprache wurden im Pflichtenheft offengelassen.

Im Pflichtenheft wurde definiert, dass das Endprodukt eine Anwendung mit einer grafischen Oberfläche ergeben muss. Durch diesen Umstand wird im Grobdesign entschieden, ob das Endprodukt eine Webanwendung oder eine Desktopanwendung werden soll. Um die Entscheidung zu fällen, wird für beide Anwendungsvarianten ein Grobbeschrieb erstellt, welcher die Anwendungen beschreibt und die Vor- und Nachteile aufzeigt. Anschliessend wird eine Nutzwertanalyse zu erstellen, damit die optimale Lösung herausgefiltert werden kann.

7.1 Grobbeschrieb Varianten

7.1.1 Webanwendung

Eine Webanwendung hat den Vorteil, dass Anpassungen auf globaler Ebene gemacht werden können, ohne dass der Benutzer etwas Neues installieren muss oder seine aktuelle Software upgraden muss. Das Wissensportfolio, welches benötigt wird, um eine sichere und intuitive Webanwendung zu erstellen ist riesig und der Zeitaufwand, um so eine Anwendung zu erstellen ist hoch und würde den Rahmen der vorgegebenen Zeit sprengen.



Abbildung 4 : Programmiersprache HTML & PHP

Das Wissensportfolio, welches für eine Webanwendung benötigt wird, beinhaltet im Grunde folgende Sprachen:

- CSS
- HTML / PHP
- JavaScript

Dieses Wissen muss zuerst aufgebaut werden und ist in grössere Projekte über mehrere Personen zu verteilen.

Die Vorteile einer Webanwendung sind vielseitig und kommen heutzutage immer mehr zum Tragen wie

- der Zugriff, welcher überall gewährleistet werden kann,
- der Vorteil, dass nichts installiert werden muss,
- die Entwicklungs- und Wartungskosten im Verhältnis gering sind, wenn die Architektur der Anwendung einfach und simpel gehalten wurde.

Die Nachteile einer Webanwendung sind jedoch genau gleich persistent und dementsprechend wichtig, wobei der grösste Nachteil ist,

- dass immer eine funktionierende Internetverbindung benötigt wird, um Zugriff auf die Anwendung zu haben.
- Zudem ist das Risiko eines Cyberangriffes auf eine Webanwendung deutlich höher, da die Sicherheitslücken des Internet gnadenlos ausgenutzt werden können.

Fazit

Wenn eine Flexible Anwendung gewünscht ist, auf die Anwender von überall her zugreifen können, ist eine Webanwendung der richtige Weg.



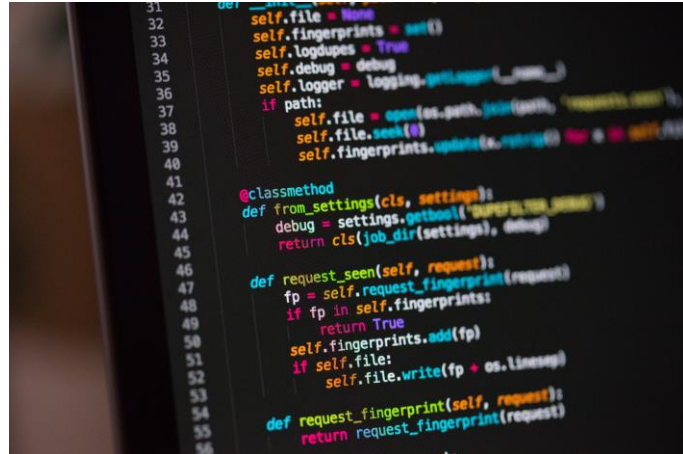
7.1.2 Desktopanwendung

Eine Desktopanwendung hat die Vorteile, welche eine Webanwendung als Nachteile hat. Grundsätzlich ist eine Desktopanwendung nicht auf einen Internetzugang angewiesen. Ausgenommen sind Anwendungen, welche auf eine API oder Internetdienste zugreifen müssen, und somit eine Kommunikation ohne Internet undenkbar ist.

Das Wissensportfolio, welches für eine Desktopanwendung benötigt wird, im Beispiel von der Programmiersprache Python benötigt ausser der Grundsprache Python keine weiteren Sprachen.

Durch diesen Umstand muss nicht das Wissen von verschiedenen Sprachen aufgebaut werden.

In grösseren Projekten ist dies meist so, dass das Projekt in Hintergrundprozessen / Oberflächen aufgeteilt wird.





7.2 Nutzwertanalyse

Anhand der Nutzwertanalyse wurden die Varianten einander gegenübergestellt und anschliessend bewertet.

7.2.1 Kriterien

Um die Nutzwertanalyse durchzuführen, wurde mithilfe einer Ideensammlung relevante Kriterien zusammengetragen. Die Kriterien wurden kritisch betrachtet und die Zehn (10) wichtigsten wurden ausgesucht und beschrieben.

Kriterien Pool				
Erweiterbar	Einfach	Modern	Wissensportfolio	Performance
Sicher	Skalierbar	Kosten	Aufwand	Lerneffekt
Ressourcen	Modular	Flexibel	Abhängigkeiten	Hardware-Ressourcen

Tabelle 12: Kriterienpool für die Nutzwertanalyse

Relevante Kriterien	Beschreibung des Kriteriums
Modern	Modernes Design der Applikation. Kein 80iger Jahre Design;
Sicherheit	Die Sicherheit der übertragenen Daten zwischen Anwendung und Benutzer, zwischen Anwendung und API;
Kosten	Die Kosten, um die Anwendung zu entwickeln, warten und unterhalten;
Aufwand	Der Aufwand (Stunden, Tage) um die Anwendung zu entwickeln;
Wissensportfolio	Das Wissensportfolio, welches vorhanden sein muss, um die Anwendung zu entwickeln;
Hardware-Ressourcen	Hardware-Ressourcen welche für die Applikation benötigt werden;
Lerneffekt	Der Lerneffekt welcher bei der Entwicklung entstehen soll;
Performance	Es dürfen keine langen Wartezeiten in der Anwendung entstehen;
Modular	Die Anwendung muss im Stande sein, dass neuen Module eingebunden werden können;
Abhängigkeiten	Abhängigkeiten von Browser oder Betriebssystem;

Tabelle 13: Priorisierte Kriterien aus dem Kriterienpool



7.2.2 Präferenzmatrix

Die Präferenzmatrix wurde gut durchdacht und lange analysiert, die Kriterien wurden einander gegenübergestellt. Das Ergebnis der gewerteten Kriterien widerspiegelt die zwei Varianten und deren Stärken und Schwächen. Mit den gewerteten Kriterien geht es weiter zur Nutzwertanalyse.

	MOD	SICH	KOS	AUF	WIS	HWR	LER	PER	MODL	ABH	
Modern (MOD)		SIC	MOD	AUF	WIS	MOD	LER	PER	MODL	ABH	
Sicherheit (SICH)			SIC	SIC	WIS	SIC	SIC	SIC	SIC	SIC	
Kosten (KOS)				AUF	WIS	KOS	KOS	PER	KOS	KOS	
Aufwand (AUF)					WIS	HWR	AUF	PER	MODL	AUF	
Wissensportfolio (WIS)						WIS	LER	PER	WIS	ABH	
Hardware-Ressourcen (HWR)							HWR	PER	HWR	ABH	
Lerneffekt (LER)								PER	LER	ABH	
Performance (PER)									PER	PER	
Modular (MODL)										ABH	
Abhängigkeiten (ABH)											
Nennungen	3	8	4	3	6	3	3	8	2	5	45
Prozentualer Anteil	7	18	9	7	13	7	7	18	4	11	100%

Tabelle 14: Präferenzmatrix

Die Punkte modern und modular sind nach langer Überlegung und den Anforderungen aus dem Pflichtenheft entsprechend nicht so schwer zu gewichten, wie die Sicherheit und die Performance. Aus diesem Grunde haben diese zwei Kriterien die niedrigste Punktezahl.



7.2.3 Nutzwertanalyse

Um ein aussagekräftiger Nutzwert zu erhalten, wurden die Kriterien entsprechend bewertet. Wie in der nachstehenden Analyse zu sehen ist, liegt der Vorschlag «Desktopanwendung» (vgl. [Kapitel 7.1.2](#)) weit vor der «Webanwendung» (vgl. [Kapitel 7.1.1](#)).

Die Teilnutzenvergabe wurde wie folgt gemacht:

1. Geringster Nutzen
2. Höchster Nutzen

	Gewichtung	Webanwendung		Desktopanwendung	
		Teilnutzen	Gesamtnutzen	Teilnutzen	Gesamtnutzen
Modern	7	2	14	1	7
Sicherheit	18	1	18	2	36
Kosten	9	2	18	1	9
Aufwand	7	1	7	2	14
Wissensportfolio	13	1	13	2	26
Hardware-Ressourcen	7	1	7	2	14
Lerneffekt	7	2	14	1	7
Performance	18	1	18	2	36
Modular	4	2	8	1	4
Abhängigkeiten	11	1	11	2	22
	100		128		175

Tabelle 15: Nutzwertanalyse

In der Tabelle 15 wird deutlich, dass für dieses Projekt und deren Anforderungen eine Desktopanwendung deutlich besser geeignet ist als eine Webanwendung. Um dies jedoch repräsentativ darzustellen, wurden noch 3 Sensitivitätsanalysen gemacht, um einen Vergleich zu machen und so die auf dieses Projekt gewinnbringendste Variante auszuwählen.



7.2.4 Sensitivitätsanalyse

Die Sensitivitätsanalyse wurde dreimal durchgeführt, im ersten Durchgang sind die Kriterien Performance, Kosten und Aufwand wie folgt verändert worden:

- Performance 18 Punkte abgezogen (von 18 auf 0)
- Kosten 9 Punkte hinzugefügt (von 9 auf 18)
- Aufwand 9 Punkte hinzugefügt (von 7 auf 16)
- Intensität 10 Punkte hinzugefügt (von 13 auf 23)

Hierbei wurde darauf geachtet, dass die Kosten gegenüber der Performance verglichen werden.

	Gewichtung	Webanwendung		Desktopanwendung	
		Teilnutzen	Gesamtnutzen	Teilnutzen	Gesamtnutzen
Modern	7	2	14	1	7
Sicherheit	18	1	18	2	36
Kosten	18	2	36	1	18
Aufwand	16	1	16	2	32
Wissensportfolio	13	1	13	2	26
Hardware-Ressourcen	7	1	7	2	14
Lerneffekt	7	2	14	1	7
Performance	0	1	0	2	0
Modular	4	2	8	1	4
Abhängigkeiten	11	1	11	2	22
	100		137		166

Tabelle 16: Erste Sensitivitätsanalyse

Ersichtlich wurde, dass die Desktopanwendung weiterhin vor der Webanwendung liegt, auch wenn einer der Grundpfeiler die Performance ohne Gewichtung (0) bewertet wird.



7.2.5 Sensitivitätsanalyse 2

Beim zweiten Durchgang wurden folgende Kriterien angepasst:

- Sicherheit 18 Punkte abgezogen (von 18 auf 0)
- Modern 5 Punkte hinzugefügt (von 7 auf 12)
- Modular 5 Punkte hinzugefügt (von 4 auf 9)
- Hardware-Ressourcen 3 Punkte hinzugefügt (von 7 auf 10)
- Abhängigkeiten 5 Punkte hinzugefügt (von 11 auf 16)

Hierbei wurde der Effekt erzwungen eine grafisch schöne, modulare und Hardware-Ressourcen intensive Anwendung zu erstellen, welche trotzdem den Fokus auf die Abhängigkeiten gelegt wird.

	Gewichtung	Webanwendung		Desktopanwendung	
		Teilnutzen	Gesamtnutzen	Teilnutzen	Gesamtnutzen
Modern	12	2	24	1	12
Sicherheit	0	1	0	2	0
Kosten	18	2	36	1	18
Aufwand	16	1	16	2	32
Wissensportfolio	13	1	13	2	26
Hardware-Ressourcen	10	1	10	2	20
Lerneffekt	7	2	14	1	7
Performance	0	1	0	2	0
Modular	9	2	18	1	9
Abhängigkeiten	16	1	16	2	32
	100		147		156

Tabelle 17: Zweite Sensitivitätsanalyse

Wie man anhand der zweiten Sensitivitätsanalyse entnehmen kann, rückt die Webanwendung in diesem Falle der Desktopanwendung auf. Auch wenn der Punkt Sicherheit nicht komplett unberücksichtigt gelassen werden kann, ist die Desktopanwendung immer noch vorne.



7.2.6 Sensitivitätsanalyse 3

Beim dritten Durchgang wurden folgende Kriterien angepasst:

- Wissensportfolio 10 Punkte abgezogen (von 13 auf 3)
- Aufwand 10 Punkte abgezogen (von 16 auf 6)
- Abhängigkeiten 10 Punkte hinzugefügt (von 16 auf 26)
- Performance 10 Punkte hinzugefügt (von 0 auf 10)

Hierbei wurde nochmals untersucht, was passiert, wenn man das Wissensportfolio und den Aufwand minimiert und die Abhängigkeiten und Performance erhöht, um eine performante und stabile Anwendung zu entwickeln.

	Gewichtung	Webanwendung		Desktopanwendung	
		Teilnutzen	Gesamtnutzen	Teilnutzen	Gesamtnutzen
Modern	12	2	24	1	12
Sicherheit	8	1	8	2	16
Kosten	18	2	36	1	18
Aufwand	6	1	6	2	12
Wissensportfolio	3	1	3	2	6
Hardware-Ressourcen	7	1	7	2	14
Lerneffekt	7	2	14	1	7
Performance	10	1	10	2	20
Modular	9	2	18	1	9
Abhängigkeiten	21	1	21	2	42
	100		147		156

Tabelle 18: Dritte Sensitivitätsanalyse

Der Umstand, dass die Desktopanwendung von der Nutzwertanalyse bis zu der letzten Sensitivitätsanalyse vor der Webanwendung liegt, zeigt klar auf welche der beiden Varianten gewählt werden sollte.

7.2.7 Logische Argumentation

Die Entscheidung eine Desktopanwendung zu entwickeln, wurde von Analyse zu Analyse klarer. Die Verschiebung der Punkte hat die Desktopanwendung immer in ein besseres Licht gerückt, selbst wenn die wichtigsten Punkte simultan unberücksichtigt blieben.



7.3 Entwicklung

Die Entwicklung der Anwendung spaltet sich auf in zwei Bereiche, welche miteinander in Verbindung stehen;

- Backend API
- Frontend

Das Backend muss als erstes fertiggestellt werden, da die Datenverarbeitung und Schnittstelle zu Navision über dies laufen. Ohne diese Schnittstelle, können im Frontend keine Daten visualisiert und angepasst werden.

Vor Beginn der Frontend-Entwicklung wurden einige Designvorschläge gemacht, welche dem Fachexperten und auch den Testpersonen (vgl. [Kapitel 6.5](#) Kommunikation\Kickoff) vorgestellt wurden, um das Design gemeinsam zu besprechen. Das endgültige Design wurde anschliessend im [Kapitel 7.3.4](#) ausgearbeitet und fertiggestellt.

7.3.1 Programmiersprache

In der Welt der Softwareentwicklung gibt es verschiedene Programmiersprachen, die passende auszuwählen ist meistens ein schwieriger und zeitintensiver Prozess, der ein grosses Wissensportfolio voraussetzt, da die verschiedenen Programmiersprachen mit ihren Vor- und Nachteilen verglichen werden müssen. Um die API und das Frontend in derselben Programmiersprache zu entwickeln, wurde die Sprache Python ausgewählt. Python ist im Jahre 2021 die meistbenutzte und meist Gefrage Sprache in der Softwareentwicklung.

Weltweit, Oktober 2021 verglichen mit Oktober 2020:

Rank	Change	Language	Share	Trend
1		Python	29.66 %	-2.1 %
2		Java	17.18 %	+0.8 %
3		JavaScript	8.81 %	+0.4 %
4		C#	7.3 %	+1.1 %
5	↑	C/C++	6.48 %	+0.7 %
6	↓	PHP	5.92 %	+0.1 %
7		R	4.09 %	+0.2 %
8		Objective-C	2.24 %	-1.2 %
9	↑	TypeScript	1.91 %	+0.1 %
10	↑↑	Kotlin	1.9 %	+0.3 %

Abbildung 6: Programmiersprache Ranking

Quelle: <https://pypl.github.io/PYPL.html>

Im Vergleich zu anderen Programmiersprachen, ist Python eine sehr benutzerfreundliche und auch anfängerfreundliche Sprache. Die Syntax, welche Python verwendet ist für die meisten Entwickler verständlich und profiliert sich hierbei gegenüber den anderen Programmiersprachen.



7.3.2 Infrastruktur

Um eine performante API aufzubauen, benötigt es einen Server, welcher jederzeit vom Firmeninternen Netzwerk erreichbar ist. Um die Ressourcen optimal einzusetzen und den Aufwand zu reduzieren hat man sich für einen Linux Server mit einem Debian Betriebssystem entschieden. Ein Linux Betriebssystem ist gegenüber einem Windows Server klar im Vorteil, da der Linux Server keine Oberfläche benötigt. Mit diesem Vorteil, können auch die Ressourcen besser verteilt werden und somit werden auch weniger Ressourcen benötigt. Die Minimalanforderung für den Server ist, 2 Prozessoren, 4 Gigabyte RAM und ~50 Gigabyte Speicher.

7.3.3 Backend API

Die Backend API dient dazu die Daten von Navision abzugleichen und diese zu verwalten, um diese später im GUI abzubilden. Die Anwendung wird mittels Endpunkte auf die API zugreifen, um die Daten zu erhalten, welche Visualisiert werden müssen.

Bevor die API erstellt werden kann, müssen jedoch die Überlegungen gemacht werden, welche Bibliotheken werden benötigt, um das Ziel einer Performanten API zu erreichen.

Nach langer Recherche im Internet und der bereits gesammelten Erfahrung im Bereich API mit Python wurde die folgende Liste erstellt:

Bibliothek «Backend»	Beschreibung der Bibliothek
FastAPI Quelle: https://fastapi.tiangolo.com/	FastAPI ist ein modernes und schnelles Web-Framework, um APIs zu erstellen;
Uvicorn Quelle: https://www.uvicorn.org/	Uvicorn ist eine schnelle und einfache ASGI Server Implementation. Diese wird benötigt, um die API am Laufen zu halten;
SQLAlchemy Quelle: https://www.sqlalchemy.org/	SQLAlchemy ist ein SQL-Werkzeugkoffer;
Zeep Quelle: https://docs.python-zeep.org/en/master/	Zeep ist ein hervorragender SOAP Client, welcher für die Anbindung an Microsoft Navision benötigt wird;
Requests Quelle: https://docs.python-requests.org/en/latest/	Requests ist in Python die erste Auswahl für einfache «http/s» Zugriffe;

Tabella 19: Backend Bibliotheken

**Projekt TRESMENT**

Mit den gewonnenen Erkenntnissen und dem errungenen Wissensstand um das Können der jeweiligen Bibliotheken, wurde eine Projektstruktur erstellt, wie die API aufgebaut werden sollte. In der nachfolgenden Abbildung 4 ist der Aufbau der Projektstruktur abgebildet.

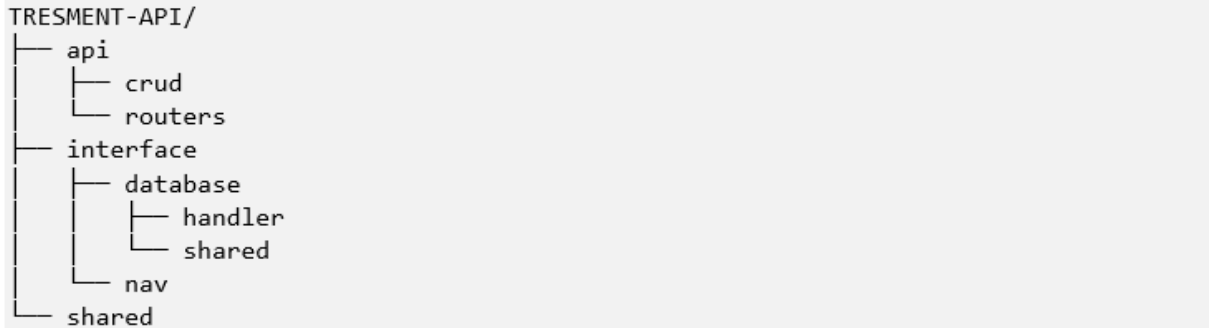


Abbildung 7: API-Projektstruktur

7.3.3.1 FastAPI

FastAPI ist für die Internetbasierte Kommunikation zwischen der Anwendung und der API zuständig. Eine API mit FastAPI zu bauen ist in wenigen Schritten erledigt und kann schnell genutzt und erweitert werden. Nachfolgend wird die Struktur der FastAPI gezeigt. Die Erklärung wird im Strukturbaum von oben nach unten beschrieben.



Abbildung 8: FastAPI Projektstruktur

In dem File **main.py** wird die API erstellt und mit den Routen „routers“ verknüpft. Um die Sicherheit zu gewährleisten ist die Einbindung der **dependencies.py** notwendig, da in dieser die Benutzerauthentifizierung definiert ist. In dem Unterordner „api“ finden sich zwei weitere Abspaltungen;

- crud
 - o Wird von den verschiedenen Routen „routers“ aufgerufen, um die Abfragen in der Datenbank korrekt zu steuern
- Routers
 - o Definiert die Endpunkte, welche von der Anwendung angesprochen werden und liefert die selektieren Daten aus den „crud“ Files an die Anfrage zurück.



7.3.3.2 Zeep

Bei einer Schnittstelle mit dem SOAP ² Protokoll, muss die gesamte XML-Struktur mitgegeben werden, welche anschliessend befüllt zum Absender kommt. Die Bibliothek Zeep ermöglicht es eine einfache und schnelle Verbindung aufzubauen, ohne dass die komplette XML-Struktur mitgegeben werden muss.

```
class GetNavData:
    def __init__(self):
        self.nav_user, self.nav_pw = get_nav_user()
        self.client = self.create_session()

    def create_session(self):
        session = Session()
        session.auth = HttpNtlmAuth(str(self.nav_user), str(self.nav_pw))
        return Client(NAV_URL,
                      transport=Transport(session=session))

    def get_employees(self):
        return self.client.service.SOAP_REQUEST_ANFRAGE("", "")
```

Abbildung 9: SOAP Request mit Python und der Bibliothek Zeep

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xml:lang="en">
  <soapenv:Header/>
  <soapenv:Body>
    <tim:CONSTANTE_NAME>
      e
      <!--Zero or more repetitions:-->
      <x50:Type>
        <x50:Parameter1?</x50:Parameter1>
        <x50:Parameter2?</x50:Parameter2>
        <x50:Parameter3?</x50:Parameter3>
      </x50:Type>
      gero
    </tim:_Resource>
  </tim:CONSTANTE_NAME>
</soapenv:Body>
</soapenv:Envelope>
```

Abbildung 10: SOAP Request über XML mit SOAPUI

Die Abbildungen 9 und 10 verdeutlichen den Unterschied zwischen der Handhabung des Netzwerkprotokolls und der Bibliothek zeep.

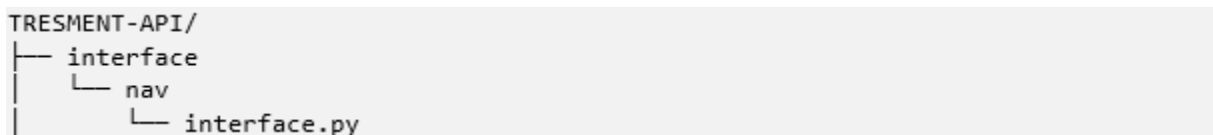


Abbildung 11: Projektstruktur Navision Schnittstelle

Interface.py ist für die gesamten Kommunikation und Anbindung an Navision zuständig. Dies ist nötig, um die korrekten Daten der Benutzer aus dem ERP-System zu laden, um diese später in die Datenbank zu schreiben, so dass die API auf die Daten zugreifen kann und keine Zeit dazwischen verloren geht. Dieser Schritt war unumgänglich, da der Zugriff nicht sonderlich schnell ist über SOAP und somit das Ziel der Performance nicht erreicht hätte werden können.

² SOAP (Simple Object Access Protocol) ist ein Netzwerkprotokoll, mit dem man Daten zwischen verschiedenen Systemen austauschen kann.



7.3.3.3 Datenbank

Um die Daten persistent zu sichern und auch zu verwalten, wird eine Datenbank eingesetzt. SQLAlchemy ist ein „Werkzeugkoffer“, der es dem Entwickler erlaubt einfache und Objektrelationale³Datenbanken aufzubauen. Als Datenbanktreiber wird die im Python Standard mitgelieferte SQLITE3 Bibliothek genutzt.

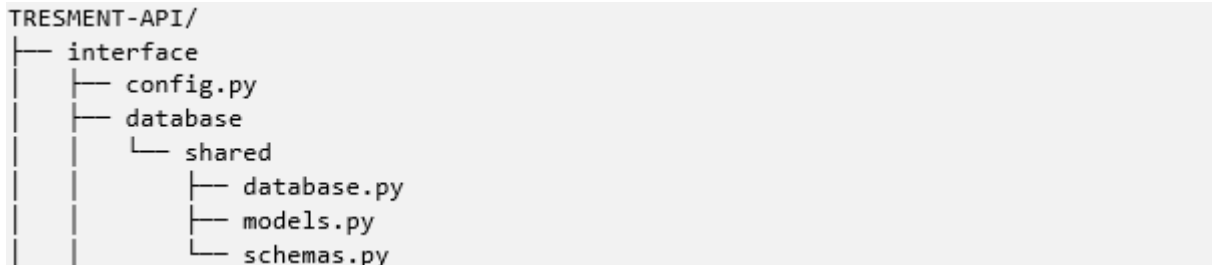


Abbildung 12: Projektstruktur Datenbank

Die Datenbank und ihre Konfigurationen werden in dem File **database.py** definiert und gestartet. Die Tabellen und zugehörigen Attributen werden als **models.py** definiert, für die Erstellung der Tabellen wurden folgende Klassen definiert:

User: Beinhaltet alle Attribute des Mitarbeiters, die Daten werden von der Navision Schnittstelle befüllt.

Absence: In der Tabelle Absence werden alle Absenzen der Mitarbeiter gespeichert. Die Tabelle Absence hat einen Fremdschlüssel (auch Foreign-Key genannt) auf die Tabelle User, um eine Verknüpfung herzustellen.

AbsenceCodes: Die Absenz Codes definieren, welche Absenzen in der Datenbank gespeichert und somit auch im Frontend angezeigt werden.

AbsenceState: Der Absenz Status besteht aus vier verschiedenen Status, welche in der Datenbank abgebildet werden müssen.

Schedule: Die Schedule ist eine Tabelle, welche im Frontend die letzten Aktualisierungen der Tabellen anzeigen kann.

Team: Wie der Name der Tabelle bereits sagt, werden hier alle Teams gespeichert. Zum einen die Default Teams welche von Navision abgeholt werden aber auch die neu erstellten Teams.

TeamUser: Positionstabelle der Tabelle Team mit allen Mitarbeitern pro Team.

AdminNav: Kommunikationsparameter für die Navision Schnittstelle.

AdminUser: Benutzer welche den Zugriff auf die Administratoroberfläche im Frontend haben.

Die Attribute für die Tabellen werden im ERM Diagramm im nächsten Kapitel definiert.

Die Schemas, welche in der API genutzt werden, werden in dem File **schemas.py** definiert und abgebildet, so dass die API auf die Datenbank zugreifen kann.

³ Objektrelationale oder kurz ORM ist eine Technik in der Softwareentwicklung, welche es ermöglicht, Objekte in eine relationale Datenbank abzulegen und diese zu verwalten



7.3.3.4 Entity Relationship Model

Das ERM (Abkürzung von „Entity Relationship Model“) Diagramm zeigt alle Klassen und Attribute auf, welche in der Datenbank angelegt werden müssen.

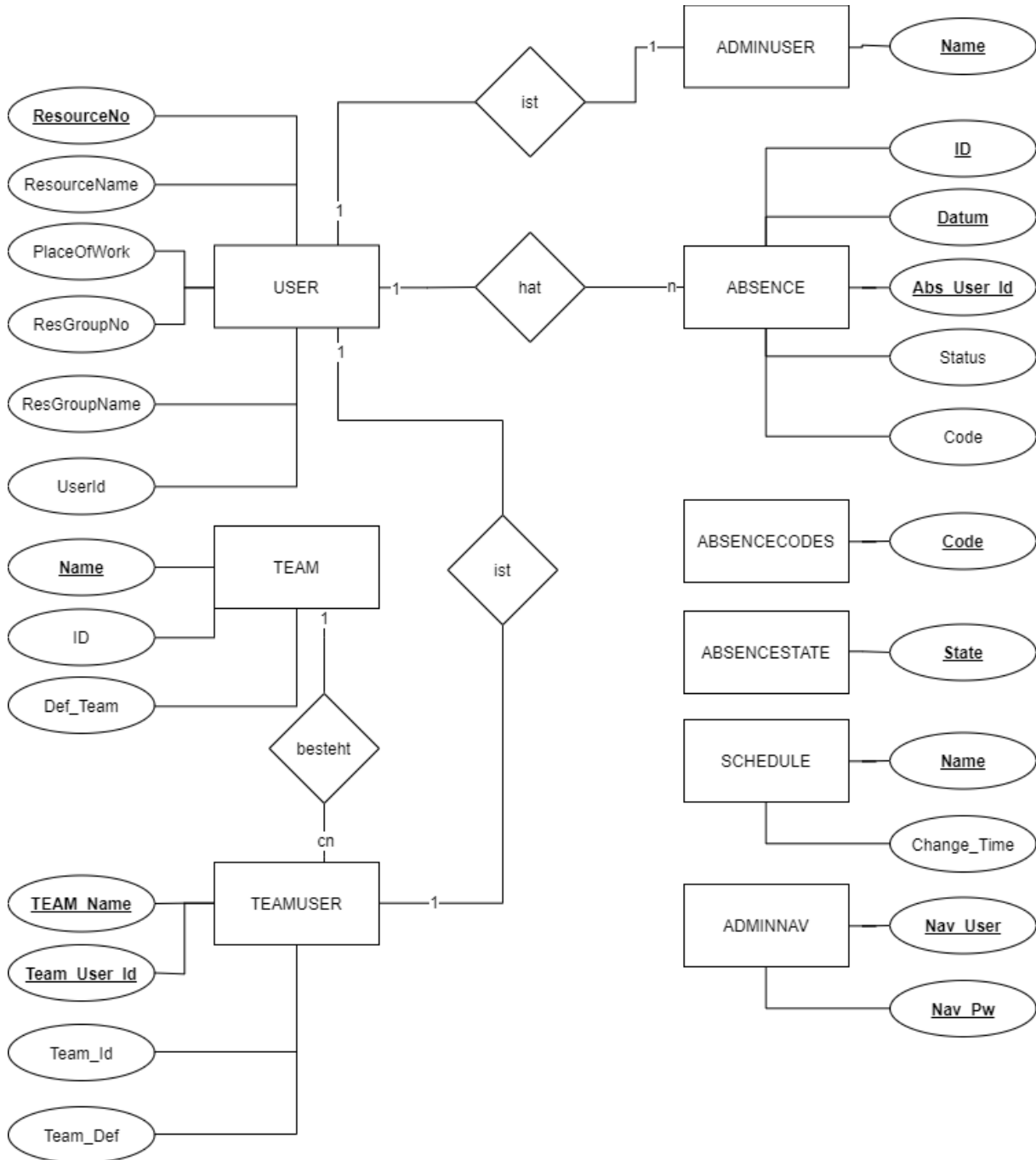


Abbildung 13: ERM Diagramm API-Datenbank



7.3.3.5 Physisches Datenmodell

Die Überführung von dem ERM Diagramm in das physische Datenmodell erleichtert den Aufbau und die Realisierung der Datenbank mit allen Klassen und Attributen.

USER

Relation	Attributname	Datentyp	Eingabe erforderlich	Verweis auf Tabelle
USER	ResourceNo	INT	Ja	
	ResourceName	INT	Ja	
	PlaceOfWork	STRING	Ja	
	ResGroupNo	INT	Ja	
	ResGroupName	STRING	Ja	
	Userld	STRING	Ja	

Einschränkungen und verhalten

Primary Key

ResourceNo

Tabelle 20 Phys. Datenmodell USER

ABSENCE

Relation	Attributname	Datentyp	Eingabe erforderlich	Verweis auf Tabelle
ABSENCE	ID	INT	Ja	
	Datum	DATE	Ja	
	Status	INT	Ja	
	Code	STRING	Ja	
	Abs_User_Id	INT	Ja	USER
	Userld	STRING	Ja	

Einschränkungen und verhalten

Primary Key

ID, Datum Abs_User_Id

Tabelle 21 Phys. Datenmodell ABSENCE

**ABSENCECODES**

Relation	Attributname	Datentyp	Eingabe erforderlich	Verweis auf Tabelle
ABSENCECODE	Code	STRING	Ja	
	HIST_ANUSER	STRING	Ja	
	HIST_ANZEIT	DATE	Ja	
	HIST_AEUSER	STRING	Ja	
	HIST_AEZEIT	DATE	Ja	

Einschränkungen und Verhalten**Primary Key**

Code

Tabelle 22 Phys. Datenmodell ABSENCECODES

ABSENCESTATE

Relation	Attributname	Datentyp	Eingabe erforderlich	Verweis auf Tabelle
ABSENCESTATE	State	INT	Ja	
	HIST_ANUSER	STRING	Ja	
	HIST_ANZEIT	DATE	Ja	
	HIST_AEUSER	STRING	Ja	
	HIST_AEZEIT	DATE	Ja	

Einschränkungen und Verhalten**Primary Key**

State

Tabelle 23 Phys. Datenmodell ABSENCESTATE

**TEAM**

Relation	Attributname	Datentyp	Eingabe erforderlich	Verweis auf Tabelle
TEAM	Name	INT	Ja	
	ID	INT	Ja	
	Def_Team	INT	Ja	
	HIST_ANUSER	STRING	Ja	
	HIST_ANZEIT	DATE	Ja	
	HIST_AEUSER	STRING	Ja	
	HIST_AEZEIT	DATE	Ja	

Einschränkungen und Verhalten**Primary Key**

Name

*Tabelle 24 Phys. Datenmodell TEAM***TEAMUSER**

Relation	Attributname	Datentyp	Eingabe erforderlich	Verweis auf Tabelle
TEAMUSER	Team_Name	INT	Ja	TEAM
	Team_Id	INT	Ja	
	Team_Def	INT	Ja	
	Team_User_Id	INT	Ja	USER
	HIST_ANUSER	STRING	Ja	
	HIST_ANZEIT	DATE	Ja	
	HIST_AEUSER	STRING	Ja	
	HIST_AEZEIT	DATE	Ja	

Einschränkungen und Verhalten**Primary Key**

Team_Name, Team_User_Id

Tabelle 25 Phys. Datenmodell TEAMUSER

**ADMINNAV**

Relation	Attributname	Datentyp	Eingabe erforderlich	Verweis auf Tabelle
ADMINNAV	Nav_User	STRING	Ja	
	Nav_Pw	STRING	Ja	
	HIST_ANUSER	STRING	Ja	
	HIST_ANZEIT	DATE	Ja	
	HIST_AEUSER	STRING	Ja	
	HIST_AEZEIT	DATE	Ja	

Einschränkungen und Verhalten**Primary Key** | Nav_User, Nav_Pw

Tabelle 26 Phys. Datenmodell ADMINNAV

ADMINUSER

Relation	Attributname	Datentyp	Eingabe erforderlich	Verweis auf Tabelle
ADMINUSER	Name	STRING	Ja	
	HIST_ANUSER	STRING	Ja	
	HIST_ANZEIT	DATE	Ja	
	HIST_AEUSER	STRING	Ja	
	HIST_AEZEIT	DATE	Ja	

Einschränkungen und Verhalten**Primary Key** | Nav_User, Nav_Pw

Tabelle 27 Phys. Datenmodell ADMINUSER

SCHEDULE

Relation	Attributname	Datentyp	Eingabe erforderlich	Verweis auf Tabelle
SCHEDULE	Name	STRING	Ja	
	Change_Time	DATE	Ja	

Einschränkungen und Verhalten**Primary Key** | Name

Tabelle 28 Phys. Datenmodell SCHEDULE



7.3.3.6 Threading

Das Kapitel Threading⁴ beschreibt den Ablauf, wie die Daten, welche von Microsoft Navision abgeholt werden, in die Datenbank geschrieben werden.

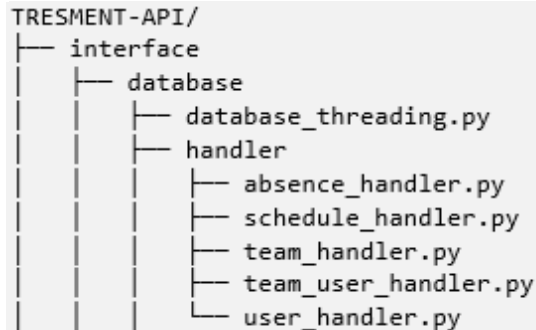


Abbildung 14: Projektstruktur Threading

Damit die Daten, welche im Frontend visualisiert werden, auch immer aktuell sind, werden die Daten in der Datenbank zyklisch überprüft und bei Unstimmigkeiten werden Daten gelöscht oder hinzugefügt.

Hierzu gibt es im **database_threading.py** mehrere Threads, welche die Daten manipulieren:

- Mitarbeiter in die Datenbank hinzufügen
- Absenzen hinzufügen/updaten/löschen
- Default Team und deren Mitarbeiter hinzufügen
- Absenzen welche älter als 1 Jahr sind aus der Datenbank löschen
- Prüfung von den Mitarbeitern, bei Änderungen wird der Mitarbeiter verändert oder gelöscht.

Die jeweiligen Aktionen zu diesen Threads sind in dem „handler“ Ordner abgelegt und pro Thread erstellt.

Durch diese Funktionalität wird die Datenbank immer aktuell gehalten. Diese Funktionalität ermöglicht es der API einen schnellen Zugriff auf die Datenbank zu machen, anstatt die Daten jedes Mal selbst von Navision abzuholen und somit wertvolle Zeit zu verlieren.

⁴ Threading oder eher Multithreading beschreibt eine parallellaufende Funktion, welche nicht von dem Hauptprozess abhängig ist.



7.3.4 Frontend

Aufgrund des Entscheides in der Nutzwertanalyse soll eine Anwendung erstellt werden, welche als Desktopanwendung aufgebaut wird. Mit der Programmiersprache Python und der GUI Bibliothek „Dear PyGui“ kann eine schlanke, performante Anwendung erstellt werden, welche auch in der Weiterentwicklung einfach ist.

Damit bei der Frontend Entwicklung nicht eine Anwendung erstellt wird, welche den Benutzern und dem Auftraggeber nicht zusagen, werden zwei Layout Entwürfe erstellt. Die Entwürfe werden anschliessend besprochen, allfällige Änderungen vorgeschlagen und dann zur Realisierung freigegeben.

Damit die im Pflichtenheft festgehaltenen Muss-Kriterien eingehalten werden, sind diese hier nochmals aufgelistet:

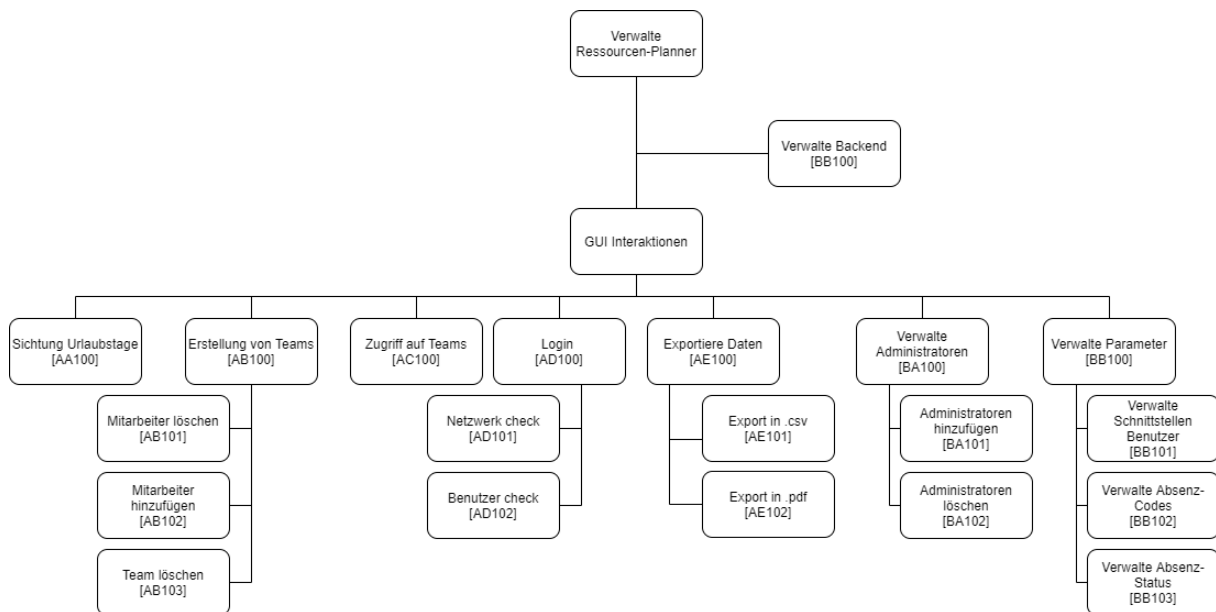


Abbildung 15: Funktionsbaum Pflichtenheft

Der Funktionsbaum zeigt auf, welche Funktionen in der Anwendung vorhanden sein müssen. Die Abkürzungen [AA*] sind im Pflichtenheft unter dem [Kapitel 14.1](#) genauer erläutert.

Bei der Entwicklung der Anwendung gibt es zwei Hauptbestandteile, die visuelle Oberfläche und die Hintergrundprozesse.

Um die Anwendung so intuitiv und flexibel wie möglich zu gestalten, werden diese separat entwickelt und so aufgebaut, dass das Hinzufügen von einem neuen Element einfach und unkompliziert ist.

7.3.4.1 Entwürfe

Es wurden verschiedene Designentwürfe für das Frontend gemacht. Diese Entwürfe und die Absprachen mit den Testern bilden am Schluss die Grundlage für das Design der Anwendung.



Abbildung 16: Entwurf 1 Oberflächen Anwendung

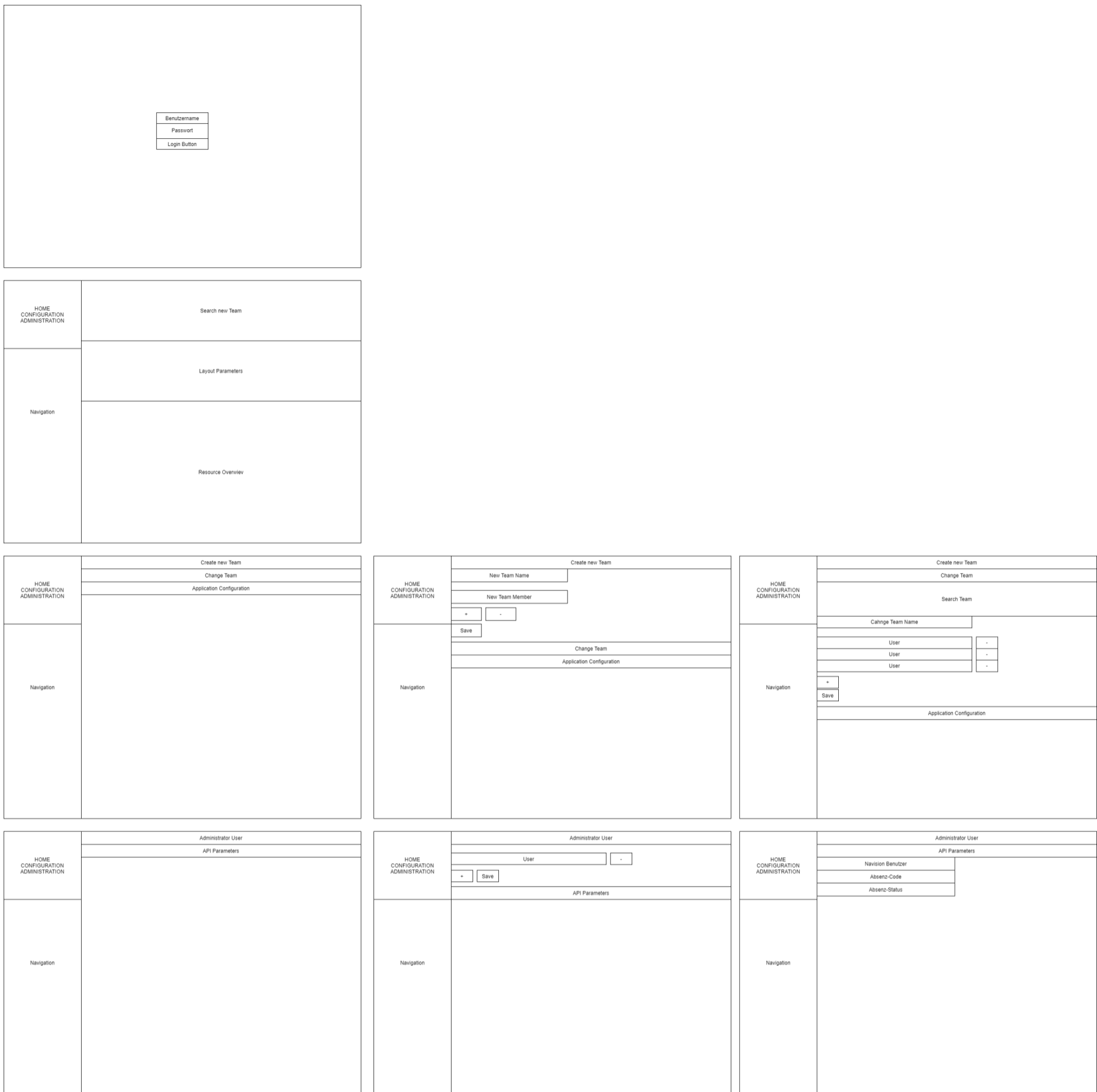


Abbildung 17: Entwurf 2 Oberflächen Anwendung



7.3.4.2 Entwicklung der Anwendung

Auch bei der Frontend Erstellung stellt sich die Frage, welche Bibliotheken dazu genutzt werden um das Ziel einer gutaussehenden, sicheren und performanten Anwendung zu erstellen.

Die Entscheidung „Dear PyGui“ als grafische Oberflächenbibliothek zu nehmen, entstand aus monatelanger Privater Erfahrung mit der Bibliothek. Die restlichen Bibliotheken sind zum Teil Standardbibliotheken oder sind für den Nutzen und die Benutzung die einzige Auswahl.

Bibliothek «Frontend»	Beschreibung der Bibliothek
Dear PyGui Quelle: https://dearpygui.readthedocs.io	Dear PyGui ist ein einfach zu bedienendes, dynamisches, GPU-beschleunigtes, plattformübergreifendes grafisches Benutzeroberflächen-Toolkit (GUI) für Python.
Idap Quelle: https://www.python-ldap.org/en/python-ldap-3.3.0/	Python Idap ermöglicht es einen Zugriff auf den Idap Server zu machen, um Daten zu prüfen oder diese auch zu verändern.
Tkinter Quelle: https://docs.python.org/3/library/tkinter.html	Tkinter ist die Standard Python GUI Bibliothek. Diese wird verwendet, um den Explorer Zugriff zu machen.
Pandas Quelle: https://pandas.pydata.org/	Pandas ist eine schnelle und leistungsstarke Datenanalyse und -manipulations Bibliothek.
Requests Quelle: https://docs.python-requests.org/en/latest/	Requests ist in Python die erste Auswahl für einfache «http/s»-Zugriffe.
SQLAlchemy Quelle: https://www.sqlalchemy.org/	SQLAlchemy ist ein SQL-Werkzeugkoffer

Tabelle 29: Frontend Bibliotheken

Hinweis

Bei der Auswahl der Bibliotheken wurde darauf geachtet, dass für die API und für die Anwendung selbst, möglichst viele Überschneidungen der Bibliotheken stattfinden. Der Vorteil: Durch viele Überschneidungen ist es möglich die Anwendung einfacher weiterzuentwickeln.

**Projekt TRESMENT**

Nachdem das Layout und die Bibliotheken definiert worden sind, kann mit der Entwicklung von der Oberfläche begonnen werden. Für die Projektstruktur wurde der Entscheidung gefällt, die Oberflächen- und Hintergrundentwicklung in verschiedenen Ordnern zu machen, um die Übersicht für Weiterentwicklungen zu garantieren.

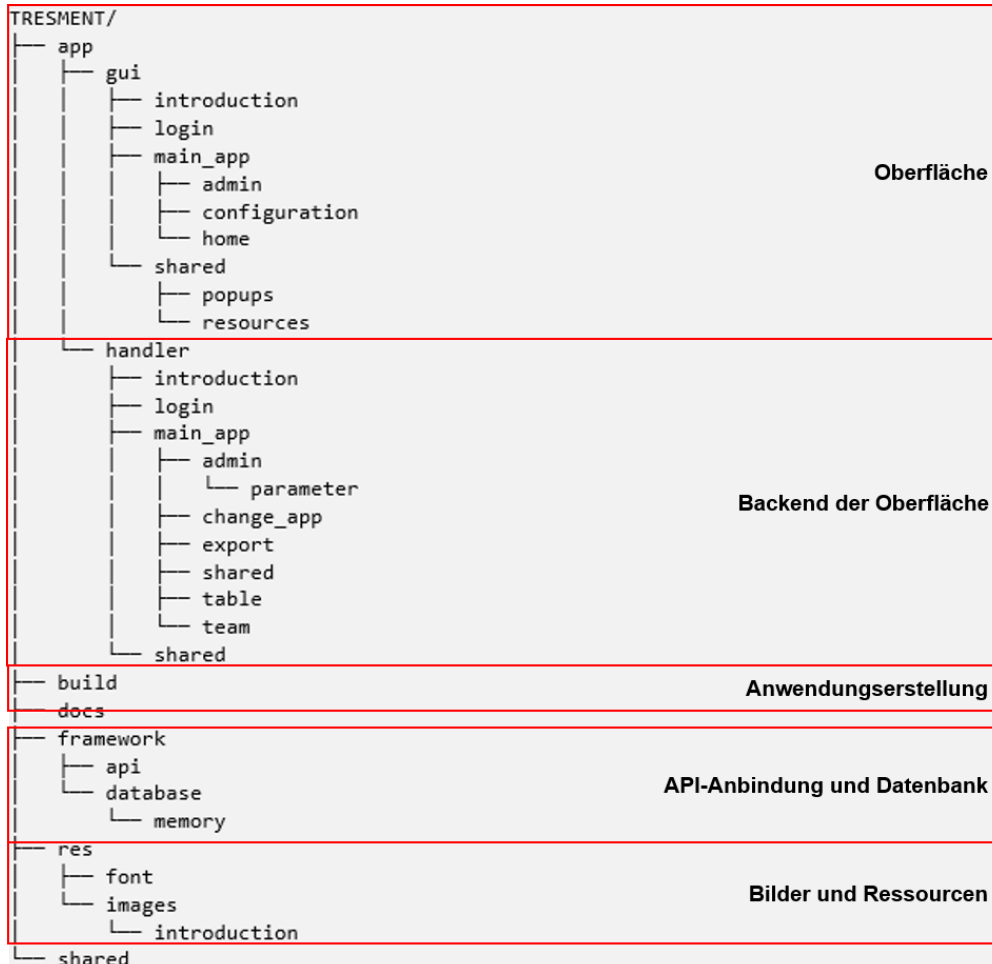


Abbildung 18: Projektstruktur GUI

Die Struktur der Anwendung ist in fünf Bereiche aufgeteilt, welche alle miteinander in Verbindung stehen.

Oberfläche; Die Oberfläche ist die reine Visualisierung der Anwendung ohne viel Logik.

Backend; Das Backend der Oberfläche beinhaltet die Logik und die Funktionalitäten der betätigten Oberflächenelemente und aufbereiten der Daten.

Anwendungserstellung; Damit die Anwendung als .exe ausgeführt werden kann muss diese zuerst aufbereitet werden.

API-Anbindung und Datenbank; Anforderungen an die API sowie Sicherung gewisser Daten in verschiedene Datenbanken.

Bilder und Ressourcen; Bilder und Schriftarten welche für die Anwendung benötigt werde.



7.3.4.2.1 Entwicklung Oberfläche

Damit die Oberflächenentwicklung und auch die Weiterentwicklung übersichtlich bleibt, werden die Hauptsegmente unterteilt, um die Abhängigkeiten zu lösen und das Verständnis aufrecht zu erhalten:

- Login-Fenster
- Einführung
- Hauptfenster
- Konfigurationen
- Administration

Die Segmente besitzen Unterpunkte, welche in dem Kapitel erläutert werden.

Login

Mit dem Login-Fenster startet die Anwendung. Um die Sicherheit zu gewährleisten, kann hier nur weitergefahren werden, wenn der Benutzer im Firmeninternen Netzwerk ist und die Logindaten mit dem LDAP-Server übereinstimmen.

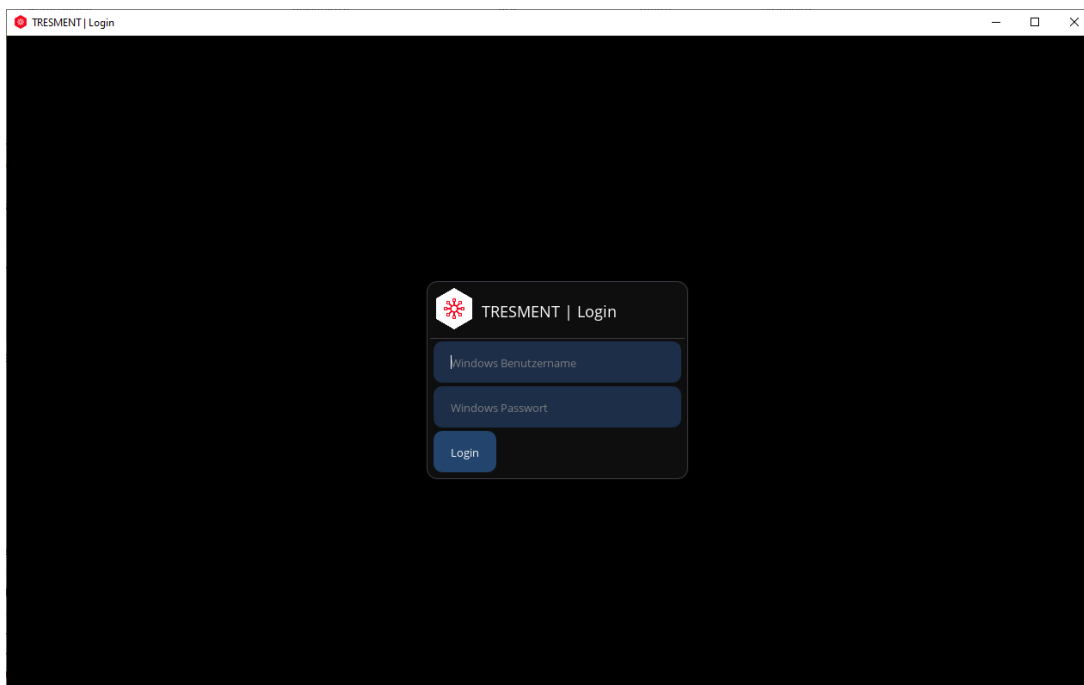


Abbildung 19: Login-Fenster



Einführung

Die Einführung soll dem Benutzer eine grobe Übersicht über die Anwendung geben. Diese besteht aus 8 Seiten und soll als Tutorial dienen, wenn sich jemand dafür entscheidet das Handbuch nicht zu lesen. Der Benutzer hat die Möglichkeit nach vorne oder nach hinten zu blättern und den Dialog wieder zu schliessen. Die Einführung wird bei jedem Start der Anwendung angezeigt, so lange bis die Benutzer aktiv den „Nicht erneut Anzeigen“ Hacken setzt. Diese Änderung wird in der Datenbank gespeichert, so dass beim nächsten Aufstarten der Anwendung der Einführungsdialog nicht mehr erscheint.

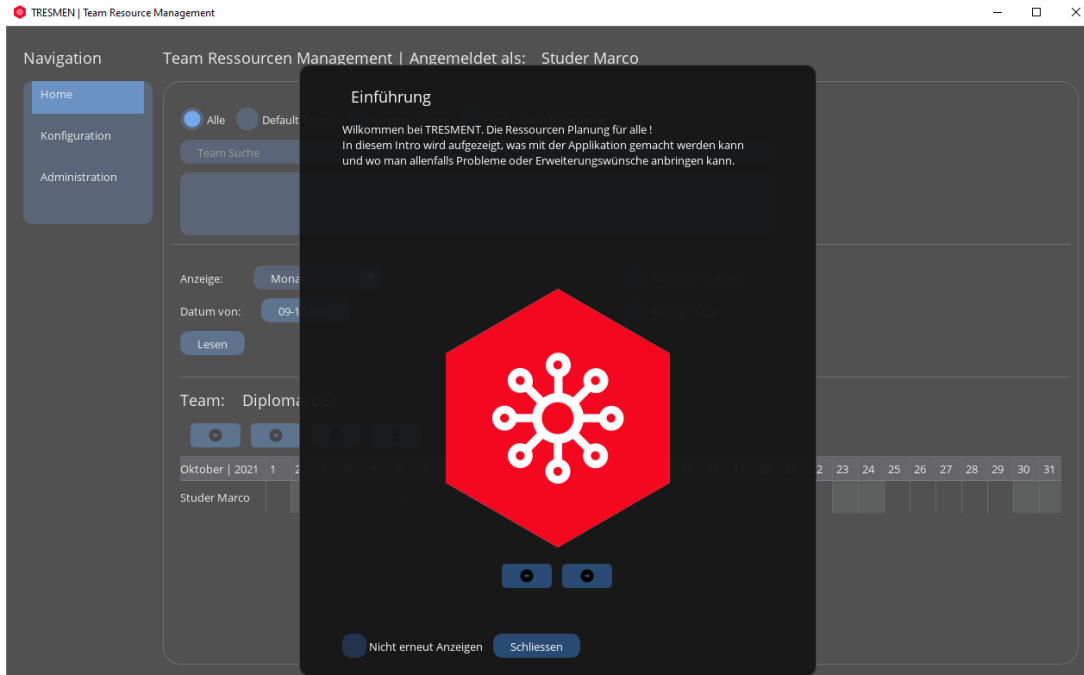


Abbildung 20: Einführung in die Anwendung



Hauptfenster

Nach dem Login erscheint das Hauptfenster, welches folgende Elemente beinhaltet:

- Navigation zu den anderen Bereichen wie Konfiguration und Administration
- Team - / Mitarbeitersuche, um die Ressourcen von einem anderen Team einzusehen
- Filter, um die Anzeige der Ressourcen zu steuern

Die Verfügbaren Filter werden benötigt um folgende Einstellungen vorzunehmen:

- Anzeige; Auswahl zwischen Wochen- / Monats- / Jahresansicht
- Datum; Erweitern / Ändern von der angezeigten Periode
- Mögliche Stunden; Erweitert die Anzeige mit einer Berechneten Summe der verfügbaren Stunden pro Mitarbeiter
- Anzeige Tage; Erweitert die Anzeige mit den Wochentagen

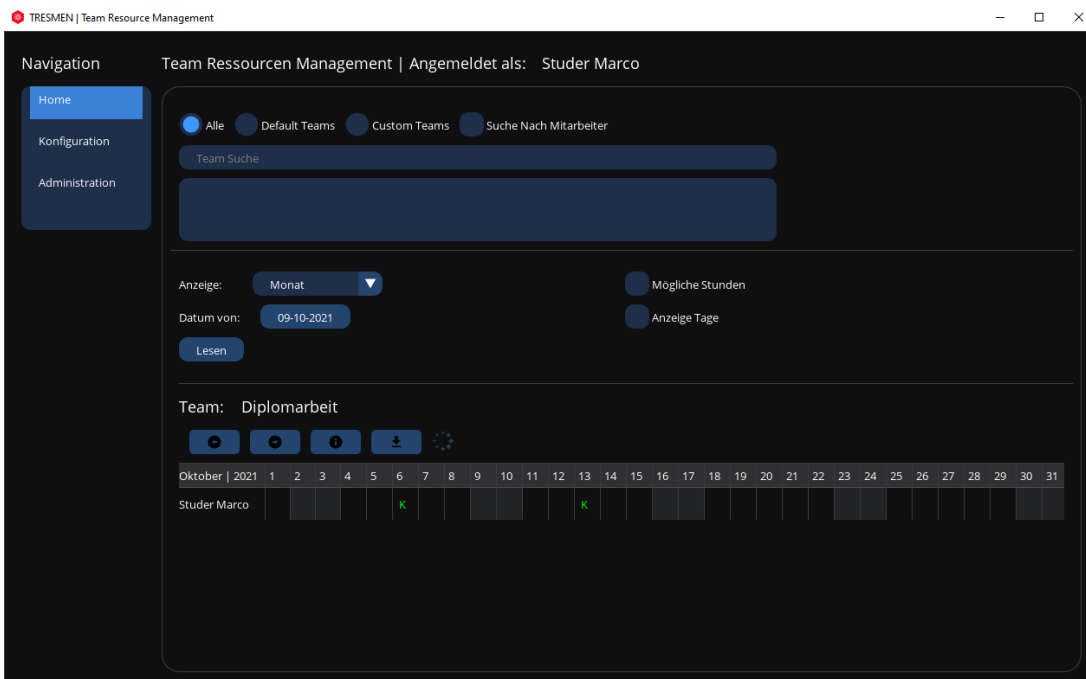


Abbildung 21: Hauptfenster

In der Wochenansicht sieht man die Arbeitstage einer Woche mit den zusätzlichen Informationen von dem Datum und der Kalenderwoche.

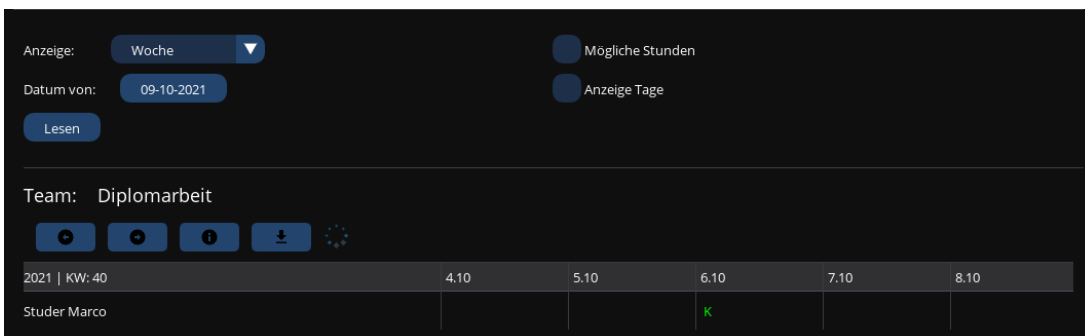


Abbildung 22: Hauptfenster Wochenansicht



Die Monatsansicht bietet dem Benutzer die Möglichkeit die Übersicht über einen gesamten Monat zu haben. Die zusätzlichen Informationen in der Monatsansicht sind der Monat, das Jahr und die Wochenenden. Sie sind farblich markiert.

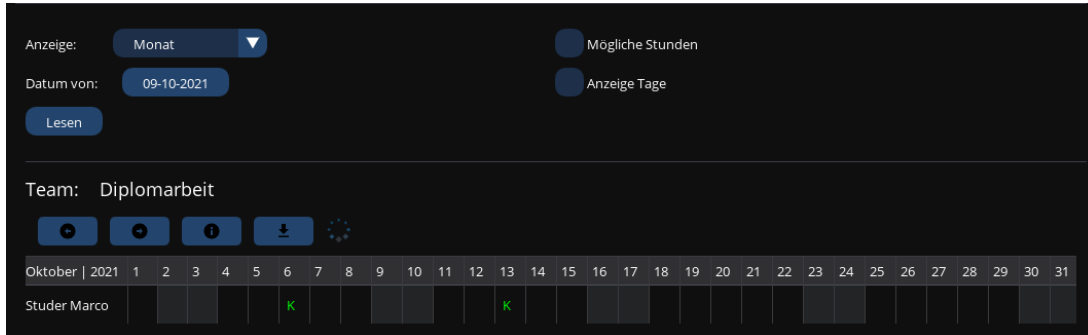


Abbildung 23: Hauptfenster Monatsansicht

Die Jahresansicht ist ein Duplikat der Monatsansicht, nur dass jeder Monat von einem Jahr angezeigt wird.

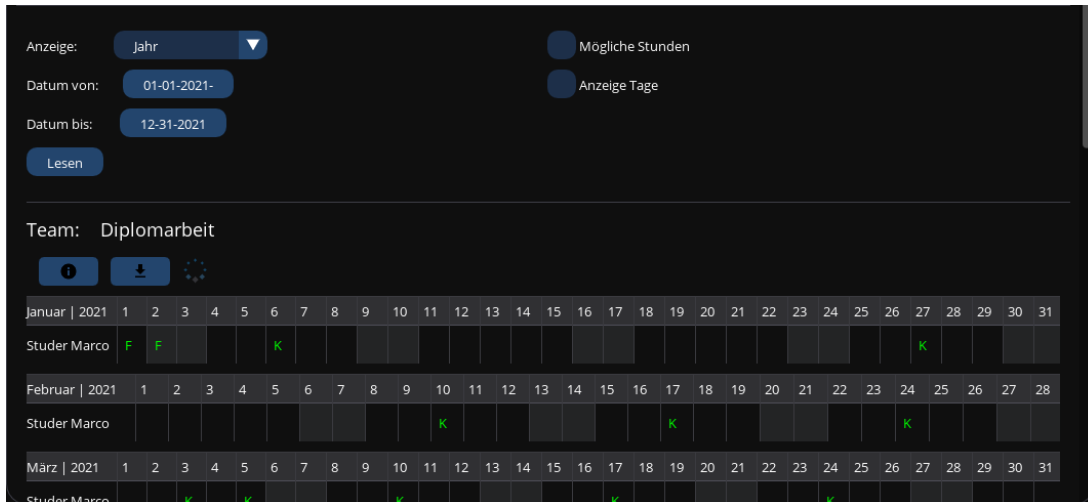


Abbildung 24: Hauptfenster Jahresansicht

Die Filtermöglichkeiten „Mögliche Stunden“ und „Anzeige Tage“ erweitern die Übersicht mit einer weiteren Spalte, um die Wochentage zu visualisieren und die möglichen Stunden für Projekte innerhalb dieser Zeit.

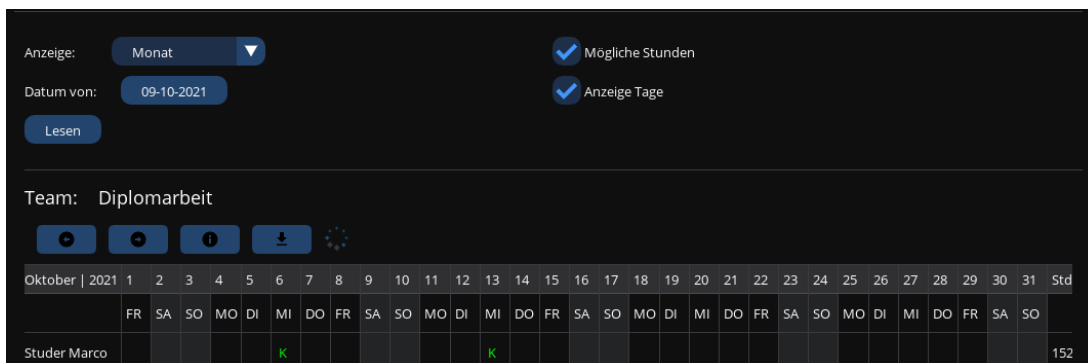


Abbildung 25: Hauptfenster Aktive Filter



Konfiguration

Die Konfigurationsübersicht zeigt die Funktionen, welche jeder Mitarbeiter durchführen kann.

Die einzelnen Untermenüs werden in den nachfolgenden Seiten erklärt und deklariert.

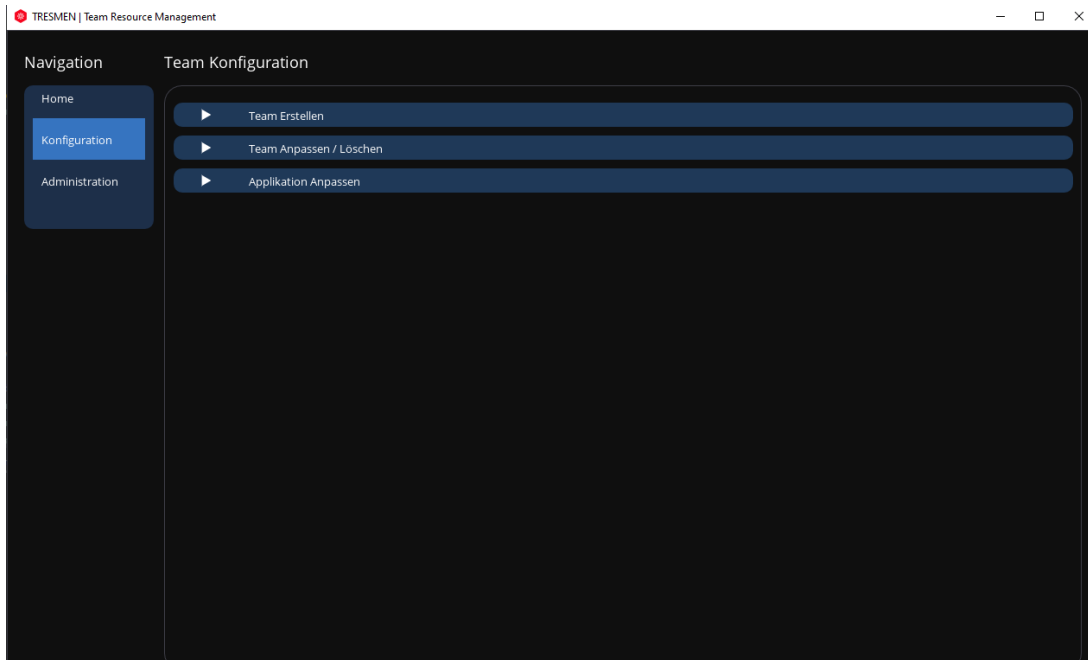


Abbildung 26: Konfigurationsübersicht

Konfiguration; Team Erstellen

Damit im agilen Format gearbeitet werden kann, müssen zuerst Teams erstellt werden. Der Name muss immer eindeutig sein und mindestens ein Team Mitglied beinhalten. Wenn man nicht weiss, wie ein Benutzer mit vollen Namen heisst, wird dies durch die Steuerung der Suche vereinfacht.

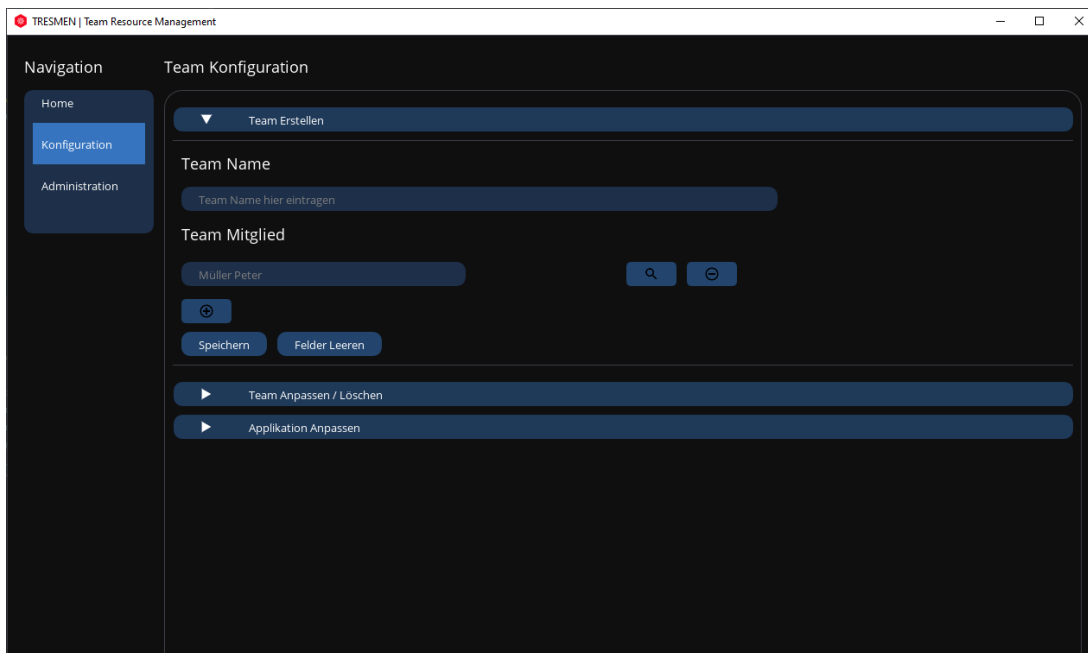


Abbildung 27: Konfiguration Team Erstellen



Konfiguration; Team Anpassen / Löschen

Jedes von einem Mitarbeiter erstellte Team kann verändert oder gar gelöscht werden. Die Standardteams, welche von Navision vorgegeben werden, können hier nicht angepasst werden, sondern müssen im Navision selbst angepasst werden.

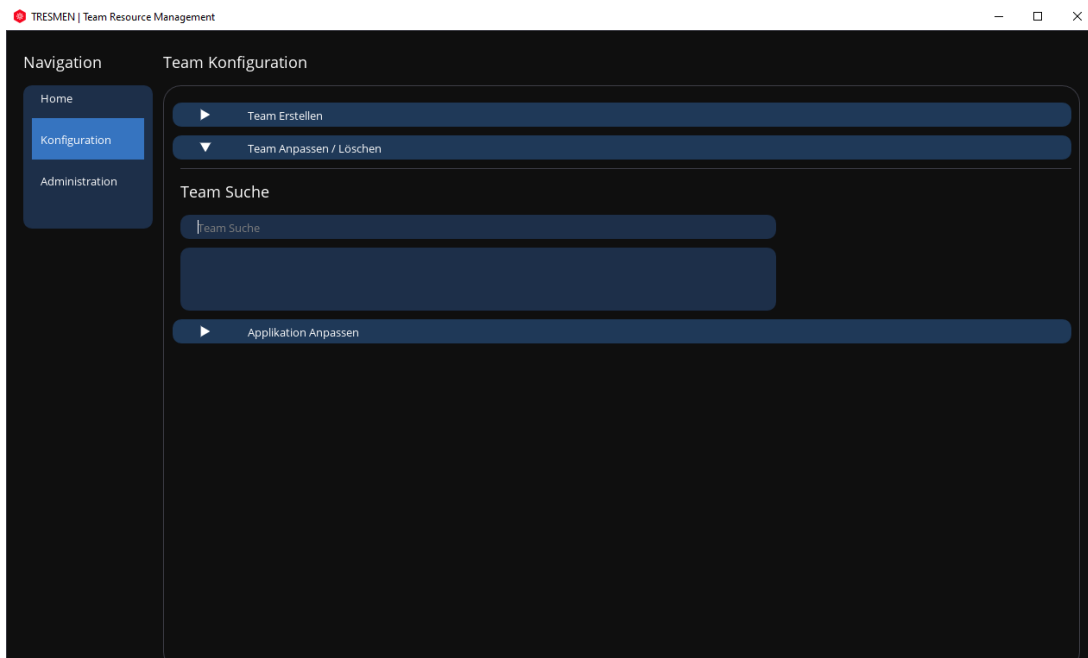


Abbildung 28: Konfiguration Team Anpassen / Löschen

Konfiguration; Anwendung Anpassen

In diesem Untermenü kann jeder Anwender Anpassungen an der Anwendung selbst vorgenommen werden, wenn jemand lieber ein helles Design verwenden möchte, sich nochmals die Einführung anschauen will oder Anpassungen an dem Standardteam oder den Standardfilter vornehmen will. Der Bereich „About“ gibt Auskunft über die Anwendung und bietet einen direkten Link zu den Handbüchern.

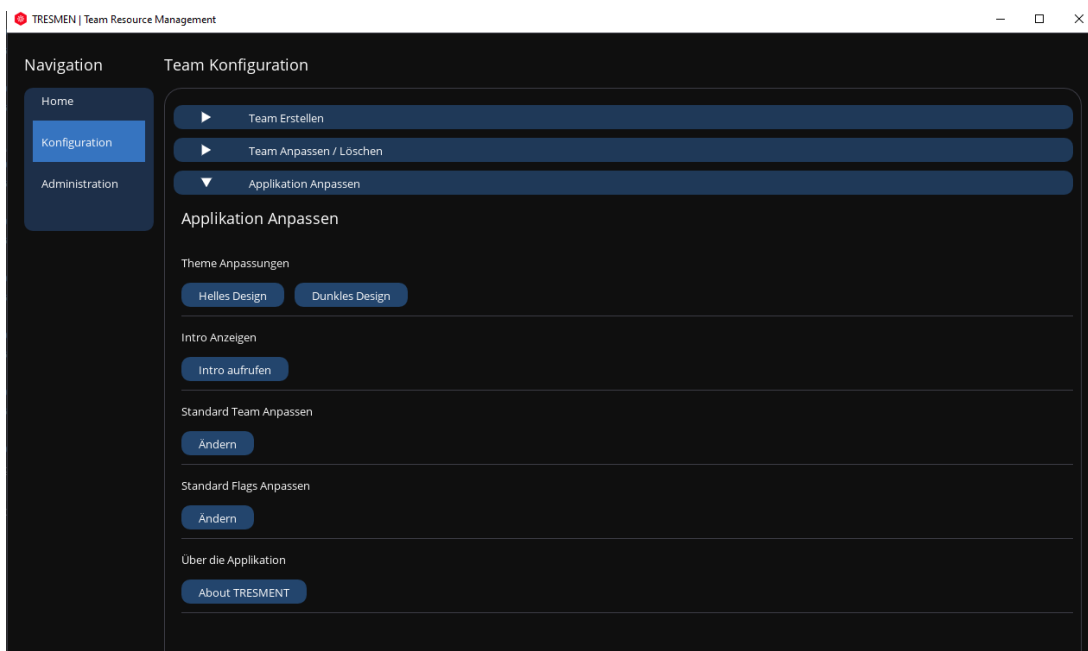


Abbildung 29: Konfiguration der Anwendung



Konfiguration; Anwendung anpassen

Mit dem Button „Helles Design“ wechselt die ganze Anwendung in das ausgewählte Design und wird so in der Datenbank gespeichert, damit die Anwendung beim nächsten Aufstarten in dem zuvor gewählten Design erscheint.

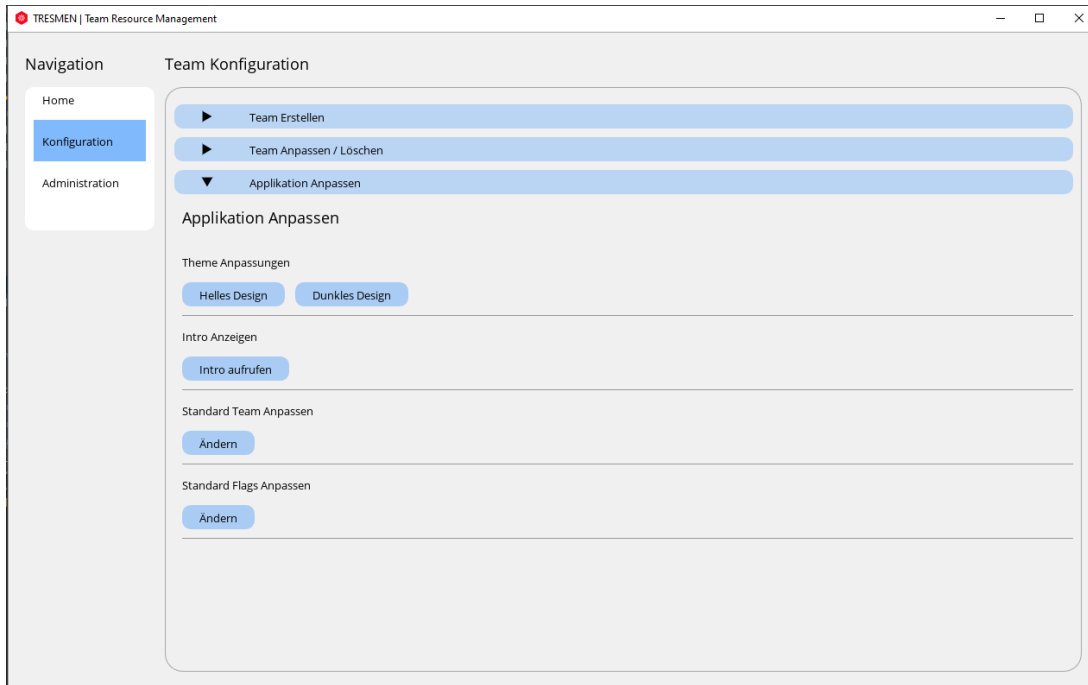


Abbildung 30: Konfiguration Helles Design

Administrator Oberfläche; Admin Benutzer

Wie im Pflichtenheft deklariert gibt es eine Administratoren-Liste. Diese Liste kann von den Entwicklern oder von den erweiterten Administratoren verändert werden.

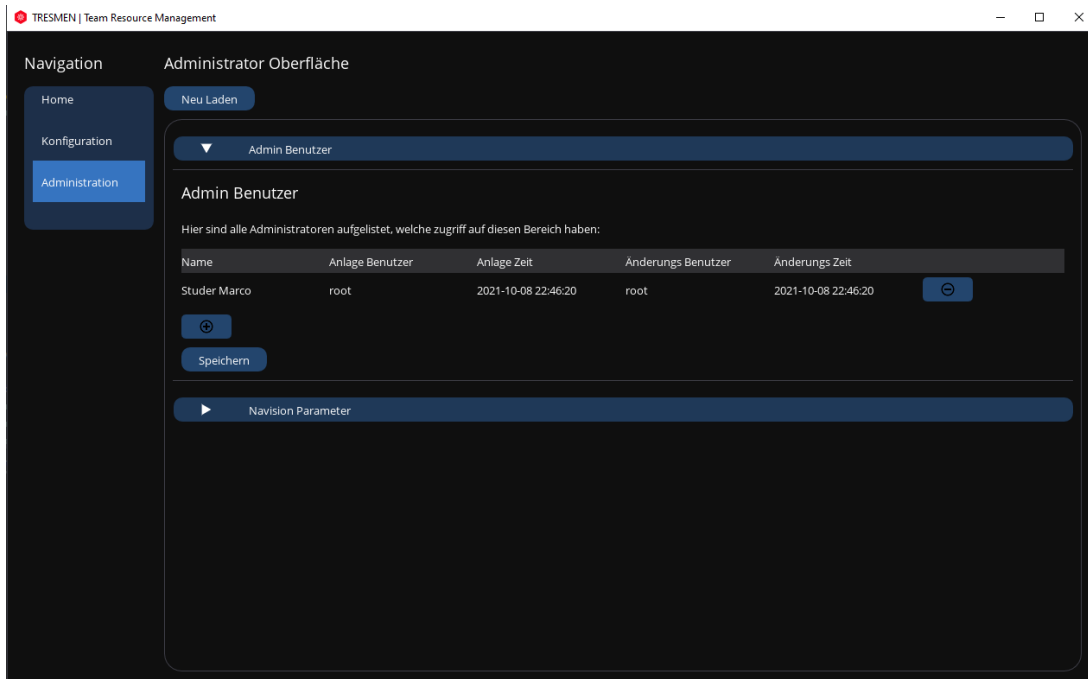


Abbildung 31: Administrator Oberfläche Administrator Benutzer Verwalten



Administrator Oberfläche; Navision Parameter

Aufgrund von sensiblen Inhalten, kann der Abschnitt API-Parameter nicht in die Dokumentation übernommen werden. Die Parameter können jedoch alle von einem Administrator und nur von einem Administrator angepasst und verändert werden.

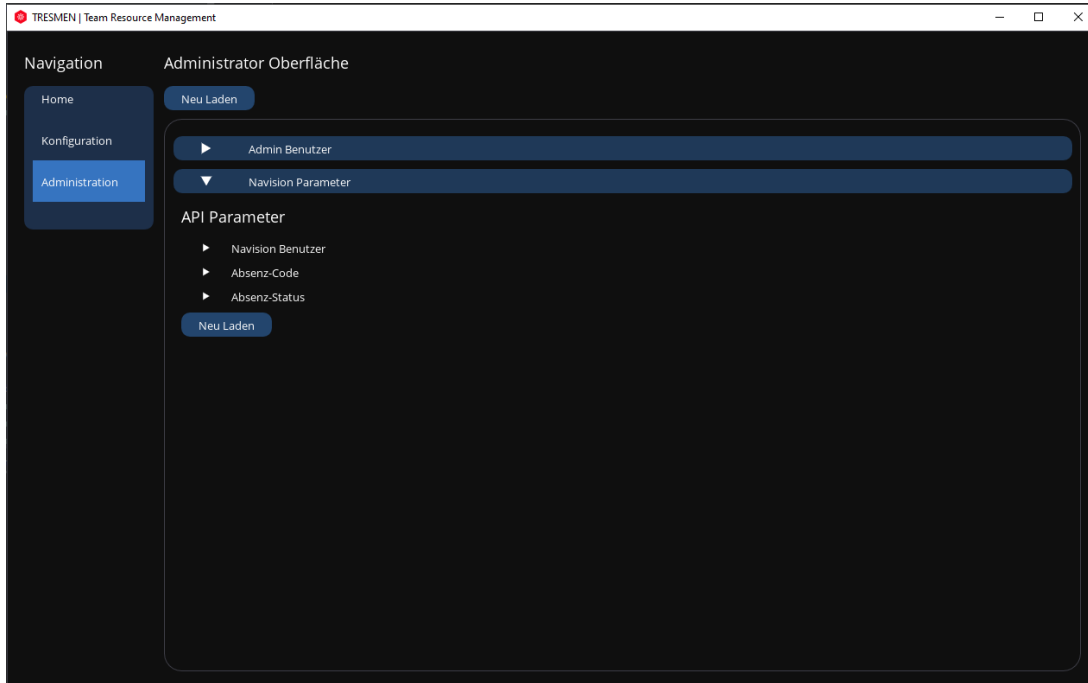


Abbildung 32: Administrator Oberfläche Navision Parameter Verwalten

Diverses;

In die Kategorie „Diverses“ werden Module wie Popupfenster für Fehlermeldungen oder einfache Konfigurations-Popups eingerichtet. Diese werden an einzelnen Orten aufgerufen oder werden mehrfach benutzt (wie im Falle der Fehlermeldungen).

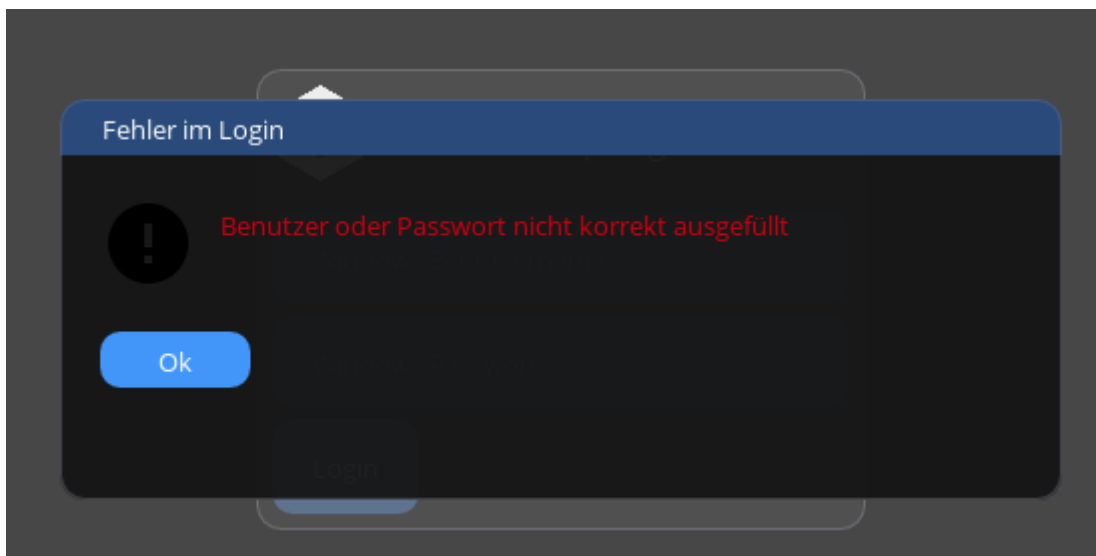


Abbildung 33: Popupfenster mit Fehlermeldung



7.3.4.2.2 Entwicklung Datenbank

Damit die Weiterentwicklung so einfach wie möglich gestaltet wird, wird die Datenbank ebenfalls mit SQLAlchemy erstellt.

Um die Anwendung so performant wie nur möglich zu machen, werden die wichtigsten Daten, nämlich die Absenzen in einer „In Memory“ Datenbank zwischengespeichert und stehen für den Zugriff zur Verfügung. Dies hat einige Vorteile:

- Die Daten sind für niemanden einsehbar, da die Datenbank nur im Memory zwischenspeichert. Sobald die Anwendung geschlossen wird, wird auch diese Datenbank gelöscht und hinterlässt keine Spuren auf dem Computer.

Durch diese Integration müssen die API-Abfragen nicht so häufig gemacht werden, da die relevanten Daten lokal geladen werden können. Somit wird auch der Server und die Internetverbindung entlastet.

- Die zweite Datenbank, welche mit der Anwendung ausgeliefert wird, ist eine interne Datenbank, welche benötigt wird, um Standardparameter abzuspeichern, so dass diese beim nächsten Aufstarten so gesetzt werden, wie sich das der Benutzer wünscht.

In Memory Datenbank

Die „In Memory“ Datenbank besitzt eine Tabelle, welche eine Spiegelung der Tabelle ABSENCE von der API ist.

Im Gegensatz zu der sichtbaren Datenbank hat die In Memory Datenbank einen Thread, welcher alle 5 Minuten die Absenzen von der API mit der eigenen Struktur vergleicht und bei Anpassungen diese übernimmt. Somit ist die In Memory Tabelle ABSENCE immer auf dem neusten Stand.

Interne Datenbank

Die Interne Datenbank der Anwendung beinhaltet 4 Tabellen. Wie man am nachfolgenden ERM-Diagramm einsehen kann, sind die Tabellen kurzgehalten und so deklariert, dass diese auf einfache Weise erweitert werden können.

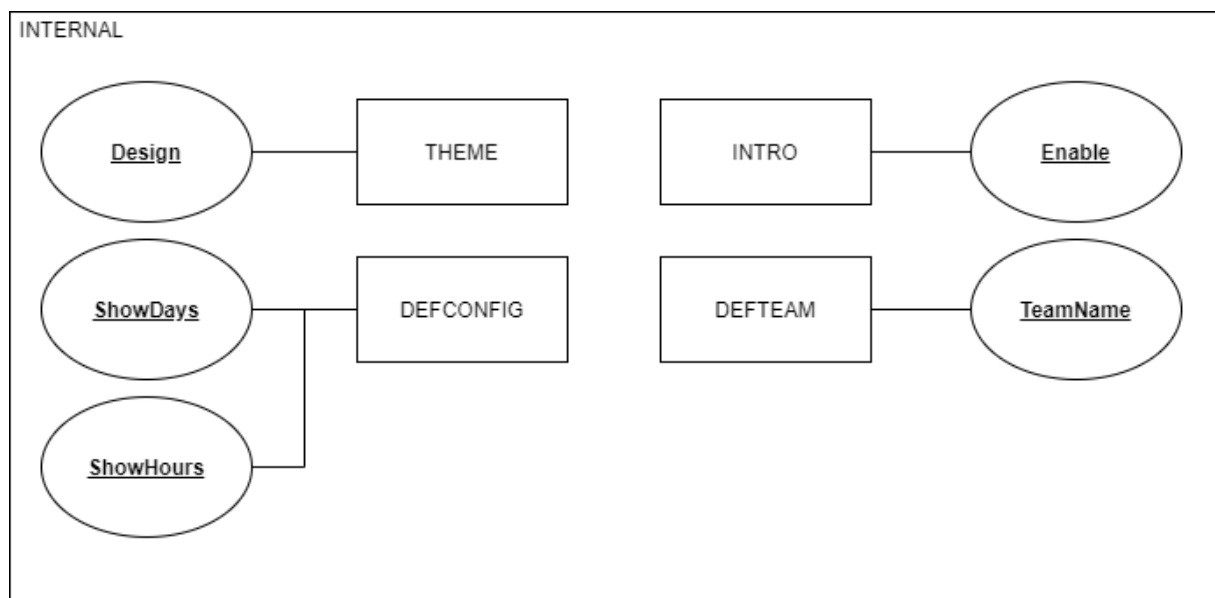


Abbildung 34: ERM Interne Datenbank



7.4 Testing

Aufgrund diverser Projektabwesenheiten und Urlaubstagen, konnte das Testing nicht an fixen Terminen durchgeführt werden. Nach dem Kickoff-Meeting haben die Testpersonen wöchentlich eine neue Version erhalten, mit einer Liste der angepassten / erweiterten Funktionen.

7.4.1 Update 24.09.2021

Neue Funktionen:

- Export;
- Ansicht mehrere Monate;
- Diverse Styling Optimierungen;
- Suche Mitarbeiter in Teams;
- Team Löschen;
- Team Anpassen angepasst (Diverse Felder hinzugefügt);
- Letzter AZE Abgleich;
- Legende der Tabelle;
- Design Wechsel;

Was wird weiter ergänzt:

- Ab dem Mittwoch (29.09.2021) wird das Login umgestellt auf ein LDAPS verifiziertes Login (Windows Benutzer / Windows Passwort);
- Bug in der API mit doppelten Absenz Einträgen wird gefixt;
- Bug Anzeige Woche wird gefixt;
- Hinzufügen PDF-Export;
- Hinzufügen Administrator Bereich;
- Funktionalität für "Verfügbare Stunden" hinzufügen;
- Button via Enter bestätigen (wird erst beim letzten Release hinzugefügt);
- Automatisches schliessen der «Header», wenn ein anderer geöffnet wird (wird erst beim letzten Release hinzugefügt);
- ggf. farbliche Markierung der Wochenenden (wird erst beim letzten Release hinzugefügt);

7.4.2 Update 29.09.2021

Neue Funktionen / Verbesserungen:

- Der Administrator Bereich ist ausgebaut und fertig;
- Bug in der API mit doppelten Absenz Einträgen gefixt;
- Login umgestellt auf ein LDAP verifiziertes Login (Windows Benutzer / Windows Passwort);
- Styling in der Export Tabelle;
- Funktionalität für "Verfügbare Stunden" hinzugefügt;
- Bug Anzeige Woche gefixt;
- Excel Export für alle Varianten;
- Login-Screen Anpassungen;
- Anpassungen, falls der Server nicht erreichbar ist;
- «Case Sensitive» ausgeschaltet;
- Text Anpassungen;
- Icon Anpassungen;



Was wird weiter ergänzt:

- PDF-Export (wenn möglich);
- Einführung Screen;
- Button via Enter bestätigen (wird erst beim letzten Release hinzugefügt);
- Automatisches schliessen der «Headers», wenn ein anderer geöffnet wird (wird erst beim letzten Release hinzugefügt);
- Ggf. Farbliche Markierung der Wochenenden (wird erst beim letzten Release hinzugefügt);

7.4.3 Update 01.10.2021

Neue Funktionen / Verbesserungen:

- Der Team Name kann neu auch angepasst werden;
- Sonderzeichen dürfen in einem Team Namen nicht gespeichert werden;
- Doppelte Einträge in den Listen wurde gefixt;
- Diverse Datum Fehler wurden behoben;
- Das Design wird jetzt gespeichert, wenn jemand auf das «Light design» umsteigt, wird dies auch gespeichert;
- Diverse kleinere Bugfixes;

7.4.4 Update 06.10.2021

Neue Features:

- Team Name ändern;
- Doppelte Einträge gefixt;
- On Enter Callbacks hinzugefügt;
- Keine Sonderzeichen im Team Namen hinzugefügt;
- Thread für Aktualisierung von dem AZE-Abgleich;
- Intro angepasst und eingebaut;
- Update Erinnerung mit automatischem Update hinzugefügt;
- Wochenende Highlight hinzugefügt;
- Styling Popup Meldungen angepasst;
- Upgrade Dear PyGui von Version 0.8.64 auf 1.0.0RC2;
 - GPU Steuerung eingebaut;
 - Automatisches schliessen der «Header», wenn ein anderer geöffnet wird;
 - Redesign Code;
- Suche nach Mitarbeiter eingebaut;
- Export Layout umgebaut;
 - Export nach PDF wird nicht Bestandteil der Arbeit sein;

7.4.5 Update 14.10.2021

- Standardteam setzen
- Standard-Konfigurationen setzen
- Dear PyGui Upgrade auf Stable Version 1.0
- Administrator Bereich mit Header Close erweitert
- Datum angepasst
- Diverse kleine Bugfixes und Performance Verbesserungen
- About Hinzugefügt
- Diverse Bugfixes



7.5 Dokumentation

Damit die Anwendung von jedem Mitarbeiter genutzt werden kann, werden diverse Handbücher verfasst. Die Handbücher werden auf die drei Bereiche Anwender, Administrator und Entwickler ausgerichtet und sind für jeden Mitarbeiter zugänglich.

Die Handbücher werden im internen Confluence hinterlegt, so dass jeder Mitarbeiter jederzeit die Möglichkeit hat in den Dokumentationen nachzuschlagen.

7.5.1 Anwender

Die Anwenderdokumentation beschreibt die Anwendung und was mit dieser möglich ist. Hierbei wird Schritt für Schritt erklärt, was gemacht werden muss, um zum Beispiel ein Team zu erstellen oder dieses zu löschen.

7.5.2 Administrator

Die Administratordokumentation bezieht sich darauf, wie sich jeder Administrator verhalten soll und wer die Hoheit über die Parameter hat. Zudem ist auch beschrieben, wie die Parameter angepasst werden müssen, sollte es Anpassungen geben.

7.5.3 Entwickler

Die Entwicklerdokumentation beinhaltet wie das Projekt weiterentwickelt werden kann. Welche IDE für die Weiterentwicklung am angenehmsten ist und wo die Passwörter zu finden sind. Der Source-Code selbst ist komplett dokumentiert und jede Funktion besitzt eine Funktionsbeschreibung und jedes File eine Einleitung.



7.6 GO – Live

Damit für den Go-Live-Termin alles funktioniert, wurde die Anwendung mehrere Wochen von ausgewählten Personen getestet und es wurden mehrere Updates der Anwendung entwickelt, so dass diese den Bedürfnissen der Anwender entspricht.

Der Go-Live findet am 18.10.2021 um 09:00 statt und wird via E-Mail an die ganze Belegschaft der Firma SSI Schäfer kommuniziert.

Hallo Zusammen !

Nach 5 langen Wochen möchte ich euch gerne meine Diplomarbeit präsentieren.
Einige haben sich bereits davon erfahren und waren auch involviert in der Ausarbeitung.

Was ist TRESMENT:
TRESMENT ist eine eigens entwickelte Software, welches jedem Mitarbeiter die Einsicht in die Abwesenheiten anderer Mitarbeiter oder Teams gewährt.
Damit die Software auch mit einer agilen Arbeitsweise etwas taugt, besteht die Möglichkeit neben den bestehenden Teams, neue Teams zu erstellen und diese anzupassen.
Zudem erspart sich jetzt jeder Teamleiter die Mühe für sein Team eine Ferienplanung zu erstellen, da jetzt jeder Mitarbeiter in einem Team seine Ferien eintragen kann und diese dann als «beantragt» angezeigt werden.
Desweiteren können Projektleiter die Liste des Teams in ein Excel exportieren und diese dann nach ihren eigenen Wünschen noch erweitern.
Damit die Sicherheit gewährleistet ist, ist es nur möglich die Anwendung zu öffnen mit dem Windows Login und wenn man im internen Firmennetzwerk ist.

Der Grundstein:
Vor einigen Monaten ist im Helpdesk das Team WAMAS-Logimat Realisierung gegründet worden, aus Mitgliedern aus verschiedenen Teams.
Um die Planung der Realisierung der Projekte durchzuführen, sind einige Schwierigkeiten aufgetreten, was die Abwesenheiten anbelangt.
Diese Schwierigkeit, war der Grundstein für meine Diplomarbeit um eine Übersicht für jeden Mitarbeiter zu schaffen, wer, wann verfügbar ist.
Dazu kommt noch der Wandel der Firma in die Richtung der agilen Arbeitsweise, wodurch neue Teams gebildet werden, welche Mitglieder aus verschiedenen Abteilungen oder Teams haben.

Die Umsetzung:
Im Juli hat die Pflichtenheftphase begonnen, welche ich im August Unterschrieben meiner Hochschule abgegeben habe und angenommen wurde.
Im September habe ich dann gestartet mit der Realisierung des Projektes, Design mit dem Firmenexperten und den ausgesuchten Testpersonen.
Die Anwendung wurde jede Woche auf die von mir erweiterten Funktionen getestet und es wurden Verbesserungen vorgeschlagen, welche für den nächsten Release umgesetzt wurden.
Hinter der Anwendung steckt eine API, welche es ermöglicht zu einem Späteren Zeitpunkt die Anwendung auch als Webanwendung zu entwickeln.

Wo finde ich die Anwendung:
Die Anwendung liegt auf unserem Server zum nutzen bereit und ist für jeden Mitarbeiter zugänglich, der Pfad für die Installation ist der folgende:
<\\NMU-SEFILE1.ssi-ch.local\Public\Software\TRESMENT>

Gibt es eine Dokumentation zu der Anwendung:
In der Anwendung selbst, gibt es eine kleine Einführung, jedoch gibt es auch eine gesamt Dokumentation, welche unter folgendem Link einsehbar ist.
<https://confluence.ssi-schaefer.ch/display/ITDOC/Team+Ressourcen+Management+-+TRESMENT>

Wo kann ich einen Fehler melden:
Wenn ihr einen Fehler findet, bin ich natürlich froh um die Information, so dass ich diesen dann beheben kann.
Die Anwendung verfügt über eine Update Information, so dass ihr immer auf dem neusten Stand seit, wenn ein Fehler behoben wurde.

Danke an alle Beteiligten
Ich möchte mich hiermit herzlich bei allen Beteiligten Bedanken, welche Inputs geliefert haben und mich in dieser Zeit unterstützt haben !
Marc Mangold als Firmenexperte.
Reto Achermann, Christian Burkhart, Michael Collavo, Martin Fehlmann als Testpersonen.
Luca Cathomen und Melanie Trottmann für die Freigabe der Schnittstellenerweiterung von LOGICO
Und natürlich unserer gesamten GL für die Freigabe des Projektes !

Abbildung 35: GO-Live E-Mail



7.7 Wirtschaftlichkeit

Das Projekt wirft keinen direkten finanziellen Gewinn ab, es soll die internen Prozesse verbessern und die Teamleiter entlasten damit Ressourcen für andere Tätigkeiten frei werden. Es können jedoch einige zeitbezogene Kosten eingespart werden, wenn die folgenden Schritte wegfallen würden:

Schritt	Tätigkeit	Dauer (ca.)
1	Erstellen von Excel für die Planung der Mitarbeiter	30 min
2	Excel ausfüllen und verteilen	15 min
3	Aufwand für jeden Mitarbeiter (gerechnet mit einem Team von 5 Personen)	30 min
4	Rückfragen von anderen Projektleitern oder Teamleitern (Pro Anfrage gerechnet)	5 min
	Total	1.3 h

Tabelle 30: Kostenübersicht

Mit dem internen Verrechnungssatz berechnet ergibt das Kosten von **CHF 136.50** für die Planung pro Teamleiter; Mitarbeiter; Projektleiter, welche Jahr für Jahr eingespart werden könnten.

Mit der Anwendung kann der Prozess auf einen Schritt reduziert werden, weil jeder Mitarbeiter die Übersicht über die An- und Abwesenheiten von allen anderen einsehen kann. Die Kosten würden sich somit um **CHF 131.25** (1.25 h x CHF 105.- / h) senken, wobei dies nur eine Schätzung des Ausmasses ist. Eine genaue Kostenschätzung ist nicht möglich, da es Jahre/Monate geben kann, bei denen der Teamleiter deutlich mehr Zeit investieren muss oder auch nicht.



8 Projektabschluss

8.1 Reflexion der Projektziele

#	Endergebnisse	Erfolgskriterien	Erfüllt in %
1.0	Die Anwendung ist im internen Firmennetzwerk für alle zugänglich;	Die Anwendung kann nicht geladen werden, sobald der Anwender nicht mit dem internen Firmennetzwerk verbunden ist;	100%
6.0	Jeder Mitarbeiter kann nur innerhalb des internen Firmennetzwerkes die Anwendung aufrufen;	Wird die Anwendung ausserhalb des internen Firmennetzwerkes aufgerufen, erscheint eine Fehlermeldung;	100%
Begründung: Die Anwendung wird nicht geladen, wenn der Benutzer ausserhalb des Firmen-internen Netzwerkes ist.			
2.0	Der Zugriff auf die Anwendung kann von jedem Nutzer ausgeführt werden;	Anwender müssen sich nicht Authentifizieren, um die Anwendung zu öffnen;	0%
Begründung: Aufgrund der Sicherheitsbedingungen wurde dieser Punkt intern besprochen und angepasst: Der Benutzer muss sich mit dem Windows Login verifizieren.			
3.0	Die Konfiguration kann nur von eingetragenen Administratoren administriert werden;	Die Administratoren werden in einer anwendungsinternen Liste eingetragen und können dort allfällige Mutationen vornehmen. Der Entwickler ist der Super-User und hat Zugriff auf sämtliche Daten;	100%
Begründung: Die Administrator Oberfläche ist nur verfügbar, wenn der Benutzer als Administrator eingetragen ist.			
4.0	Jeder Mitarbeiter kann eigene Teams erstellen, um die beantragten und freigegebenen Abwesenheiten anzuzeigen;	Jeder Anwender kann in der Anwendung ein Team erstellen;	100%
Begründung: Nach dem GO-Live wurden direkt etliche Teams erstellt, ohne dass eine Nachfrage an mich gekommen ist.			
5.0	Jeder Mitarbeiter kann seine eigenen Teams erstellen, verwalten und löschen;	Jeder Anwender kann innerhalb der Anwendung ein Team verwalten und auch löschen;	100%
Begründung: Nach dem GO-Live wurden direkt etliche Teams erstellt, angepasst oder auch direkt gelöscht ohne dass eine Nachfrage an mich gekommen ist.			
7.0	Alle Funktionen, sowie der Aufbau der Anwendung werden im Confluence dokumentiert;	Ein neuer Bereich für die Dokumentation im Confluence wird für das Projekt erstellt;	100%
8.0	Es wird eine Benutzeranleitung für die Anwender erstellt;	Die Anleitung wird im Confluence abgelegt und ist für alle Anwender zugänglich;	100%
Begründung: Bereich ist erstellt und jeder Mitarbeiter hat Zugriff auf den Bereich.			

**Projekt TRESMENT**

#	Endergebnisse	Erfolgskriterien	Erfüllt in %
1.0	Das Design und Anforderungen der Anwendung werden mit den ausgewählten Anwendern abgesprochen;	Bedürfnisse der ausgesuchten Test Anwender werden vor der Realisierung ermittelt und die Applikation wird darauf aufgebaut;	100%
Begründung: Bedürfnisse wurden beim Kick-Off Meeting besprochen und in das Layout und die Funktionalität miteinbezogen.			
2.0	Es werden wöchentliche Meetings mit den Anwendern abgehalten, um diese über den Fortschritt zu informieren;	Funktions- und Designanpassungen werden den Anwendern laufend präsentiert, um allfällige Änderungswünsche vorzunehmen. Meetings werden fortlaufen terminiert;	100%
Begründung: Die Meetings mit den Testpersonen wurden wöchentlich gehalten, um das Feedback zu erhalten.			
3.0	Meeting alle 2 Wochen mit Fachexperte und Auftraggeber;	Der Zwischenstand wird dem Fachexperten und Auftraggeber in wöchentlichen Meetings präsentiert;	100%
Begründung: Der Zwischenstand wurde alle zwei Wochen mit dem internen Fachexperten festgestellt und besprochen.			

Tabelle 31: Reflexion der Projektziele



8.2 Reflexion und Erkenntnisse

8.2.1 Zusammenfassung

Die Projektziele konnten alle sorgfältig erreicht werden. Durch die gesammelte Erfahrung über die drei Jahre sowie dem Erlernten aus dem Unterricht des Projektmanagements und den Inputs aller erlernten Module konnte die Erarbeitung der Diplomarbeit erfolgreich erstellt werden.

8.2.2 Weg zum Ziel

Am Anfang war es eine nützliche Idee mit Chancen und Risiken. Die Komplexität zeigte sich in der Integrität und dem Datenschutz der Anwendung. Personenbezogene Daten, die vor dem Zugriff von aussen zu schützen sind, stellten beim Design und Entwicklung entsprechende Anforderungen. Ebenfalls war der Zeitfaktor von 5 Wochen zur Realisierung eine Prüfung und ob das Endresultat in der ersten Version den beabsichtigten Nutzen erfüllen kann.

Damit war der erste Stein als Weg zum Gipfel gelegt.

8.2.3 Methoden und Hilfsmittel

Die gesamte Arbeit wurde mittels dem Office 365/Teams geschrieben, sodass jeder Beteiligte jederzeit Zugriff auf das Dokument hatte und dies zugleich auf der Cloud abgespeichert wurde. Allerdings bereitete dies ab einer gewissen Seitenanzahl ein wenig Schwierigkeiten, da das Dokument lange und immer wieder neu geladen werden musste.

Das Hauptinstrument für die erarbeitete Diplomarbeit war einerseits das erhaltene Skript des Projektmanagements und andererseits das erlernte der letzten drei Jahre, insbesondere der Bereich System- und Softwareengineering.

8.2.4 Planung

Die Projektplanung konnte durch ein vorgängig erstelltes Inhaltsverzeichnis im Projekt Dokument im Word Schritt für Schritt geplant werden. Das Layout des Dokumentes (Schrift, Format etc.) wurde bereits vorbestimmt, sodass direkt in die Arbeit geschrieben werden konnte. So entstanden eine gewisse Schlüssigkeit und ein klarer Aufbau. Ebenfalls waren die Meilensteine und die Projektziele sehr nützlich, damit die Ziele und Schwerpunkte zielgerecht angestrebt werden konnten.

8.2.5 Einzelarbeit

Die Übersicht in diesem Projekt aufrecht zu erhalten, gestaltete sich als eine Herausforderung, denn die Gefahren waren die Wichtigkeiten und Prioritäten nicht schlüssig zu gestalten. Der Projektplan hatte hierbei sehr geholfen, die Pendenzen fristgerecht abzuarbeiten, so dass nicht am Ende der Erfolg ausgeblieben ist. Die Dokumentation der Arbeit war der schwierigste Teil, da dies keine technischen Spezifikationen oder ein Pflichtenheft ist. Hierbei konnte ich aus den letzten drei Jahren der Ausbildung viel lernen und mitnehmen, was mir das Schreiben und auch das Planen dieser Arbeit erleichtert hatte.



9 Quellenverzeichnis

9.1 Webquellen

Zweck	Quelle
W3Schools Tutorials und Inputs	https://www.w3schools.com/sql/
W3Schools Simulation der Befehle	https://www.w3schools.com/sql/trysql.asp?filename=trysql_select_all
Python Dokumentation	https://docs.python.org/3/
Bild Titelseite	https://www.teko.ch/de
Programmiersprache HTML & PHP Abbildung 2	https://unsplash.com/photos/ieic5Tq8YMk
Programmiersprache Python Abbildung 3	https://unsplash.com/photos/OqtafYT5kTw
Analyse Programmiersprachen	https://pypl.github.io/PYPL.html

10 Verzeichnisse

10.1 Abbildungsverzeichnung

Abbildung 1: Projektorganisation.....	13
Abbildung 2: Projektstrukturplan.....	15
Abbildung 3: Projektablaufplan.....	16
Abbildung 4 : Programmiersprache HTML & PHP	19
Abbildung 5 : Programmiersprache Python.....	20
Abbildung 6: Programmiersprache Ranking.....	27
Abbildung 7: API-Projektstruktur	29
Abbildung 8: FastAPI Projektstruktur	29
Abbildung 9: SOAP Request mit Python und der Bibliothek Zeep	30
Abbildung 10: SOAP Request über XML mit SOAPUI	30
Abbildung 11: Projektstruktur Navision Schnittstelle	30
Abbildung 12: Projektstruktur Datenbank	31
Abbildung 13: ERM Diagramm API-Datenbank	32
Abbildung 14: Projektstruktur Threading	37
Abbildung 15: Funktionsbaum Pflichtenheft	38
Abbildung 16: Entwurf 1 Oberflächen Anwendung.....	39
Abbildung 17: Entwurf 2 Oberflächen Anwendung.....	40
Abbildung 18: Projektstruktur GUI	42
Abbildung 19: Login-Fenster	43
Abbildung 20: Einführung in die Anwendung	44
Abbildung 21: Hauptfenster	45
Abbildung 22: Hauptfenster Wochenansicht	45
Abbildung 23: Hauptfenster Monatsansicht.....	46
Abbildung 24: Hauptfenster Jahresansicht.....	46
Abbildung 25: Hauptfenster Aktive Filter	46
Abbildung 26: Konfigurationsübersicht.....	47
Abbildung 27: Konfiguration Team Erstellen	47



Abbildung 28: Konfiguration Team Anpassen / Löschen	48
Abbildung 29: Konfiguration der Anwendung	48
Abbildung 30: Konfiguration Helles Design	49
Abbildung 31: Administrator Oberfläche Administrator Benutzer Verwalten	49
Abbildung 32: Administrator Oberfläche Navision Parameter Verwalten	50
Abbildung 33: Pop-upfenster mit Fehlermeldung	50
Abbildung 34: ERM Interne Datenbank	51
Abbildung 35: GO-Live E-Mail	55
Abbildung 36: Navision Parameter	96

10.2 Tabellenverzeichnis

Tabelle 1: Beruflicher Werdegang	9
Tabelle 2: Projektauftrag	10
Tabelle 3: Projektbeschreibung	10
Tabelle 4: Systemziele / Endergebnisse und Erfolgskriterien	11
Tabelle 5: Vorgehensziele / Endergebnisse und Erfolgskriterien	12
Tabelle 6: Projekttyp, Organisation und Genehmigung	12
Tabelle 7: Projektplanung und Entscheid	13
Tabelle 8: Projektrollen	14
Tabelle 9: Übersicht Projektdauer Ressourcen	17
Tabelle 10: Übersicht Woche Ressourcen	17
Tabelle 11: Kommunikationsplan	18
Tabelle 12: Kriterienpool für die Nutzwertanalyse	21
Tabelle 13: Priorisierte Kriterien aus dem Kriterienpool	21
Tabelle 14: Präferenzmatrix	22
Tabelle 15: Nutzwertanalyse	23
Tabelle 16: Erste Sensitivitätsanalyse	24
Tabelle 17: Zweite Sensitivitätsanalyse	25
Tabelle 18: Dritte Sensitivitätsanalyse	26
Tabelle 19: Backend Bibliotheken	28
Tabelle 20 Phys. Datenmodell USER	33
Tabelle 21 Phys. Datenmodell ABSENCE	33
Tabelle 22 Phys. Datenmodell ABSENCECODES	34
Tabelle 23 Phys. Datenmodell ABSENCESTATE	34
Tabelle 24 Phys. Datenmodell TEAM	35
Tabelle 25 Phys. Datenmodell TEAMUSER	35
Tabelle 26 Phys. Datenmodell ADMINNAV	36
Tabelle 27 Phys. Datenmodell ADMINUSER	36
Tabelle 28 Phys. Datenmodell SCHEDULE	36
Tabelle 29: Frontend Bibliotheken	41
Tabelle 30: Kostenübersicht	56
Tabelle 31: Reflexion der Projektziele	58



11 Glossar

Bezeichnung	Beschreibung
TRESMENT	Beschreibt den Projektnamen, welcher die gekürzte Variante von Team Res -sourcen Management ist.
API	Schnittstelle für Drittanbieter, dessen Angebote in der Applikation angezeigt werden.
LVS	Abkürzung für Lagerverwaltungssoftware.
MFR	Abkürzung für den Materialflussrechner, welcher das Herzstück eines Automatisierungsbereiches ist.
SPS	Abkürzung für Speicherprogrammierbare Steuerung, welche zum Beispiel genutzt wird, um Fördererelemente zu steuern.
Confluence	Kommerzielle Wiki-Software welche der Dokumentation von Prozessen oder Softwares dient.
IDE	Beschreibt die Abkürzung einer Integrierten Entwicklungsumgebung.
Python	Programmiersprache zum Entwickeln von Anwendungen jeglicher Art.
GUI	«Graphical User Interface» / Grafische Benutzeroberfläche.
SOAP	Netzwerkprotokoll, um Daten zu verschicken.
Thread	Separate Ausführung eines Befehles / Ablaufes. Entkoppelung von dem Hauptprozess.
CSS	Cascading Style Sheets, Stylesheet-Sprache für das Design zusammen mit HTML und JavaScript
HTML	Hypertext Markup Language, Grundprogrammiersprache jeder Website
JavaScript	Skriptsprache, um HTML Seite dynamischer aufzubauen
Debian	Freies Betriebssystem welches auf dem GNU-Prinzip (UNIX ähnliches Betriebssystem) und dem Linux-Kernel besteht



12 Redlichkeitserklärung

Die Verfasserinnen und Verfasser bestätigen mit ihrer Unterschrift, dass die vorliegende Arbeit selbstständig, ohne fremde Hilfe und ohne Benutzung anderer als die angegebenen Hilfsmittel angefertigt wurde.

Die aus fremden Quellen (einschliesslich elektronischer Quellen) direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Die Arbeit ist in gleicher oder ähnlicher Form noch nicht vorgelegt worden.

Unterschrift:

Datum/Ort: 25.10.2021 / HO

Marco Studer

M. Studer



13 Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mich während der Anfertigung dieser Diplomarbeit unterstützt und motiviert haben.

Zuerst gebührt mein Dank Benjamin Bäni (TEKO Diplomdozent) und Marc Mangold, welche die Diplomarbeit betreut und begutachtet haben. Für jegliche Kritik und Anregungen war ich stets dankbar und offen und möchte mich hiermit nochmals bedanken.

Ein besonderer Dank gilt allen Test Personen, welche die Anwendung zu dem gemacht haben, was diese jetzt ist. Ohne den Input und das Feedback von Christian Burkhart, Reto Achermann, Michael Collavo und Martin Fehlmann wäre das Ergebnis nicht so sehenswert wie es jetzt ist. Somit möchte ich mich hier nochmals bei euch bedanken, dass ihr euch die Zeit genommen habt, um das Projekt zu einem Erfolg zu führen.

Ebenfalls möchte ich mich bei der gesamten Geschäftsleitung der Firma SSI Schäfer Schweiz bedanken, für die Freigabe dieser Diplomarbeit und natürlich auch die Unterstützung, welche mir angeboten wurde.

Abschliessend möchte ich mich bei meiner Partnerin Tanja Utz und meinen Eltern bedanken, welche mir durch das Studium geholfen haben, immer ein offenes Ohr für meine Probleme hatten und schlussendlich die Diplomarbeit durchgelesen haben und mir wertvolles Feedback gegeben haben.



14 Anhang

14.1 Pflichtenheft

TEAM RESSOURCEN MANAGEMENT

Diplomarbeit

Projektteam:
Studiengang:
Klasse
Abgabe Datum:
Diplomlehrer:

Marco Studer
Technischer Informatiker HF
OTIN-18-T-a TEKÖ Olten
13.08.2021
Benjamin Bani



Inhaltsverzeichnis

1	Beschreibung	68
2	Ausgangslage	68
3	Ziele	69
3.1	Systemziele	69
3.2	Vorgehensziele	70
3.3	Rahmenbedingungen.....	70
3.4	Abgrenzungen.....	70
4	Auftragserklärung	71
4.1	Lastenheft.....	71
4.1.1	Zielbestimmung	71
4.1.2	Produkteinsatz.....	71
4.1.2.1	Produktfunktionen	72
4.1.3	Produktdaten	73
4.1.4	Produkteleistungen.....	74
4.1.5	Qualitätsanforderungen.....	74
4.1.6	Sensibilität	74
4.1.7	Ergänzungen	74
4.2	Pflichtenheft	75
4.2.1	Use Case Diagramm	76
4.2.2	Use Case Diagramm – Beschreibung.....	77
4.2.3	Funktionsbaum.....	81
5	Backend Prozesse	82
5.1	Informationserarbeitung	82
5.2	Navision Schnittstelle	82
5.2.1	Prozess Modell.....	82



6	Entity Relationship Model	83
6.1	Informationserarbeitung	83
6.2	ERM Diagramm	83
7	Strategiebezug und Umsetzung von Vorgaben	84
7.1	Strategiebezug	84
7.2	Umsetzung von Vorgaben	84
8	Rechtliche Grundlagen	84
9	Mittelbedarf	84
9.1	Personalaufwand	84
9.2	Sachmittel	84
9.3	Kosten (CHF)	85
10	Planung und Organisation	85
10.1	Projektplanung	85
10.3	Projektorganisation	86
11	Risiken & Chancen	88
11.1	Risiken	88
11.2	Chancen	88
12	Konsequenzen	89
12.1	Bei Projektfreigabe	89
12.2	Bei Projektfreigabe zu einem späteren Zeitpunkt	89
12.3	Bei ausbleibender / abgelehnter Projektfreigabe	89



1 Beschreibung

Vor mehr als 80 Jahren wurde der Grundstein für die heute international operierende SSI SCHÄFER Gruppe gelegt. Verteilt auf sechs Kontinente entwickeln mehr als 10.500 Mitarbeiter innovative Konzepte und Lösungen in den Branchen der Kunden und gestalten so die Zukunft der Intralogistik. Dabei nutzt das Unternehmen die Chancen der digitalen Vernetzung und erhöht somit die Effizienz der globalen Lieferkette. Mit der aufstrebenden Digitalisierung verändert sich auch SSI Schäfer am Standort Muen von einer Wasserfall Projektdurchführung zu einer agilen Softwareentwicklung für die kommenden Projekte.

Es wird in einem ersten Schritt die Arbeitsorganisation der vier Teams von einer klassischen Wasserfall Projektbearbeitung in eine agile Projektbearbeitung umorganisiert. Die Führungsorganisation bleibt in der hierarchischen Ausrichtung.

Um die Planung der Abwesenheiten und Ferien für die Scrumteams zu vereinfachen, soll ein Tool entwickelt werden, welches die Mitarbeiter in den Teams bestmöglich unterstützt. Dies ist nötig, da unsere Scrumteams sich in hierarchischen Einheiten befinden. Aktuell kann nur der Teamleiter die Abwesenheiten sehen, wie sie in der Zeiterfassung vom Mitarbeiter beantragt wurden. Die Mitarbeiter können keine Abwesenheiten der anderen einsehen. Was dazu führt, dass der Prozess der Abwesenheitsverteilung viele Ressourcen in Anspruch nimmt (Teamleiter, Projektleiter, Mitarbeiter).

2 Ausgangslage

Bei der Planung des Urlaubes für das nächste Jahr benutzt jedes Team seine eigenen Mittel, um sicher zu stellen, dass nicht jeder Mitarbeiter zur gleichen Zeit im Urlaub ist. Meist verwenden die Teams eine eigens gestaltete Excel Liste oder versuchen via Outlook die Abwesenheiten im Griff zu haben. Mit der kommenden Umstrukturierung in die agile Softwareentwicklung führt dies noch mehr zu Komplikationen, um die Abwesenheiten zu planen. Es besteht die Möglichkeit, dass sich Mitarbeiter aus vier verschiedenen Teams in einem Scrumteam zusammenfinden. Für die Planung muss der Projektleiter mit dem Teamleiter Rücksprache halten, um den Überblick über die Ressourcen zu haben.

Dieser ungenaue und aufwändige Prozess soll mit einer Softwareanwendung erleichtert werden. Die Software umfasst eine Schnittstelle zum Microsoft Dynamics Navision⁵ und einer Visualisierung, die die beantragten und freigegebenen Urlaubstage jedes Mitarbeitenden der SSI Schäfer Schweiz anzeigt.

Im nachfolgenden Dokument wird jeweils von Urlaubstagen gesprochen, hierbei ist immer die Rede von beantragten und freigegebenen Urlaubstagen.

⁵ Microsoft Dynamics NAV ist eine Standardsoftware für ERP(Enterprise-Resource-Planning)-Systeme. Wenn im Dokument die Rede von Navision ist, ist Microsoft Dynamics NAV gemeint.



3 Ziele

3.1 Systemziele

Richtziel: Eine neue Applikation für das «Team Ressourcen Management» ist erstellt

Nr.	Kategorie	Beschreibung	Messgrösse	Priorität*
1	Funktionalität	Applikation ist im internen Firmennetzwerk aufrufbar	Die Applikation kann nicht geladen werden, sobald man nicht im internen Firmennetzwerk ist.	1
2	Funktionalität	Der Zugriff auf die Applikation kann von jedem Benutzer ausgeführt werden.	Anwender müssen sich nicht Authentifizieren, um die Applikation zu öffnen	1
3	Funktionalität	Die Konfiguration kann nur von eingetragenen Administratoren verwaltet werden.	Die Administratoren werden in einer Liste eingetragen und können dort auch weitere hinzufügen. Der Entwickler ist der Super-User und hat ebenfalls Zugriff auf die Administratoren.	1
4	Funktionalität	Jeder Mitarbeiter kann eigene Teams erstellen, um die beantragten und freigegebenen Urlaubstage anzuzeigen	Jeder Anwender kann in der Applikation ein Team erstellen	1
5	Funktionalität	Jeder Mitarbeiter kann seine eigenen Teams verwalten und auch löschen	Jeder Anwender kann in der Applikation ein Team verwalten und auch löschen.	1
6	Funktionalität	Jeder Mitarbeiter kann nur innerhalb des internen Firmennetzwerkes die Applikation aufrufen	Der Anwender bekommt eine Fehlermeldung, wenn die Applikation ausserhalb des internen Firmennetzwerkes aufgerufen wird.	1
11	Wartung	Alle Funktionen, sowie der Aufbau der Applikation werden im Confluence dokumentiert	Ein neuer Bereich für die Dokumentation im Confluence wird für das Projekt erstellt	2
12	Handhabung	Es wird eine Anleitung für die Anwender erstellt	Die Anleitung wird im Confluence abgelegt und ist für alle Anwender einsehbar	2

* **Priorität: 1 = hoch, 2 = mittel, 3 = tief**



3.2 Vorgehensziele

Nr.	Kategorie	Beschreibung	Messgrösse	Priorität*
1	Stakeholder	Design und Anforderungen der Applikation werden mit den ausgewählten Anwendern abgesprochen	Bedürfnisse der drei Anwender werden vor der Realisierung ermittelt und die Applikation wird darauf aufgebaut	1
2	Stakeholder	Wöchentliche Meetings mit den Anwendern	Funktions- & Designanpassungen werden den Anwendern laufend präsentiert, um allfällige Änderungswünsche vorzunehmen. Meetings werden fortlaufen terminiert	1
4	Stakeholder	Meeting alle 2 Wochen mit Firmenexperte & Auftraggeber	Der Zwischenstand wird dem Firmenexperten & Auftraggeber in wöchentlichen Meetings präsentiert	1

* **Priorität: 1 = hoch, 2 = mittel, 3 = tief**

3.3 Rahmenbedingungen

Für die Entwicklung und Planung des Projektes werden keine weiteren Softwarelizenzen benötigt und es kann alles mit den von SSI Schäfer bereitgestellten Tools erledigt werden.

Am Standort Neunkirch oder Muhen kann eine ESXi⁶ Infrastruktur genutzt werden.

3.4 Abgrenzungen

Die Analyse der Zeit- und Kostenersparnis sowie die Anpassung der Prozesse nach Fertigstellung der Applikation, ist nicht Teil dieses Projekts.

⁶ VMware ESXi ist ein Typ-1-Hypervisor der Enterprise-Klasse, der von VMware für die Bereitstellung und Bereitstellung virtueller Computer entwickelt wurde.



4 Auftragserklärung

4.1 Lastenheft

Das Lastenheft wird erstellt, um ein gegenseitiges Verständnis darzubieten für das Projekt. Dieses Lastenheft enthält alle Anforderungen, die das zu erstellende System bereitstellen soll. Die Anforderungen wurden durch den Input des Auftraggebers in das Lastenheft eingearbeitet

Projektname	TRESMENT – Team Ressource Management
Verantwortlicher Mitarbeiter	Studer Marco
Digitaler Ablageort des Dokuments	Microsoft Teams / Sharepoint Office 365

4.1.1 Zielbestimmung

Der Auftrag besteht aus folgenden Punkten welche zu erfüllen sind:

- Graphical User Interface ⁷Funktionalitäten
 - Ansicht Urlaubstage in verschiedenen Designs;
 - Möglichkeit eigene Teams erstellen und auch bestehende erstellte Teams bearbeiten;
 - Ansicht Urlaubstage von verschiedenen Teams;
 - Zugriff nur für Mitarbeiter der SSI Schäfer Schweiz;
 - Möglichkeit schaffen die Urlaubstage in ein anderes Format zu exportieren;
- Die Datenaufbereitung im Backend⁸ muss so erstellt werden, dass diese Daten auch für künftige Projekte genutzt werden können;
- Die Administratoren können Parameter wie Schnittstellenbenutzer oder diverse Codes anpassen und wenn nötig von Benutzern erstellte Teams löschen;
- Die Dokumentation muss wie folgt gegliedert und aufgebaut sein;
 - Benutzerhandbuch für Installation und Umgang mit der Applikation;
 - Entwicklerhandbuch für das Aufsetzen der Umgebung und die Weiterentwicklung;
 - Administratorhandbuch für das Anpassen von Parametern und löschen von Teams;

4.1.2 Produkteinsatz

Das Produkt kann von jedem Mitarbeiter innerhalb der SSI Schäfer AG Schweiz genutzt werden.

Explizit von:

- Mitarbeiter
- Administratoren
- Entwickler

⁷ Graphical User Interface (kurz GUI) ist die Grafische Benutzeroberfläche einer Applikation.

⁸ Der Begriff Backend wird genutzt, um einen Hintergrundprozess zu beschreiben/benennen.



Projekt TRESMENT

4.1.2.1 Produktfunktionen

AA: GUI; Sichtung Urlaubstage

Folgende Funktionalitäten werden bei der Visualisierung der Daten benötigt:

- Ansicht Abwesenheiten pro Woche
- Ansicht Abwesenheiten pro Monat
- Ansicht Abwesenheiten pro Jahr
- Ansicht „mögliche produktive Stunden“ pro Mitarbeiter

AB: GUI; Erfassung von Teams

Folgende Funktionalitäten benötigt das Erstellen von neuen Team Ansichten:

- Erstellen von eigenem Team
- Hinzufügen von Teammitgliedern
- Teammitgliedern hinzufügen
- Teammitglieder löschen

AC: GUI; Zugriff auf alle Teams:

- Suche nach Teams

AD: GUI; Login:

- Anmelden mittels Navision Namen
- Anmelden nur im internen Netzwerk möglich

AE: GUI; Exportieren der Daten in ein anderes Tool:

- Abwesenheiten Exportieren in ein anderes Format

BA: GUI; Verwaltung von Administratoren

- Hinzufügen von Administratoren
- Löschen von Administratoren

BB: GUI; Verwaltung von Parameter

Die Administratoren müssen folgende Aktionen in der Applikation erledigen können:

- Anpassung von Navision Schnittstellen Benutzer
- Anpassung der von Navision bereitgestellten Absenz-Codes
- Anpassung der von Navision bereitgestellten Absenz-Status
- Erstellte Teams löschen

C: Backend Schnittstelle

Die Datenaufbereitung muss so ausgelegt sein, so dass die Daten auch für weitere Projekte genutzt werden können damit nicht noch eine weitere Schnittstelle zu Navision hergestellt werden muss:

- Schnittstelle zu Microsoft Dynamics Navision erstellen
- REST API erstellen

D: Applikation Dokumentation

Die Dokumentation für die Applikation und das Backend muss zwingend folgende Punkte umfassen:

- Benutzerhandbuch für Anwender
- Entwicklungsdokumentation Applikation & Backend für Entwickler
- Administratorhandbuch für die Administratoren



4.1.3 Produktdaten

AA: GUI; Ansicht Urlaubstage

- Monat
- Jahr
- Tag
- Abwesend
- Verfügbar
- Anzahl verfügbare Stunden

AB: Erstellen von Teams

- Team Name
- Vorname Nachname von Team Mitglied

ABA: GUI; Verwaltung von Teams

- Vorname Nachname von Team Mitglied
- Button für Hinzufügen
- Button für Entfernen

AC: GUI; Zugriff auf alle Teams:

- Suchfunktion nach Teams

AD: GUI; Login:

- Vorname Nachname

AE: GUI; Exportieren der Daten in ein anderes Tool:

- Button für Abwesenheiten Exportieren

BA: GUI; Verwaltung von Administratoren

- Suche nach Mitarbeiter
- Mitarbeiter als Administrator Hinzufügen
- Mitarbeiter als Administrator Löschen

BB: GUI; Verwaltung von Parameter

- Navision Benutzer
- Navision Passwort
- Absenz-Code hinzufügen
- Absenz-Code löschen
- Absenz-Status hinzufügen
- Absenz-Status löschen
- Taste, um erstellte Teams zu löschen



C: Backend Schnittstelle

- Mitarbeiter Nummer
- Mitarbeiter Name
- Arbeitsort
- Team Nummer
- Team Name
- Historie Felder
- Absenz Nummer
- Absenz Datum

D: Applikation Dokumentation

- Benutzerhandbuch Applikation
- Entwicklerhandbuch Applikation
- Entwicklerhandbuch backend
- Administrationshandbuch Applikation

4.1.4 Produkteleistungen

Performance: Die Ansicht der Abwesenheiten muss rasch einsehbar sein.

Konsistenz: Die Daten sind über die gesamte Anwendung konsistent und weisen keine Redundanten auf. Die Anwendung erzwingt die Erfassung aller notwendigen Felder.

4.1.5 Qualitätsanforderungen

Die effiziente Bedienung und performante Laufzeit der Applikation haben oberste Priorität. An nächster Stelle folgt die Korrektheit der Daten und die Robustheit der Applikation.

Die Anwender sollen ohne Schulung und Einweisung die Applikation intuitiv bedienen können. Die Applikation muss auf Windows einwandfrei funktionieren.

4.1.6 Sensibilität

Der Datenschutz und die Kriterien der Datenhaltung ist oberste Priorität. Die Daten dürfen nicht ausserhalb des Firmennetzwerks empfangen werden können.

4.1.7 Ergänzungen

Realisierung

Das Endprodukt und die Anwendung sehen keine spezifische Programmiersprache vor.

Skalierbarkeit

Die Anwendung muss jederzeit mit neuen Features erweiterbar sein.



4.2 Pflichtenheft

Das Pflichtenheft beinhaltet alle «ME» und «KE» Kriterien. Dabei handelt es sich um sogenannte «Muss erfüllt» und «Kann erfüllt» Pflichten.

#	Beschreibung	ME KE
1	Use-Case Diagramm ist erstellt	ME
2	Use-Case Beschreibung ist erstellt	ME
3	Funktionsbaum ist erstellt	ME
4	Backend Prozess ist erstellt	ME
6	Das Pflichtenheft wird dem Auftraggeber übergeben	ME
7	ERM-Diagramm ist erstellt	KE

Legende:

ME: Muss erfüllt werden

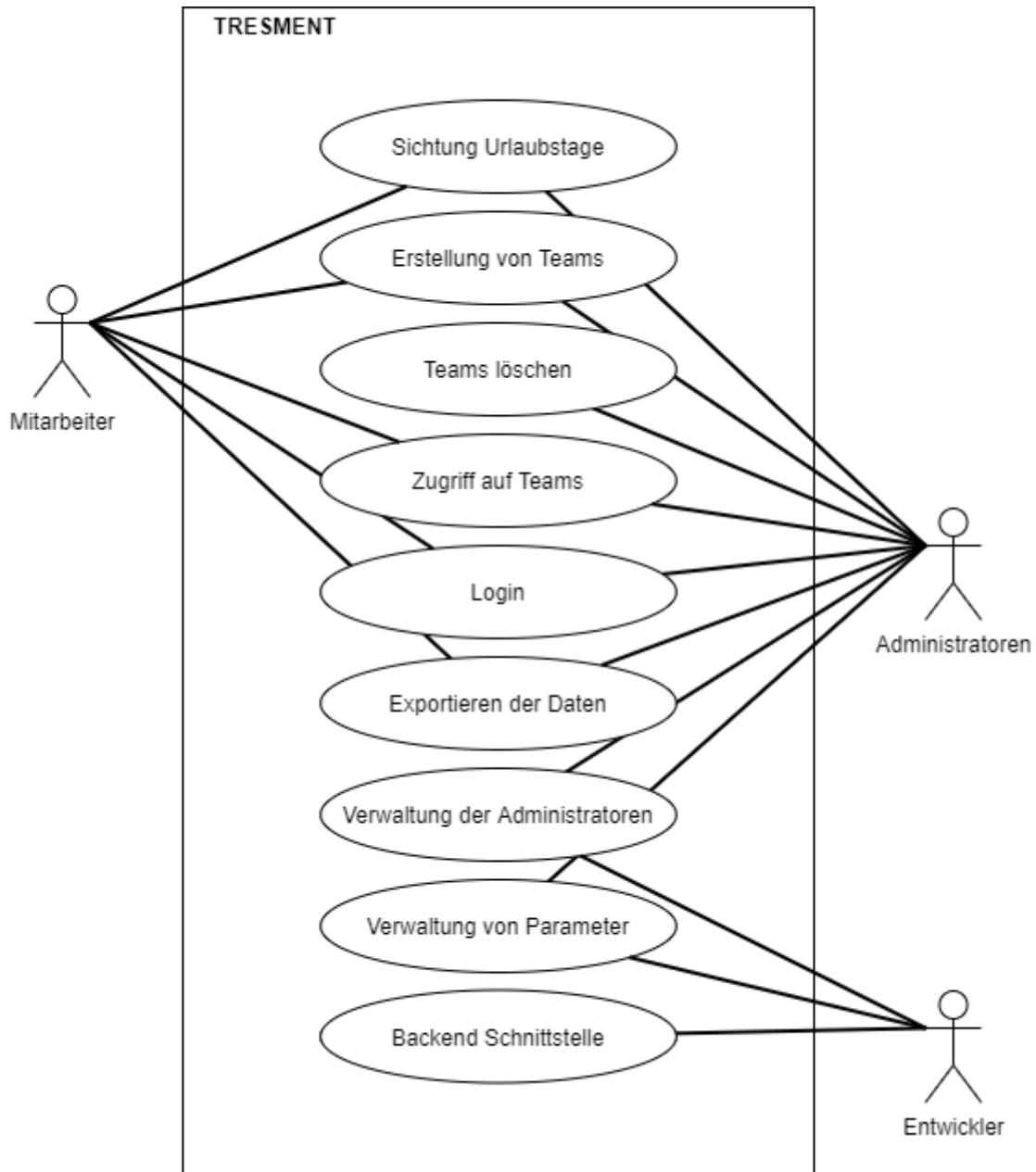
KE: Kann erfüllt werden



4.2.1 Use Case Diagramm

Das Use Case Diagramm für das Team Ressource Management umfasst die Akteure „Mitarbeiter, Administratoren und Entwickler“

Den Akteuren ist jeweils der zugelassene und angedachte „Operative- Prozess“ assoziiert und stellt damit die Kernaktivitäten der Applikation in der Verwendung dar.





4.2.2 Use Case Diagramm – Beschreibung

[AA100] Sichtung Urlaubstage	
Ziel	Der Benutzer kann in verschiedenen Ansichten die Urlaubstage einsehen.
Kategorie	Operativ
Vorbedingung	Mitarbeiter ist im internen Netzwerk eingeloggt
Nachbedingung	Erfolg: Der Benutzer kann die Urlaubstage einsehen
Akteure	Mitarbeiter, Administrator
Auslösendes Ereignis	Der Benutzer wählt eine Ansichtsmethode aus.
Beschreibung	<ol style="list-style-type: none">1. Ansicht für die Sichtung der Urlaubstage wird ausgewählt2. Die Applikation stellt die Informationen wie beschrieben dar.
Erweiterungen	<ol style="list-style-type: none">1. Die Ansicht kann auch die verfügbaren Stunden anzeigen [AA101]

[AB100] Erstellung von Teams	
Ziel	Der Benutzer kann ein Team erstellen und Mitglieder hinzufügen.
Kategorie	Operativ
Vorbedingung	Mitarbeiter ist in die Applikation eingeloggt. [AD100]
Nachbedingung	Erfolg: Der Benutzer kann ein neues Team anlegen
Akteure	Mitarbeiter, Administrator
Auslösendes Ereignis	Neues Team hinzufügen
Beschreibung	<ol style="list-style-type: none">1. Neues Team hinzufügen2. Teammitglieder hinzufügen
Erweiterungen	Änderungen <ol style="list-style-type: none">1. Mitarbeiter löschen aus bestehendem Team [AB101]2. Mitarbeiter hinzufügen in bestehendes Team [AB102]3. Administratoren können bestehende Teams löschen [AB103]

**Projekt TRESMENT****[AC100] Zugriff auf Teams**

Ziel	Der Benutzer kann nach erfassten Teams suchen.
Kategorie	Operativ
Vorbedingung	Mitarbeiter ist in die Applikation eingeloggt. [AD100]
Nachbedingung	Erfolg: Der Benutzer hat Zugriff auf alle erfassten Teams
Akteure	Mitarbeiter, Administrator
Auslösendes Ereignis	Suche nach Team
Beschreibung	<ol style="list-style-type: none">1. Eingabe des Teams in der Suchfunktion2. Gesuchtes Team wird angezeigt

[AD100] Login

Ziel	Der Benutzer kann sich nur im internen Netzwerk in die Applikation anmelden.
Kategorie	Operativ
Vorbedingung	Mitarbeiter ist im Navision erfasst
Nachbedingung	Erfolg: Der Benutzer kann sich in die Applikation anmelden und diese benutzen
Akteure	Mitarbeiter, Administrator
Auslösendes Ereignis	Der Benutzer meldet sich mit seinen Logindaten an
Beschreibung	<ol style="list-style-type: none">1. Anmelden in die Applikation
Erweiterungen	<ol style="list-style-type: none">1. Fehler wenn der Benutzer nicht im internen Netzwerk ist [AD101]2. Fehler wenn der Benutzer nicht vorhanden ist [AD102]

**[AE100] Exportieren der Daten**

Ziel	Die ausgelesenen Daten können in ein anderes Tool welches die Dateiformate .csv oder .pdf verwenden können exportiert werden.
Kategorie	Operativ
Vorbedingung	Mitarbeiter ist in der Applikation eingeloggt [AD100] und im internen Netzwerk
Nachbedingung	Erfolg: Die Absenzen werden in ein anderes Tool exportiert.
Akteure	Mitarbeiter, Administrator
Auslösendes Ereignis	Der Mitarbeiter betätigt den «Export» Button
Beschreibung	<ol style="list-style-type: none">1. Team auswählen und Ansicht festlegen2. Exportieren Button betätigen
Erweiterungen	Export <ol style="list-style-type: none">1. Export ist möglich in folgendem Datenformat: .csv [AE101]2. Export ist möglich in folgendem Datenformat: .pdf [AE102]
Alternativen	-

[BA100] Verwaltung von Administratoren

Ziel	Die Administratoren können den Admin Bereich aufrufen und weitere Administratoren hinzufügen oder diese entfernen.
Kategorie	Operativ
Vorbedingung	Mitarbeiter ist in der Applikation eingeloggt [AD100], verfügt über die nötigen Berechtigungen und ist im internen Netzwerk eingeloggt
Nachbedingung	Erfolg: Weitere Administratoren wurden hinzugefügt oder entfernt
Akteure	Administrator, Entwickler
Auslösendes Ereignis	Der Mitarbeiter betätigt den Administrator Reiter.
Beschreibung	<ol style="list-style-type: none">1. Navigation mit dem Administrator Reiter2. Suche nach neuem Administrator
Erweiterungen	Änderung <ol style="list-style-type: none">1. Hinzufügen von Administratoren [BA101]2. Löschen von Administratoren [BA102]
Alternativen	-

**[BB100] Verwaltung von Parameter**

Ziel	Die Administratoren können den Admin Bereich aufrufen und die Parameter verändern.
Kategorie	Operativ
Vorbedingung	Mitarbeiter ist in der Applikation eingeloggt [AD100], verfügt über die nötigen Berechtigungen und ist im internen Netzwerk eingeloggt
Nachbedingung	Erfolg: Parameter können von Administratoren angepasst werden.
Akteure	Administrator
Auslösendes Ereignis	Der Mitarbeiter betätigt den Administrator Reiter.
Beschreibung	1. Navigation mit dem Administrator Reiter
Erweiterungen	Änderung 1. Anpassung von Schnittstellen Benutzer [BB101] 2. Anpassung von Absenz-Codes [BB102] 3. Anpassung von Absenz-Status [BB103]
Alternativen	-

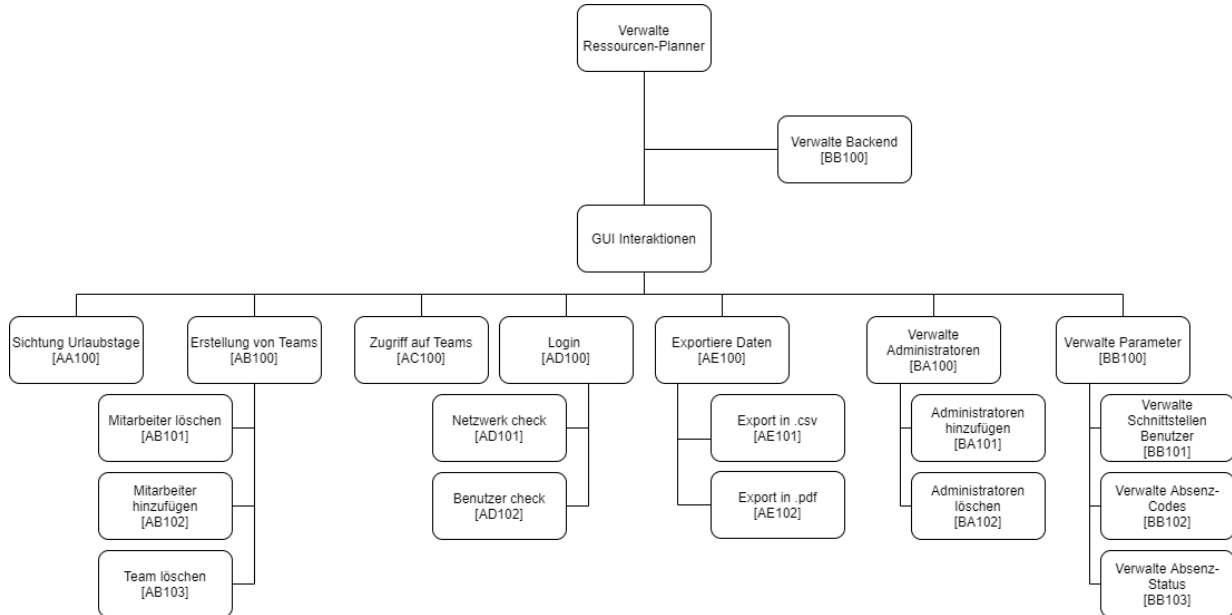
[BB100] Backend Schnittstelle

Ziel	Die Schnittstelle zu Navision ist korrekt abgesichert und aufrufbar über mehrere Ressourcen.
Kategorie	Operativ
Vorbedingung	Mitarbeiter ist im internen Netzwerk
Nachbedingung	-
Akteure	Entwickler
Auslösendes Ereignis	Aufrufen der REST-API mit dem entsprechenden Auth-Key.
Beschreibung	1. Backend ist mit einem Auth-Key abgesichert 2. REST-API ist aufrufbar im internen Netzwerk



4.2.3 Funktionsbaum

Der Funktionsbaum oder auch Funktionsstrukturbaum genannt umfasst in der Verwaltung- und Verwendung des Team Resource Management und des Backend folgende Kernfunktionen:



Kurzer Hinweis: Bei der Funktionsstruktur AB100, AD100, AE100, BA100, BB100 werden die zugeordneten Erweiterungen jeweils mit der Funktionsbezeichnung X1XX und einer separaten Funktionslinie dargestellt.



5 Backend Prozesse

5.1 Informationserarbeitung

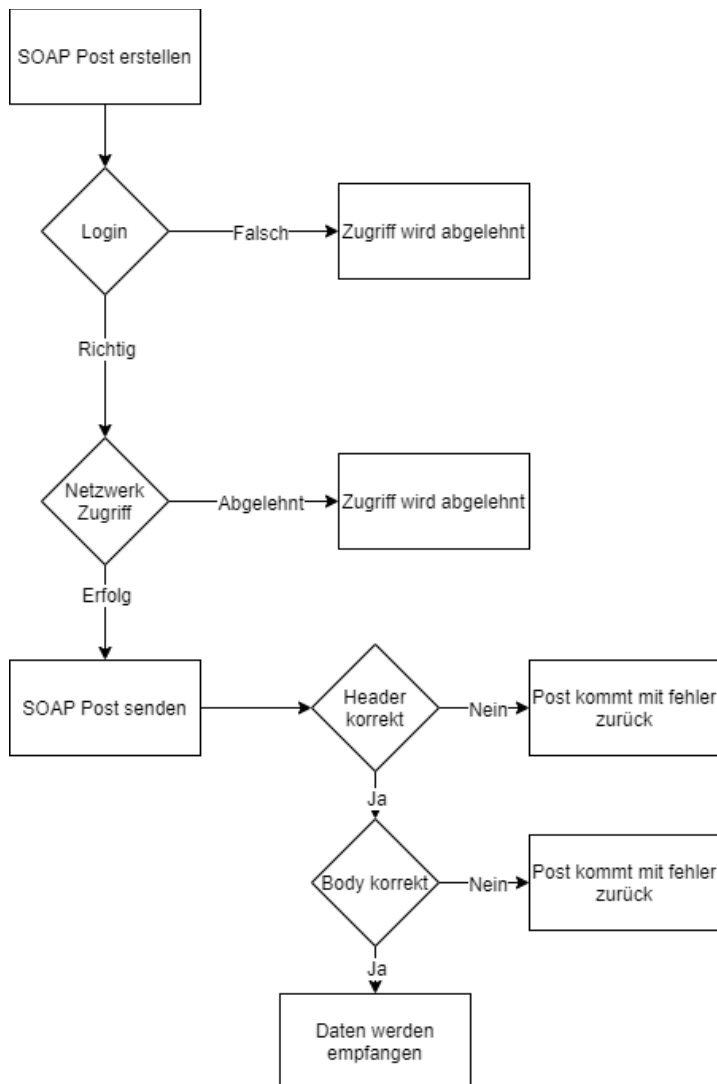
Die Schnittstelle zu Microsoft Dynamics Navision muss mit der Firma Logico abgestimmt werden, um eine Verbindung aufzubauen. Der Zugriff zu der Schnittstelle kann nur mit der SOAP Post Methode gemacht werden.

5.2 Navision Schnittstelle

Für die Schnittstelle werden folgende Punkte benötigt:

- Schnittstellen Benutzer
- Schnittstellen Passwort
- Erstellung von einem SOAP⁹ Post

5.2.1 Prozess Modell



⁹ SOAP (ursprünglich für *Simple Object Access Protocol*) ist ein Netzwerkprotokoll, um Daten zwischen Systemen auszutauschen



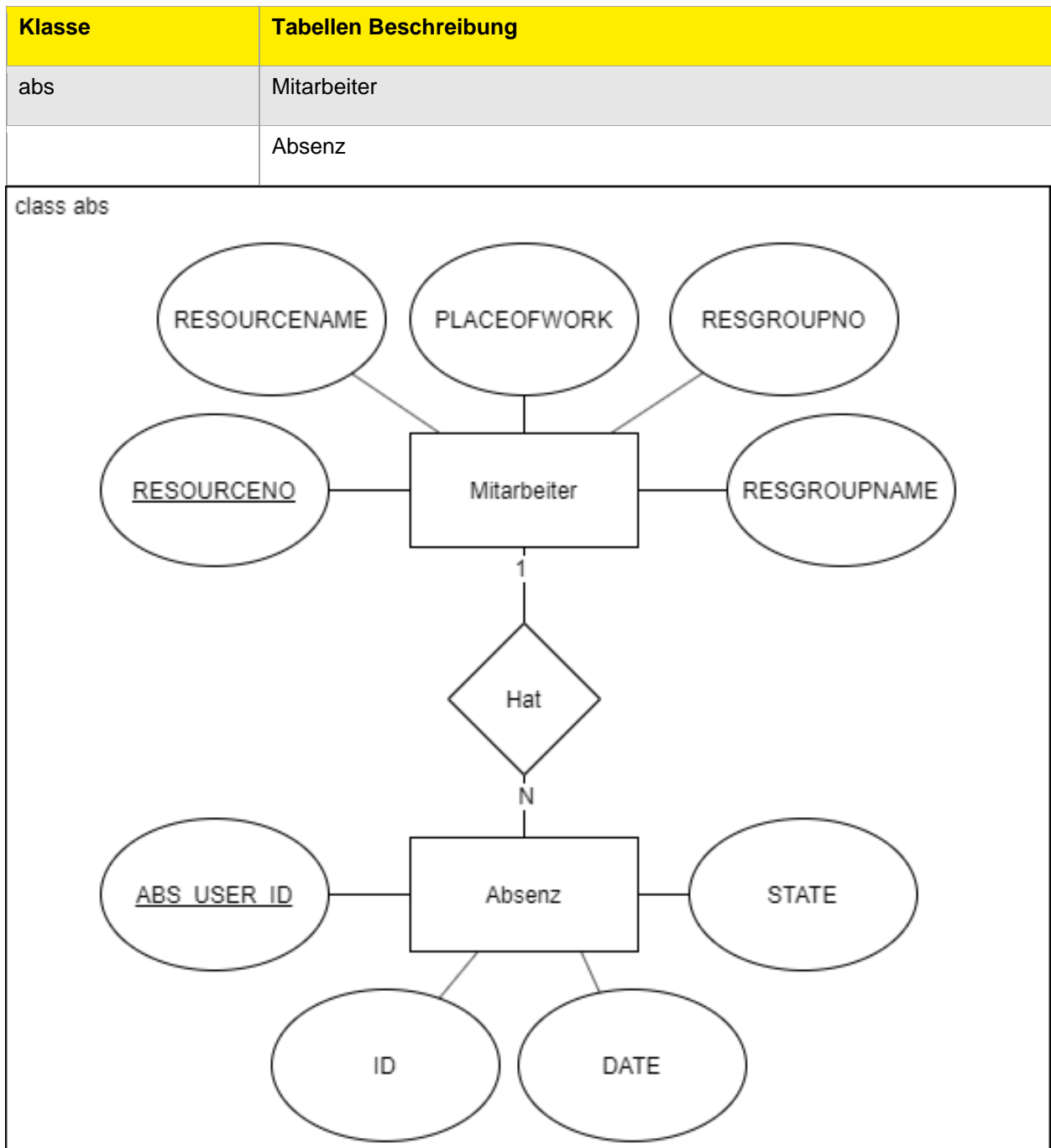
6 Entity Relationship Model

6.1 Informationserarbeitung

Das ERM kann bereits designed werden, da die Datenkonstellation von Microsoft Dynamics Navision bereits bekannt ist.

6.2 ERM Diagramm

Das ERM Beinhaltet folgende Klassen und Tabellen und ist unterhalb abgebildet. Weitere Klassen werden während der Diplomarbeit erarbeitet, festgelegt und dokumentiert.





7 Strategiebezug und Umsetzung von Vorgaben

7.1 Strategiebezug

Die SSI Schäfer Schweiz arbeitet laufend an der Verbesserung von internen Prozessen, somit entspricht das Projekt Team Ressource Management der Firmenphilosophie. Ebenso kommt es der stark ausgelasteten Teamleiter & Projektleitern entgegen, die Ressourcenplanung und Urlaubsplanung zu vereinfachen.

7.2 Umsetzung von Vorgaben

Das Design der Applikation soll dem Corporate Design der SSI Schäfer AG entsprechen.

8 Rechtliche Grundlagen

Das Projekt Team Ressource Management muss unter Einhaltung der aktuellen Datenschutz- und Datensicherheitsbestimmungen durchgeführt werden, da mit Mitarbeiterinformationen umgegangen wird.

9 Mittelbedarf

9.1 Personalaufwand

Phase	Geplant (Manntage ¹⁰)
Initialisierung*	5
Konzept	5
Realisierung Applikation	16
Realisierung Backend	7
Handbücher	3
Review mit Anwender	1
Einführung	2
Total	39
* Vorleistung (IST)	

9.2 Sachmittel

Sachmittel werden keine Benötigt, zusätzliche Hard- oder Software sowie Lizenzen müssen weder für die Entwicklung noch den Betrieb beschafft werden.

¹⁰ Ein Manntag entspricht 8 Stunden



9.3 Kosten (CHF)

Die Kosten wurden mit dem internen Verrechnungssatz von CHF 105.- / h berechnet.

Phase	Geplant (Manntage)
Initialisierung*	4'200.-
Konzept	4'200.-
Realisierung Applikation	13'440.-
Realisierung Backend	5'880.-
Handbücher	2'520.-
Review mit Anwender	840.-
Einführung	1'680.-
Total	32'760.-
* Vorleistung (IST)	

10 Planung und Organisation

10.1 Projektplanung

Meilensteine	Geplant
Projektfreigabe	13.08.2021
Fertigstellung Planung & Konzept	13.09.2021
Vorbereitung Infrastruktur	20.09.2021
Fertigstellung Front-End	30.09.2021
Fertigstellung Back-End	18.09.2021
Go-Live	18.10.2021
Projektabschluss	22.10.2021



10.3 Projektorganisation

Auftraggeber	
Name	Marc Mangold
Beruf	Softwareentwickler MAS IT
Funktion / Vertretende Organisationseinheit	Leiter Helpdesk Customer Services
E-Mail	marc.mangold@ssi-schaefer.com
Telefon	+41 62 739 34 65
Adresse	Kesslerstrasse 1 5037 Muhen

Projektleiter	
Name	Marco Studer
Funktion / Vertretende Organisationseinheit	Software Engineer Helpdesk
E-Mail	marco.studer@ssi-schaefer.com
Telefon	+41 62 739 34 08

Fachexperte	
Name	Marc Mangold
Funktion / Vertretende Organisationseinheit	Leiter Helpdesk
E-Mail	marc.mangold@ssi-schaefer.com
Telefon	+41 62 739 34 65



Anwender	
Name	Michael Collavo
Funktion / Vertretende Organisationseinheit	Sales Consultant
E-Mail	michael.collavo@ssi-schaefer.com
Telefon	+41 62 739 34 44

Anwender	
Name	Christian Burkhart
Funktion / Vertretende Organisationseinheit	Service Account Manager
E-Mail	christian.burkhart@ssi-schaefer.com
Telefon	+41 62 739 34 81

Anwender	
Name	Martin Fehlmann
Funktion / Vertretende Organisationseinheit	Software Engineer
E-Mail	martin.fehlmann@ssi-schaefer.com
Telefon	+41 62 739 34 36



11 Risiken & Chancen

11.1 Risiken

Nr.	Beschreibung	EW	AG	RZ	Massnahmen	Verantw.	Termin
1	Sicherheit der Applikation	1	2	2	Applikation nur im internen Netz erreichbar / Authentifizierung über Benutzer	PL	04.10.2021
2	Verzögerungen aufgrund von Schwierigkeiten beim Programmieren	2	2	6	Bei Schwierigkeiten Hilfe beim Firmenexperten beantragen & Stakeholder über Verzögerungen informieren	PL	04.10.2021

Legende: **EW = Eintrittswahrscheinlichkeit: 1 Niedrig / 2 Mittel / 3 Hoch**
AG = Auswirkungsgrad: 1 Niedrig / 2 Mittel / 3 Hoch
RZ = Risikozahl: EW x AG

11.2 Chancen

Nr.	Beschreibung	EW	AG	CZ	Massnahmen	Verantw.	Termin
1	Zeit- und Kostensparnis nach Fertigstellung des Projekts	3	3	9	Erfolgreiche Fertigstellung des Projekts, Einführung der Anwender	PL	22.10.2021

Legende: **EW = Eintrittswahrscheinlichkeit: 1 Niedrig / 2 Mittel / 3 Hoch**
AG = Auswirkungsgrad: 1 Niedrig / 2 Mittel / 3 Hoch
CZ = Chancenzahl: EW x AG



12 Konsequenzen

12.1 Bei Projektfreigabe

Bei Projektfreigabe kann das Projekt wie geplant gestartet werden.

12.2 Bei Projektfreigabe zu einem späteren Zeitpunkt

Das Projekt ist nicht zeitkritisch und könnte auch zu einem späteren Zeitpunkt gestartet werden, allerdings sollten die vorgegebenen Termine der STFW eingehalten werden.

12.3 Bei ausbleibender / abgelehnter Projektfreigabe

Würde das Projekt abgelehnt werden, könnte alternativ über den Einkauf & Einführung eines bestehenden Tools mit ähnlichem Funktionsumfang diskutiert werden.

Projektauftrag genehmigt:

13.9.2021 B. Bani

Rolle	Name	Ort, Datum	Unterschrift
Projektleiter	Marco Studer	MU, 10.08.2021	M. Studer
Auftraggeber	Marc Mangold	MU, 10.08.2021	Mangold
CFO	Luca Cathomen	MU, 10.08.2021	Luca Cathomen



14.2 Handbücher

Die Handbücher wurden 1 zu 1 aus dem Confluence exportiert und passen dementsprechend nicht zu 100% ins Word Format.

Handbuch Team Ressourcen Management

Benutzerhandbuch
Administratorhandbuch
Entwicklerhandbuch



Inhaltsverzeichnis

1	Administratorhandbuch	94
2	Benutzerhandbuch	97
3	Entwicklerhandbuch	109
3.1	Voraussetzungen	109
3.2	Benutze Libraries API	109
3.3	Benutze Libraries GUI	110
3.4	API-Handbuch	110
3.4.1	Serverzugang	111
3.4.2	Start der API	111
3.4.3	Projektstruktur	111
3.4.4	Tipps und Tricks	114
3.5	GUI Handbuch.....	115
3.5.1	Source-Code	115
3.5.2	Auswahl der IDE.....	115
3.5.3	Start der Applikation	119
3.5.4	Projektstruktur	121
3.5.5	Bilden der Anwendung	124



Team Ressourcen Management
Diplomarbeit von Marco Studer

Projektteam:	Marco Studer
Studiengang:	Technischer Informatiker HF
Klasse	OTIN-18-T-a TEKO Olten
Abgabe Datum:	25.10.2021
Diplomlehrer:	Benjamin Bäni
Firmen-Experte:	Marc Mangold
Programmiersprache:	Python
Ablageort:	T:\Software\TRESMENT

Ausgangslage

Bei der Planung des Urlaubes für das nächste Jahr benutzt jedes Team seine eigenen Mittel, um sicher zu stellen, dass nicht jeder Mitarbeiter zur gleichen Zeit im Urlaub ist. Meist verwenden die Teams eine eigens gestaltete Excel Liste oder versuchen via Outlook die Abwesenheiten im Griff zu haben. Mit der kommenden Umstrukturierung in die agile Softwareentwicklung führt dies noch mehr zu Komplikationen, um die Abwesenheiten zu planen. Es besteht die Möglichkeit, dass sich Mitarbeiter aus vier verschiedenen Teams in einem Scrumteam zusammenfinden. Für die Planung muss der Projektleiter mit dem Teamleiter Rücksprache halten, um den Überblick über die Ressourcen zu haben.

Dieser ungenaue und aufwändige Prozess soll mit einer Softwareanwendung erleichtert werden. Die Software umfasst eine Schnittstelle zum Microsoft Dynamics Navision und einer Visualisierung, die die beantragten und freigegebenen Absenzen jedes Mitarbeitenden der SSI Schäfer Schweiz anzeigt.

Pflichtenheft



20210910 Pflich...arco_Studer.pdf

Für was kann die Anwendung genutzt werden ?

Die Anwendung dient zum Planen der Urlaube in den fixen Teams wie auch in neu geschaffenen Teams für die Projekte.



Eine weitere Anwendung ist die Allgemeine Übersicht der verfügbaren Ressourcen in einem Team wie auch das Exportieren in ein Excel.

Was bietet die Anwendung für Erweiterungsmöglichkeiten ?

Die Anwendung selbst kann bis ins unendliche weiterentwickelt werden, da die Anwendung auf stabilen Libraries im Python Umfeld aufgebaut wurde.

Ein weiteres Ziel kann sein die Anwendung in eine Webbasierte Applikation umzuwandeln.

Hierfür wurde bereits eine API geschaffen, um dies zu ermöglichen.

Wo kann ich einen Fehler melden ?

Fehler können gerne alle an [Studer Marco](#) gemeldet werden.

Handbücher

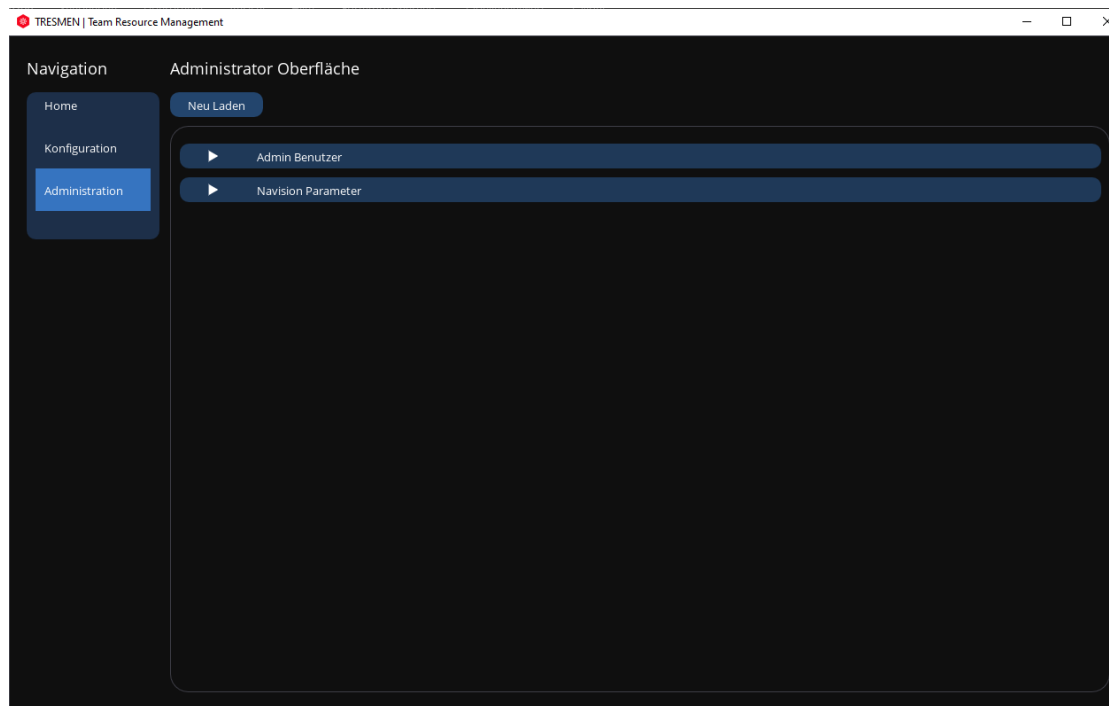
- [Administratorhandbuch](#)
- [Benutzerhandbuch](#)
- [Entwicklerhandbuch](#)



1 Administratorhandbuch

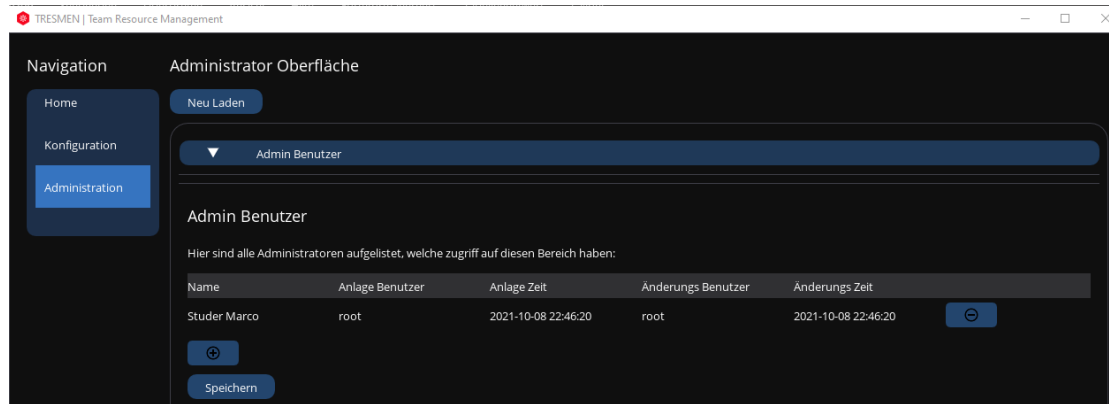
Das Administratorhandbuch baut auf dem [Benutzerhandbuch](#) auf.

Als Administrator gibt es einen zusätzlichen Menü Punkt, welcher es ermöglicht gewisse API-Parameter anzupassen.



Die Administratoroberfläche ist nur zugänglich für Benutzer welche als Administrator Benutzer hinterlegt sind.

Diese Hinterlegung kann jeder Entwickler auf der API-Seite machen oder jeder der bereits ein Administrator ist.



Administrator Benutzer

Ein Administrator Benutzer kann von einem Administrator oder einem Entwickler hinzugefügt werden.

Das Modifizieren der Administrator Benutzer ist auf demselben Prinzip aufgebaut wie bei der

[Team Erstellung / Team Anpassung](#).

- Um einen weiteren Administrator hinzuzufügen gibt es jedoch nur eine Option: Der Komplette Name kann in dem Feld eingetragen werden

Um weitere Mitglieder hinzuzufügen, kann entweder das "Plus" Icon gedrückt werden oder man bestätigt mit der "Enter Taste".

Wenn ein Mitglied zu viel hinzugefügt wurde, kann dieser mit dem "Minus" Icon wieder entfernt werden.

- **Speichern** Die Manipulation an den Administratoren wird gespeichert

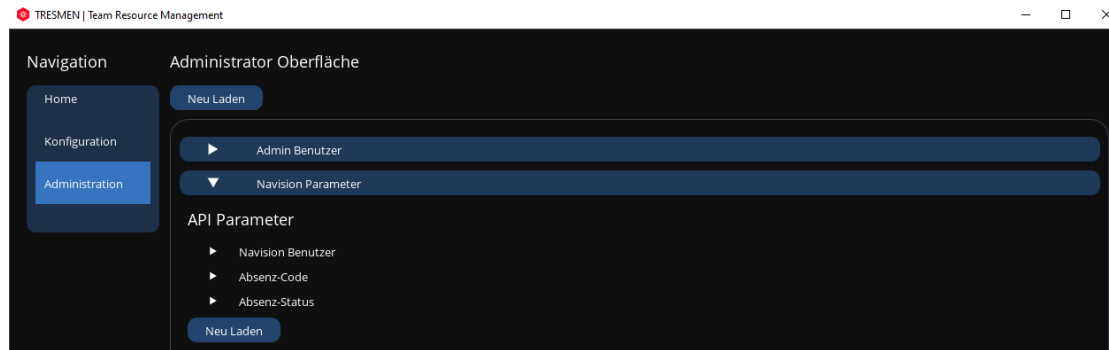


Abbildung 36: Navigation Parameter

Navision Parameter

Die Navision Parameter dürfen nur mit Zustimmung vom HR erweitert oder umgestellt werden.

- **Navision Benutzer**

Log-In Daten für die Schnittstelle von der API zum Navision. Diese dürfen nur angepasst werden, wenn etwas an dem Benutzer angepasst werden muss !

- **Absenz-Code**

Die Absenz-Codes definieren welche Absenzen in der Ressourcen-Tabelle angezeigt werden.

- **Absenz-Status**

definiert den Absenz-Status von Navision, als Beispiel 1 = freigegeben



2 Benutzerhandbuch

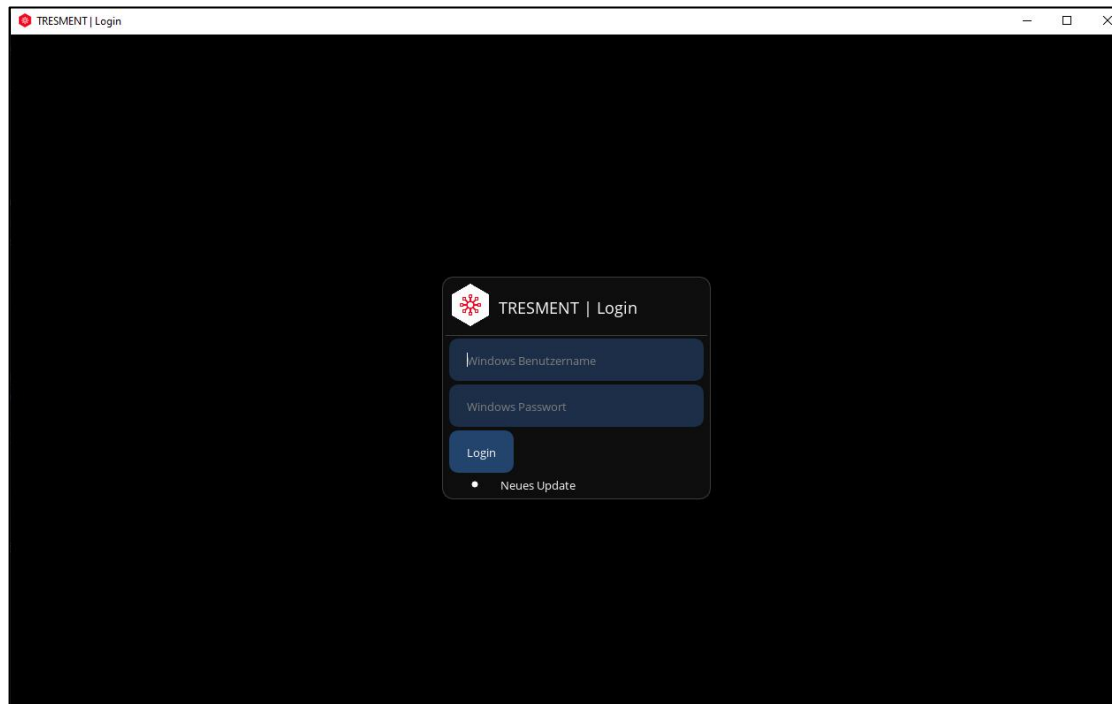
Das Benutzerhandbuch ist der Grundstein für alle Anwender für die Applikation. Hier werden alle Grundfunktionalitäten erklärt und beschrieben.

Der Vermittelte Inhalt in dem Handbuch wurde nach den Oberflächen gegliedert:

- [Login](#)
- [Home](#)
 - [Navigation:](#)
 - [Team Suche:](#)
 - [Filter:](#)
 - [Interaktion mit der Ansicht:](#)
 - [Ansicht der Ressourcen:](#)
- [Konfiguration](#)
 - [Team Erstellen:](#)
 - [Team Anpassen / Löschen](#)
 - [Applikation Anpassen](#)



Login

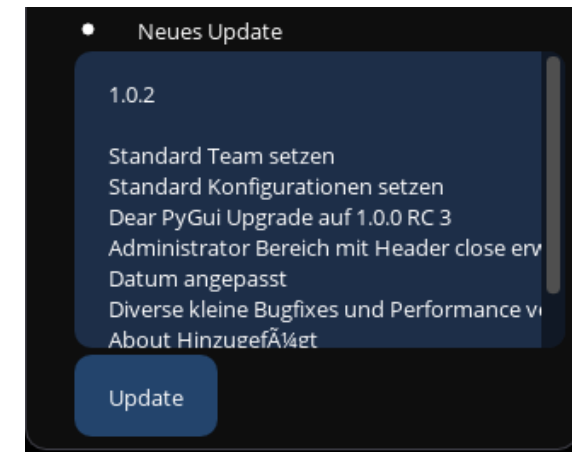


- Das Login Fenster besteht aus 4 Elementen:
Anmeldedaten Benutzername
- Anmeldedaten Passwort
- Login Taste oder bestätigen mit Enter
- Neues Update herunterladen, wenn dies verfügbar ist.

Damit die Sicherheit gewährleistet ist, wird das Log-In via Windows Benutzername gemacht (Beispiel: mstuder).

Dass auch jeder immer auf dem neusten Stand bleibt, wird jeder Benutzer informiert, sobald ein neues Release verfügbar ist.

Sobald man die Update Taste betätigt, wird die neueste Version heruntergeladen und installiert.



Home

- Das Home Fenster besteht aus den meisten Elementen und wird somit unterteilt in:
- Die Navigation
- Die Team Suche
- Die Filtermöglichkeiten
- Die Interaktionen mit der Ansicht
- Die Ansicht der Ressourcen

Navigation:

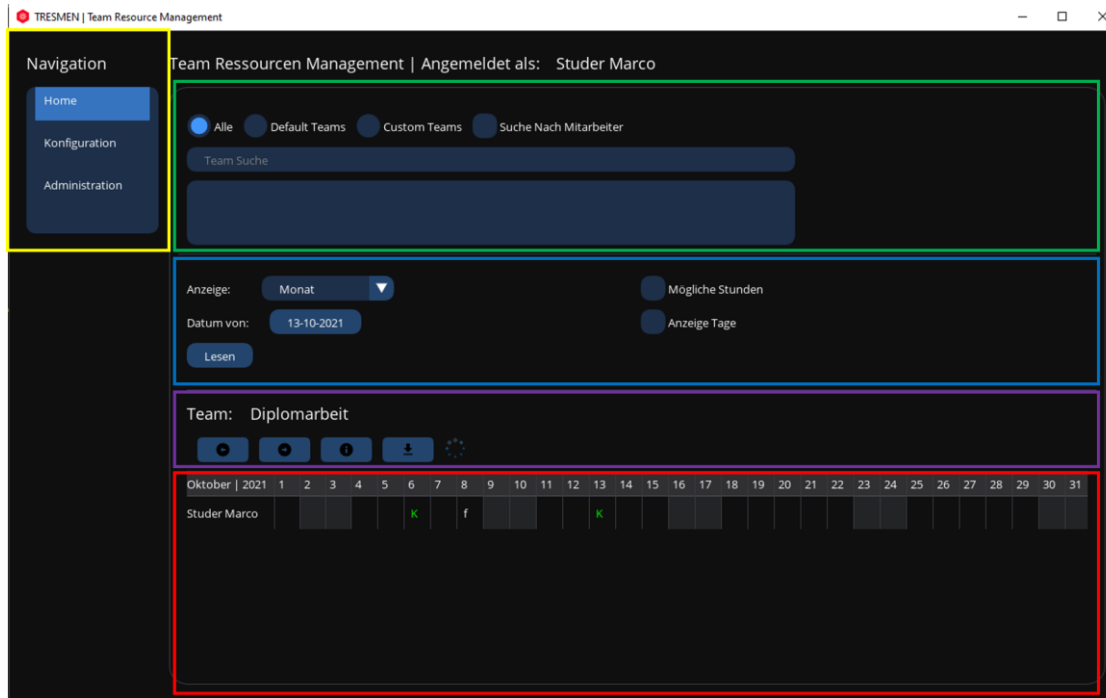
Die Navigation dient dazu zwischen den Bereichen hin und her zu springen.

Im Home Bereich findet ihr die Hauptanzeige und unter Konfigurationen können Teams erstellen, angepasst oder gar gelöscht werden. Zudem besteht die Möglichkeit einige Anpassungen an der Applikation selbst zu machen, diese werden zu einem Späteren Zeitpunkt erklärt.



Team Suche:

- Die Team Suche wird benötigt, falls man die Ressourcen von einem anderen Team einsehen möchte. Um alle Teams anzuzeigen, kann der * benutzt werden. Hierbei gibt es die Abstufung zwischen:
 - Alle
 - Alle Teams werden angezeigt, auch die selbst erstellten Teams
 - Default Teams
 - Es werden nur Teams angezeigt, welche vom Datenstamm (Navision) angelegt wurden
 - Custom Teams
 - Es werden nur Teams angezeigt, welche von einem Benutzer angelegt wurden
 - Suche nach Mitarbeiter
 - Wenn man nicht weiss in welchem Team sich ein Mitarbeiter befindet, kann man diesen über diese Suche herausfinden.



Filter:

- Die Filter welche gesetzt und verändert werden können sind folgende:
- Anzeige
 - Anzeige eines einzelnen Monats; einer Woche; oder eines ganzen Jahres
- Datum von
 - Datumsauswahl welchen Monat, Woche, Jahr man einsehen möchte
- Mögliche Stunden
 - Anzeige in der Ressourcen Tabelle wie viele Stunden ein Mitarbeiter zur Verfügung hat (ACHTUNG: Hierbei sind Mitarbeiter welche nicht 100% Arbeiten nicht korrekt)
- Anzeige Tage
 - Erweitert die Anzeige in der Ressourcen Tabelle um die Wochentage
- Lesen
 - Übernimmt die Filter und berechnet die Ressourcen Tabelle neu

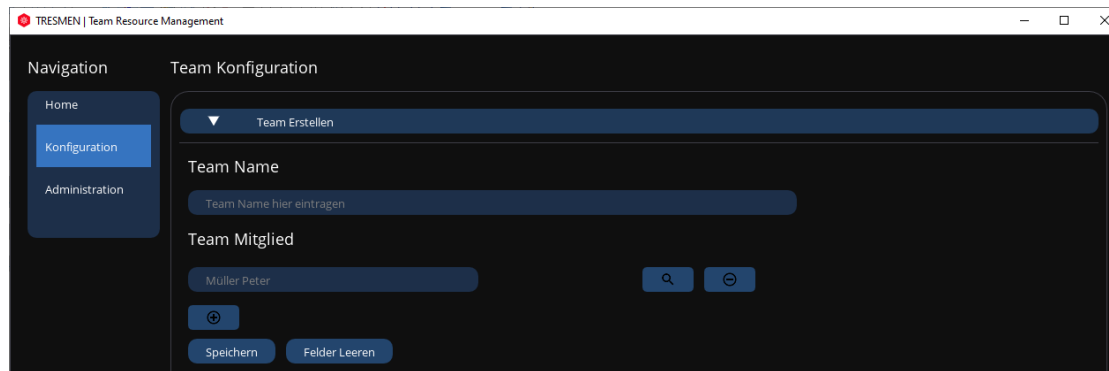


Konfiguration



Das Konfigurationsfenster ermöglicht es dem Anwender eigene Teams zu erstellen, diese zu verwalten und wenn nötig diese auch zu löschen.

Neben den Konfigurationen für die Teams ist es auch möglich diverse Anpassungen an der Applikation zu machen.



Team Erstellen:

Um ein Team zu erstellen, wird nicht viel benötigt.

- **Team Name**

Der Team Name muss individuell eingegeben werden und darf nicht bereits vorhanden sein. Dies wird durch eine grüne (Team Name OK) oder rote (Team Name bereits vorhanden) Farbe dargestellt

- **Team Mitglied**

Es gibt 2 verschiedene Wege ein Team Mitglied hinzuzufügen:



- Der Komplette Name kann in dem Feld eingetragen werden
- Suchfunktion (Lupen Icon) verwenden, welches nach einem Benutzer suchen lässt

Um weitere Mitglieder hinzuzufügen, kann entweder das "Plus" Icon gedrückt werden oder man bestätigt mit der "Enter Taste".

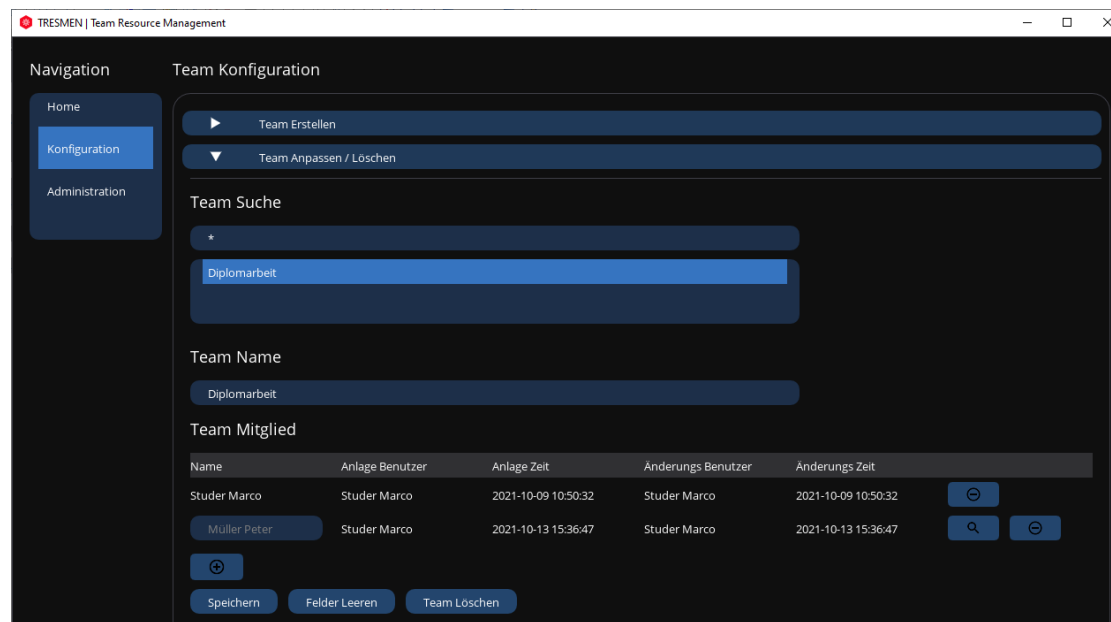
Wenn ein Mitglied zu viel hinzugefügt wurde, kann dieser mit dem "Minus" Icon wieder entfernt werden.

- **Speichern**

Das Erstellte Team wird gespeichert und ist nun für alle anderen Benutzer verfügbar

- **Felder Leeren**

Alle Felder werden geleert und es wird nichts gespeichert



Team Anpassen / Löschen

Um ein Bestehendes Team anzupassen oder dies gar zu löschen, muss dies zuerst ausgewählt werden in der Team Suche. Um alle Teams anzuzeigen, kann * benutzt werden.

- **Team Name**

Der Team Name kann angepasst werden, fällt jedoch unter dieselbe Prüfung wie beim Erstellen eines Teams. Dies wird durch eine grüne (Team Name OK) oder rote (Team Name bereits vorhanden) Farbe dargestellt

- **Team Mitglied**

Team Mitglieder können hinzugefügt oder gelöscht werden. Das hinzufügen und löschen funktioniert gleich wie beim Team Erstellen.

- Hinzufügen mit der "Enter" Taste oder dem "Plus" Icon
- Entfernen mit dem "Minus" Icon

- **Speichern**

Das Team wird mit den Änderungen gespeichert und ist für alle anderen Benutzer



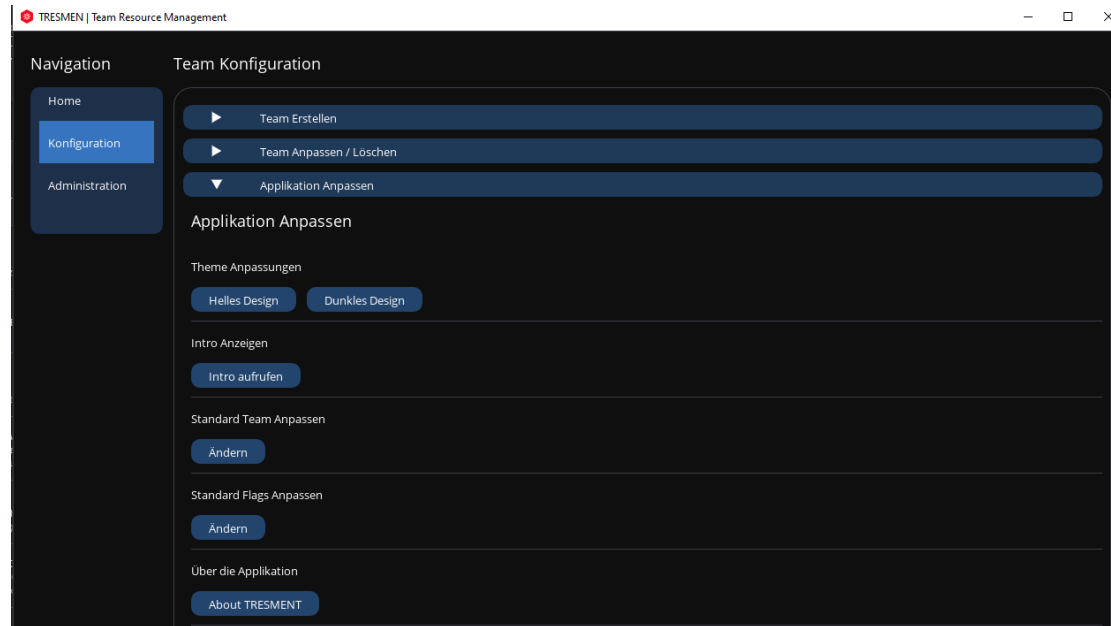
verfügbar

- **Felder Leeren**

Alle Felder werden geleert und es wird nichts gespeichert

- **Team Löschen**

Das Team wird gelöscht und ist ab diesem Zeitpunkt nicht mehr für andere Benutzer verfügbar.



Applikation Anpassen

Das Applikation Anpassen Untermenü bietet einige Anpassungen an der Applikation selbst oder an der Handhabung der Applikation.

- **Theme Anpassungen**

Das Design kann zwischen Dunkel & Hell umgestellt werden und wird für das nächste Aufstarten gespeichert

- **Intro Anzeigen**

Das Einführungsfenster kann nochmals aufgerufen werden, um die Funktionalitäten nochmals zu prüfen

- **Standard Team Anpassen**

Das Standard Team, welches beim aufstarten der Anwendung angezeigt wird, wird hier geändert

- **Standard Flags Anpassen**

Die Flags von dem Hauptfenster "Mögliche Stunden", "Anzeige Tage" können von angepasst werden, dass diese beim aufstarten aktiv sind oder nicht.



- **Über die Applikation**

Infofenster über die Anwendung, Beteiligte, Libraries und direkter Zugang zu den Handbüchern.



3 Entwicklerhandbuch

Willkommen beim Handbuch für die Weiterentwicklung der Anwendung TRESMENT.

Die Anwendung und die API wurde komplett in Python entwickelt, um die Weiterentwicklung in einer Sprache zu halten.

Für die Entwicklung wurde die IDE Pycharm verwendet, welche Gratis unter folgendem Link heruntergeladen werden kann : <https://www.jetbrains.com/de-de/pycharm/download/#section=windows>

Die Entwicklung kann jedoch in jedem beliebigen Editor fortgefahren werden.

3.1 Voraussetzungen

Um die Anwendung weiterzuentwickeln wird Python 3.9 benötigt.

3.2 Benutze Libraries API

Library	Beschreibung der Library
FastAPI https://fastapi.tiangolo.com/	FastAPI ist ein modernes und schnelles Web-Framework, um APIs zu erstellen;
Uvicorn https://www.uvicorn.org/	Uvicorn ist eine schnelle und einfache ASGI Server Implementation. Diese wird benötigt, um die API am Laufen zu halten;
SQLAlchemy https://www.sqlalchemy.org/	SQLAlchemy ist ein SQL-Werkzeugkoffer;
Zeep https://docs.python-zeep.org/en/master/	Zeep ist ein hervorragender SOAP Client, welcher für die Anbindung an Microsoft Navision benötigt wird;



Requests https://docs.python-requests.org/en/latest/	Requests ist in Python die erste Auswahl für einfache «http/s» zugriffe;
---	--

3.3 Benutze Libraries GUI

Library	Beschreibung der Library
Dear PyGui https://dearpygui.readthedocs.io	Dear PyGui ist ein einfach zu bedienendes, dynamisches, GPU-beschleunigtes, plattformübergreifendes grafisches Benutzeroberflächen-Toolkit (GUI) für Python.
ldap https://www.python-ldap.org/en/python-ldap-3.3.0/	Python ldap ermöglicht es einen Zugriff auf den ldap Server zu machen, um Daten zu prüfen oder diese auch zu verändern.
Tkinter https://docs.python.org/3/library/tkinter.html	Tkinter ist die Standard Python GUI Bibliothek. Diese wird verwendet, um den Explorer Zugriff zu machen.
Pandas https://pandas.pydata.org/	Pandas ist eine schnelles und leistungsstarkes Datenanalyse und -Manipulation Bibliothek.
Requests https://docs.python-requests.org/en/latest/	Requests ist in Python die erste Auswahl für einfache «http/s» zugriffe.
SQLAlchemy https://www.sqlalchemy.org/	SQLAlchemy ist ein SQL-Werkzeugkoffer

3.4 API-Handbuch

Das API-Handbuch beschreibt die API und wie diese weiterentwickelt werden kann.



Die API wurde mittels Python erstellt und benötigt die Libraries welche im Entwicklerhandbuch abgebildet sind.

3.4.1 Serverzugang

Die API läuft auf einem Server in unserem Netzwerk (*****).

Der Zugang zu diesem Server kann im Helpdesk angefragt werden.

3.4.2 Start der API

Die API kann auf zwei Varianten gestartet werden:

- python3 main.py
- ./tresment.sh

3.4.3 Projektstruktur

Die Projektstruktur wird von Top-Down erklärt.

```
TRESMENT-API/  
├── api  
│   ├── crud  
│   └── routers  
├── interface  
│   ├── database  
│   │   ├── handler  
│   │   └── shared  
│   └── nav  
└── shared
```

3.4.3.1 Toplevel

Auf der Toplevel Ebene (TRESMENT-API/) befindet sich der Startpunkt der API.

- main.py



- Startet die API und die Threads für die Datenbank
- requirements.txt
 - Beschreibt die benötigten Libraries für die API
- TRESMENT-API.pid
 - Prozess PID File für das Starten und Stoppen der API
- TRESMENT.db
 - Datenbank der API
- tresment-restart.sh
 - CSH Script zum neustarten der API
- tresment.sh
 - CSH Script zum Beenden oder starten der API

3.4.3.2 API

Unter dem Ordner API befinden sich die SQL Statements für die API und auch die API-Endpunkte, wie auch die Abhängigkeiten.

- crud
 - req_crud.py
 - Enthält alle SQL-Anforderungen welche von der API angefordert werden.
- routers
 - get_requests.py
 - Enthält alle GET Endpunkte der API
- dependencies.py



- Beinhaltet die Benutzerabfrage für die API, welche bei jedem Mal aufgerufen wird.

3.4.3.3 INTERFACE

Im Interface werden zwei Punkte behandelt, die Schnittstelle zu Navision und die interne Schnittstelle zu der Datenbank.

- config.py
 - Enthält Standard Parameter, wenn die Datenbank mal gelöscht wird
- database
 - Beinhaltet die komplette Datenbankabwicklung
 - Datenbank erstellen
 - Modells der Datenbank
 - Schema der Datenbank
 - Threads, um die Datenbank zu befüllen oder löschen
- nav
 - interface.py
 - Wickelt die Schnittstelle zum Navision ab

3.4.3.4 SHARED

Im Shared Verzeichnis wird bisher nur die Logging Funktion aufgerufen, welche ebenfalls auf dem Server Log Files generiert.



3.4.4 Tipps und Tricks

3.4.4.1 Datenbank im Debugmodus starten

Die Datenbank kann im Debugmodus gestartet werden, so dass diese alles loggt, was passiert, hierfür muss im File database.py unter (interface/database/shared) die Markierte Zeile auskommentiert werden.:

Codeblock 1 database.py

```
engine = create_engine(  
    SQLALCHEMY_DATABASE_URL, connect_args={"check_same_thread": False},  
    #echo=True #Für Debugg zwecke einschalten  
)
```

3.4.4.2 Entwicklung in eigener Umgebung

Um die API-Lokal zu entwickelt, muss der Source-Code vom Server heruntergeladen werden und anschliessend muss die IP im File main.py angepasst werden:

Codeblock 2 main.py

```
VON:  
if __name__ == "__main__":  
    models.Base.metadata.create_all(bind=engine)  
    uvicorn.run("main:app", host="*****", port=****, log_level="info",  
callback_notify=init())  
AUF:  
if __name__ == "__main__":  
    models.Base.metadata.create_all(bind=engine)  
    uvicorn.run("main:app", host="127.0.0.1", port=5001, log_level="info", call-  
back_notify=init())
```

mit dieser Anpassung läuft die API nun lokal auf dem Computer.



3.5 GUI Handbuch

Das GUI Handbuch beschreibt das GUI und wie diese weiterentwickelt werden kann.

Das GUI wurde mittels Python erstellt und benötigt die Libraries welche im Entwicklerhandbuch abgebildet sind.

3.5.1 Source-Code

Der Source-Code ist aktuell unter \\MU-SEFILE1.ssi-ch.local\Public\Software\TRESMENT\app_data\source\app abgelegt.

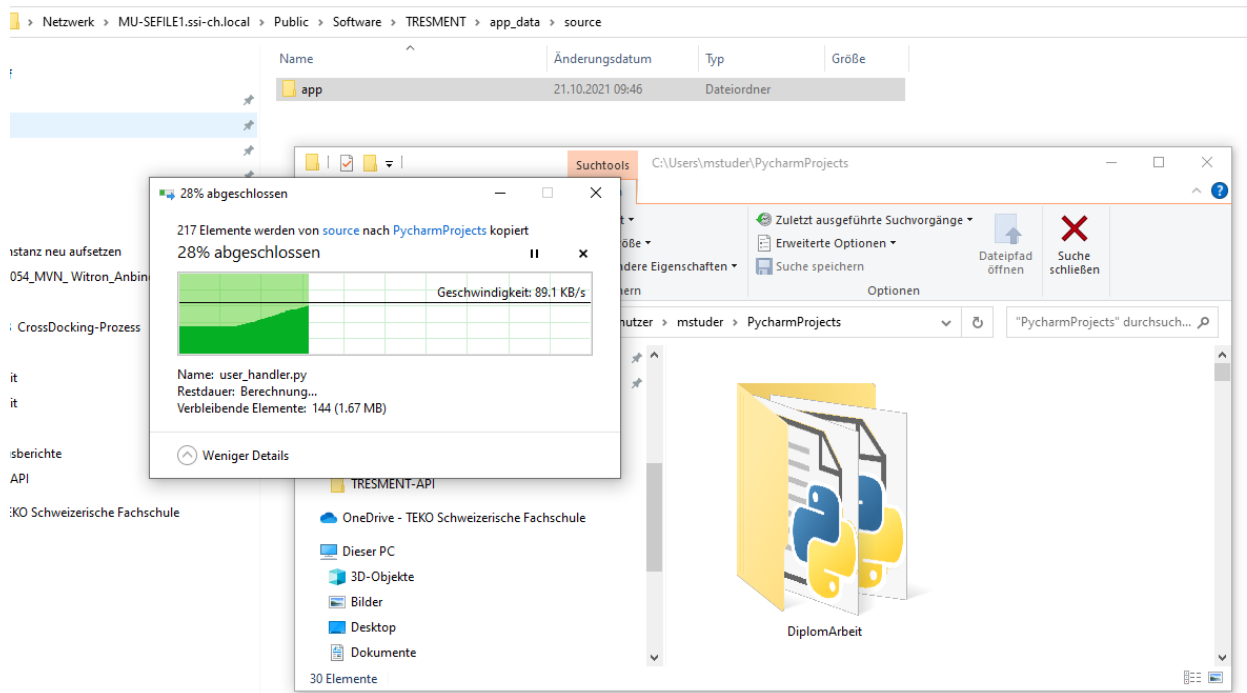
In einem späteren Zeitpunkt wird der Source in unser internes .git Repository übernommen.

3.5.2 Auswahl der IDE

Als IDE kann ich Pycharm empfehlen, welche unkompliziert zu bedienen ist.

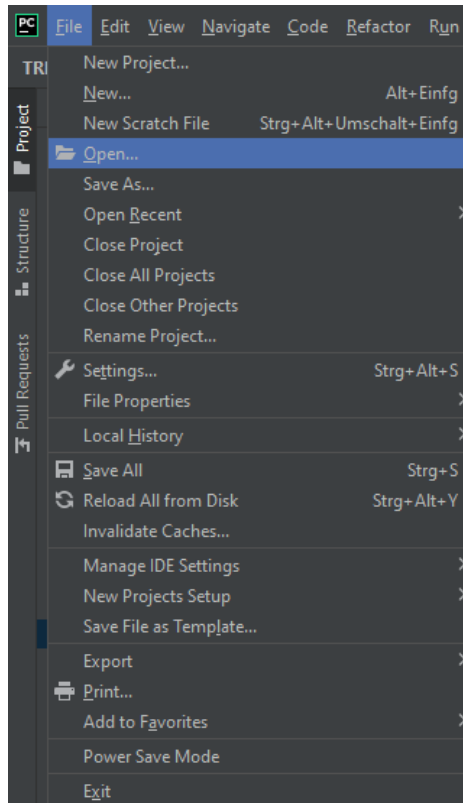
Die Anwendung kann jedoch auf mit einem normalen Editor weiterentwickelt oder bearbeitet werden.

Solange der Code noch nicht im .git ist, sollte der Source-Code lokal kopiert werden.

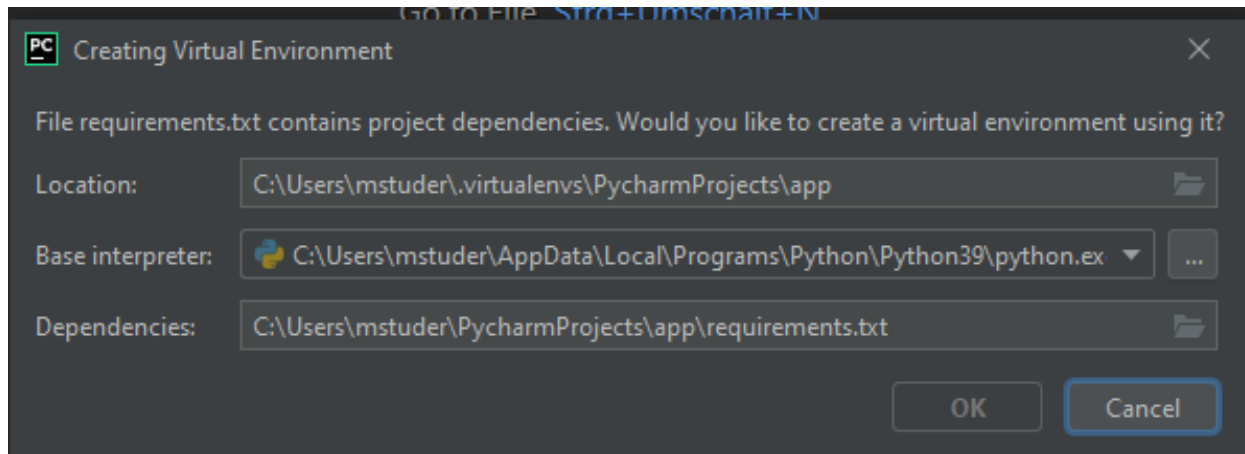


3.5.2.1 Arbeiten mit PyCharm

Um das Projekt in Pycharm zu öffnen müssen folgende Schritte befolgt werden:



-
- Projektordner öffnen

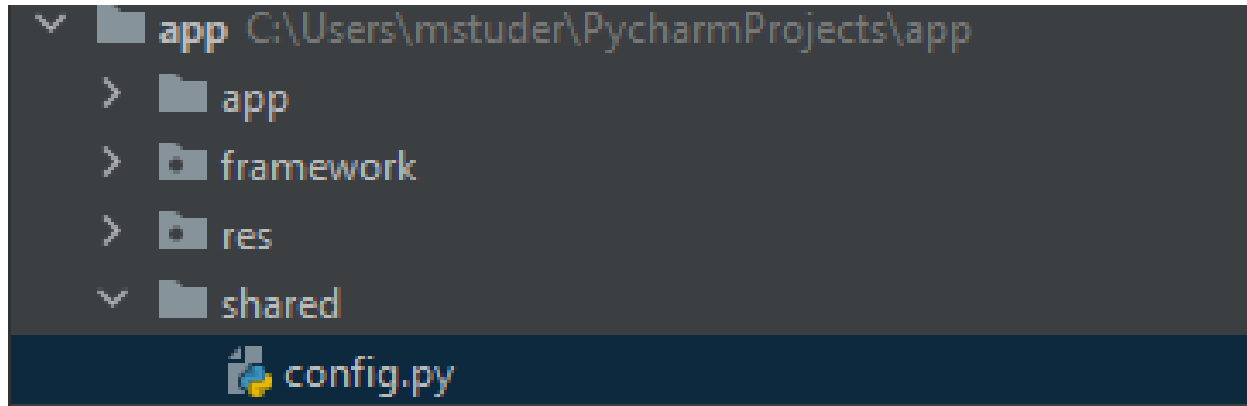


- Virtual Environment anlegen lassen von PyCharm.
- **Wichtig ist hierbei, dass der richtige Python Interpreter ausgewählt wird (PYTHON 3.9+)**
Anschliessend werden alle wichtigen Pakete direkt in das Virtual Environment installiert, so dass die Applikation anschliessend gestartet werden kann.

Die Library ldap muss separat installiert werden, mit folgendem Befehl in dem Terminal:

- `pip install .\python_ldap-3.3.1-cp39-cp39-win_amd64.whl`

Um die Anwendung starten zu können, muss zuerst noch eine Konfiguration vorgenommen werden:



Codeblock 3 Anpassen API User

```
""" -----  
-----  
* PROJECT:  TRESMENT  
* PACKAGE:  SHARED  
* FILE:     config.py  
* CONTENT:  Stores the API Credentials and the API Call  
* STATIC:  
-----  
----- """  
  
NAV_USER = ""  
NAV_PASSWD = ""
```

Der Benutzer und das Passwort können im Helpdesk angefragt werden.

3.5.3 Start der Applikation

Um die Anwendung zu starten, kann man dies entweder im File main.py via Rechtsklick:



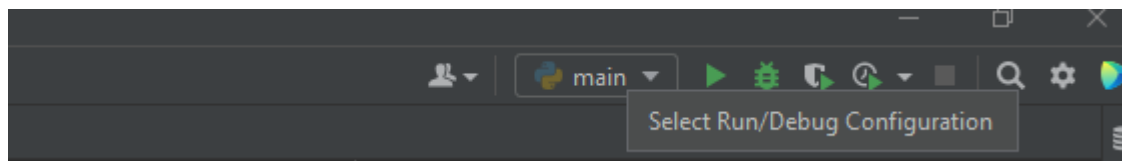
```
def init():
    """
    Initial function to build
    """
    models.Base.metadata.create_all(bind=engine)
    memory.Base.metadata.create_all(bind=db_mem.engine)
    init_gui(__version__)

if __name__ == '__main__':
    __version__ = '1.0.2'
    __author__ = 'Marco Studer'
    init()

init()

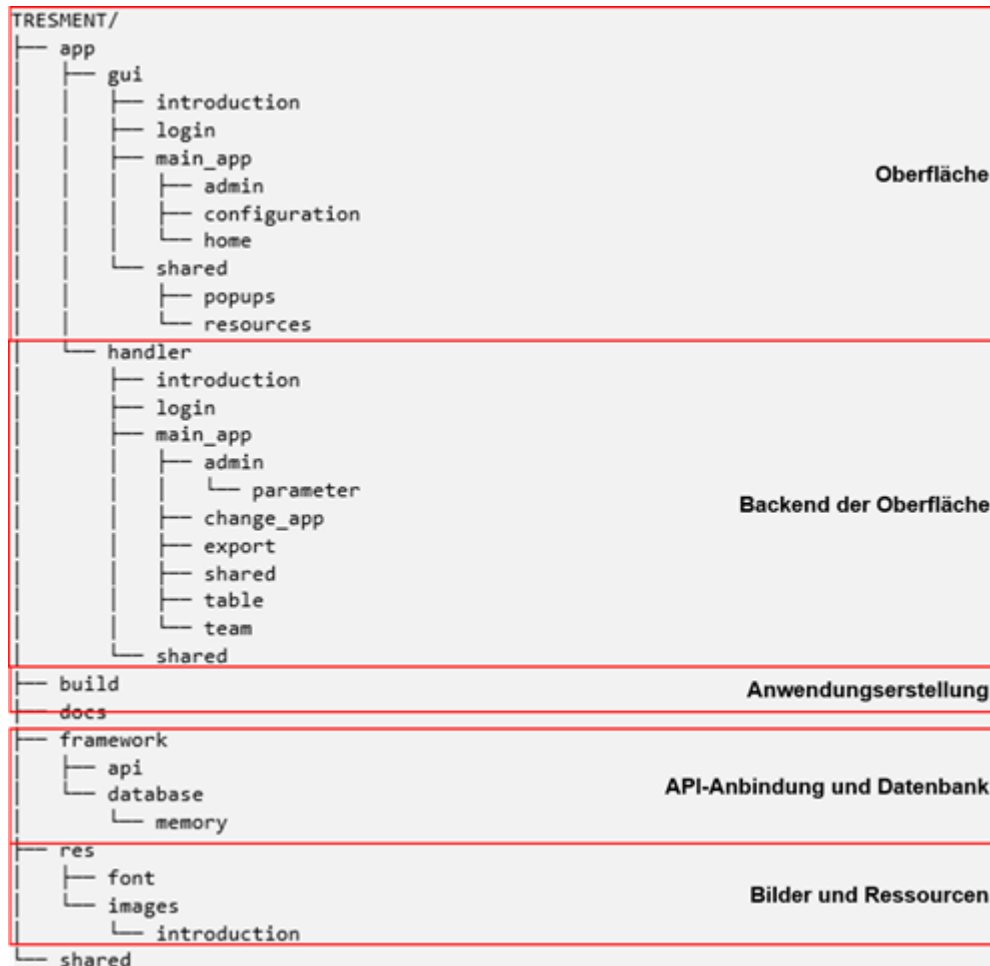
python.exe C:/Users/mstuder/PycharmProjects/app/main.py
```

Oder über den Start Knopf ganz oben rechts:





3.5.4 Projektstruktur



3.5.4.1 Toplevel

Auf der Toplevel Ebene (app/) befindet sich der Startpunkt der API.



- main.py
 - Startet die Applikation
- requirements.txt
 - Beschreibt die benötigten Libraries für die API
- TRM.db
 - Datenbank der API

3.5.4.2 Die Oberfläche

Unter dem Ordner app/gui befinden sich alle Oberflächenfiles, welche für die statische Erzeugung der Oberfläche dient.

Diese ist aufgeteilt in die verschiedenen Bereiche der Oberfläche:

- Introduction
 - Einführung Fenster, welches nach dem Login eingeblendet wird
- login
 - Login-Fenster welches beim Aufstarten der Anwendung angezeigt wird
- main_app
 - admin
 - Administrator Bereich
 - configuration
 - Team und Applikation Konfiguration Bereich



- home
 - Hauptfenster mit der Navigation
- shared
 - popups
 - Die Verschiedenen Popupmenüs
 - resources
 - Schriftarten, Bilder und Handler welche global genutzt werden

3.5.4.3 Das Backend der Oberfläche

Unter dem Ordner app/handler befinden sich alle Hintergrundprozesse oder Funktionen der Oberfläche, welche für Aktionen oder dynamische Bereitstellungen dienen.

- introduction
 - Konstanten und Bilder für den Einführungsdialog
- login
 - Alle Login Funktionen und den Switch zum Hauptfenster
- main_app
 - Alle Funktionen welche in der Applikation benötigt werden

3.5.4.4 API-Anbindung und Datenbank

Die API-Anbindung, sowie die Datenbanken sind im Verzeichnis Framework

Die API-Anbindung wird in dem Unterordner api abgehandelt

Die Datenbank Erstellung der normalen Datenbank ist im Unterordner database zu finden, wobei die "in Memory" Datenbank unter dem Verzeichnis database/memory zu finden ist.



Codeblock 5 Application Build

```
Build vorgang starten ? J/N    $ J
C:\Users\mstuder\PycharmProjects\app\build
Nuitka-Options:INFO: Used command line options: --mingw64 --onefile --windows-
icon-from-ico=../res/images/logo_small_icon_only_inverted.ico --follow-imports --
plugin-enable=numpy --enable-plugin=anti-bloat --noinclude-pytest-mode=nofollow --
noinclude-setuptools-mode=nofollow --plugin-enable=pylint-warnings --plugin-
enable=tk-inter --windows-disable-console ../main.py
Nuitka-Options:INFO: Following all imports is the default for onefile mode and
need not be specified.
Nuitka:INFO: Starting Python compilation with Nuitka '0.6.16.4' on Python '3.9'
commercial None.
Nuitka:INFO: Completed Python level compilation and optimization.
Nuitka:INFO: Generating source code for C backend compiler.
Nuitka:INFO: Running data composer tool for optimal constant value handling.
Nuitka:INFO: Running C level backend compilation via Scons.
Nuitka-Scons:INFO: Mismatch between Python binary ('C:\\Users\\mstuder\\.virtua-
lenvs\\PycharmProjects\\app\\Scripts\\python.exe' -> 'pei-x86-64') and C compiler
('C:\\Users\\mstuder\\.nimble\\bin\\gcc.exe' -> None) arches, ignored!
Nuitka-Scons:INFO: Backend C compiler: C:\Users\mstuder\AppData\Local\Nuitka\Nu-
itka\gcc\x86_64\10.2.0-11.0.0-8.0.0-r5\mingw64\bin\gcc.exe (gcc.exe).
```

Der erste Build der Anwendung dauert ca. 40 Minuten da die ganze Applikation mit NUITKA zurück nach C kompiliert wird.

Beim nächsten Build muss dieser Vorgang nicht wiederholt werden.



Codeblock 6 Application Build

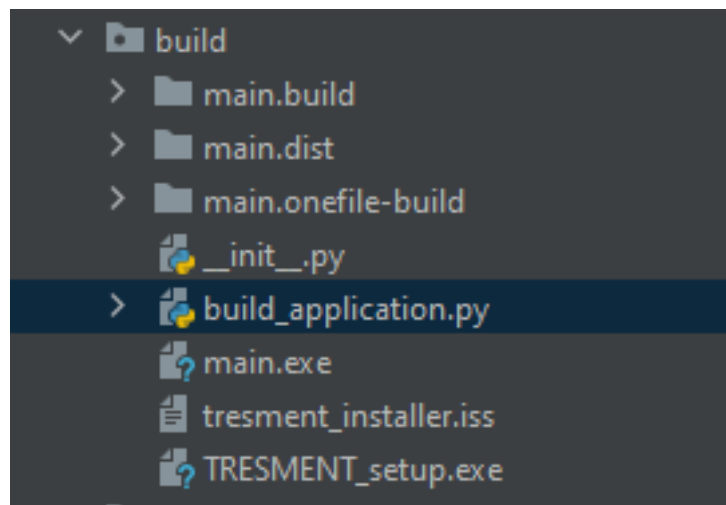
```
C:\Users\mstuder\VIRTUAL~3\PYCHAR~1\app\lib\SITE-P~1\nuitka\build\include\nu-
itka/tracing.h:40: warning: "NUITKA_PRINT_TRACE" redefined
  40 | #define NUITKA_PRINT_TRACE(value)
      |
static_src\OnefileBootstrap.c:61: note: this is the location of the previous defi-
nition
  61 | #define NUITKA_PRINT_TRACE(arg)
      |
Nuitka-Scons:INFO: Compiled 1 C files using ccache.
Nuitka-Scons:INFO: Cached C files (using ccache) with result 'cache miss': 1
Nuitka-Onefile:INFO: Keeping onefile build directory 'main.onefile-build'.
Nuitka-Postprocessing:INFO: Adding 6 icon(s) from icon file
'../res/images/logo_small_icon_only_inverted.ico'.
Nuitka:INFO: Keeping dist folder 'main.dist' for inspection, no need to use it.
Nuitka:INFO: Keeping build directory 'main.build'.
Nuitka:INFO: Successfully created 'main.exe'.
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!
Build Abgeschlossen !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!
Kopiere alle Bilder / etc. in den dist. folder...
*res* Verzeichnis kopiert
-----
-----
Start Inno Setup ? J/N    $
```

Mit dem bestätigen, dass ein Inno Setup gebildet wird, wird die Anwendung Inno Setup Compiler geöffnet und die Anwendung wird in eine Setup.exe umgewandelt



```
Untitled - Inno Setup Compiler 6.1.2 [Compiling]
File Edit View Build Run Tools Help
|
Preprocessing
  Reading file: C:\Program Files (x86)\Inno Setup 6\USPPBulltins.iss
  Parsing [Setup] section, line 18
  Parsing [Setup] section, line 19
  Parsing [Setup] section, line 20
  Parsing [Setup] section, line 22
  Parsing [Setup] section, line 23
  Parsing [Setup] section, line 24
  Parsing [Setup] section, line 25
  Parsing [Setup] section, line 28
  Parsing [Setup] section, line 29
  Parsing [Setup] section, line 30
  Parsing [Setup] section, line 31
  Parsing [Setup] section, line 32
  Parsing [Setup] section, line 33
  Parsing [Setup] section, line 34
  Reading file (WizardImageFile)
    Reading file: C:\Program Files (x86)\Inno Setup 6\WIZMODERNIMAGE.BMP
  Reading file (WizardSmallImageFile)
    Reading file: C:\Program Files (x86)\Inno Setup 6\WIZMODERNSMALLIMAGE.BMP
  Preparing Setup program executable
    Updating icons (SETUP.E32)
    Updating version info (SETUP.E32)
  Determining language code pages
  Parsing [Language] section, line 37
    Reading file: C:\Program Files (x86)\Inno Setup 6\Languages\German.isl
  Messages in script file
  Reading default messages from Default.isl
  Parsing [Language] section, line 37
    Reading file: C:\Program Files (x86)\Inno Setup 6\Languages\German.isl
  Parsing [LangOptions], [Messages], and [CustomMessages] sections
  Messages in script file
```

Die Installation .exe TRESMENT_setup.exe liegt jetzt bereit in dem build Verzeichnis.



Damit der Installer auch direkt für alle verfügbar ist, kann dieser Automatisch auf den Server kopiert werden:



Codeblock 7 Application Build

```
Installer auf den Server Kopieren ? J/N    $J
-----
-----
Copy Old Installer to the Archive
Copy the new Installer to the Server
-----
-----
-----
Build Vorgang abgeschlossen
-----
-----
-----
-----
Process finished with exit code 0
```



14.3 Projektstatusbericht

Projekt: TRESMENT – Team Ressource Management

Stautsbericht: 13.09 – 17.09.2021

Projektleiter Marco Studer	Projektziele Eine neue Anwendung für das «Team Ressourcen Management» ist erstellt	Verteiler • Benjamin Bäni • Marc Mangold
--------------------------------------	--	---

Gesamt- beurteilung	Projektverlauf 	Projektklima 	Termine 	Risiken 	Ressourcen
Tendenz					

Aktueller Projektstand <ul style="list-style-type: none"> • Projektinitialisierung abgeschlossen • Projektplanung abgeschlossen • Projektrealisierung: <ul style="list-style-type: none"> • Nutzwertanalyse durchgeführt • Entscheid gefällt eine Desktopanwendung zu erstellen. • Backend-API Design und Realisierung begonnen 	Was läuft gut? <ul style="list-style-type: none"> • Dokumentationsstruktur aufgebaut und die ersten wichtigen Elemente hinzugefügt. • Start Backend-API begonnen Was läuft nicht gut? <ul style="list-style-type: none"> • Hardware defekt festgestellt, somit ist die Produktivität für 1 Tag gesunken.
---	---

Geplante nächste Schritte / getroffene Massnahmen <ul style="list-style-type: none"> • Design und Dokumentation der API fertigstellen. • API Realisierung fertigstellen.

Projekt-Statusbericht; Stefan Thöni, Josef Räber



Projekt: TRESMENT – Team Ressource Management

Stautsbericht: 17.09 – 24.09.2021

Projektleiter Marco Studer	Projektziele Eine neue Anwendung für das «Team Ressourcen Management» ist erstellt	Verteiler • Benjamin Bäni			
Gesamtbeurteilung	Projektverlauf 	Projektklima 	Termine 	Risiken 	Ressourcen
Tendenz					
Aktueller Projektstand <ul style="list-style-type: none">Realisierung Backend und Frontend	Was läuft gut? <ul style="list-style-type: none">Backend-API fertiggestellt in kürzerer ZeitFrontent in der Entwicklung Was läuft nicht gut? <ul style="list-style-type: none">-				
Geplante nächste Schritte / getroffene Massnahmen <ul style="list-style-type: none">Dokumentation erweiternFrontent weiterentwickeln , Dokumentation erweitern					

Projekt-Statusbericht; Stefan Thöni, Josef Räber



Projekt: TRESMENT – Team Ressource Management

Stautsbericht: 24.09 – 01.10.2021

Projektleiter Marco Studer	Projektziele Eine neue Anwendung für das «Team Ressourcen Management» ist erstellt	Verteiler • Benjamin Bäni			
Gesamtbeurteilung	Projektverlauf 	Projektklima 	Termine 	Risiken 	Ressourcen
Tendenz					
Aktueller Projektstand <ul style="list-style-type: none">• Realisierung Frontend• Dokumentation Frontend	Was läuft gut? <ul style="list-style-type: none">• Backend-API fertig dokumentiert• Dokumentationsfortschritt Frontend Was läuft nicht gut? <ul style="list-style-type: none">• Aufgrund von vielen Pendenzen bei der Arbeit sinken die Ressourcen und die Arbeiten müssen auf mehr Randzeiten / Abendzeiten verschoben werden.				
Geplante nächste Schritte / getroffene Massnahmen <ul style="list-style-type: none">• Dokumentation erweitern• Frontend weiterentwickeln , Dokumentation erweitern					

Projekt-Statusbericht; Stefan Thöni, Josef Räber



Projekt: TRESMENT – Team Ressource Management

Stautsbericht: 01.10 – 08.10.2021

Projektleiter Marco Studer	Projektziele Eine neue Anwendung für das «Team Ressourcen Management» ist erstellt	Verteiler • Benjamin Bäni
--------------------------------------	--	-------------------------------------

Gesamt-Beurteilung	Projektverlauf	Projektklima	Termine	Risiken	Ressourcen
Tendenz					

Aktueller Projektstand <ul style="list-style-type: none"> • Realisierung Frontend • Dokumentation Frontend 	Was läuft gut? <ul style="list-style-type: none"> • Frontend nahezu fertig gestellt Was läuft nicht gut? <ul style="list-style-type: none"> • Aufgrund von vielen Pendenzen bei der Arbeit sinken die Ressourcen und die Arbeiten müssen auf mehr Randzeiten / Abendzeiten verschoben werden. • Die Dokumentation von dem Frontend hängt etwas hinterher.
--	--

Geplante nächste Schritte / getroffene Massnahmen <ul style="list-style-type: none"> • Dokumentation erweitern • Frontend weiterentwickeln , Dokumentation erweitern • Go Live vorbereiten

Projekt-Statusbericht; Stefan Thöni, Josef Räber



Projekt: TRESMENT – Team Ressource Management

Stautsbericht: 08.10 – 15.10.2021

Projektleiter Marco Studer	Projektziele Eine neue Anwendung für das «Team Ressourcen Management» ist erstellt	Verteiler • Benjamin Bäni			
Gesamt- beurteilung	Projektverlauf 	Projektklima 	Termine 	Risiken 	Ressourcen
Tendenz					
Aktueller Projektstand <ul style="list-style-type: none">• Dokumentation nahezu fertiggestellt• Entwicklerhandbuch noch verfeinern• Vorbereitungen Go-Live• Korrekturlesen und layouting	Was läuft gut? <ul style="list-style-type: none">• Applikation fertig entwickelt und fertig getestet• Handbücher nahezu fertiggestellt Was läuft nicht gut? <ul style="list-style-type: none">• -				
Geplante nächste Schritte / getroffene Massnahmen <ul style="list-style-type: none">• Entwicklerhandbuch fertigstellen• Go-Live am Montag					

Projekt-Statusbericht; Stefan Thöni, Josef Räber



Projekt: TRESMENT – Team Ressource Management

Stautsbericht: 15.10 – 22.10.2021

Projektleiter Marco Studer	Projektziele Eine neue Anwendung für das «Team Ressourcen Management» ist erstellt	Verteiler • Benjamin Bäni			
Gesamtbeurteilung	Projektverlauf	Projektklima	Termine	Risiken	Ressourcen
Tendenz					
Aktueller Projektstand <ul style="list-style-type: none">• Dokumentation fertiggestellt• Handbücher fertiggestellt• Go-Live durchgeführt• Korrekturlesen erledigt• Formatierung erledigt	Was läuft gut? <ul style="list-style-type: none">• Gutes Feedback beim Go-Live erhalten• Dokumentation fertiggestellt Was läuft nicht gut?				
Geplante nächste Schritte / getroffene Massnahmen <ul style="list-style-type: none">• Schlusswort zu Ende schreiben• Nochmaliges durchlesen und Prüfung der Formatierung• Abgabe der Diplomarbeit					

Projekt-Statusbericht; Stefan Thöni, Josef Räber



Kontakt:

Marco Studer

Meisenweg 25

4528 Zuchwil

E-Mail: studer.marco@hotmail.com / marco.studer@ssi-schaefer.com

