

V1: “Mein Dashboard” Seite mit Infos befüllen

IPA-Daten

Firma	WebGate Consulting AG
Abteilung	IBM / MS-Azure Team
Autor	Nathan Fahrni
Ausgabedatum	08.12.2022
Projektvorgehen	Sequenziell mit der Projektvorgehensmethode XY
Version	X 0.5

Beteiligter Personenkreis

In der Genehmigung	
Valid-Experte	Barbara Hegnauer
In der Durchführung	
Verantwortliche Fachkraft	Mirko Eberlein
Hauptexperte	Mirko Eberlein
Zweit Experte	Andrin Seifert
Zur Information, Kenntnis	
Berufsbilder	Barbara Hegnauer

Änderungskontrolle, Prüfung, Genehmigung

Version	Datum	Name	Beschreibung
Vorlage	2017-10-10	A.Mueller	Dokumentvorlage QV2018, Version V1.0
X0.2	2022-12-05	N.Fahrni	Vorbereitung Dok Probe IPA
X0.3	2022-12-06	N.Fahrni	Dokumentation Projektmethode & Testkonzepte
X0.4	2022-12-07	N.Fahrni	Realisierung dokumentiert
X0.5	2022-12-08	N.Fahrni	Kleinere anpassungen

Verwendete Abkürzungen

Abkürzung	Bedeutung
IPA	Individuelle praktische Arbeit
OdA	Organisation der Arbeitswelt
QV	Qualifikationsverfahren
VP	Verantwortliche Fachkraft
VZ	Verzeichnis
VSC	Visual Studio Code

Inhaltsverzeichnis

Inhaltsverzeichnis.....	2
Abbildungsverzeichnis.....	4
Tabellenverzeichnis.....	4
1. Kurzfassung des IPA-Berichtes	5
1.1 Ausgangssituation	5
1.2 Umsetzung	5
Teil 1: Ablauf Organisation und Umfeld	6
2. Aufgabenstellung	7
2.1 Titel der Arbeit	7
2.2 Thematik.....	7
2.3 Ausgangslage.....	7
2.4 Detaillierte Aufgabenstellung	7
2.5 Mittel und Methoden inklusive Projektmethode	7
2.5.1 Projektmethode: Wasserfall	7
2.5.2 Mittel.....	8
2.6 Vorkenntnisse.....	8
2.7 Vorarbeiten.....	8
2.8 Arbeiten in den letzten 6 Monaten	8
3. Detailliertes Projektvorgehen	9
3.1.1 Projektmethode: Wasserfall	9
3.1.2 Phasen:	9
3.2 Projektorganisation	10
3.2.1 Projektrollen	11
4. Zeitplan	12
5. Organisation der IPA.....	13
5.1 Datensicherung der IPA.....	13
6. Arbeitsjournal	14
7. Abschlussbericht	20
7.1 Vergleich Ist/Soll.....	20
7.2 Mittelbedarf.....	20
7.3 Realisierungsbericht	20
7.4 Testbericht.....	20
7.5 Fazit zum IPA (Projekt)	20
7.6 Persönliches Fazit	20
7.7 Schlussreflexion	21
8. Unterschriften und Abnahmeprotokoll	22
Teil 2: Projektdokumentation	23
9. Initialisierung.....	24
9.1 Studie; Ist-Zustand.....	24
9.2 Persönliche Vorgehensziele	24
9.3 Projektziele	24

9.4	Anforderungen.....	25
9.4.1	Funktionale Anforderungen.....	25
9.4.2	Nicht funktionale Anforderungen.....	25
9.4.3	Variantenentscheid.....	25
10.	Konzept.....	26
10.1	Mockup.....	26
10.2	Testkonzept.....	27
10.2.1	Testziele.....	27
10.2.2	Testmethoden und Testfälle	27
11.	Realisierung	29
11.1	Allgemeines Layout vom Dashboard	29
11.2	Score comparison Components.....	29
11.3	Open Assessment	31
11.4	Testprotokoll.....	32
11.4.1	Testfall.....	32
11.4.2	Testergebnis.....	33
11.4.3	Testwürdigung	34
11.4.4	Testfazit.....	34
11.5	Einführung vorbereiten	35
12.	Einführung.....	36
13.	Literatur und Quellenverzeichnis.....	37
14.	Glossar	38
15.	Anhang	39

Abbildungsverzeichnis

Abbildung 1: WebGate Logo	6
Abbildung 2: Wasserfallmethode Diagramm	9
Abbildung 3, Demo Hermes 5.1 IPA	9
Abbildung 4: Projektorganisation.....	10
Abbildung 5: Dokumentationsablage via Onedrive	13
Abbildung 6: Dokumentationsablage Git	13
Abbildung 7: Mockup	26
Abbildung 8: Screenshot vom Terminal, nachdem der Command ausgeführt wurde.	27
Abbildung 9: Screenshot vom Terminal, nachdem der Command fertig ist.	27
Abbildung 10: Allgemeines Layout	29
Abbildung 11: Score comparison Teil	29
Abbildung 12: Netz- und Liniendiagramm	30
Abbildung 13: Open Assessments	31
Abbildung 14: erfolgreiches Test ergebnis	34
Abbildung 15: andere Darstellung der erfolgreichen Tests	34

Tabellenverzeichnis

Tabelle 1: Projektrollen.....	11
Tabelle 2: Testcase T-001	32
Tabelle 3: Testcase T-002.....	32
Tabelle 4: Testcase T003	32
Tabelle 5: Testcase T-004.....	33
Tabelle 6: Testergebnis	33
Tabelle 7: Quellen	37
Tabelle 8: Glossar	38

1. Kurzfassung des IPA-Berichtes

Folgend wird kurz der IPA-Bericht zusammengefasst.

1.1 Ausgangssituation

Im Rahmen von der Probe-IPA der WebGate Consulting AG habe ich folgende Aufgabestellung durchgeführt. Die Aufgabenstellung bestand daraus das die "Mein Dashboard" Seite von der Mitarbeiter Aktien Index (MAX) Applikation, welche wir in den letzten Monaten gebaut haben.

«MAX» ist eine Applikation, besteht aus einem Backend (API) und einem Frontend (UI). MAX wird dem Kunden als Software-as-a-Service (SaaS) zur Verfügung gestellt.

Dieser Auftrag beinhaltet die folgenden Punkte: Eigener Score und Unternehmensdurchschnittsscore. Mein eigener Score wird vom zuletzt ausgefüllten Assessment sein. Der Unternehmensdurchschnittsscore wird einmal pro Quartal berechnet und gespeichert. Damit man die Entwicklung in einem Linien-Graphen sehen kann. Auch sieht gibt es ein Netzdiagramm von meinen letzten Assessments (besteht aus dem Mittelwert zwischen Selbst- und Fremdeinschätzung zusammengestellt). Auf der Dashboard Seite kann ich sehen, dass ich noch offene Assessments habe, welche nicht ausgefüllt sind. Dieser Teil des Task ist nur ein Bonus, das heisst dass es die niedrigste Priorität hat.

1.2 Umsetzung

Dashboard Seite erstellen, alle Punkte aus der Aufgabenstellung umsetzen, anhand des Designs im Mockup. Während der Umsetzung wurde besonders auf Erweiterbarkeit der Applikation geschaut. Dies vereinfacht Erweiterungen/Änderungen in der Zukunft.

Teil 1: Ablauf Organisation und Umfeld

IPA-Projektname:

V1: "Mein Dashboard" Seite mit Infos befüllen



Abbildung 1: WebGate Logo

Autor: Nathan Fahrni

2. Aufgabenstellung

Mein Dashboard Seite erstellen und anhand des Mockups erstellen.

2.1 Titel der Arbeit

V1: "Mein Dashboard" Seite mit Infos befüllen

2.2 Thematik

Titel: V1: "Mein Dashboard" Seite mit Infos befüllen

Beschreibung:

Mit dieser Erweiterung von der MAX-Applikation wollen wir eine bessere User experience kreieren. Die Dashboard Seite gibt dem User eine schnelle und klare Übersicht über sein «MAX-Score» und über noch auszufüllende Assessments.

Die Dashboard Seite wird auch den Standard "Home" page auf welcher man beim Applikationsstart ankommt.

2.3 Ausgangslage

Arbeitsgebiet: Applikationsentwicklung

Programmiersprache: C#, Typescript

2.4 Detaillierte Aufgabenstellung

Die Dashboard Seite anhand des Mockups umsetzen.

Es gibt zwei Score Darstellungen, auf der linken Seite wird der eigene Score vom zuletzt ausgefüllten Assessment angezeigt und auf der rechten Seite wird der

Unternehmensdurchschnitt aller im letzten Quartal ausgefüllten Assessments dargestellt.

Dies führt zu einem einfachen Vergleich der eigenen Leistung im Vergleich zu dem Unternehmensdurchschnitt. Bei diesem Vergleich wird immer der aktuelle Score genutzt.

Es wird eine Trend Darstellung des Unternehmensdurchschnitts geben in Form eines Liniendiagramm. Dieses wird die letzten vier Quartale darstellen (Q4 2021, Q1 2022, Q2 2022, Q3 2022). Bei dem Liniendiagramm werden nur abgeschlossene Quartale angezeigt.

Das heisst, das aktuelle Quartal wird nicht im Diagramm zu sehen sein.

Es wird ein Netz/Radardiagramm haben, welches den Mittelwert (Fremd- & Eigeneinschätzung) hat, dieses wird die Kernwerte des zuletzt ausgefüllten Assessments anzeigen. Das Netzdiagramm hat tiefere Priorität gegenüber den zuvor erwähnten Punkten.

Beide diese Diagramme werden mithilfe von Apex Charts generiert werden.

(<https://apexcharts.com/angular-chart-demos/line-charts/basic/>)

2.5 Mittel und Methoden inklusive Projektmethode

2.5.1 Projektmethode: Wasserfall

Ich werde detaillierter auf die Wasserfall Projektmethode eingehen bei Kapitel «**Projektmethode: Wasserfall**».

2.5.2 Mittel

Ich werde mit der bereits Bestehenden Angular Applikation den Auftrag erfüllen. Auch werde ich unsere «Definition of done» verwenden, um die Fertigkeit des Auftrags zu prüfen. Ich werde die Funktionalität mit Typescript und HTML umsetzen, das Styling mit SCSS.

2.6 Vorkenntnisse

- Angular 1.5 Jahre Erfahrung
- HTML 3 Jahre Erfahrung
- CSS/SCSS 3 Jahre Erfahrung
- Typescript 1.5 Jahre Erfahrung
- Javascript 3 Jahre Erfahrung
- C# 1 Jahr Erfahrung
- Apexcharts.js (<https://apexcharts.com/angular-chart-demos/line-charts/basic/>) 4 Monate Erfahrung

2.7 Vorarbeiten

Ich habe von der Projektleiterin die Mockups bereits im Voraus erhalten, sodass ich mir bereits ein Bild der Arbeit machen konnte, welche auf mich zukommt. Auch haben wir das Backend (BE) bereits im Voraus auf diesen Stand gebracht.

2.8 Arbeiten in den letzten 6 Monaten

Zut:me Applikation (Frontend development, Backend development)

MAX (Frontend development, Backend development)

Goodmeeting V2 (Frontend development, Backend development)

3. Detailliertes Projektvorgehen

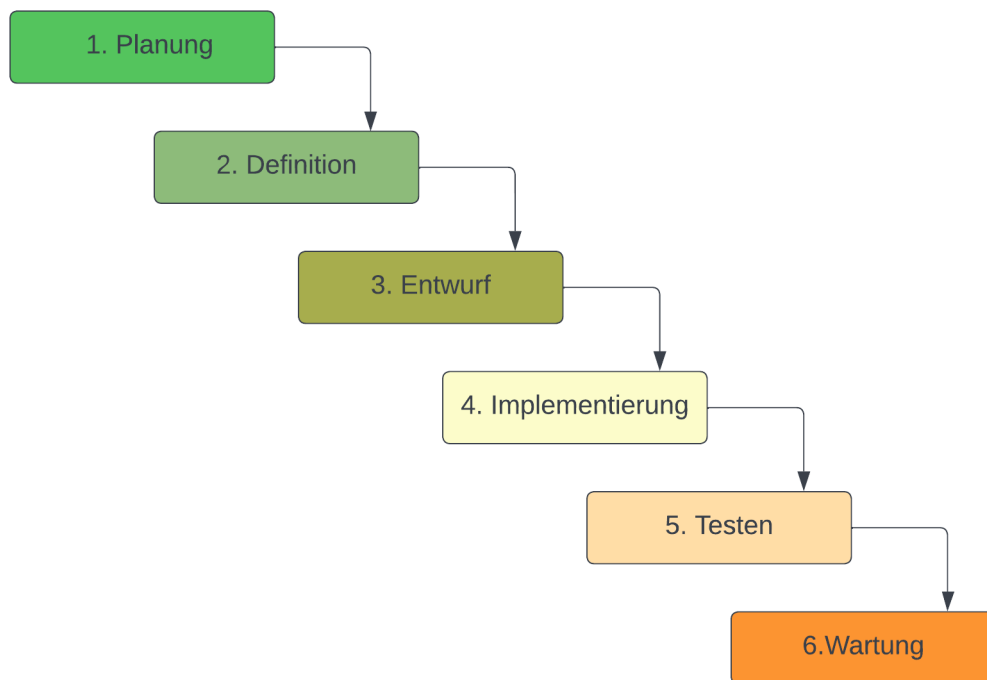


Abbildung 2: Wasserfallmethode Diagramm

Ich habe mich für die Wasserfallmethode entschieden. Weil es für diesen Auftrag meiner Meinung nach am meisten Sinn macht einen klaren Ablauf zu haben und eine Abhängigkeit von den Verschiedenen Phasen.

Abbildung 3, Demo Hermes 5.1 IPA

3.1.1 Projektmethode: Wasserfall

Die Wasserfallmethode ist eine strukturierte Methodik, die aufgrund ihrer präzisen Arbeitsweise häufig genutzt wird. Die Phasen der Methodik werden klar definiert und dokumentiert, was jede einzelne Phase umfasst. In der Wasserfallmethodik gibt es zwei Varianten, eine hat fünf Phasen, die andere hat sechs Phasen, wobei jede Phase die Grundlage der nächsten bildet. Alle Phasen sind klar voneinander getrennt.

3.1.2 Phasen:

Phase 1: Planung

Als erster Schritt gilt die Planung oder Ist-Analyse. Hier wird die Ausgangssituation festgestellt.

Phase 2: Definition

Hier wird die Softwarearchitektur definiert, sowie auch das Softwaredesign und -spezifikationen. Dies besteht aus Systemmodellen oder/und Sequenzdiagramme. Bei den Spezifikationen gibt es drei Varianten:

- **Informale Spezifikation** in Prosa (natürlicher Sprache)
- **Formale Spezifikation** in spezieller Modellierungssprache
- **Exemplarische Spezifikation** mit Testfällen

Phase 3: Entwurf

In der Entwurfsphase wird ermittelt welches Problem es zu lösen gibt. Mann erstellt einen Entwurf der Lösung auf. Das sichert ein Vorgehen mit System im nächsten Schritt.

Phase 4: Implementation

Dann wird die Implementation der Lösung vorgenommen.

Phase 5: Testen

Das Produkt wird auf einer Testumgebung in Betrieb genommen, die Testumgebung ist eine kontrollierte und sichere Umgebung.

Phase 6: Wartung

Der letzte Schritt hat kein festgelegtes Enddatum und beschreibt die Instandhaltung nach Implementation.

Ich habe die Phase sechs nicht in der IPA durchlaufen, weil ich ein festgelegtes Enddatum der IPA habe. Und weil diese Phase 6 den Rahmen meiner Aufgabe überschreiten würde.

3.2 Projektorganisation

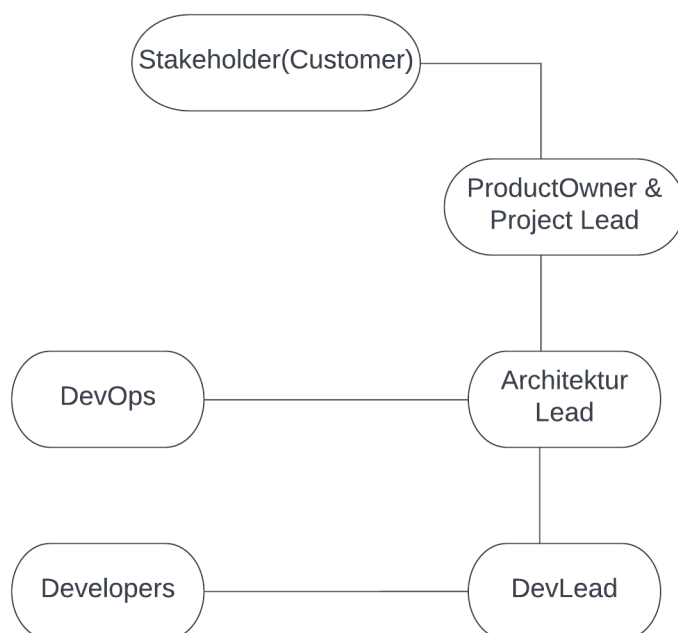


Abbildung 4: Projektorganisation

3.2.1 Projektrollen

Tabelle 1: Projektrollen

Rollenbeschreibung der IPA	
Stakeholder:	Ist der Kunde
Product Owner & Project lead	Projektleiter und Projektverantwortlicher
Architektur Lead	Für die Architektur verantwortlich
DevOps:	Development und Operations für einen schnelleren und effizienteren Workflow
DevLead	Hauptverantwortlicher Developer
Developers	Entwickler der Lösung

4. Zeitplan

Zeitplan			Tag 1		Tag 2		Tag 3		Tag 4		Tag 5	
			Montag		Dienstag		Mittwoch		Donnerstag		Freitag	
			05.12.2022		06.12.2023		07.12.2024		08.12.2025		09.12.2026	
			V	N	V	N	V	N	V	N	V	N
Phase initialisierung		in h	Phase initialisierung									
Kickoff Probe IPA	SOLL	1	1									
	IST	1	1									
Zeitplan erstellen + planen	SOLL	2		2	1							
	IST	2		2								
Dokumentation Vorlage machen	SOLL	4	2	2								
	IST	3	2	1								
Variantenentscheid	SOLL	1				1						
	IST	0.5				0.5						
Phase Konzept			Phase Konzept									
Testkonzept erstellen	SOLL	3			2		1	2				
	IST	1			1		1					
Phase Realisierung			Phase Realisierung									
Allgemeines Layout Mein Dashboard	SOLL	4				2						
	IST	2				2						
Score comparison Components	SOLL	4.5					2	2.5				
	IST	3					2	1				
Line chart	SOLL	4						4				
	IST	2						2				
Radarchart	SOLL	4							4			
	IST	1						1				
Open Assessments Notifcation	SOLL	3.5							2	1.5		
	IST	3							3			
Test-Cases umsetzen	SOLL	1								1		
	IST	0.5								0.5		
Andere Tätigkeiten			Andere Tätigkeiten									
Arbeitsjournal & Zeitplan aktualisieren	SOLL	2.5		0.5		0.5		0.5		0.5		
	IST	2		0.5		0.5		0.5		0.5		
Dokumentation Schreiben	SOLL	-										
	IST	11.5		1	2.5	2	2	1	1	2		
Dokumentation beenden	SOLL	2								2	3	
	IST	0										
Präsentation vorbereiten & präsentieren	SOLL	5.5			0.5						4	1
	IST	5.5			0.5						4	1
Tests durchführen	SOLL	2								2		
	IST	0								0		

Zeit Pro Phase

Meilensteine

Projekt start

1

Testkonzept/e erstellt

2

Dokumentations Abgabe

3

Probe IPA Präsentation

4

Total SOLL:

38.00

h

Total IST:

36.00

h

5. Organisation der IPA

Die Dokumenten Ablage wird mithilfe von zwei Varianten sichergestellt. Mit einem Github repository manuell und mit Word autosave direkt in mein Onedrive. Mein Arbeitsplatz befindet sich im Büro der WebGate Consulting AG in Dietikon.

5.1 Datensicherung der IPA

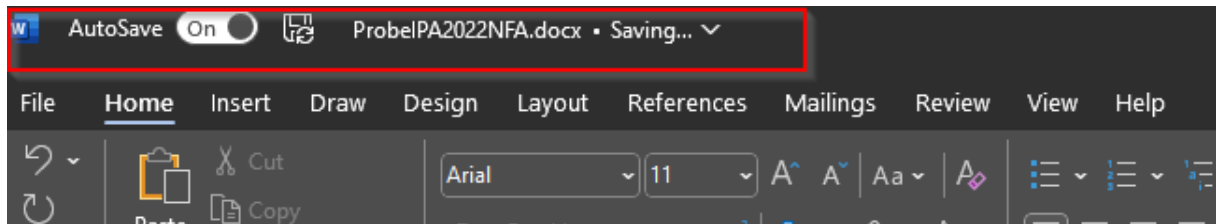


Abbildung 5: Dokumentationsablage via Onedrive

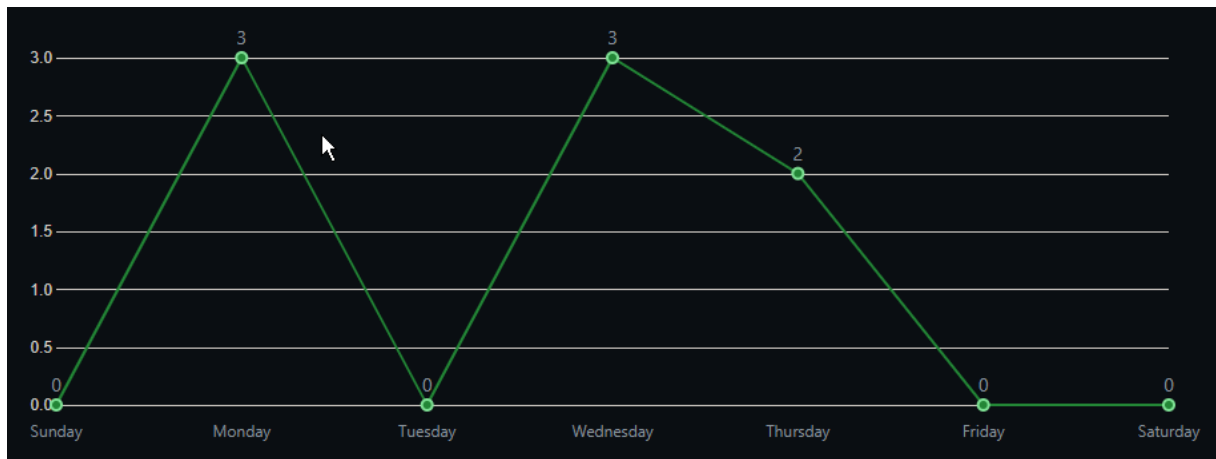


Abbildung 6: Dokumentationsablage Git

Ich speichere die Dokumentation via Autosave direkt in meinem Onedrive von der Arbeit. Auch pushe ich die Dokumentation auf mein Github Repository.

6. Arbeitsjournal

Die Festlegungen dieses Dokuments gelten im Projekt.

Gemäss Art. 5 Absatz 2 der Wegleitung über die individuelle praktische Arbeit (IPA) an Lehrabschlussprüfungen des BBT vom 27. August 2001 gilt:

„Die zu prüfende Person führt ein Arbeitsjournal. Sie dokumentiert darin täglich das Vorgehen, den Stand der Prüfungsarbeit, sämtliche fremde Hilfestellungen (auch das Internet ist eine Hilfestellung) und besondere Vorkommnisse wie z.B. Änderungen der Aufgabenstellung, Arbeitsunterbrüche, organisatorische Probleme, Abweichungen von der Soll-Planung.“

Das Arbeitsjournal zur IPA ist zwingend zu führen und den Experten und Fachvorgesetzten vorzulegen. Das Arbeitsjournal ist täglich sinngemäss und korrekt auszufüllen.

Das Arbeitsjournal dient der Nachvollziehbarkeit der von den Lernenden ausgeführten Arbeiten und wird als Teil der IPA in die Bewertung mit einbezogen.

Tätigkeiten 05.12.2022	Beteiligte Personen	Aufwand geplant (Std)	Aufwand effektiv (Std)
<i>Probe IPA Kickoff</i>	NFA, CSC, BHE, ASE	0.75	1.16
<i>Vorbereitung Dokument und Repository</i>	NFA	4	3
<i>Zeitplan erstellen und Planen</i>	NFA	2	2
<i>Dokumentieren</i>	NFA	0	1
<i>Arbeitsjournal schreiben</i>	NFA	0.5	0.5
Total:		7.25	7.66
Tages Ablauf			
<p>Im Kickoff haben wir den Ablauf dieser nächsten 5 Tage besprochen, wir haben die IPA-Kriterien noch überflogen, Projektmanagement Methoden besprochen.</p> <p>Bei der Vorbereitung von Dokument und Repository habe ich mir auf meinem Private GitHub Account ein Repository erstellt, um die Dokumentation da hochzuladen. Auch fing ich an die Dokumentation auszufüllen mit Informationen, welche ich bereits habe.</p>			
Hilfestellungen			
<p>https://github.com/NKFahrni/WGC-Probe-IPA-Dok Dokumentation Repository</p> <p>https://webgate.atlassian.net/wiki/spaces/WGCLER/pages/1189937153/IPA+DDU+2021 IPA Vorlage von Dominik Dudczak</p>			
Reflexion			
<p><u>Was lief gut</u></p> <p>Ich konnte den Zeit plan beenden. Auch konnte ich alles erledigen, was ich geplant hatte.</p> <p><u>Was lief weniger gut</u></p> <p>Ich konnte bei Word kein Blocksatz machen, weil es dann den Text auseinandergerissen wird.</p> <p><u>Deine Erkenntnisse von heute</u></p> <p>Es ist viel mehr Aufwand, als erwartet eine Dokumentation vorzubereiten.</p>			
Nächste Schritte			
<p>Projektmethode auswählen, herausfinden was der beteiligte Person Kreis ist und was mit Ausgabe Datum gemeint ist. Testkonzepte erstellen.</p>			

Tätigkeiten 06.12.2022	Beteiligte Personen	Aufwand geplant (Std)	Aufwand effektiv (Std)
Dokumentation	NFA	0	2.5
Präsentations Tipps & Tricks	NFA, CSC; MSC, MEB, BHE	0.5	0.5
<i>TestKonzept entwerfen</i>	NFA	2	1
<i>Allgemeines Layout Mein Dashboard</i>	NFA	2	2
<i>Dokumentation</i>	NFA	0	2
Total:		4.50	8.00
Tages Ablauf			
Mir ist aufgefallen, dass ich noch sehr viel zu dokumentieren habe. Und ich noch nicht so weit bin wie erwartet. Ich kam sehr gut mit dem Basic Layout voran. Zeitplan anpassen, der Varianten entschied musste ich noch verschieben auf heute Nachmittag.			
Hilfestellungen			
https://www.appvizer.de/magazin/organisation-planung/projektmanagement/wasserfall-modell David Bürgler => Fragen zu backend Calls			
Reflexion			
<u>Was lief gut</u> Umsetzen des Mockups im code life super. <u>Was lief weniger gut</u> Dokumentieren war sehr anstrengend und langsam <u>Deine Erkenntnisse von heute</u> Ich muss noch vieles Dokumentieren.			
Nächste Schritte			
Weiter machen mit der Realisierungs Phase und die Chartfactories schreiben. Dokumentation weiterführen, und die Testkonzepte verfeinern und die Test cases schreiben in code und Doku. Die Projektmethode fertig beschreiben.			

Tätigkeiten 07.12.2022	Beteiligte Personen	Aufwand geplant (Std)	Aufwand effektiv (Std)
Dokumentation	NFA	0	1.5
Testcases	NFA	1	1
<i>Realisierung</i>	NFA	4.5	4.5
<i>Dokumentation</i>	NFA	0	0.75
<i>Zeitplan & Arbeitsjournal Updaten</i>	NFA	0.5	0.25
Total:		6.00	8.00
Tages Ablauf			
<p>Ich habe mit dem Teil der Dokumentation begonnen, welchen ich Gestern nicht beenden konnte. Dann habe ich die Test-Cases geschrieben und verbessert. Dann habe ich noch gecoded und schon ziemlich alles gemacht. Dies war sehr unerwartet, ich dachte es würde länger gehen. Ich konnte beide Charts bereits einfügen. Auch die Daten für die beiden Charts sind bereits aus dem Backend und richtige Daten. Ich muss nur noch die Logik und Navigation hinter dem «Offene Assessments» Feld machen.</p>			
Hilfestellungen			
<p>David Bürgler um Hilfe bei dem Code verstehen gefragt. Cedric hat geholfen mit CSS flexbox. https://css-tricks.com/snippets/css/a-guide-to-flexbox/</p>			
Reflexion			
<p>Alle todos der Dokumentation von Gestern konnten gemacht werden.</p> <p><u>Was lief gut</u> Schneller als erwartet gearbeitet.</p> <p><u>Was lief weniger gut</u> Ich konnte mich nicht sehr lange am Stück konzentrieren.</p> <p><u>Deine Erkenntnisse von heute</u> Ich war sehr schnell im Umsetzen der Funktionalität und des Layouts. Hat viel weniger Zeit gekostet als gedacht. David war sehr hilfreich, da er sehr schnell das Backend nach meinen Wünschen angepasst hat.</p>			
Nächste Schritte			
<p>Dokumentation weiter machen und eine Checkliste anfangen, damit Cedric und ich unsere Dokumentation nach dieser Checkliste prüfen können.</p>			

Tätigkeiten 08.12.2022	Beteiligte Personen	Aufwand geplant (Std)	Aufwand effektiv (Std)
Realisierung, Offene Assessments	NFA	2	3
Checkliste machen/beginnen	NFA	1	1
<i>Überprüfung realisierung</i>	NFA	1.5	0.5
<i>Testcases umsetzen im code</i>	NFA	1	0.5
<i>Dokumentieren & überprüfen</i>	NFA	2	2
Total:		7.50	7.00
Tages Ablauf			
Ich habe das letzte Stück der Realisierung umgesetzt und dann die Hälfte der Checkliste für die Dokumentation erstellt. Ich habe die Realisierung noch einmal geprüft und die Test-Cases umgesetzt und durchgeführt. Die Dokumentation überprüft, und fertig dokumentiert			
Hilfestellungen			
David zu API calls gefragt, die er geschrieben hat.			
Reflexion			
<u>Was lief gut</u> Ich konnte die Checkliste mit Cedric Schacher beenden.			
<u>Was lief weniger gut</u> Dokumentieren brachte wieder viel Aufwand mit sich. Das Einfügen vom Excel ins Word gab etwas Probleme.			
<u>Deine Erkenntnisse von heute</u> Word und Excel können sich sehr interessant verhalten und dacht viel Zeit kosten			
Nächste Schritte			
Präsentation vorbereiten, Präsentieren			

Tätigkeiten 09.12.2022	Beteiligte Personen	Aufwand geplant (Std)	Aufwand effektiv (Std)
Präsentation vorbereiten	NFA	4	5
Präsentieren	NFA	1	1
Total:	NFA	5.00	6.00
Tages Ablauf			
Präsentation Vorbereiten, Präsentieren.			

7. Abschlussbericht

7.1 Vergleich Ist/Soll

Die Umsetzung hat so funktioniert wie geplant, bis auf einen Punkt beim Liniendiagramm da sieht es etwas komisch aus, wenn man über den ersten Datenpunkt hovers, dann hat es keinen Titel im Hover drin. Durch dies sieht das Hover Feld nicht gut aus. Aber ansonsten sieht die Umsetzung gleich aus wie das Mockup aus.

7.2 Mittelbedarf

Es mussten keine zusätzlichen Mittel beschafft werden ich konnte die bereits bestehenden Verwenden.

7.3 Realisierungsbericht

Bei der Realisation gab es keine wirklichen Probleme ausser einigen kleineren Styling Probleme, die ich aber gefixt habe. Diese Probleme waren nicht vorauszusehen, weil es Probleme sind, welche erst beim Responsive machen, auftreten. Auch ist mir aufgefallen, dass es nicht gut aussieht, wenn der erste Datenpunkt im Liniendiagramm keinen Titel hat, weil es beim Hover Effekt nur ein kleines graues Fenster anzeigt. Ansonsten verlief alles wie geplant, sogar viel schneller als geplant.

7.4 Testbericht

Alle Tests verliefen wie geplant und waren erfolgreich. Die Funktionen haben die korrekte Funktionalität. Und Bonus zum Testing war, dass der Dashboard Component 100% Coverage hat.

7.5 Fazit zum IPA (Projekt)

Da dies eine IPA in einem sehr eingeschränkten Zeitfenster ist, war die Zeitplanung der grösste Stolperstein, weil ich vieles in der Dokumentation unterschätzte. Einmal einen kleinen Stolperstein als ich die Seite Responsive machte, dies hat zu kurzer Zeitplan Änderung geführt. Ansonsten gab es nicht Probleme oder Unterbrechungen. Die Tests liefen genau wie geplant. Beim Zeitplan war zu viel Zeit auf die Realisierung verplant und zu wenig auf die Dokumentation.

7.6 Persönliches Fazit

Mit 1 ½ Erfahrung mit Angular, Typescript und SCSS war dieser Auftrag keine Stresssituation. Ich konnte mir gut vorstellen, wie ich die Dashboard Seite aufbauen musste. Ich hatte etwas Respekt vor den zwei Diagrammen, weil das letzte Mal als ich mit Apex-

Charts etwas gemacht hatte, war es sehr Anspruchsvoll. Doch diesmal ging es sehr schnell und ohne Probleme. Ich konnte das Netz-, wie auch das Liniendiagramm innert kürzester Zeit auf der Applikation wie gewünscht ausgeben. Dies hat mich sehr gefreut, da ich sehen konnte das ich vieles gelernt habe, seit ich das letzte Mal Apex-Charts benutzte.

Ich habe auch gemerkt, wie einfach mir das Layout fiel, die Flexbox keywords kannte ich bereits und wusste auch welche wie und wo einzusetzen sind. Ich konnte mir auch bildlich gut vorstellen, wie ich das HTML aufbauen muss. Ich würde mir das nächste Mal im zeit plan die Soll stunden für die Dokumentation auch eintragen und auch würde ich diese sehr hoch einschätzen um nicht noch einmal die Dokumentation so zu unterschätzen wie bei der Probe IPA. Auch würde ich mir die Checkliste und die Dokumentation Vorlage vorher vorbereiten und nicht erst fertig vorbereiten, wenn das Projekt angefangen hat.

7.7 Schlussreflexion

Die Arbeit war im Dokumentationsteil viel grosser als erwartet, auch war mir nicht immer klar was ich nun wie dokumentieren soll. Beim Realisieren habe ich keine Probleme gehabt. Bei meiner nächsten Arbeit würde ich noch viel mehr nach der Projektmethode arbeiten. Auch würde ich die Anforderungen and die Dokumentation vorher anschauen und das Dokument dem entsprechend vorbereiten.

8. Unterschriften und Abnahmeprotokoll

Die lernende Person bestätigt mit ihrer Unterschrift diese IPA aus Eigenleistung erbracht und nach den Vorgaben der Prüfungskommission Informatik Kanton Bern erstellt zu haben. Die Angaben im Arbeitsjournal entsprechen dem geleisteten Arbeitsaufwand.

Teil 2: Projektdokumentation

Hier werde ich ausführlich auf das Projekt und dessen Umsetzung eingehen. Ich habe die Realisierung und deren Schritte so gut es geht dokumentiert.

9. Initialisierung

Im folgenden Teil wird, der IST- und der SOLL zustand, beschrieben. Es werden Verschiedene Anforderungen und Ziele beschrieben.

9.1 Studie; Ist-Zustand

Aktuell gibt es keine «My Dashboard» Seite bei der MAX-Applikation. Das heisst der Ist Zustand ist momentan nichts.

9.2 Persönliche Vorgehensziele

Ich möchte mehr über Apexcharts und wie man diese am besten anwendet lernen. Auch ist ein Ziel von mir, dass ich mich an den Zeitplan halte und diesen genau befolge. Ein weiteres ist, dass ich das Design exakt dem Mockup nachbaue.

9.3 Projektziele

Dokumentation bis am 8.12.2022 um 23:59 abgegeben haben.

Mit der Präsentation am 09.12.2022 um 12:00 fertig sein, sodass ich noch Zeit habe diese zu verbessern falls nötig.

Am Tag 1 soll um 17:00 der Zeitplan fertig sein.

Am Tag 2 sollen um 17:00 die Testkonzepte stehen.

Am Tag 4 soll die Dokumentation um 23:59 abgegeben sein.

Am Tag 5 soll die Präsentation um 15:30 gehalten werden.

9.4 Anforderungen

Die «My Dashboard» Seite soll dem Mockup entsprechen. My Index wird angezeigt, Unternehmens Durchschnittsindex wird angezeigt, das Netzdiagramm zeigt den Mittelwert der Fremd- und Selbsteinschätzung, der Liniengraph zeigt den Verlauf des Unternehmens Durchschnittsindex in Quartalen an. Falls ich offene Assessments habe, wird dies kommuniziert/angezeigt. Davon gibt es zwei Varianten, einmal offene in denen ich evaluiert werde und einmal in denen ich jemanden anderen evaluiere.

9.4.1 Funktionale Anforderungen

Die beiden Indexe werden korrekt angezeigt. Der Liniengraph ist mit den Durchschnittscores von den letzten 4 Quartalen befüllt. Das Netzdiagramm mit dem Mittelwert der Fremd- und Selbsteinschätzung des letzten Assessments. Das Anzeigen, ob ich noch offene/auszufüllende Assessments habe wird mit drei Varianten kommen, Variante eins: Wenn kein Assessment offen ist, wird das Feld dies kommunizieren. Zweite Variante: Wenn es Assessments hat, wo ich mich selbst einschätzen muss, dann wird das Feld dies Kommunizieren und den Router link auf dies Anpassen. Variante drei: Wenn es Assessments hat, bei denen ich jemanden anderen einschätze, dann wird dies im Feld kommuniziert und der Router link geht zur entsprechenden Seite.

9.4.2 Nicht funktionale Anforderungen

Das Design von der Seite entspricht dem Design im Mockup. Das Design ist responsive. Die Seite hat einen Loading status, mit welchem verhindert werden kann, dass man Fehler im Frontend generiert.

9.4.3 Variantenentscheid

Bei dieser Aufgabe wurde das Design bereits im Voraus durch den Auftraggeber, Thalia Fox, festgelegt. Da dieser Teil in einem Projekt stattfindet, muss das Design dem Entsprechen. Mit dem wurde der Variantenentscheid bereits von der Auftraggeberin übernommen und lässt kein Spielraum für eigene Varianten.

10. Konzept

Konzepte gab es hier fast keine, da ich eine Aufgabe in einer bereits bestehenden Applikation erledigte. Deswegen habe ich nur das Testkonzept machen müssen. Ich werde jedoch noch ein Screenshot vom Mockup einfügen.

10.1 Mockup



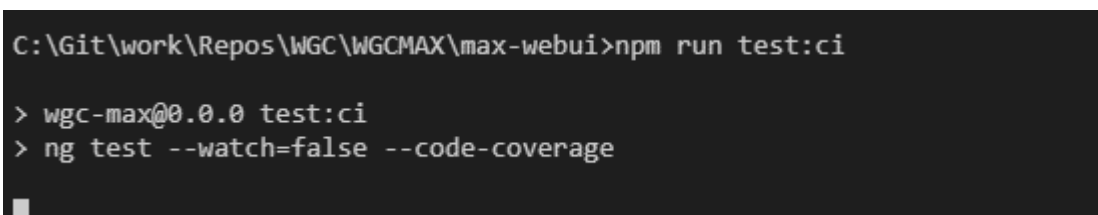
Abbildung 7: Mockup

10.2 Testkonzept

Hier habe ich mich für Unit Tests in Angular entschieden, da diese zur definition of done gehören. Die Unit Tests in Angular werden mit folgenden Tools sein: Karma, Jasmine. Dies sind beides Testing Tools für Angular mit diesen kann man auch Unit Tests schreiben. Die Unit Tests werden sehr simpel sein, da ich auf der Dashboard Seite nicht allzu viel Funktionalität habe. Jasmine ist unser Test-Framework, mit dem wir Tests schreiben. Karma ist die Laufzeitumgebung (runtime) für die Tests.

10.2.1 Testziele

Alle Unit Tests können erfolgreich ausgeführt werden und stimmen alle. Auch muss die Test coverage über 50% sein bei dem neu geschriebenen Code erwartetes Ergebnis sieht so ähnlich wie in folgenden Abbildungen aus.



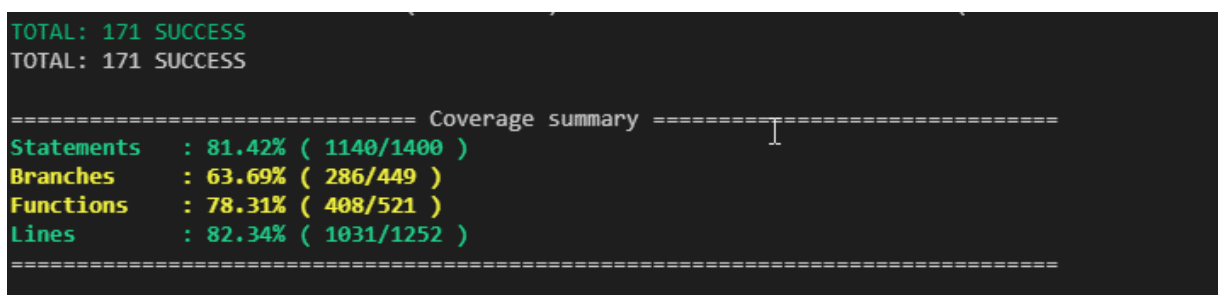
```
C:\Git\work\Repos\WGC\WGCMAX\max-webui>npm run test:ci

> wgc-max@0.0.0 test:ci
> ng test --watch=false --code-coverage
```

Abbildung 8: Screenshot vom Terminal, nachdem der Command ausgeführt wurde.

Mit dem Command «npm run test:ci» sollte folgendes erscheinen:

Wenn die Tests erfolgreich durchgelaufen sind, sollte es so ähnlich wie in folgender Abbildung aussehen:



```
TOTAL: 171 SUCCESS
TOTAL: 171 SUCCESS

===== Coverage summary =====
Statements   : 81.42% ( 1140/1400 )
Branches     : 63.69% ( 286/449 )
Functions    : 78.31% ( 408/521 )
Lines        : 82.34% ( 1031/1252 )
=====
```

Abbildung 9: Screenshot vom Terminal, nachdem der Command fertig ist.

10.2.2 Testmethoden und Testfälle

Hier werde ich meine vier Testfälle so gut es geht beschreiben. Ich werde die Testfälle alle als Unit Tests durchführen.

Fall 1: getLastFourQuarters()

Prüfe, ob die Service Funktion das richtige returned. Und dies dann mit den Mock Daten vergleichen. Prüfen, ob die Quartale korrekt sortiert sind und dass ich die korrekten vier Quartale kriege.

Fall 2: getLastAssessmentScore()

Prüfe, ob die Service Funktion das richtige Assessment returned und dann in der getLastAssessmentScore Funktion auch der Score ausgelesen wird. Prüfe, ob dies mit den Mock Daten übereinstimmt.

Fall 3: getCompanyAverage()

Prüfe, ob die Service Funktion den Unternehmensdurchschnitt returned. Prüfe, ob die getCompanyAverage Funktion den richtigen Durchschnitt ausliest.

Fall 4: getOpenAssessments()

Prüfe, ob die Service Funktion die Assessments returned. Prüfe, ob die Filter Funktion auch nur die Assessments, welche noch auszufüllen sind, returned.

11. Realisierung

In diesem Teil der Projektdokumentation werde ich im Detail auf die verschiedenen Entscheidungen und Challenges in der Realisierung des Projektes eingehen.

11.1 Allgemeines Layout vom Dashboard



Abbildung 10: Allgemeines Layout

Dieser Arbeitsschritt war sehr schnell gemacht, ich brauchte deutlich weniger Zeit als erwartet, dies ist sehr nützlich, da ich noch die Chartfactories schreiben muss und einige Test-Cases. Dies ist immer ein grosser Zeitaufwand. Auch muss ich im nächsten Schritt noch die einzelnen Components ausfindig machen, das heisst ich werde noch die einzelnen Elemente aus der HTML-Datei nehmen, um die Indexe anzuzeigen.

11.2 Score comparison Components

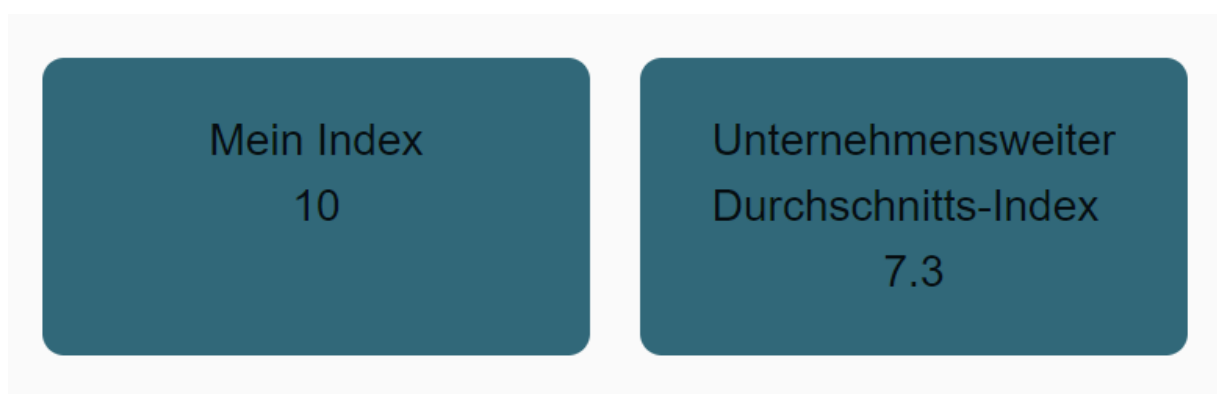


Abbildung 11: Score comparison Teil

Die beiden Indexe aus dem Dashboard Component herausnehmen und in zwei Subcomponents packen. Dies verkleinert den Dashboard Component um einiges. Dies war der Plan doch habe ich gesehen, dass diese beiden Teile nur sehr klein sind und keine

eigenen Components brauchen. Deswegen habe ich mich entschieden alle im Dashboard Component zu Lösen. Netz- und Liniendiagramm

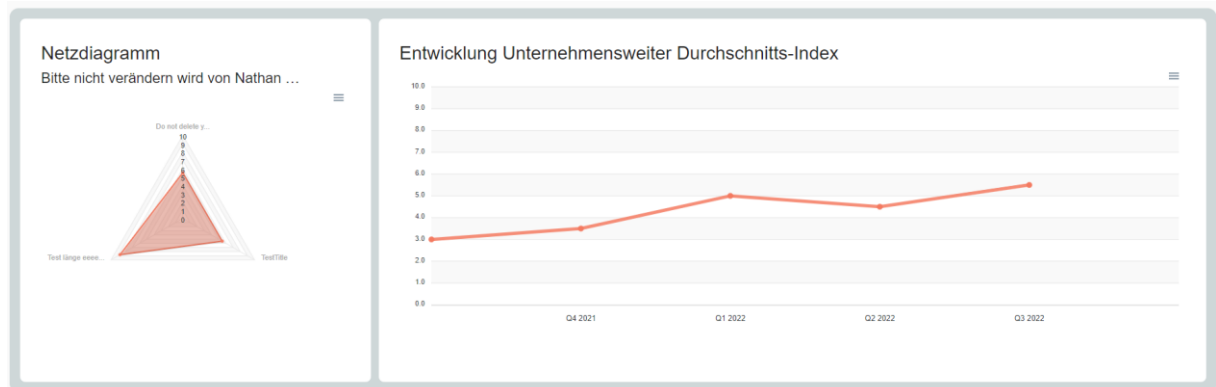


Abbildung 12: Netz- und Liniendiagramm

Die Diagramme sind auch schon fertig. Da ich heute mit David Bürgler noch kleine Änderungen am Backend besprochen hatte, erwartete ich, dass die Änderungen nicht so schnell umgesetzt wurden. Jedoch konnte ich bereits innert kürzester Zeit auf die neuen API-Calls zugreifen und die Daten im Frontend darstellen. Auch die Responsivness der Seite habe ich noch einmal überarbeitet. Dies verhält sich nun viel Benutzer freundlicher. Beim Netzdiagramm gab es ein wenig Probleme als ich versuchte nur den einen Datensatz mit dem Mittelwert der beiden Einschätzungen darzustellen. Also musste ich vieles neu schreiben. Doch diese Probleme haben mich nur kurz aufgehalten. Das Liniendiagramm ist so weit fertig, ich habe hier jedoch noch einen Problempunkt gefunden. Der im Design vom Mockup nicht ersichtlich war. Die Diagramme haben eine Hover Funktion, in der der Name des Datenpunktes widergespiegelt wird. Der allererste Punkt hat aber im Mockup keinen Titel/Name das heisst dieses Hover Feld ist sehr klein und sieht fehl am Platz aus. Ich persönlich bin dafür das dies nicht so bleibt und wir dies einfach wie alle anderen Datenpunkte machen und auch den Namen/Titel anzeigen.

Ich habe bereits fast alles aus dem Mockup und dem Task in Jira erfasst, ich muss nur noch den Teil mit den Offenen Assessments fertig machen. Also die Logik und Navigation dahinter.

11.3 Open Assessment



Abbildung 13: Open Assessments

Das Open Assessment element auf der Seite hat erstaunlicherweise fast am meisten Zeit gekostet, da ich sehr viele Daten nehmen und überprüfen musste. Auch hat es so viele unterschiedliche darstellungsmöglichkeiten, dass ich diese noch im HTML ausarbeiten musste. Dies stellte eine komplexe Aufgabe dar. Der Text passt sich der Offenen Assessment situation an.

Testprotokoll

Die Testprotokolle fassen die Testergebnisse zusammen. In diesem Teil werden die Test-Cases genauer beschrieben.

11.4.1 Testfall

Tabelle 2: Testcase T-001

ID / Bezeichnung	T-001	getLast5QuartersScore()
Beschreibung	Prüfe, ob die Funktion das richtige returned. Und dies dann mit den Mock Daten vergleichen. Prüfen, ob die Quartale korrekt sortiert sind und dass ich die korrekten vier Quartale kriege.	
Testvoraussetzung	Mit «npm run test:ci» können die Tests durchlaufen, ohne abubrechen.	
Testschritte	npm run test:ci	
Erwartetes Ergebnis	Der Test für getLastFourQuarters läuft erfolgreich durch	

Tabelle 3: Testcase T-002

ID / Bezeichnung	T-002	getNewestAssessment()
Beschreibung	Prüfe, ob die Funktion das richtige Assessment returned und dann in der getLastAssessmentScore Funktion auch der Score ausgelesen wird. Prüfe, ob dies mit den Mock Daten übereinstimmt.	
Testvoraussetzung	Mit «npm run test:ci» können die Tests durchlaufen, ohne abubrechen.	
Testschritte	npm run test:ci	
Erwartetes Ergebnis	Der Test für getLastAssessmentScore läuft erfolgreich durch	

Tabelle 4: Testcase T003

ID / Bezeichnung	T-003	getCurrentCompanyAverage()
Beschreibung	Prüfe, ob die Funktion den Unternehmensdurchschnitt returned. Prüfe, ob die getCompanyAverage Funktion den richtigen Durchschnitt ausliest.	
Testvoraussetzung	Mit «npm run test:ci» können die Tests durchlaufen, ohne abubrechen.	
Testschritte	npm run test:ci	
Erwartetes Ergebnis	Der Test für getCompanyAverage läuft erfolgreich durch	

Tabelle 5: Testcase T-004

ID / Bezeichnung	T-004	getOpenAssessments()
Beschreibung	Prüfe, ob die Service Funktion die Assessments returned. Prüfe, ob die Filter Funktion auch nur die Assessments, welche noch auszufüllen sind, returned.	
Testvoraussetzung	Mit «npm run test:ci» können die Tests durchlaufen, ohne abzubrechen.	
Testschritte	npm run test:ci	
Erwartetes Ergebnis	Der Test für getOpenAssessments läuft erfolgreich durch	

11.4.2 Testergebnis

Tabelle 6: Testergebnis

Tester	Nathan Fahrni
Datum Testdurchführung	08.12.2022
Fehlerklasse (Testergebnis)	keine
Fehlerbeschreibung	keine

11.4.3 Testwürdigung

Die tests waren wie zu erwarten erfolgreich, alle funktionen haben das Returned was sie sollten.

11.4.4 Testfazit

Die Unit Tests waren erfolgreich, weil ich die Tests so schrieb wie die Funktionalitätsanforderungen im Auftrag waren.

Als ich `npm run test:ci` ausgeführt habe, kriegte ich wie erwartet eine erfolgreiche response, siehe folgende Abbildungen:

```
TOTAL: 177 SUCCESS
TOTAL: 177 SUCCESS

===== Coverage summary =====
Statements   : 82.19% ( 1205/1466 )
Branches     : 64.42% ( 297/461 )
Functions    : 79.34% ( 438/552 )
Lines        : 83.1% ( 1092/1314 )
=====
```

Abbildung 14: erfolgreiches Test ergebnis

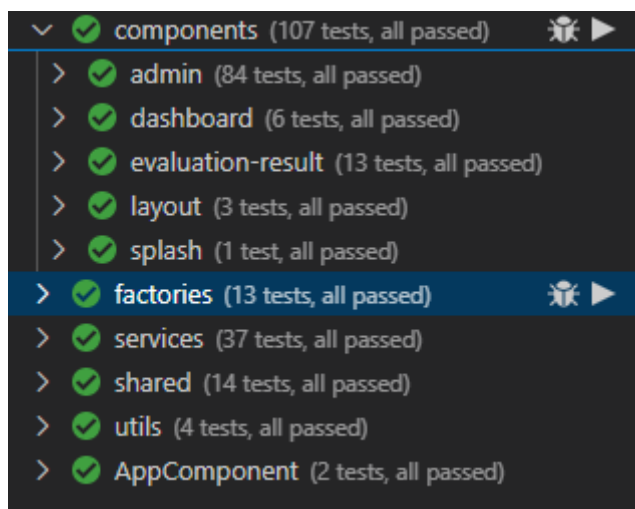


Abbildung 15: andere Darstellung der erfolgreichen Tests

11.5 Einführung vorbereiten

Bei der Applikation wird dies die neue Homepage, das heisst man muss dies in den Applikation routen anpassen. Auch wird das die Fallbackroute, das heisst es wird immer auf diese zurückfallen, falls eine Route aufgerufen wird die es nicht (mehr) gibt. Ansonsten gibt es keine Änderungen Einfluss auf die Applikation haben.

12. Einführung

Dies ist ein sehr einfacher Teil, da es nichts zum Installieren gibt. Hier muss ich einen Pullrequest (PR) erstellen und dieser die verschiedenen Checks in unserer Buildpipeline bestehen. Darunter fallen Buildet die Applikation ohne Fehler, Quality Gate (code Qualität gecheckt), ob mehr als 50% des neuen Codes getestet ist.

13. Literatur und Quellenverzeichnis

Tabelle 7: Quellen

CSS flexbox.	https://css-tricks.com/snippets/css/a-guide-to-flexbox/
Wasserfallmethode	https://www.appvizer.de/magazin/organisation-planung/projektmanagement/wasserfall-modell
IPA-Vorlage von Dominik Dudczak	https://webgate.atlassian.net/wiki/spaces/WGCLER/pages/1189937153/IPA+DDU+2021
Dokumentation Repository	https://github.com/NKFahrni/WGC-Probe-IPA-Dok
Apexcharts	(https://apexcharts.com/angular-chart-demos/line-charts/basic/)

14. Glossar

Tabelle 8: Glossar

Begriff	Bedeutung
IPA	Individuelle praktische Arbeit
MAX	Mitarbeiter Aktien Index – WebGate interne Applikation. Ein Task davon ist meine Probe IPA
OdA	Organisation der Arbeitswelt
QV	Qualifikationsverfahren
VP	Verantwortliche Fachkraft
VSC	Visual Studio Code
VZ	Verzeichnis

15. Anhang

Der restliche Code ist auf dem WGMAX-353 Branch zu finden. Link:

https://dev.azure.com/WebGateConsultingAG/wgc_MAX/_git/max-webui?version=GBfeature%2FWGMAX-453

15.1 Component

```
import { Component, OnInit } from '@angular/core';
import { MAT_TOOLTIP_DEFAULT_OPTIONS } from '@angular/material/tooltip';
import { DashboardLinechartFactory } from
'src/app/factories/charts/dashboardLinechartFactory';
import { roundToOneDecimal } from 'src/app/factories/charts/getScore';
import { AssessmentTemplate } from 'src/app/models/assessmentTemplate.model';
import { LineChartOptions } from 'src/app/models/lineChartOptions';
import { DashboardRadarChartOptions } from
'src/app/models/radarChartOptions.model';
import { EvaluatorAssessmentService } from 'src/app/services/evaluator-
assessment.service';
import { ReleasedAssessmentService } from 'src/app/services/released-
assessment.service';
import { customTooltipOptions } from 'src/app/utils/custom-tooltip-options';
import { DashboardRadarChartFactory } from
'../../factories/charts/dashboardRadarChartFactory';
import { CompanyAverage } from ' ../../models/companyAverage';
import { MyAssessment } from ' ../../models/myAssessment.model';
import { MyAssessmentService } from ' ../../services/my-assessment.service';
import { QuarterScoresService } from ' ../../services/quarter-scores.service';

@Component({
  selector: 'app-dashboard',
  templateUrl: './dashboard.component.html',
  styleUrls: ['./dashboard.component.scss'],
  providers: [{ provide: MAT_TOOLTIP_DEFAULT_OPTIONS, useValue:
customTooltipOptions }],
})
export class DashboardComponent implements OnInit {
  radarChartOptions!: DashboardRadarChartOptions;
  lineChartOptions!: LineChartOptions;
  newestAssessment!: MyAssessment;
  companyAverage!: number;
  scoresLoaded: boolean = false;
  isLinechartLoaded: boolean = false;
  myAssessments!: MyAssessment[];
  myAssesmentAsEvaluator!: MyAssessment[];

  constructor(
```

```

    private releasedAssessmentService: ReleasedAssessmentService,
    private myAssessmentService: MyAssessmentService,
    private evaluatorAssessmentService: EvaluatorAssessmentService,
    private dashboardRadarchartFactory: DashboardRadarchartFactory,
    private dashboardLinechartFactory: DashboardLinechartFactory,
    private quarterScoresService: QuarterScoresService,
  ) { }

  ngOnInit(): void {
    this.getMyIndex();
    this.getevaluatorAssessmentService();
  }

  getchartOptions(): void {
    this.getCurrentCompanyAverage();
    this.radarchartOptions = this.getRadarchartOptions('#f47a60');
    this.generateLinechartOptions();
  }

  getRadarchartOptions(color: string): DashboardRadarChartOptions {
    return
    this.dashboardRadarchartFactory.getDashboardRadarchartOptions(this.newestAsses
    sment.coreValues, color)
  }

  generateLinechartOptions(): void {
    this.quarterScoresService.getLast5QuartersScore().subscribe((scores:
    CompanyAverage[]) => {
      this.linechartOptions =
    this.dashboardLinechartFactory.getLinechartOptions(scores);
      this.isLinechartLoaded = true;
    });
  }

  getMyIndex(): void {
    this.myAssessmentService.getMyAssessments(localStorage.getItem('userId')
    || '').subscribe((myAssessments: MyAssessment[]) => { //not right call?
      this.myAssessments = myAssessments;
      this.newestAssessment = this.getNewestAssessment(myAssessments);
      this.getchartOptions();
    });
  }

  getNewestAssessment(myAssessments: MyAssessment[]): MyAssessment {
    let newestAssessment = myAssessments[0];
    myAssessments.forEach((value: MyAssessment) => {
      if (value.terminatedDateTime !== null &&
    value.terminatedDateTime.localeCompare(newestAssessment.terminatedDateTime)
    === 0) {
        newestAssessment = value;
      }
    });
  }

```



```

    }
  })
  return newestAssessment;
}

getCurrentCompanyAverage(): void {
  this.releasedAssessmentService.getCompanyScore(this.newestAssessment.templateId).subscribe((companyScore: AssessmentTemplate) => {
    this.companyAverage = roundToOneDecimal(companyScore.average);
    this.scoresLoaded = true;
  });
}

amountOfMyOpenAssessments():MyAssessment[] {
  return this.myAssessments.filter((assessment:MyAssessment) =>
!assessment.requesterHasFinished);
}

amountOfOpenAssessmentsAsSupervisor():MyAssessment[]{
  return this.myAssesmentAsEvaluator.filter((assessment:MyAssessment) =>
!assessment.requesterHasFinished);
}

getevaluatorAssessmentService():void{
  this.evaluatorAssessmentService.getMyAssesmentAsEvaluator(localStorage.getItem('userId') || '').subscribe((myAssessments:MyAssessment[])=>{
    this.myAssesmentAsEvaluator = myAssessments;
  });
}
}

```

15.2 HTML

```

<div class="fullHeightNoNavbar dashboard-container p4">
  <div class="indexPanels backgroundPrimary dashboardHeading p4">
    <div class="">{{"dashboard.myIndex" | translate}}</div>
    <div *ngIf="scoresLoaded">{{newestAssessment.evaluatedScore}}</div>
  </div>
  <div class="indexPanels backgroundPrimary dashboardHeading p4">
    <div class="top p15 pr5">
      <!--I DONT KNOW HOW TO CHANGE THIS, IT LOOKS DUMB IN THE BROWSER IM GOING TO FIGURE IT OUT SOON -->
      <div class="textleft">
        {{"dashboard.companyAverage" | translate}}
      </div>
    </div>
    <div *ngIf="scoresLoaded">{{companyAverage}}</div>
  </div>

```

```

<!--OPEN ASSESSMENTS-->
<div class="openAssessments backgroundAccent p4">
  <div class="dashboardHeading">{{"dashboard.openAssessmentsTitles" |
translate}}</div>
  <ng-container *ngIf="scoresLoaded">
    <div *ngIf="!amountOfMyOpenAssessments().length &&
!amountOfOpenAssessmentsAsSupervisor().length">
      {{"dashboard.noOpenAssessments" |
translate}}</div>
    <div *ngIf="amountOfMyOpenAssessments().length ||
amountOfOpenAssessmentsAsSupervisor().length">
      {{"dashboard.openAssessments.firstPart" | translate}}
      <!--MyAssessments Link-->
      <a *ngIf="amountOfOpenAssessmentsAsSupervisor().length"
routerLink="../activeAssessments"
class="link">{{amountOfOpenAssessmentsAsSupervisor().length}}
      {{amountOfOpenAssessmentsAsSupervisor().length === 1
?
"dashboard.openAssessments.singleAssessmentsAsEvaluator":"dashboard.openAsses
sments.multipleAssessmentsAsEvaluator"
| translate}}</a>
      <span *ngIf="amountOfMyOpenAssessments().length &&
amountOfOpenAssessmentsAsSupervisor().length">
{{"dashboard.openAssessments.and"| translate}} </span>
      <!--akactiveAssessments Link-->
      <a *ngIf="amountOfMyOpenAssessments().length"
routerLink="../myAssessments"
class="link">{{amountOfMyOpenAssessments().length}}
      {{amountOfMyOpenAssessments().length === 1
?'dashboard.openAssessments.singleMyAssessment':'dashboard.openAsses
sments.multipleMyAssessment' |
translate}}</a>
      {{"dashboard.openAssessments.lastPart" | translate}}
    </div>
  </ng-container>
</div>
<!--RADARCHART-->
<div class="chartContainer backgroundAccent p2">
  <div class="radarChartLastAssessment p4">
    <div class="dashboardHeading">{{"dashboard.radarChart" |
translate}}</div>
    <div class="titleAssessment text-overflow subheading"
*ngIf="scoresLoaded"
matTooltip="{{newestAssessment.defaultTitle}}">
{{newestAssessment.defaultTitle}}
    </div>
    <div class="titleAssessment textFont mt5" *ngIf="scoresLoaded &&
newestAssessment.coreValues.length < 3">
      {{"dashboard.cantshowRadarchartText" | translate}}</div>
  </div>
</div>

```

```

    <apx-chart *ngIf="scoresLoaded && newestAssessment.coreValues.length >
2" #radarChart
      [series]="radarchartOptions.series" [chart]="radarchartOptions.chart"
[xaxis]="radarchartOptions.xaxis"
      [markers]="radarchartOptions.markers" [fill]="radarchartOptions.fill"
[title]="radarchartOptions.title"
      [yaxis]="radarchartOptions.yaxis"
[plotOptions]="radarchartOptions.plotOptions"
      [colors]="radarchartOptions.colors" id="radarchartOptions">
    </apx-chart>
  </div>
  <!-- LINECHART -->
  <div class="linechartcompanyAveragePerQuarter p4">
    <div class="dashboardHeading">{{"dashboard.linechartTitle" |
translate}}</div>
    <apx-chart *ngIf="isLinechartLoaded" [series]="linechartOptions.series"
[chart]="linechartOptions.chart"
      [xaxis]="linechartOptions.xaxis" [yaxis]="linechartOptions.yaxis"
[dataLabels]="linechartOptions.dataLabels"
      [grid]="linechartOptions.grid" [stroke]="linechartOptions.stroke"
[markers]="linechartOptions.markers"
      [colors]="linechartOptions.colors"></apx-chart>
    </div>
  </div>
</div>

```

15.3 SCSS

```

.dashboard-container {
  display: flex;
  gap: 32px;
  flex-direction: row;
  flex-wrap: wrap;
  justify-content: flex-start;
  align-content: center;
}

.indexPanels {
  display: flex;
  justify-content: flex-start;
  flex-direction: column;
  align-items: center;
  align-content: center;
  flex-wrap: wrap;
  border-radius: 14px;
}

```

```
width: calc(20% - 32px);
min-width: 350px;
}

.top {
  display: flex;
  justify-content: flex-start;
  flex-direction: column;
  align-items: center;
  align-content: center;
  flex-wrap: wrap;
  border-radius: 14px;
}

.textleft{
  margin: auto;
  text-align: left;
  overflow: hidden !important;
  text-overflow: ellipsis !important;
  white-space: inherit !important;
}

.openAssessments {
  border-radius: 14px;
  width: calc(60% - 32px);
  min-width: 500px;
  min-height: 190px;
}

.chartContainer {
  display: flex;
  border-radius: 14px;
  gap: 12px;
  width: calc(100% - 12px);
  flex-direction: row;
  justify-content: center;
  flex-wrap: wrap;
}

.radarChartLastAssessment {
  flex-grow: 1;
  width: calc(30% - 12px);
  min-width: 400px;
  min-height: 550px;
  border-radius: 8px;
  background-color: white;
}

.linechartcompanyAveragePerQuarter {
  flex-grow: 2;
  width: calc(70% - 12px);
  min-width: 600px;
```

```
min-height: 550px;
border-radius: 8px;
background-color: white;
}

.titleAssessment{
  max-width: 400px !important;
  min-width: 200px !important;
}

.link{
  text-decoration: underline
}
```

Zeitplan			Tag 1		Tag 2		Tag 3		Tag 4		Tag 5	
			Montag		Dienstag		Mittwoch		Donnerstag		Freitag	
			05.12.2022		06.12.2023		07.12.2024		08.12.2025		09.12.2026	
			V	N	V	N	V	N	V	N	V	N
Phase initialisierung			Phase initialisierung									
		in h										
Kickoff Probe IPA	SOLL	1	1									
	IST	1	1									
Zeitplan erstellen + planen	SOLL	2		2	1							
	IST	2		2								
Dokumentation Vorlage machen	SOLL	4	2	2								
	IST	3	2	1								
Variantenentscheid	SOLL	1				1						
	IST	0.5				0.5						
Phase Konzept					Phase Konzept							
Testkonzept erstellen	SOLL	3			2		1	2				
	IST	1			1		1					
Phase Realisierung					Phase Realisierung							
Allgemeines Layout Mein Dashboard	SOLL	4				2						
	IST	2				2						
Score comparison Components	SOLL	4.5					2	2.5				
	IST	3					2	1				
Line chart	SOLL	4						4				
	IST	2						2				
Radarchart	SOLL	4							4			
	IST	1						1				
Open Assessments Notification	SOLL	3.5							2	1.5		
	IST	3							3			
Test-Cases umsetzten	SOLL	1								1		
	IST	0.5								0.5		
Andere Tätigkeiten			Andere Tätigkeiten									
Arbeitsjournal & Zeitplan aktualisieren	SOLL	2.5		0.5		0.5		0.5		0.5		
	IST	2		0.5		0.5		0.5		0.5		
Dokumentation Schreiben	SOLL	-										
	IST	11.5		1	2.5	2	2	1	1	2		
Dokumentation beenden	SOLL	2								2	3	
	IST	0										
Präsentation vorbereiten & präsentieren	SOLL	5.5			0.5						4	1
	IST	5.5			0.5						4	1
Tests durchführen	SOLL	2								2		
	IST	0								0		

Zeit Pro Phase

Meilensteine

Projekt start

1

Testkonzept/e erstellt

2

Dokumentations Abgabe

3

Probe IPA Präsentation

4

Total SOLL:

38.00

h

Total IST:

36.00

h