|  |
| --- |
|  |

|  |
| --- |
|  |

IPA Bericht von Janosch Lio

Ablösung Mitarbeiter Skillmatrix

Kandidat Janosch Lio

Kornweg 10

5722 Gränichen

Fachvorgesetzter Dominik Baltzer

Hauptexperte Michelle Schär

Nebenexperte Dirk Decher

Abgabedatum 21.03.2024

Über dieses Dokument

Ablage

|  |  |
| --- | --- |
| Geschäftsnummer | - |
| Geschäftstitel | - |

Verteiler

|  |  |  |  |
| --- | --- | --- | --- |
| Name / Vorname | Kurzzeichen | Organisation | Rolle |
| Janosch Lio | LJA | CM Informatik AG | Auszubildender Applikationsentwickler |

Konventionen

In diesem Dokument wurden die folgenden Konventionen verwendet:

|  |  |
| --- | --- |
| Was | Beschrieb |
| Gleichstellung | Es wird innerhalb des vorliegenden Dokuments jeweils eine Form von Personenbezeichnungen (z.B. Projektleitende, Mitarbeitende etc.) verwendet, die alle Geschlechter einschliesst. |
| Grün & ??? | Grün hinterlegter Text betrifft Punkte, welche noch besprochen werden müssen |
| Gelb & !!! | Gelb hinterlegter Text betrifft Punkte, welcher bei der Umsetzung speziell beachtet werden müssen. |

Inhaltsverzeichnis

[1 Aufgabenstellung 8](#_Toc161908230)

[1.1 Titel der IPA 8](#_Toc161908231)

[1.2 Ausgangslage 8](#_Toc161908232)

[1.3 Detaillierte Aufgabenstellung 8](#_Toc161908233)

[1.4 Mittel und Methoden 9](#_Toc161908234)

[1.5 Vorkenntnisse 9](#_Toc161908235)

[1.6 Vorarbeiten 9](#_Toc161908236)

[1.7 Neue Lerninhalte 9](#_Toc161908237)

[1.8 Arbeiten in den letzten 6 Monaten 9](#_Toc161908238)

[1.9 Startblock 10](#_Toc161908239)

[1.10 IPA Arbeitstage 10](#_Toc161908240)

[1.11 Termine 10](#_Toc161908241)

[2 Organisation der Arbeitsergebnisse 11](#_Toc161908242)

[2.1 Versionsverwaltung von Dokumenten 11](#_Toc161908243)

[2.1.1 Ordnerstruktur 11](#_Toc161908244)

[2.1.2 Backup Strategie 12](#_Toc161908245)

[2.2 Versionsverwaltung von Code 12](#_Toc161908246)

[2.3 Arbeitsplatz 12](#_Toc161908247)

[3 Projektorganisation 13](#_Toc161908248)

[3.1 Beteiligte Personen 13](#_Toc161908249)

[3.1.1 Kandidat 13](#_Toc161908250)

[3.1.2 Verantwortliche Fachkraft 13](#_Toc161908251)

[3.1.3 Hauptexperte 13](#_Toc161908252)

[3.1.3.1 Nebenexperte 13](#_Toc161908253)

[3.2 Organigramm 14](#_Toc161908254)

[3.3 Projektmanagementmethode IPERKA 14](#_Toc161908255)

[3.3.1 IPERKA Projektphasen 15](#_Toc161908256)

[3.3.2 IPERKA: Zu beachten 16](#_Toc161908257)

[4 Zeitplan 17](#_Toc161908258)

[4.1 Meilensteine 17](#_Toc161908259)

[5 Arbeitsjournale 19](#_Toc161908260)

[5.1 01.03.2024 19](#_Toc161908261)

[5.2 04.03.2024 20](#_Toc161908262)

[5.3 06.03.2024 21](#_Toc161908263)

[5.4 07.03.2024 22](#_Toc161908264)

[5.5 08.03.2024 23](#_Toc161908265)

[5.6 11.03.2024 24](#_Toc161908266)

[5.7 13.03.2024 25](#_Toc161908267)

[5.8 14.03.2024 26](#_Toc161908268)

[5.9 15.03.2024 27](#_Toc161908269)

[5.10 18.03.2024 28](#_Toc161908270)

[5.11 20.03.2024 29](#_Toc161908271)

[5.12 21.03.2024 30](#_Toc161908272)

[6 Kurzfassung 32](#_Toc161908273)

[6.1 Kurze Ausgangssituation 32](#_Toc161908274)

[6.2 Umsetzung 32](#_Toc161908275)

[6.3 Ergebnis 32](#_Toc161908276)

[7 Informieren 33](#_Toc161908277)

[7.1 Vorhandene Skillmatrix 33](#_Toc161908278)

[8 Planen 35](#_Toc161908279)

[8.1 Nicht funktionale Anforderungen 35](#_Toc161908280)

[8.2 Funktionale Anforderungen 35](#_Toc161908281)

[8.3 Systemgrenzen / Schnittstellen 36](#_Toc161908282)

[8.3.1 Azure EntraID 36](#_Toc161908283)

[8.3.2 Blazor 36](#_Toc161908284)

[8.3.2.1 Blazor Server 36](#_Toc161908285)

[8.3.2.2 Blazor WebAssembly 37](#_Toc161908286)

[8.4 Entity Framework Core 37](#_Toc161908287)

[8.5 GUI-Mockup 38](#_Toc161908288)

[8.5.1 Skillmatrix Page 38](#_Toc161908289)

[8.5.2 Skillverwaltung Page 39](#_Toc161908290)

[8.5.2.1 Neue Skillkategorie Modal 40](#_Toc161908291)

[8.5.2.2 Skillkategorie löschen Modal 40](#_Toc161908292)

[8.5.2.3 Neuer Skill Modal 41](#_Toc161908293)

[8.5.2.4 Skill löschen Modal 41](#_Toc161908294)

[8.5.3 Benutzerverwaltung Page 42](#_Toc161908295)

[8.5.3.1 Neuer Mitarbeiter Modal 43](#_Toc161908296)

[8.5.3.2 Mitarbeiter löschen Modal 43](#_Toc161908297)

[8.6 Blazor Frontend Components 44](#_Toc161908298)

[8.7 Use Case Diagramm 46](#_Toc161908299)

[8.8 Datenbankdiagramm 46](#_Toc161908300)

[8.8.1 Mitarbeiter Entity 47](#_Toc161908301)

[8.8.2 MitarbeiterSkill Entity 47](#_Toc161908302)

[8.8.3 Skill Entity 47](#_Toc161908303)

[8.8.4 Skillkategorie Entity 47](#_Toc161908304)

[8.8.5 Log Entity 48](#_Toc161908305)

[8.9 Gesamtsystem / Systemgrenzen 48](#_Toc161908306)

[8.10 Testkonzept 49](#_Toc161908307)

[8.10.1 Testumgebung 49](#_Toc161908308)

[8.10.2 Testfälle 49](#_Toc161908309)

[9 Entscheiden 60](#_Toc161908310)

[9.1 Blazor Hosting Modelle 60](#_Toc161908311)

[9.1.1 Kommentar zur Entscheidung des Hosting-Modells 60](#_Toc161908312)

[9.2 Blazor Komponenten Bibliothek 61](#_Toc161908313)

[9.2.1 Kommentar zur Entscheidung der Komponenten-Bibliothek 61](#_Toc161908314)

[9.3 Entscheidungsgespräch 62](#_Toc161908315)

[10 Realisieren 63](#_Toc161908316)

[10.1 Erstellung des Blazor Projektes 63](#_Toc161908317)

[10.1.1 Installation MudBlazor 63](#_Toc161908318)

[10.1.2 Erstellung des MainLayout 63](#_Toc161908319)

[10.1.3 Erstellung des Nav Menus 64](#_Toc161908320)

[10.2 Datenbank erstellen 64](#_Toc161908321)

[10.2.1 Datenbank verbinden 64](#_Toc161908322)

[10.3 Datenbankobjekte / Entities erstellen 65](#_Toc161908323)

[10.3.1 Migration durchführen 66](#_Toc161908324)

[10.4 Authorization und Bootstrapping 67](#_Toc161908325)

[10.4.1 Bootstrapping 68](#_Toc161908326)

[10.4.2 Authorization Handler 68](#_Toc161908327)

[10.4.3 NavMenu Visibilität 69](#_Toc161908328)

[10.4.4 Zugriffsverweigerung 70](#_Toc161908329)

[10.5 Pages und Komponenten 70](#_Toc161908330)

[10.5.1 Bearbeiten von Einträgen 70](#_Toc161908331)

[10.5.2 DTOs 71](#_Toc161908332)

[10.5.3 Services 72](#_Toc161908333)

[10.5.4 Modale 72](#_Toc161908334)

[10.5.5 Skillsverwaltung Page 73](#_Toc161908335)

[10.5.6 Benutzerverwaltung Page 74](#_Toc161908336)

[10.5.7 Skillmatrix Page 74](#_Toc161908337)

[10.5.7.1 Datenabfrage für die Skillmatrix Page 74](#_Toc161908338)

[10.5.7.2 Editieren der Selbsteinschätzung 75](#_Toc161908339)

[10.5.7.3 Filtrierung 76](#_Toc161908340)

[10.6 Logging 77](#_Toc161908341)

[10.7 Technische Dokumentation 77](#_Toc161908342)

[10.8 Arbeits- und Fachmethodik 78](#_Toc161908343)

[11 Kontrollieren 79](#_Toc161908344)

[11.1 Testprotokoll 79](#_Toc161908345)

[11.2 Testfazit 84](#_Toc161908346)

[12 Auswerten 85](#_Toc161908347)

[12.1 Ausbaumöglichkeiten 85](#_Toc161908348)

[12.1.1 Hinzufügen eines Skills 85](#_Toc161908349)

[12.1.2 Übersicht 85](#_Toc161908350)

[12.1.3 Sortierung der Skillkategorien festlegen 85](#_Toc161908351)

[12.2 Soll / Ist Vergleich 86](#_Toc161908352)

[13 Schlusswort 87](#_Toc161908353)

[14 Glossar 88](#_Toc161908354)

[15 Verzeichnisse 89](#_Toc161908355)

[15.1 Tabellenverzeichnis 89](#_Toc161908356)

[15.2 Abbildungsverzeichnis 90](#_Toc161908357)

[16 Angang 92](#_Toc161908358)

Teil 1: Umfeld und Ablauf

# Aufgabenstellung

Beschreibung der Arbeit aus PkOrg

## Titel der IPA

Ablösung Mitarbeiter Skillmatrix

## Ausgangslage

CM Informatik benutzt ein unter den Mitarbeitern geteiltes Excel-Sheet in welchem alle Mitarbeiter eine Selbstbeurteilung in verschiedensten Fähigkeiten führen. Die bewerteten Fähigkeiten erstrecken sich dabei von Wissen in firmenspezifischen Fachanwendungen über Sprachkenntnissen bis zu Technologien. Dies soll unseren Mitarbeitern dabei helfen, bei Fragen oder Problemen die richtigen Ansprechpartner zu finden.

## Detaillierte Aufgabenstellung

Das in der Ausgangslage erwähnte Excel-Sheet soll durch eine Webanwendung abgelöst werden.

Die erwartete Anwendung muss dabei folgende Bestandteile beinhalten:

Einen Skill-Administations-Bereich in welchem Administrator-Benutzer neue Skill-Kategorien und dazugehörige Skills erfassen, bearbeiten und löschen können.

Einen User-Administrationsbereich in welchem Administrator-Benutzer neue Mitarbeiter erfassen, bearbeiten und löschen können, sowie Mitarbeiter zu Administrator-Benutzer machen können. Es ist eine Lösung zu finden, wie ein erstmaliges einrichten der Benutzer sichergestellt werden kann (Bootstraping).

Ein Skill-Edit-Bereich wo jeder Mitarbeiter sich selbst beurteilen kann, in jedem Skill auf einer Skala von 0 bis 4. Nur Administrator-Benutzer können auch andere Benutzer-Skills bearbeiten.

Einen Skill-Übersichts-Bereich wo alle Mitarbeiter mit ihren Skills angezeigt werden können. Hier soll man einerseits nach Mitarbeiter per Namen suchen können, aber auf die Übersicht nach Skill-Kriterien filtrieren können - z.B. alle Mitarbeiter mit SQL-Skill höher als 3 und C# höher als 2.

Der Benutzer der Applikation soll mittels Microsoft Entra ID authentifiziert werden und so für die Bearbeitung eines Mitarbeiterdatensatzes oder die Administrationsbereiche autorisiert werden.

Alle Daten sollen in einer MsSQL Datenbank persistiert werden.

Dort ist ebenfalls eine Logtabelle zu führen und relevante Benutzerinteratktionen sowie Fehlerereignisse sollen protokolliert werden.

Relevante Parameter - unter anderem für die Entra ID Anbindung sowie Datenbankzugriff - sind in einem Konfigurationsfile zu führen.

Es ist eine kurze technische Dokumentation zu erstellen in welcher die Einrichtung der Anwendung und der Konfigurationsparameter erklärt werden. Eine Benutzerdokumentation wird nicht erwartet.

## Mittel und Methoden

Microsoft Blazor

MsSql

C#

Microsoft Entity Framework

GitHub

## Vorkenntnisse

Entwicklung in C# in den letzten 6 Monaten (davor VBA)

Mehrjährige Erfahrung mit MsSQL

GitHub ist das vom Betrieb eingesetzte Versionskontrollsystem

## Vorarbeiten

Aufsetzen GitHub Repo

Einrichten App-Registration auf Azure-Tennant für Login-Integration

## Neue Lerninhalte

Integration Microsoft Entra ID

Microsoft Entity Framework

## Arbeiten in den letzten 6 Monaten

In den letzten 6 Monaten hat der Kandidat diverse Tasks wie Bugfixing oder neue Features für die CMI-Lösung in C# erarbeitet. Eines der grösseren Aufträge war das Hinzufügen eines neue custom Wizards. Dafür wurden inklusive Tests, zwei Arbeitstage aufgewendet.

Zusätzlich dazu hat der Kandidat ein Übungsprojekt in Blazor umgesetzt.

## Startblock

**API Startblock 1:** 26.02.2024 – 01.03.2024  
**PA-Durchführung:** 26.02.2024 – 30.03.2024

## IPA Arbeitstage

* Freitag, 01.03.2024
* Montag, 04.03.2024
* Mittwoch, 06.03.2024
* Donnerstag, 07.03.2024
* Freitag, 08.03.2024
* Montag, 11.03.2024
* Mittwoch, 13.03.2024
* Donnerstag, 14.03.2024
* Freitag, 15.03.2024
* Montag, 18.03.2024
* Mittwoch, 20.03.2024
* Donnerstag, 21.03.2024

## Termine

* 1. Besuchstag Hauptexperte, 06.03.2024:15:45-16:45
* 2. Besuchstag Hauptexperte, 15.03.2024: 13:30-14:30
* Präsentation IPA, 05.04.2024: 13:30-17:00

# Organisation der Arbeitsergebnisse

Im folgenden Abschnitt ist die Versionsverwaltung und mein Arbeitsplatz beschrieben.

## Versionsverwaltung von Dokumenten

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte BeschreibungFür die Versionierung und Verwaltung von Code verwendet die CMI GitHub. Sämtlicher Code und Dokumente, welche im Rahmen dieser IPA erstellt wurden, werden mit diesem Versionsverwaltungssystem verwaltet.

Es wird täglich ein Arbeitsjournal mit dem aktuellen Datum erstellt und hochgeladen.

Alle Änderungen von Dokumenten können in der Commit-History nachvollgezogen werden. Durch die Commits sind diese versioniert und können wiederhergestellt werden.

Abbildung 1 Versionsverwaltung GitHub

### Ein Bild, das Text, Screenshot, Schrift, Zahl enthält. Automatisch generierte BeschreibungOrdnerstruktur

Abbildung 2 Ordnerstruktur

|  |  |
| --- | --- |
| Ordner | Inhalt |
| IPA Dokumente | Das Stammverzeichnis für alle IPA-Dokumente |
| Balsamiq | Balsamiq Mockups |
| Dokumentation | Der IPA-Bericht, der jeden Tag aktualisiert wird |
| Draw.io | Diagramme und Abbildungen, die mit Draw.io erstellt wurden |
| Tagesjournal | Alle Tagesjournale |
| Zeitplan | Der Zeitplan |
| Dbdiagram.io | Das Datenbankdiagramm, das mit dbdiagram.io erstellt wurde |

Tabelle 1 Legende Ordnerstruktur

### Backup Strategie

Ein Bild, das Text, Screenshot, Zahl, Schrift enthält.

Automatisch generierte BeschreibungDas ganze Stammverzeichnis «IPA Dokumente» wird jeden Tag zusätzlich noch auf meiner privaten Ablage auf Google Drive gespeichert. Auf diese Dokumente kann ich jederzeit und von überall zugreifen. Ab dem Start der Realisierungsphase wird auch sämtlicher Sourcecode hochgeladen.

Abbildung 3 Backup Strategie

## Versionsverwaltung von Code

Das Projekt wird in Visual Studio programmiert und in der Realisierungsphase mindestens täglich mit Commits versioniert. Somit kann auf jeden Stand der Versionen zurückgegriffen werden.

## Arbeitsplatz

Ein Bild, das Im Haus, Bürogebäude, Mobiliar, Fenster enthält.

Automatisch generierte BeschreibungIn der gesamten IPA-Zeit werde ich in Aarau vor Ort arbeiten wo auch Dominik Baltzer immer anwesend sein wird. In der CMI haben wir dynamische Arbeitsplätze, das heisst jeder kann sich dort hinsetzen und arbeiten, wo er will. Währen der IPA werde ich jedoch probieren diesen Platz für mich zu «reservieren».

Abbildung 4 Arbeitsplatz

# Projektorganisation

In folgendem Abschnitt wird die Projektorganisation beschrieben.

## Beteiligte Personen

In diesem Kapitel werden alle beteiligten Personen mit Kontaktdaten und Rollen aufgeführt. Auftragnehmer, Projektleiter, Entwickler und Tester ist der Kandidat.

### Kandidat

|  |  |
| --- | --- |
| Vorname Name | Janosch Lio |
| E-Mail | Janosch.lio@cmiag.ch |
| Telefon Geschäft | +41 43 355 33 99 |
| Telefon Privat | +41 79 554 24 54 |
| Rollen | Kandidat / Auftragnehmer / Projektleiter / Entwickler / Tester |

Tabelle 2 Beteiligte Person Janosch Lio

### Verantwortliche Fachkraft

|  |  |
| --- | --- |
| Vorname Name | Dominik Baltzer |
| E-Mail | Dominik.baltzer@cmiag.ch |
| Telefon Geschäft | +41 43 355 33 99 |
| Telefon Privat | +41 79 795 82 26 |
| Rollen | Verantwortliche Fachkraft / Auftraggeber / Betreuer |

Tabelle 3 Beteiligte Person Dominik Baltzer

### Hauptexperte

|  |  |
| --- | --- |
| Vorname Name | Michelle Schär |
| E-Mail | michelle.schaer21@gmail.com |
| Telefon Privat | +41 79 855 50 95 |
| Rollen | Hauptexperte |

Tabelle 4 Beteiligte Person Michelle Schär

#### Nebenexperte

|  |  |
| --- | --- |
| Vorname Name | Dirk Decher |
| E-Mail | pex@decher-it-services.ch |
| Rollen | Nebenexperte |

Tabelle 5 Beteiligte Person Dirk Decher

## Organigramm

Ein Bild, das Text, Screenshot, Visitenkarte, Schrift enthält.

Automatisch generierte BeschreibungIm folgenden Organigramm wird die Konstellation der Projektorganisation veranschaulicht.

Abbildung 5 Organigramm

## Projektmanagementmethode IPERKA

Für die IPA habe ich IPERKA als Projektmanagementmethode gewählt, da ich vor allem an der Berufsfachschule BBB schon einige Projekte nach diesem Ablauf erstellt habe.

Ein Bild, das Text, Diagramm, Screenshot, Schrift enthält.

Automatisch generierte BeschreibungJeder Buchstabe der Methode steht für den jeweiligen Ablaufschritt.

Abbildung 6 IPERKA Methode

### IPERKA Projektphasen

|  |  |  |
| --- | --- | --- |
| Phase | Beschreibung | Fragen |
| Informieren | Der Auftrag wird geklärt und Informationen werden beschaffen und sortiert. Es sollen das wesentliche erkannt werden. | * Wer will von wem was? * Wie lautet der Auftrag oder die Aufgabe? * Wozu dient das Produkt? * Wie muss es sein? * Wer nutzt es? * Wann muss es beendet sein? |
| Planen | Die Ziele sollen definiert werden und ein Lösungsweg soll bestimmt werden. Es soll ein Arbeitsplan und einen dazugehörigen Zeitplan vorgenommen werden. | * Welche Prioritäten hat der Auftrag? * Welche Ressourcen stehen zur Verfügung? (Material- Infrastruktur-, Personal- und Zeitbedarf) * Ist die zeitliche Planung realistisch? * Sind die Meilensteine zeitlich definiert? * Welches sind die Prüfkriterien? |
| Entscheiden | Es soll entschieden werden welchen Lösungsweg man einschlägt. | * Macht meine Idee Sinn? * Kann ich mein Vorhaben fortsetzen? * Kann ich die Verantwortung tragen? |
| Realisieren | In dieser Phase soll das geplante und entschiedene umgesetzt werden. Es wird ebenfalls entschieden, ob das Projekt evtl. abgebrochen wird. | * Werden die geplanten Schritte auch tatsächlich umgesetzt? * Stimmt Mittel und Wege in Bezug auf das Ziel? * Gehen die Teilnehmenden nach Plan und Absprache vor oder gehen sie eigene Wege? * Gibt es Teilerfolge? * Werden Meilensteine erreicht? * Muss mehr „Dampf“ aufgesetzt werden? |
| Kontrollieren | Der Arbeitsauftrag soll nochmals durchgelesen werden und das Produkt anhand der erstellten Testfälle getestet werden. | * Können die Ziele und Teilziele erreicht werden? * Stimmt die Qualität des Produktes, bewährt sich das Produkt in Tests? * Sind alle Teile der Abmachungen berücksichtigt? * Werden Missstände offen kommuniziert? * Wie sind die Rückmeldungen an die Teilverantwortlichen? * Ist das Produkt aktuell, sind allfällige Änderungen nachgetragen und Abmachungen eingehalten worden? |
| Auswerten | Es wird eine Reflexion über das Produkt erstellt und Optimierungen / Erkenntnisse sollen festgehalten werden. | * Sind alle Ziele erreicht worden? * Wo gibt es Verbesserungsideen zum Produkt, zum Prozess, zur Zusammenarbeit, zum Umgang im Team? * Was machen wir das nächste Mal anders/besser? * Haben wir uns aufgrund dieser Erfahrung verändert/etwas gelernt/ Erkenntnisse gewonnen? * Wie wird sichergestellt, dass Verbesserungsmassnahmen bei künftigen Aufträgen beachtet werden? |

Tabelle 6 IPERKA Phasen

### IPERKA: Zu beachten

Ein Bild, das Text, Diagramm, Reihe enthält.

Automatisch generierte BeschreibungDa IPERKA in 6 Phasen unterteilt ist kann man schnell meinen, dass jeder Schritt exakt vom anderen abgetrennt ist. Dies ist jedoch nicht der Fall. Es ist nicht immer möglich, die Schritte deutlich voneinander zu trennen und sie können sich zeitlich überlappen.

Abbildung 7 IPERKA Phasen

Es ist möglich, zu planen, auch wenn die Informationsphase noch nicht abgeschlossen ist.

# Zeitplan

Der Zeitplan wurde in Excel umgesetzt und ist auf der Y-Achse nach IPERKA strukturiert. Jede Phase beinhaltet einen Detailbereich, die eine Beschreibung für die Tätigkeit beinhaltet.

Die X-Achse bildet die Zeit ab wobei jeder Tag in 4 Spalten für 2 Stunden aufgeteilt wurde. Zusätzlich wurde jeweils ein grauer Bereich für potenzielle Überstunden ergänzt.

|  |  |
| --- | --- |
| Zelle | Bedeutung |
|  | Geplante Zeit (Soll) |
|  | Tatsächliche Zeit (Ist) |
|  | Meilenstein |
|  | Potenzielle Überstunden |

Tabelle 7 Legende Zeitplan

## Meilensteine[[1]](#footnote-1)

Meilensteine (auch bekannt als Prüf- oder Messpunkte) zeigen wichtige Punkte im Projektverlauf und definieren den Zeitpunkt, an welchem ein bestimmtes Ziel oder Ereignis erreicht sein sollte. Sie zeigen den Weg bis zum Projektabschluss und welcher nur durch das Erreichen der Meilensteine erfolgreich abgeschlossen werden kann.

Ich habe drei Meilensteine definiert.

|  |  |  |  |
| --- | --- | --- | --- |
| Meilenstein | Bedeutung | IPERKA-Phasen | Datum |
| Konzept | Es wurde ein Konzept für das Realisieren erstellt. | Informieren, Planen, Entscheiden | 08.03.2024 |
| Realisierung | Das Projekt wurde realisiert. | Realisieren | 18.03.2024 |
| Abgabe | Das Projekt wurde kontrolliert, umgesetztes wurde getestet und ausgewertet. | Kontrollieren, Auswerten | 21.03.2024 |

Tabelle 8 Meilensteine



Abbildung 8 Zeitplan

# Arbeitsjournale

## 01.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Zeitplan und Tagesjournal erstellen | 2 | 2 | 0 | Nein (wird mit HEX am 06.03.2024 besprochen) |
| 2 | Grundgerüst Dokumentation erstellen | 1 | 1 | 0 | Ja |
| 3 | Dokumentation Teil 1 inkl. Organisation | 2 | 2 | 0 | Ja |
| 4 | Analyse der Skillsmatrix (Excel) und informieren über Blazor | 0.5 | 0.5 | 0 | Nein |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Zeitplan und Tagesjournal erstellen | 08:00 – 10:00 | 2 | 2 |
| Grundgerüst Dokumentation erstellen | 10:00 – 11:00 | 1 | 1 |
| Dokumentation Teil 1 inkl. Organisation | 11:00 – 12:00  13:00 – 14:00 | 2 | 2 |
| Dokumentation schreiben | 14:00 – 15:00  15:00 – 16:00 | 2 | 2 |
| Analyse der Skillsmatrix (Excel) und Informieren über Blazor | 16:00 – 16:30 | 0.5 | 0.5 |
| Arbeitsjournal | 16:30 – 17:00 | 0.5 | 0.5 |

|  |
| --- |
| Probleme |
| Ich bin heute sehr gut mit meinen geplanten Arbeiten vorangekommen und hatte eigentlich keine Probleme beim Starten. Worüber ich jedoch gestolpert bin, war am Morgen die Zeitplanung, die komischerweise nicht aufging. Da ich jedoch genügend Zeit für die Erstellung des Zeitplanes und Arbeitsjournals eingeplant habe, konnte ich das Problem ohne Überschreitung der geplanten Zeit lösen. |

|  |
| --- |
| Hilfestellung / Überzeiten |
| Ich musste keine Überzeiten aufschreiben was mich erleichtert hat. Am Morgen habe ich mit Dominik den Zeitplan nochmals überflogen und angeschaut, ob dieser so realistisch ist. |

|  |
| --- |
| Tagesreflektion |
| Der Start der IPA lief am Morgen, obwohl ich etwas nervös war, sehr gut. Ich konnte alle meine geplanten Arbeiten zeitgemäss erledigen / starten. Ich habe jedoch bedenken, ob ich genügend Zeit für die Phase «Informieren» eingeplant habe. Ich konnte heute zwar schon starten, denke aber, dass ich mehr als 1 Stunde insgesamt brauchen werde. Ich hoffe, dass ich durch diese Fehlplanung nicht zu sehr in Verzug gerate. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* Voraus
* **Im Plan**
* In Verzug

## 04.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Analyse der Skillsmatrix (Excel) und Informieren über Blazor | 0.5 | 0.5 | 0 | Ja |
| 2 | Anforderungsanalyse | 1 | 1 | 0 | Ja |
| 3 | GUI-Mockup (mit Abnahmesitzung) | 3 | 3 | 0 | Ja |
| 4 | Testfälle definieren | 1 | 1 | 0 | Nein |
| 5 | Dokumentation schreiben | 2 | 2 | 0 | Nein |
| 6 | Tagesjournal schreiben | 0.5 | 0.5 | 0 | Ja |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Analyse der Skillsmatrix (Excel) und Informieren über Blazor | 08:00 – 08:30 | 0.5 | 0.5 |
| Anforderungsanalyse | 08:30 – 09:30 | 1 | 1 |
| GUI-Mockup | 09:30 – 12:00 | 2.5 | 2.5 |
| Dokumentation schreiben | 13:00 – 14:00 | 1 | 1 |
| Testfälle definieren | 14:00 – 15:00 | 1 | 1 |
| Abnahmesitzung | 15:00 – 15:30 | 0.5 | 0.5 |
| Dokumentation schreiben | 15:30 – 16:30 | 1 | 1 |
| Tagesjournal schreiben | 16:30 – 17:00 | 0.5 | 0.5 |

|  |
| --- |
| Probleme |
| - |

|  |
| --- |
| Hilfestellung / Überzeiten |
| Ich musste keine Überzeiten aufschreiben. Um 15:00 Uhr habe ich mit Dominik eine Abnahmesitzung bezüglich des Mockups gemacht, wo wir das Mockup gemeinsam anschauten, und Anpassungen diskutierten. |

|  |
| --- |
| Tagesreflektion |
| Ich konnte heute alle geplanten Arbeiten plangemäss erledigen, ohne zu schnell oder zu langsam zu sein. Ich bin nicht mehr so nervös wie am ersten Tag, was die Arbeit etwas erleichtert hat. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* Voraus
* **Im Plan**
* In Verzug

## 06.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Expertenbesuche (Online) | 1 | 1 | 0 | Ja |
| 2 | Testfälle definieren | 0.5 | 0.5 | 0 | Ja |
| 3 | Dokumentation schreiben | 2 | 2 | 0 | Ja |
| 4 | Arbeitsjournal | 0.5 | 0.5 | 0 | Nein |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Testfälle | 13:00 – 13:30 | 0.5 | 0.5 |
| Dokumentation schreiben | 13:30 – 15:30 | 2 | 2 |
| Tagesjournal | 15:30 – 15:45  16:45 – 17:00 | 0.5 | 0.5 |
| Expertenbesuche (Online) | 15:45 – 16:45 | 1 | 1 |

|  |
| --- |
| Probleme |
| - |

|  |
| --- |
| Hilfestellung / Überzeiten |
| Bei den Testfällen war ich unsicher, ob ich bei den Funktionalen Anforderungen, nicht Funktionale Anforderungen aufgelistet habe. Ich habe dann mit Dominik die Anforderungen angeschaut und konnten gemeinsam entscheiden zu welcher Art von Anforderung diese gehörten. |

|  |
| --- |
| Tagesreflektion |
| Heute Morgen war ich in der Berufsschule und hatte Sport. Ich konnte am Nachmittag alle geplanten Arbeiten erledigen und die Testfälle abschliessen. Ich bin froh und erleichtert, dass ich nicht im Verzug bin und dem Plan nicht hinterherkomme. Am Nachmittag hatte das Expertengespräch mit Frau Schär, was sehr gut lief. Wir haben die Bewertungskriterien nochmals angeschaut und konnte bei Unsicherheiten nochmals detaillierter Fragen stellen. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* Voraus
* **Im Plan**
* In Verzug

## 07.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Testfälle definieren | 0.5 | 0.5 | 0 | Ja |
| 2 | Mögliche Lösungswege erfassen | 1 | 0.5 | -0.5 | Ja |
| 3 | Datenbankdiagramm | 1 | 1.5 | +0.5 | Ja |
| 4 | Entscheiden von Lösungswegen | 2 | 2 | 0 | Ja |
| 5 | Zeitpuffer für die Dokumentation | 2 | 2 | 0 | Ja |
| 6 | Projekt erstellen | 1 | 1 | 0 | Ja |
| 7 | Tagesjournal schreiben | 0.5 | 0.5 | 0 | Ja |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Testfälle fertigstellen | 08:00 – 08:30 | 0.5 | 0.5 |
| Mögliche Lösungswege erfassen | 08:30 – 09:00 | 0.5 | 1 |
| Datenbankdiagramm erstellen | 09:00 – 10:30 | 1.5 | 1 |
| Entscheiden von Lösungswegen | 10:30 – 12:00  13:00 – 13:30 | 2 | 2 |
| Zeitpuffer für die Dokumentation (Dokumentation anpassen) | 13:30 – 15:30 | 2 | 2 |
| Projekt erstellen | 15:30 – 16:30 | 1 | 1 |
| Tagesjournal schreiben | 16:30 – 17:00 | 0.5 | 0.5 |

|  |
| --- |
| Probleme |
| Ein Problem des heutigen Tages war, dass ich nicht wusste welches Tool ich für die Erstellung des Datenbankdiagrammes verwenden sollte. In der Schule habe «Visual Paradigm» verwendet, jedoch lief die Lizenz dieser Software schon ab. Ich habe dann ein Online-Tool namens «dbdiagram.io» gefunden. Da dies neu für mich war, habe ich mehr Zeit als geplant für das Erstellen des Datenbankdiagrammes aufgewendet was zum Glück kompensiert werden konnte, da ich beim Erfassen der Lösungswege weniger Zeit als geplant verwendet habe. |

|  |
| --- |
| Hilfestellung / Überzeiten |
| - |

|  |
| --- |
| Tagesreflektion |
| Heute konnte ich wieder nach Plan arbeiten. Beim Erstellen des Datenbankdiagrammes habe ich etwas länger gebraucht als geplant, was mich für eine kurze Zeit etwas verunsicherte, bin dann aber wieder nachgekommen. Am Nachmittag habe ich mit Dominik noch eine kleine Sitzung bezüglich meiner Entscheidungen gemacht, die sehr gut verlief. Er war mit meinen Entscheidungen einverstanden. Anschliessend konnte ich das Projekt erstellen und vorbereiten. Somit habe ich für die Realisierungsphase die grobe Grundlage erstellt. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* Voraus
* **Im Plan**
* In Verzug

## 08.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Datenbank, Datenbankobjekte erstellen und konfigurieren | 2 | 2 | 0 | Ja |
| 2 | Authorization und Bootstrapping | 2 | 2 | 0 | Ja |
| 3 | Skill-Admin Page und dazugehörige Services | 1.5 | 1.5 | 0 | Nein |
| 4 | Dokumentation schreiben | 2 | 2 | 0 | Ja |
| 5 | Arbeitsjournal | 0.5 | 0.5 | 0 | Ja |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Datenbank, Datenbankobjekte erstellen und konfigurieren | 08:00 – 10:00 | 2 | 2 |
| Authorization und Bootstrapping | 10:00 – 12:00 | 2 | 2 |
| Skill-Admin Page und dazugehörige Services | 13:00 – 14:30 | 1.5 | 1 |
| Dokumentation schreiben | 14:30 – 16:30 | 2 | 2 |
| Arbeitsjournal | 16:30 – 17:00 | 0.5 | 0.5 |

|  |
| --- |
| Probleme |
| Heute hatte ich ein kleines Problem, das mich kurz aufgehalten hatte. Nachdem ich die Relationen und Entities in der OnModelCreating Methode definiert hatte wollte ich die Migration mittels Command «Add-Migration» ausführen. Dort wurde ich darauf hingewiesen, dass dies nicht richtig funktionierte. Das hat mich kurz zeit gekostet, bin dann aber trotzdem rechtzeitig fertig geworden und konnte das Problem lösen. |

|  |
| --- |
| Hilfestellung / Überzeiten |
| Wie oben bei den Problemen beschrieben, hat die Migration nicht funktioniert wie gewünscht. Zur hilfestellung habe ich ChatGPT gebraucht. Ich habe ihm den Code der vier Entities «Skill», «Skillkategorie», «Mitarbeiter» und «MitarbeiterSkill» mitgegeben und gefragt, was denn nicht richtig sei. Er hat mich auf einen Flüchtigkeitsfehler hingewiesen, und zwar, dass ich in der «MitarbeiterSkill» Entity die id mit dem Datentyp «string» versehen habe. Nachdem dies angepasst wurde hat die Migration einwandfrei funktioniert. |

|  |
| --- |
| Tagesreflektion |
| Der heutige Tag verlief wieder nach Plan. Am Morgen hatte ich zwar ein kleines Problem mit der Migration, was jedoch schnell behoben werden konnte, ohne in Verzug zu geraten. Am Nachmittag konnte ich mit der Programmierung der ersten Page beginnen was soweit gut verlief. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* Voraus
* **Im Plan**
* In Verzug

## 11.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Skill-Admin Page und dazugehörige Services | 1.5 | 4 | +2.5 | Ja |
| 2 | User-Admin Page und dazugehörige Services | 3 | 1.5 | -1.5 | Ja |
| 3 | Skill-Edit Page | 1 | 0 | -1 | Nein |
| 4 | Dokumentation schreiben | 2 | 2 | 0 | Ja |
| 5 | Arbeitsjournal | 0.5 | 0.5 | 0 | Ja |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Skill-Admin Page und dazugehörige Services | 08:00 – 12:00 | 4 | 2 |
| User-Admin Page und dazugehörige Services | 13:00 – 14:30 | 1.5 | 3 |
| Dokumentation schreiben | 14:30 – 16:30 | 2 | 2 |
| Arbeitsjournal | 16:30 – 17:00 | 0.5 | 0.5 |

|  |
| --- |
| Probleme |
| Heute hatte ich das Problem, dass beim Manuellen Refresh der Browser Page anschliessend keine Daten mehr geladen werden. |

|  |
| --- |
| Hilfestellung / Überzeiten |
| Das oben beschriebene Problem konnte ich mit ChatGPT lösen. Ich habe folgenden Prompt eingegeben: «Ich habe eine Blazor Server webapplikation. Wenn ich die Seite manuell aktualisiere, laden anschliessend keine Daten mehr. Hier die Fehlermeldung in der Konsole: Uncaught (in promise) DOMException: Failed to execute 'setStartBefore' on 'Range': the given Node has no parent.”. Darauf hat er mir vorgeschlagen “OnAfterRenderAsync” zu verwenden anstatt “OnInitializedAsync”, da «OnInitializedAsync» nach dem Initialisieren der Komponente aufgerufen wird, jedoch nicht vor dem Rendern. Dies kann Probleme verursachen. |

|  |
| --- |
| Tagesreflektion |
| Der heutige Tag lief vor allem am Anfang etwas holprig, da ich vier Stunden für die Skillsverwaltungs-Page aufgewendet habe, jedoch nur 1.5 Stunden für diesen Tag eingeplant hatte. Hier bin ich etwas in Stress geraten, konnte mich jedoch trotzdem beherrschen und die Seite fertig programmieren. Die verlorene Zeit konnte ich zum glück mit der Programmierung der Benutzerverwaltung-Page wieder gut machen, da ich für diese Seite nur 1.5 Stunden brauchte. Eigentlich wäre heute noch das Programmieren der Skillseditier-Page geplant gewesen, jedoch fällt diese Aufgabe komplett weg, da ich die Funktionalität des Editieren / der Selbsteinschätzung direkt in die Skillmatrix Page implementiere. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* Voraus
* **Im Plan**
* In Verzug

## 13.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Skill-Edit Page | 1.5 | 0 | -1.5 | Ja |
| 2 | Dokumentation schreiben | 2 | 2 | 0 | Ja |
| 3 | Arbeitsjournal | 0.5 | 0.5 | 0 | Ja |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Dokumentation schreiben | 13:00 – 15:00 | 4 | 2 |
| Skillmatrix Page mit Filtering | 15:00 – 16:30 | 1.5 | 0 |
| Arbeitsjournal | 16:30 – 17:00 | 0.5 | 0.5 |

|  |
| --- |
| Probleme |
| - |

|  |
| --- |
| Hilfestellung / Überzeiten |
| Während ich mit der Skillmatrix-Page angefangen habe, habe ich mich gefragt, ob ich verschachtelte DTOs oder alleinstehende DTOs erstellen sollte. Es geht darum, ob ich für die Page «Skillmatrix» ein MitarbeiterDTO erstellen soll, das eine Liste von Skillkategorien beinhaltet, was wiederum eine Liste mit Skills enthält (also 3 Levels). Daraufhin habe ich Dominik für seinen Rat gefragt. Er meinte, ein Mix zwischen den beiden wäre gut. Das heisst, dass ich die Mitarbeiter allein lade, und anhand des Angeklickten Mitarbeiters, die SkillkategorieDTO mit einer Liste von Skills lade. Das hat für mich Sinn gemacht. |

|  |
| --- |
| Tagesreflektion |
| Heute Morgen war ich wieder an der Berufsschule. Ich habe zuerst mit der Bearbeitung und dem Überarbeiten der Dokumentation gestartet. Da ich für heute die Skill-Edit Page geplant hatte, diese jedoch in die Skillmatrix-Page integriert wird und somit eigentlich wegfällt, konnte ich mit der Erstellung der Skillmatrix-Page anfangen. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* **Voraus**
* Im Plan
* In Verzug

## 14.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Skill-Edit Page | 0.5 | 0 | -0.5 | Ja |
| 2 | Skillmatrix Page mit Filtering | 4 | 5.5 | +1.5 | Nein |
| 3 | Logging | 1 | 1 | 0 | Ja |
| 4 | Dokumentation schreiben | 2 | 1 | -1 | Ja |
| 5 | Arbeitsjournal | 0.5 | 0.5 | 0 | Ja |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Skillmatrix Page mit Filtering | 08:00 – 12:00  13:00 – 14:30 | 5.5 | 4 |
| Dokumentation schreiben | 14:30 – 15:30 | 1 | 2 |
| Logging | 15:30 – 16:30 | 1 | 1 |
| Arbeitsjournal schreiben | 16:30 – 17:00 | 0.5 | 0.5 |

|  |
| --- |
| Probleme |
| Heute hatte ich das Problem, dass ich 1.5 Stunden länger für die Skillmatrix-Page brauchte, als ich eingeplant hatte. Ich habe so lange gebraucht, da die Skillkategorie und Skills beim Auswählen eines neuen Mitarbeiters nicht entsprechend geladen wurden. Ich habe den Debugger verwendet und beim Klick-Even eines Mitarbeiters einen Haltepunkt gesetzt. Somit habe ich den Fehler gefunden und konnte ihn lösen. Leider bin ich noch nicht zum Dokumentieren gekommen. Dort werde ich morgen noch nachholen. |

|  |
| --- |
| Hilfestellung / Überzeiten |
| Als ich beim Logging-Teil angekommen bin, habe ich Dominik gefragt, ob ich das so in Ordnung ist. Er stimmte zu, jedoch hat er mit zum Loggen noch «SeriLog» empfohlen. Ich habe mich mal reingelesen und mich dafür entschieden, «Serilog» als Logger zu verwenden. |

|  |
| --- |
| Tagesreflektion |
| Heute bin ich am Mittag etwas in Stress gekommen, da ich für die Skillmatrix Page länger brauchte als geplant. Dokumentieren konnte ich diese auch noch nicht. Für diese Situation habe ich den Realisations Puffer eingebaut, den ich morgen verwenden werde um dies noch nachzuholen. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* Voraus
* Im Plan
* **In Verzug**

## 15.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Readme: Technische Dokumentation zur Einrichtung der Anwendung | 2 | 2 | 0 | Ja |
| 2 | Zeitpuffer für die Realisierung | 2 | 2 | 0 | Ja |
| 3 | Expertenbesuche (Online) | 1 | 0.5 | -0.5 | Ja |
| 4 | Kontrolle des Projektes | 0.5 | 0.5 | 0 | Ja |
| 5 | Dokumentation schreiben | 2 | 2 | 0 | Ja |
| 6 | Arbeitsjournal | 0.5 | 0.5 | 0 | Ja |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Readme: Technische Dokumentation zur Einrichtung der Anwendung | 08:00 – 10:00 | 2 | 2 |
| Zeitpuffer für die Realisierung | 10:00 – 12:00 | 2 | 2 |
| Testfälle Dokumentieren | 13:00 – 13:30 | 0.5 | 0 |
| Expertenbesuche (Online) | 13:30 – 14:00 | 0.5 | 1 |
| Dokumentation schreiben | 14:00 – 15:00  15:00 – 16:00 | 2 | 2 |
| Kontrolle des Projekts | 16:00 – 16:30 | 0.5 | 0.5 |
| Arbeitsjournal schreiben | 16:30 – 17:00 | 0.5 | 0.5 |

|  |
| --- |
| Hilfestellung / Überzeiten |
| Heute war ich am Schreiben der Technischen Dokumentation zur Einrichtung der Skillmatrix. Ich habe diese mit Dominik angeschaut und ein paar Fragen zur CMI-Vorlage gestellt und was alles drin sein muss. Er hat mir ebenfalls empfohlen noch ein SQL-Script oder eine eingebaute Funktionalität zu erweitern, die das Aktualisieren der Datenbank (Migrationen) ermöglicht. Ich habe mich für die Funktionalität entschieden. Diese ist dokumentiert jedoch hier kurz erklärt: Im Program.cs File wird die Methode «Migrate» auf dem SkillmatrixDbContext aufgerufen, was die Datenbank erstellt, falls sie noch nicht vorhanden ist und alle Migrationen, die noch nicht auf der Datenbank sind, ausführt. |

|  |
| --- |
| Tagesreflektion |
| Der Heutige Tag lief ganz gut. Ich konnte die Technische Dokumentation zur Einrichtung der Skillmatrix abschliessen und hatte anschliessend im Realisationspuffer noch Zeit, die Skillmatrix Page zu dokumentieren, wofür ich gestern leider keine Zeit mehr hatte. Somit konnte ich meinen Planungsverzug wieder nachholen und mich auf die bevorstehenden Aufträge konzentrieren. Ich hatte am Nachmittag ebenfalls noch das Expertengespräch mit Frau Schär was sehr gut verlief. Wir hatten nur 30 Minuten was mir ermöglichte noch mehr Zeit in die Testfälle zu stecken. Diese hatte ich eigentlich erst auf den 18.03.2024 geplant. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* **Voraus**
* Im Plan
* In Verzug

## 18.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Kontrolle des Projektes | 1.5 | 1 | -0.5 | Ja |
| 2 | Testfälle dokumentieren | 2 | 0.5 | -1.5 | Ja |
| 3 | Dokumentation erweitern (inkl. Tagesjournal) | 1 | 1 | 0 | Ja |
| 4 | Selbstreflexion | 1.5 | 1.5 | 0 | Ja |
| 5 | Dokumentation schreiben | 1.5 | 3.5 | +2 | Ja |
| 6 | Arbeitsjournal | 0.5 | 0.5 | 0 | Ja |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Kontrolle des Projektes | 08:00 – 09:00 | 1 | 1.5 |
| Testfälle dokumentieren | 09:00 – 09:30 | 0.5 | 2 |
| Dokumentation erweitern (inkl. Tagesjournal) | 09:30 – 10:30 | 1 | 1 |
| Selbstreflexion | 10:30 – 12:00 | 1.5 | 1.5 |
| Dokumentation schreiben | 13:00 – 16:30 | 3.5 | 1.5 |
| Arbeitsjournal schreiben | 16:30 – 17:00 | 0.5 | 0.5 |

|  |
| --- |
| Probleme |
| - |

|  |
| --- |
| Hilfestellung / Überzeiten |
| Ich habe die Dokumentation am Freitag an Joel Weiss von meinem Scrum Team, der vor drei Jahren ebenfalls seine Lehre abgeschlossen hat, zugereicht und ihn gefragt, ob er diese nach Rechtschreibfehler und auch Formatierungsfehler kontrollieren kann. Seine Vorschläge habe ich heute angepasst und danke ihm dafür. |

|  |
| --- |
| Tagesreflektion |
| Heute verlief der Tag ruhig und konnte wieder alle meine geplanten Aufgaben abschliessen. Ich habe heute etwas mehr Zeit in die Dokumentation gesteckt, da ich diese noch kontrollierte und verschiedene Rechtschreibfehler korrigiert habe. Ich habe ebenfalls den Code nochmals überarbeitet, Kommentare eingefügt und das Naming verschiedener Methoden angepasst. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* Voraus
* **Im Plan**
* In Verzug

## 20.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Selbstreflexion | 0.5 | 0.5 | 0 | Ja |
| 2 | Auswertung Zeitplan | 1 | 1 | 0 | Ja |
| 3 | Dokumentation schreiben | 2 | 2 | 0 | Ja |
| 4 | Arbeitsjournal | 0.5 | 0.5 | 0 | Ja |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Kontrolle des Projektes | 13:00 – 13:30 | 0.5 | 0.5 |
| Auswertung Zeitplan | 13:30 – 14:30 | 1 | 1 |
| Dokumentation schreiben | 14:30 – 16:30 | 2 | 2 |
| Arbeitsjournal schreiben | 16:30 – 17:00 | 0.5 | 0.5 |

|  |
| --- |
| Probleme |
| - |

|  |
| --- |
| Hilfestellung / Überzeiten |
| Ich habe am Nachmittag Frau Schär kontaktiert und sie bezüglich der Code Abgabe gefragt, ob ich den selbst erstellten Code in eine Word Datei kopieren soll oder ob ich die Code Files in einen Ordner einfügen kann, da mir dies Dominik so vorgeschlagen hat und ich es so auch besser finde. Sie erwähnte, dass sie diesen Weg auch akzeptiert. |

|  |
| --- |
| Tagesreflektion |
| Heute Morgen hatte ich an der BBB, Sport. Am Nachmittag habe ich die Dokumentation mit dem Abbildungsverzeichnis und Tabellenverzeichnis erweitert und allfällige Rechtschreibfehler und Formatierungsfehler angepasst. Ich bin heute wieder mit all meinen geplanten Arbeiten durchgekommen und freue mich morgen auf die Abgabe der IPA. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* Voraus
* **Im Plan**
* In Verzug

## 21.03.2024

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Geplante Tagesziele | | | | | |
| **Nr.** | **Ziel** | **Soll** | **Ist** | **Abweichung** | **Abgeschlossen** |
| 1 | Kontrolle und Abgabe der Dokumentation | 3.5 | 3.5 | 0 | Ja |
| 2 | Arbeitsjournal | 0.5 | 0.5 | 0 | Ja |

|  |  |  |  |
| --- | --- | --- | --- |
| Tätigkeit | Zeit | Aufwand | Ziel |
| Kontrolle und Abgabe der Dokumentation | 08:00 – 11:30 | 3.5 | 3.5 |
| Arbeitsjournal schreiben | 11:30 – 12:00 | 0.5 | 0.5 |

|  |
| --- |
| Probleme |
| Ich habe gemerkt, dass der ganze Text in Textgrösse 9 geschrieben wurde, was etwas klein ist. Ich habe deshalb sämtlichen Text auf Schriftgrösse 10 gestellt, was die logischerweise die Formatierung änderte. Ich konnte dies jedoch rechtzeitig noch anpassen. |

|  |
| --- |
| Hilfestellung / Überzeiten |
| - |

|  |
| --- |
| Tagesreflektion |
| Heute habe ich die Dokumentation nochmals durchgelesen und die Kriterien dazu angeschaut und kontrolliert. Ich habe das FArbeit Dokument ausgedruckt und signiert. Ich bin froh, dass ich nun fertig bin mit der Dokumentation und mich nun auf die Präsentation, die Berufsschule und die BM konzentrieren kann. |

**Ich binEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte BeschreibungEin Bild, das Screenshot, Text, Schrift, weiß enthält.

Automatisch generierte Beschreibung:**

* Voraus
* **Im Plan**
* In Verzug

Teil 2: Projekt

# Kurzfassung

Folgend die Kurzfassung.

## Kurze Ausgangssituation

Die CM Informatik verwendet eine Excel-Datei namens «Skillmatrix», in der Mitarbeiter sich in verschiedenen vorgegebenen Skills einschätzen können. Dies soll dazu dienen, dass man bei Fragen auf die richtige Person mit dem vorhandenen Wissen zu finden.

Das Problem dieser Excel-Matrix ist, dass sie sehr unübersichtlich ist und nicht mehr aktuell geführt wird.

Die Excel-Skillmatrix soll als Webanwendung abgelöst werden. Es soll möglich sein, Mitarbeiter, Skillkategorien und Skills in einer Liste / Tabelle verwalten zu können, sowie Mitarbeiterbewertungen für jeden Skill zu erfassen. Ausserdem soll mit einer Suchfunktion schnell ein Mitarbeiter gefunden werden, der bestimmte Skillkriterien erfüllt.

## Umsetzung

Das Produkt der IPA wird mit Blazor Server, einem .NET-Front-End-Webframework, realisiert. In der Entscheidungsphase wurde zwischen Blazor Server und Blazor WebAssembly entschieden. Für die Komponenten wurde zwischen MudBlazor und Blazor Bootstrap entscheiden, wobei MudBlazor gewählt wurde.

Mittels Azure Entra ID werden die Benutzer authentifiziert.

Es wird das Entity Framework Core verwendet, das die Daten in einer MSSQL Datenbank speichert, verwaltet und abruft.

## Ergebnis

Das Produkt entspricht allen erfassten Anforderungen. Der Benutzer kann sich beim Besuchen der Webapplikation mit seinem Office 365 Account der CM Informatik einloggen. Falls dieser von einem Administrator erfasst wurde, kann er seine Selbsteinschätzung in den verschiedenen Skills eintragen.

Administratoren können Mitarbeiter, Skillkategorien und Skills erfassen, bearbeiten und löschen. Sie können ebenfalls von jedem Mitarbeiter die Selbsteinschätzung anpassen.

Mittels Bootstraping kann im appsettings.json die Mail des Benutzers angegeben werden, die beim Anmelden immer Admin Rechte haben soll. Dies ermöglicht das erstmalige Einrichten der Benutzer / Mitarbeiter.

# Informieren

Folgend die Informierungsphase.

## Vorhandene Skillmatrix

Die CMI verwaltet eine riesige Excel-Skillmatrix im CMI-Richclient in der sich jeder Mitarbeiter in den vorhandenen Skills selbst beurteilen kann.

Die Skillmatrix wurde grundsätzlich erstellt, damit Mitarbeiter bei Fragen in einem bestimmten Bereich auf die richtige Person mit dem Vorhandenen Wissen zugehen können.

Auf der Y-Achse sind alle Kategorien mit den mit den jeweiligen Skills aufgelistet. Ein Beispiel dafür ist die Kategorie «Sprachen» in der Französisch, Italienisch und Englisch enthalten sind.

Auf der X-Achse sind oben alle Firmenabteilungen mit dem jeweiligen Kürzel der Mitarbeiter aufgelistet. Es ist ebenfalls ein Bereich «Skillsverteilung» vorhanden, der angibt wie viele Mitarbeiter sich in den jeweiligen Skills mit welchem Level eingeschätzt haben. Die Mitarbeiter können sich von 1 bis 4 in den Skills selbst beurteilen.

Folgende Tabelle / Legende zeigt die Kriterien der Levels, um die Einschätzung der Mitarbeiter zu erleichtern.

|  |  |  |
| --- | --- | --- |
| Level | Kriterien | Symbol |
| 0 | Kennt das Thema nicht und hat keine Erfahrung | Ein Bild, das Kreis enthält.  Automatisch generierte Beschreibung |
| 1 | Kennt das Thema, hat aber noch keine Erfahrung | Ein Bild, das Kreis, Symbol enthält.  Automatisch generierte Beschreibung |
| 2 | Hat sich in mind. einem Projekt mit dem Thema befasst und sich Know-how angeeignet |  |
| 3 | Konnte seine Erfahrung in diversen Projekten anwenden und erweitern | Ein Bild, das Symbol, Logo, Kreis, Design enthält.  Automatisch generierte Beschreibung |
| 4 | Ist ein Experte | Ein Bild, das Kreis enthält.  Automatisch generierte Beschreibung |

Tabelle 9 Skillmatrix Levels

Ein Bild, das Text, Screenshot, parallel, Zahl enthält.

Automatisch generierte BeschreibungNachfolgend zeigt ein Screenshot einen Teil der Skillmatrix.

Abbildung 9 Excel-Skillmatrix

Es ist erkenntlich, dass die Tabelle / Matrix sehr unübersichtlich und unvollständig ist und begründet auch, warum diese abgelöst werden soll. Folgend die Pro und Kontras einer Excel Liste:

Pro:

* Schnell und billig erstellt

Contra:

* Schlechte Skalierung
* Nicht Up-To-Date
* Nicht übersichtlich

Die Möglichkeit die Skillmatrix im CMI Richclient zu integrieren wäre vorhanden, hier haben wir uns aber bewusst dagegen entschieden. Die Skillmatrix soll ein abgegrenztes System sein.

# Planen

In diesem Kapitel werden die Anforderungen aus dem PkOrg in funktionale und nicht funktionale Anforderungen unterteilt, um im später folgendem Testkonzept Testfälle extrahieren zu können.

## Nicht funktionale Anforderungen

* Die Benutzeroberfläche sollte intuitiv und einfach zu bedienen sein, um eine einfache Navigation zu gewährleisten
* Der Code soll gut strukturiert und kommentiert sein, was die Wartbarkeit und die Weiterentwicklung erleichtert
* Alle Daten sollen in einer MsSQL-Datenbank gespeichert werden
* Es soll eine Logtabelle geführt werden, die Benutzerinteraktionen und Fehlerereignisse in einer Tabelle protokolliert
* Azure Entra ID-Anbindung und Datenbank Verbindung sollen in einem Konfigurationsfile geführt sein
* Es soll eine kurze technische Dokumentation erstellt werden, die das Einrichten der Applikation erklärt

## Funktionale Anforderungen

|  |  |
| --- | --- |
| Anforderungsnummer | Anforderung |
| AN-01 | Administratoren können neue Skill-Kategorien und dazugehörige Skills erfassen, bearbeiten und löschen |
| AN-02 | Administratoren können neue Mitarbeiter erfassen, bearbeiten und löschen |
| AN-03 | Administratoren können Mitarbeiter zu Administrator-Benutzer «befördern» |
| AN-04 | Mitarbeiter sollen sich selbst in den vorhandenen Skills von 0 bis 4 bewerten können |
| AN-05 | Administratoren sollen die Möglichkeit haben die Skills von anderen Benutzern zu verändern |
| AN-06 | Es soll eine Skill-Übersicht mit allen Mitarbeitern und ihren Skills vorhanden sein. |
| AN-07 | Auf der Skill-Übersicht soll man nach Namen suchen können und Mitarbeiter nach bestimmten Skill-Kriterien filtern können.  Beispiel: alle Mitarbeiter mit SQL-Skill höher als 3 und C# höher als 2 |
| AN-08 | Benutzer der Anwendung sollen mittels Microsoft Entra ID authentifiziert werden. |
| AN-09 | Das erstmalige Einrichten von Benutzern ist erforderlich (Bootstraping) |
| AN-10 | Fehleingaben in Eingabefelder werden überprüft und der Benutzer wird darauf hingewiesen |

Tabelle 10 Funktionale Anforderungen

## Systemgrenzen / Schnittstellen

Folgend die Systemgrenzen und Schnittstellen der IPA.

### Azure EntraID[[2]](#footnote-2)

Die CMI verwendet bereits Azure, um verschiedene Unternehmensanwendungen zu verwalten. Ein Beispiel dafür ist die Unternehmensanwendung «Kapplanung». Diese wird verwendet, um die Kapazitätsplanung in den Scrum Teams zu veranschaulichen und zu verwalten.

Azure Entra ID war früher bekannt als Azure AD und ist ein Dienst, der Identitäten und Zugriffe in der Cloud verwaltet. Es verwaltet Benutzeridentitäten und Zugriffsrechte für Cloud-Anwendungen. Unternehmen, wie auch die CMI, können mit Azure Entra ID eine zentrale Identitätsquelle für Benutzer erstellen und verwalten.

Mit Azure Entra ID sind Single Sign-On (SSO), Multi-Faktor-Authentifizierung, also beispielsweise mit einem Authenticator auf dem Handy, Identitäts- und Zugriffsverwaltung und Geräteverwaltung möglich.

### Blazor[[3]](#footnote-3)

Blazor ist ein kostenloses Open-Source-Webframework, das es ermöglicht interaktive Websiten zu gestalten und mit C# Funktionalitäten zu programmieren. Es gehört zur schon bekannten und viel genutzter ASP.NET Core Familie und macht den Umgang mit Blazor somit vertraut. Dabei gibt es zwei verschiedene Blazor Hosting-Modelle.

#### Blazor Server

Ein Bild, das Text, Screenshot, Diagramm, Design enthält.

Automatisch generierte BeschreibungBei Blazor Server wird die Anwendung auf dem Server ausgeführt. Die Benutzerinteraktionen werden mit SignalR, eine Microsoft-Bibliothek für Echtzeitkommunikation in Webanwendungen, zwischen dem Browser und dem Server übertragen.

Abbildung 10 Blazor Server

#### Blazor WebAssembly

Ein Bild, das Text, Screenshot, Rechteck, Diagramm enthält.

Automatisch generierte BeschreibungBei Blazor WebAssembly wird die ganze Anwendung vollständig im Browser ausgeführt. Der C#-Code wird in WebAssembly kompiliert, was ermöglicht, dass die Anwendung in einer abgesicherten Sandbox läuft.

Abbildung 11 Blazor WebAssembly

## Entity Framework Core[[4]](#footnote-4)

Entity Framework Core ist ein Open-Source-Datenzugriffs-Framework von Microsoft für .Net und dient als ein objektrelationales Mapping (ORM) Tool. Mit Entity Framework Core können Entwickler mit Daten in einer Datenbank arbeiten, ohne dabei direkt auf die Datenbank selbst zugreifen zu müssen. Sie arbeiten hingegen mit .Net Objekten, also C#-Klassen. Folgend werde ich einige Funktionen von Entity Framework Core in einer Tabelle auflisten.

|  |  |
| --- | --- |
| Funktionsname | Funktionalität |
| LINQ-Abfragen | Ermöglicht das Schreiben von Abfragen in C#, die dann während Laufzeit in SQL-Abfragen übersetzt werden und Daten von der Datenbank abrufen. |
| Migrationen | Kann als Versionsverwaltungssystem von Datenbank-Schematas bezeichnet werden. Somit müssen keine SQL-Skripte erstellt werden |
| Unterstützung verschiedener Datenbanken | Entity Framework Core unterstützt viele verschiedene Datenbanken wie beispielsweise SQL Server, SQLite, PostgreSQL, MySQL und weitere. In meiner IPA werde ich MSSQL verwenden. |

Tabelle 11 Entity Framework Core

## GUI-Mockup

Das GUI (Graphical User Interface) habe ich mit Blasamiq erstellt, da ich in der Schule einige Mockups mit diesem Tool schon erstellt habe. Alle Mockups befinden sich in einem PDF-Dokument im Ordner «Balsamiq».

### Skillmatrix Page

Die Skillmatrix basiert auf einem Master-Detail Konzept mit mehreren Tabellen. Beim Klick auf einen User werden alle Skillkategorien mit den jeweiligen Skills des Users geladen. In der Skillkategorie Tabelle kann man nun die Kategorie auswählen, in der man sich einschätzen möchte. Rechts neben der Skillkategorie-Tabelle erscheinen dann alle Skills, die die Kategorie beinhaltet. Administratoren können die Skills von jedem Mitarbeiter bearbeiten. Das Editieren der Selbsteinschätzung erfolgt mit einem Inline-Editor. In einer Suchleiste über der Mitarbeitertabelle kann nach Mitarbeitern gesucht werden.

Ein Bild, das Text, Screenshot, Software, parallel enthält.

Automatisch generierte Beschreibung

Abbildung 12 Mockup Skillmatrix

Über der User Tabelle befindet sich ein Filterbereich, in dem man dynamisch Filter hinzufügen und löschen kann. Links wählt man einen Skill aus, nachdem man filtrieren möchte und rechts kann anschliessend der Mindestwert für diesen Skill gewählt werden. Im Screenshot oben werden also alle Mitarbeiter, die in C# mindestens eine 3 und maximal eine 4 und in JavaScript mindestens eine 2 haben, angezeigt.

### Skillverwaltung Page

Ein Bild, das Text, Screenshot, Software, Diagramm enthält.

Automatisch generierte BeschreibungIn der Skillverwaltung Page können Administrator Benutzer Skillkategorien und Skills hinzufügen, bearbeiten und Löschen. In der linken Tabelle sind alle Skillkategorien aufgelistet und in der rechten Tabelle alle Skills. Das Bearbeiten der Skilkategorien und Skills erfolgt wie in der Skillmatrix mit einem InLine-Editor. In der Spalte «Aktionen» der beiden Tabellen können Skillkategorien und Skills mit einem Lösch-Button gelöscht werden. Über den jeweiligen Tabellen können neue Skillkategorien und Skills mit einem Hinzufügen-Button hinzugefügt werden.

Abbildung 13 Mockup Skillverwaltung

#### Neue Skillkategorie Modal

Ein Bild, das Text, Screenshot, Software, Display enthält.

Automatisch generierte Beschreibung

Abbildung 14 Mockup Neue Skillkategorie

Beim Hinzufügen einer neuen Skillkategorie öffnet sich ein Modal-Fenster, indem man den Namen der neuen Skillkategorie in einem Textfeld erfassen kann.

#### Skillkategorie löschen Modal

Ein Bild, das Text, Screenshot, Software, Display enthält.

Automatisch generierte Beschreibung

Abbildung 15 Mockup Skillkategorie löschen

Beim Löschen einer Skillkategorie öffnet sich ein Modal-Fenster, das Bestätigen will, ob der Benutzer die Kategorie mit der Anzahl der dazugehörigen Skills wirklich löschen möchte.

#### Neuer Skill Modal

Ein Bild, das Text, Screenshot, Software, Multimedia-Software enthält.

Automatisch generierte Beschreibung

Abbildung 16 Mockup Neuer Skill

Beim Hinzufügen eines neuen Skills öffnet sich ein Modal-Fenster, indem man den Namen des neuen Skills in einem Textfeld erfassen kann. In einem Dropdown kann man die Kategorie des neuen Skills festlegen.

#### Skill löschen Modal

Ein Bild, das Text, Screenshot, Software, Display enthält.

Automatisch generierte Beschreibung

Abbildung 17 Mockup Skill löschen

Beim Löschen eines Skills öffnet sich ein Modal-Fenster, das Bestätigen will, ob der Benutzer den Skill wirklich löschen möchte.

### Benutzerverwaltung Page

Auf der Page «Benutzerverwaltung» können Administratoren neue Mitarbeiter erfassen, bearbeiten, löschen und ebenfalls zu Administrator Benutzer befördern, jedoch dieses Recht auch wieder entfernen. In einer Suchleiste über der Tabelle kann nach Mitarbeitern gesucht werden.

Ein Bild, das Text, Screenshot, Software, Display enthält.

Automatisch generierte Beschreibung

Abbildung 18 Mockup Benutzerverwaltung

#### Neuer Mitarbeiter Modal

Ein Bild, das Screenshot, Text, Software, Multimedia-Software enthält.

Automatisch generierte Beschreibung

Abbildung 19 Mockup Neuer Mitarbeiter

Beim Hinzufügen eines neuen Mitarbeiters öffnet sich ein Modal-Fenster, indem man den Namen, Vornamen und Email des neuen Mitarbeiters in Textfeldern erfassen kann. In einer Checkbox «IsAdmin» kann beim Erstellen schon festgelegt werden, ob der Mitarbeiter Adminrechte hat.

#### Mitarbeiter löschen Modal

Ein Bild, das Screenshot, Text, Software, Display enthält.

Automatisch generierte Beschreibung

Abbildung 20 Mockup Mitarbeiter Löschen

Beim Löschen eines Mitarbeiters öffnet sich ein Modal-Fenster, das Bestätigen will, ob der Administrator Benutzer den Mitarbeiter wirklich löschen möchte.

## Blazor Frontend Components[[5]](#footnote-5)[[6]](#footnote-6)

Da ich das Mockup nun erstellt habe, stellt sich jetzt die Frage, welche UI Components ich verwende, um die Applikation zu gestalten. Für Blazor gibt es viele kostenlose Component Libraries. Ich analysiere zwei Kandidaten, die für mich in Frage kommen würden. Blazor Bootstrap und MudBlazor. Dabei werde ich die Tabellen-Komponente der beiden vergleichen.

Bootstrap ist ein Open-Source-Framework, das sich für die Entwicklung von Webanwendungen eignet. Bootstrap kann unabhängig von der verwendeten Plattform oder dem Framework verwendet werden. Blazor Bootstrap hingegen ist speziell für die Verwendung mit Blazor-Framework geeignet.

MudBlazor ist ebenfalls eine Open-Source-Bibliothek, wurde speziell für die Entwicklung mit dem Blazor-Framework entwickelt und basiert auf dem Material Design von Google.

In folgender Tabelle werden die beiden Tabellen-Komponente verglichen.

|  |  |  |
| --- | --- | --- |
| Bibliothekname | Tabelle / Grid | Funktionalitäten |
| Blazor Bootstrap | Abbildung 21 Tabelle Blazor Bootstrap | Design:   * Das Design der Tabelle gefällt mir gut * Einfach   Suchfunktion:   * Keine Suchleiste * Integrierter Filter   Pagination:   * Der Paginator kann aktiviert oder deaktiviert werden * Anstelle des Paginators kann auch eine Scrollbar aktiviert werden   InLine Edit:   * Bietet keinen InLine-Editor |
| MudBlazor | Abbildung 22 Tabelle MudBlazor | Design:   * Das Design der Tabelle gefällt mir sehr gut * Schlicht und einfach   Suchfunktion:   * Die MudBlazor Tabelle bietet eine integrierte Suchleiste   Pagination:   * Der Paginator kann aktiviert oder deaktiviert werden * Anstelle des Paginators kann auch eine Scrollbar aktiviert werden   InLine Edit:   * Die Tabelle bietet einen schon integrierten InLine-Editor |

Tabelle 12 Blazor Frontend Components

## Use Case Diagramm

Um die Interaktivitäten der verschiedenen Benutzer der Applikation mit der Webanwendung zu veranschaulichen, erstelle ich ein Use Case Diagramm.

Abbildung 23 Use Case Diagramm

Ein Bild, das Text, Diagramm, Kreis, Screenshot enthält.

Automatisch generierte BeschreibungBeim Mitarbeiter wird erwartet, dass dieser schon von einem Administrator erfasst wurde. Falls ein Mitarbeiter noch nicht erfasst wurde, unterscheidet sich dieser vom erfassten Mitarbeiter lediglich dadurch, dass dieser sich nicht selbst einschätzen kann.

## Datenbankdiagramm

Ein Bild, das Text, Screenshot, Design enthält.

Automatisch generierte BeschreibungDas folgende Datenbankdiagramm wurde mit dbdiagram.io erstellt. Folgende Tabellen wurden erstellt.

Abbildung 24 Datenbankdiagramm

### Mitarbeiter Entity

|  |  |  |
| --- | --- | --- |
| Property | Typ | Funktion |
| MitarbeiterId | Integer (Primary Key) | Legt die eindeutige ID fest |
| Name | Varchar | Der Name des Mitarbeiters |
| Vorname | Varchar | Der Vorname des Mitarbeiters |
| Email | Varchar | Die Email des Mitarbeiters |
| IsAdmin | Bool | Legt fest, ob der Mitarbeiter Admin ist oder nicht |

Tabelle 13 Mitarbeiter Entity

### MitarbeiterSkill Entity

|  |  |  |
| --- | --- | --- |
| Property | Typ | Funktion |
| MitarbeiterId | Integer (Teil des Composite Keys) | Die MitarbeiterId ist Teil des Composite Keys |
| SkillId | Integer (Teil des Composite Keys) | Die SkillId ist Teil des Composite Keys |
| Selbsteinschaetzung | integer | Ist die Selbsteinschätzung der Mitarbeiter von 0 bis 4 |

Tabelle 14 MitarbeiterSkill Entity

### Skill Entity

|  |  |  |
| --- | --- | --- |
| Property | Typ | Funktion |
| SkillId | Integer (Primary Key) | Legt die eindeutige ID fest |
| KategorieId | Integer (Foreign Key) | Ist die Fremd ID der Skillkategorie |
| SkillName | Varchar | Der Name des Skills |

Tabelle 15 Skill Entity

### Skillkategorie Entity

|  |  |  |
| --- | --- | --- |
| Property | Typ | Funktion |
| SkillkategorieId | Integer (Primary Key) | Legt die eindeutige ID fest |
| SkillkategorieName | Varchar | Der Name der Skillkategorie |

Tabelle 16 Skillkategorie Entity

### Log Entity

|  |  |  |
| --- | --- | --- |
| Property | Typ | Funktion |
| LogId | Integer (Primary Key) | Legt die eindeutige ID fest |
| EreignisTyp | Varchar | Der Ereignis Typ des Logs (bsp «Error», «Information», «Warning») |
| EreignisDatum | Datetime | Das aktuelle Datum und die aktuelle Zeit des Logeintrages |
| BenutzerEmail | Varchar | Die Email des Benutzers, die diesen Logeintrag bewirkte |
| Nachricht | Varchar | Die Nachricht / Exception / Stacktrace des Logeintrages |

Tabelle 17 Log Entity

Die MitarbeiterSkill Tabelle hat einen Composite Key, der sich aus der MitarbeiterId und der SkillId zusamensetzt. Diese kann als die Zwischentabelle für eine Many-To-Many Relation von Mitarbeiter und Skill betrachtet werden.

Ebenfalls wurde eine Logtabelle erfasst, die unabhängig vom System existiert.

## Gesamtsystem / Systemgrenzen

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte BeschreibungDas Gesamtsystem besteht aus dem User Interface (UI) und der Blazor Server Web Applikation, die Azure Entra ID als Identity Provider verwendet. Die Blazor Server Web Applikation verwendet Entity Framework Core, um mit der Datenbank zu kommunizieren. Folgendes Diagramm zeigt diese Konstellation:

Abbildung 25 Gesamtsystem

## Testkonzept

Mein Testkonzept ist eine technische Spezifikation, die einzelne Testfälle bei der Softwareprüfung beschreibt. Dabei wird eine Anforderung / Voraussetzung angegeben und anhand dieser einen Input und der erwartete Output festgelegt. Diese Testfälle werden vor der Realisierung geschrieben, da diese nicht der Applikation, sondern anhand der Anforderungen angepasst sein sollten. In der Phase «Kontrollieren» werden diese dann ausgeführt und kontrolliert.

In der Schule habe ich dieses Testkonzept so gelernt und angewendet.

### Testumgebung

In Folgender Tabelle wird die Testumgebung beschrieben.

|  |  |
| --- | --- |
| Produkt | Blazor Webanwendung (Skillmatrix) |
| **Datenbank** | CMI\_Skillmatrix |
| **Gerät** | HP ZBook Power G9 |
| **Betriebssystem** | Windows 10 |
| **Browser & Version** | Version 122.0.6261.129 (Offizieller Build) (64-Bit) |
| **Testdatum** | 15.03.2024 |

Tabelle 18 Testumgebung

### Testfälle

Nachfolgend sind alle Testfälle anhand der funktionalen Anforderungen definiert worden.

|  |  |
| --- | --- |
| Testfall-Nr | 001 |
| Anforderungsnummer | AN-01 |
| Szenario | Ein Administrator möchte eine neue Skill-Kategorie hinzufügen |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet |
| Testschritte | 1. Der Administrator navigiert zur Page «Skillsverwaltung» 2. Der Administrator klickt unter dem Titel «Skillkategorien» auf den Button «Skillkategorie Hinzufügen» 3. Der Administrator gibt im Modal Fenster den Kategorienamen «Sprachen» ein 4. Der Administrator klickt auf den «OK» Button |
| Erwartetes Ergebnis | Die erstellte Skill-Kategorie mit dem Namen «Sprachen» erscheint in der Skill-Kategorie Tabelle |

Tabelle 19 Testfall 001

|  |  |
| --- | --- |
| Testfall-Nr | 002 |
| Anforderungsnummer | AN-01 |
| Szenario | Ein Administrator möchte einen neuen Skill hinzufügen |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet * eine Skillkategorie namens «Sprachen» ist vorhanden |
| Testschritte | 1. Der Administrator navigiert zur Page «Skillsverwaltung» 2. Der Administrator klickt unter dem Titel «Skills» auf den Button «Skill Hinzufügen» 3. Der Administrator gibt im Modal Fenster den Skillnamen «Englisch» ein 4. Der Administrator wählt die Skillkategorie «Sprachen» im Dropdown aus 5. Der Administrator klickt auf den «OK» Button |
| Erwartetes Ergebnis | Der erstellte Skill mit dem Namen «Englisch» und der Skillkategorie «Sprachen» erscheint in der Skill Tabelle |

Tabelle 20 Testfall 002

|  |  |
| --- | --- |
| Testfall-Nr | 003 |
| Anforderungsnummer | AN-01 |
| Szenario | Ein Administrator möchte eine Skillkategorie löschen |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet * eine Skillkategorie namens «Sprachen» ist vorhanden * ein Skill namens «Englisch» ist vorhanden |
| Testschritte | 1. Der Administrator navigiert zur Page «Skillsverwaltung» 2. Der Administrator klickt in der Skillkategorie Tabelle in der Spalte «Aktionen» auf den Löschbutton 3. Es öffnet sich ein Modalfenster, das folgenden Text beinhaltet: «Möchten Sie die Skillkategorie «Sprachen» mit 1 Skill(s) wirklich löschen? 4. Der Administrator klickt auf «Löschen» |
| Erwartetes Ergebnis | Die Skillkategorie mit dem Namen «Sprachen» ist in der Skillkategorie Tabelle nicht vorhanden  Der Skill «Englisch» ist in der Skill Tabelle nicht vorhanden |

Tabelle 21 Testfall 003

|  |  |
| --- | --- |
| Testfall-Nr | 004 |
| Anforderungsnummer | AN-01 |
| Szenario | Ein Administrator möchte einen Skill löschen |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet * ein Skill namens «Englisch» ist vorhanden |
| Testschritte | 1. Der Administrator navigiert zur Page «Skillsverwaltung» 2. Der Administrator klickt in der Skills Tabelle in der Spalte «Aktionen» auf den Löschbutton 3. Es öffnet sich ein Modalfenster, das folgenden Text beinhaltet: «Möchten Sie den Skill «Englisch» wirklich löschen? 4. Der Administrator klickt auf «Löschen» |
| Erwartetes Ergebnis | Der Skill mit dem Namen «Englisch» ist in der Skill Tabelle nicht vorhanden |

Tabelle 22 Testfall 004

|  |  |
| --- | --- |
| Testfall-Nr | 005 |
| Anforderungsnummer | AN-01 |
| Szenario | Ein Administrator möchte einen Skill bearbeiten |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet * eine Skillkategorie namens «Sprachen» ist vorhanden * ein Skill namens «Englisch» ist vorhanden |
| Testschritte | 1. Der Administrator navigiert zur Page «Skillsverwaltung» 2. Der Administrator klickt in der Skill Tabelle auf den Skillnamen «Englisch» 3. Es erscheint einen In-Line Editor 4. Der Administrator gibt in der Skillname Spalte den neuen Namen «Französisch» ein 5. Der Administrator klickt auf den Häkchen Button |
| Erwartetes Ergebnis | Der Skill «Französisch» ist in der Skill Tabelle vorhanden |

Tabelle 23 Testfall 005

|  |  |
| --- | --- |
| Testfall-Nr | 006 |
| Anforderungsnummer | AN-01 |
| Szenario | Ein Administrator möchte eine Skillkategorie bearbeiten |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet * eine Skillkategorie namens «Sprachen» ist vorhanden * ein Skill namens «Englisch» mit der Skillkategorie «Sprachen» ist vorhanden |
| Testschritte | 1. Der Administrator navigiert zur Page «Skillsverwaltung» 2. Der Administrator klickt in der Skillkategorie Tabelle auf die Skillkategorie «Sprachen» 3. Es erscheint ein In-Line Editor 4. Der Administrator gibt in der Skillname Spalte den neuen Namen «Languages» ein 5. Der Administrator klickt auf den Häkchen Button |
| Erwartetes Ergebnis | Die Skillkategorie «Languages» ist in der Skillkategorie Tabelle vorhanden  Der Skill Englisch hat neu die Skillkategorie «Languages» |

Tabelle 24 Testfall 006

|  |  |
| --- | --- |
| Testfall-Nr | 007 |
| Anforderungsnummer | AN-02 |
| Szenario | Ein Administrator möchte einen neuen Mitarbeiter hinzufügen |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet |
| Testschritte | 1. Der Administrator navigiert zur Page «Benutzerverwaltung» 2. Der Administrator klickt in unter dem Titel «Benutzerverwaltung» auf den Button «Mitarbeiter Hinzufügen» 3. Es öffnet sich ein Modal Fenster mit den Textfeldern «Name», «Vorname», «Email» und einer Checkbox «IsAdmin» 4. Der Administrator gibt den Namen «Lio», den Vornamen «Janosch», die Email «janosch.lio@cmiag.ch» ein und klick auf die Checkbox «IsAdmin» |
| Erwartetes Ergebnis | Der neue Mitarbeiter mit folgenden Eigenschaften ist in der Mitarbeiter Tabelle vorhanden:  Name: «Lio»  Vorname: «Janosch»  Email: «janosch.lio@cmiag.ch»  IsAdmin: «true» |

Tabelle 25 Testfall 007

|  |  |
| --- | --- |
| Testfall-Nr | 008 |
| Anforderungsnummer | AN-02 |
| Szenario | Ein Administrator möchte einen Mitarbeiter bearbeiten |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet * ein Mitarbeiter mit folgenden Eigenschaften ist vorhanden:   Name: «Klee»  Vorname: «Alexander»  Email: «alexander.klee@cmiag.ch»  IsAdmin: «false» |
| Testschritte | 1. Der Administrator navigiert zur Page «Benutzerverwaltung» 2. Der Administrator klickt auf den Namen «Klee» 3. Es erscheint ein In-Line Editor 4. Der Administrator gibt den neuen Namen «Hans» ein 5. Der Administrator klickt auf den Häkchen Button |
| Erwartetes Ergebnis | Der Mitarbeiter mit folgenden Eigenschaften ist in der Mitarbeiter Tabelle vorhanden:  Name: «Hans»  Vorname: «Alexander»  Email: «alexander.klee@cmiag.ch»  IsAdmin: «false» |

Tabelle 26 Testfall 008

|  |  |
| --- | --- |
| Testfall-Nr | 009 |
| Anforderungsnummer | AN-02 |
| Szenario | Ein Administrator möchte einen Mitarbeiter löschen |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet * ein Mitarbeiter mit folgenden Eigenschaften ist vorhanden:   Name: «Klee»  Vorname: «Alexander»  Email: «alexander.klee@cmiag.ch»  IsAdmin: «false» |
| Testschritte | 1. Der Administrator navigiert zur Page «Benutzerverwaltung» 2. Der Administrator klickt in der Spalte «Aktionen» auf den Löschbutton 3. Es öffnet sich ein Modal Fenster, das folgenden Text beinhaltet: «Möchten Sie den Mitarbeiter «Alexander Klee» wirklich löschen?» 4. Der Administrator klickt auf den Button «Löschen» |
| Erwartetes Ergebnis | Der Mitarbeiter mit folgenden Eigenschaften ist nicht mehr in der Mitarbeiter Tabelle vorhanden:  Name: «Klee»  Vorname: «Alexander»  Email: «alexander.klee@cmiag.ch» IsAdmin: «false» |

Tabelle 27 Testfall 009

|  |  |
| --- | --- |
| Testfall-Nr | 010 |
| Anforderungsnummer | AN-03 |
| Szenario | Ein Administrator möchte einen Mitarbeiter zum Administrator «befördern» |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet * ein Mitarbeiter mit folgenden Eigenschaften ist vorhanden:   Name: «Klee»  Vorname: «Alexander»  Email: «alexander.klee@cmiag.ch»  IsAdmin: «false» |
| Testschritte | 1. Der Administrator navigiert zur Page «Benutzerverwaltung» 2. Der Administrator klickt auf den Namen «Klee» 3. Es erscheint ein In-Line Editor 4. Der Administrator klickt auf die Checkbox in der Spalte «IsAdmin» 5. Der Administrator klickt auf den Häkchen Button |
| Erwartetes Ergebnis | Der Mitarbeiter mit folgenden Eigenschaften ist in der Mitarbeiter Tabelle vorhanden:  Name: «Klee»  Vorname: «Alexander»  Email: «alexander.klee@cmiag.ch»  IsAdmin: «true» |

Tabelle 28 Testfall 010

|  |  |
| --- | --- |
| Testfall-Nr | 011 |
| Anforderungsnummer | AN-04 |
| Szenario | Ein Mitarbeiter möchte eine Selbsteinschätzung geben |
| Testvoraussetzung | * ein Mitarbeiter ist mit folgenden Eigenschaften erfasst   Name: «Lio»  Vorname: «Janosch»  Email: «janosch.lio@cmiag.ch»  IsAdmin: «false»   * eine Skillkategorie namens «Sprachen» mit einem Skill «Englisch» ist vorhanden |
| Testschritte | 1. Der Mitarbeiter navigiert zur Page «Start» 2. Der Mitarbeiter klickt in der Mitarbeitertabelle auf seinen Namen 3. In der Skillkategorietabelle klickt der Mitarbeiter auf Sprachen 4. In der Skilltabelle klickt der Mitarbeiter auf den Skill «Englisch» 5. Der Mitarbeiter gibt seine Selbsteinschätzung mit der Nummer 3 ein 6. Der Mitarbeiter klickt auf den Häkchen Button |
| Erwartetes Ergebnis | Der Mitarbeiter hat nun folgende Werte in der Skillmatrix: Skillkategorie: «Sprachen»  Skillname: «Englisch»  Selbsteinschätzung: «3» |

Tabelle 29 Testfall 011

|  |  |
| --- | --- |
| Testfall-Nr | 012 |
| Anforderungsnummer | AN-05 |
| Szenario | Ein Administrator Benutzer möchte die Selbsteinschätzung eines Mitarbeiters bearbeiten |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet * ein Mitarbeiter ist mit folgenden Eigenschaften erfasst   Name: «Lio»  Vorname: «Janosch»  Email: «janosch.lio@cmiag.ch»  IsAdmin: «false»   * eine Skillkategorie namens «Sprachen» ist vorhanden * ein Skill namens «Englisch» mit der Skillkategorie «Sprachen» ist vorhanden * Der Mitarbeiter hat sich in diesem Skill mit 3 eingeschätzt |
| Testschritte | 1. Der Administrator navigiert zur Page «Start» 2. Der Administrator klickt in der Mitarbeitertabelle auf den Mitarbeiter mit Namen «Lio», Vorname «Janosch» 3. In der Skillkategorietabelle klickt der Mitarbeiter auf Sprachen 4. In der Skilltabelle klickt der Mitarbeiter auf den Skill «Englisch» 5. Der Administrator gibt in der Spalte «Selbsteinschätzung» die Zahl 2 ein 6. Der Administrator klickt auf den Häkchen Button |
| Erwartetes Ergebnis | Der Mitarbeiter hat nun folgende Werte in der Skillmatrix: Skillkategorie: «Sprachen»  Skillname: «Englisch»  Selbsteinschätzung: «2» |

Tabelle 30 Testfall 012

|  |  |
| --- | --- |
| Testfall-Nr | 013 |
| Anforderungsnummer | AN-06 |
| Szenario | Ein Benutzer möchte die Skill-Übersicht (Skillmatrix) von allen Mitarbeitern und ihren Skills sehen |
| Testvoraussetzung | * Ein Benutzer ist angemeldet * ein Mitarbeiter ist mit folgenden Eigenschaften erfasst   Name: «Lio»  Vorname: «Janosch»  Email: «janosch.lio@cmiag.ch»  IsAdmin: «false»   * eine Skillkategorie namens «Sprachen» ist vorhanden * ein Skill namens «Englisch» mit der Skillkategorie «Sprachen» ist vorhanden * Der Mitarbeiter hat sich in diesem Skill mit 3 eingeschätzt |
| Testschritte | 1. Der Benutzer navigiert zur Page «Start» |
| Erwartetes Ergebnis | Der Benutzer sieht nun die Skill-Übersicht (Skillmatrix). Es sieht den Mitarbeiter mit Namen «Lio» und Vornamen «Janosch» und kann seine Skills anschauen. |

Tabelle 31 Testfall 013

|  |  |
| --- | --- |
| Testfall-Nr | 014 |
| Anforderungsnummer | AN-07 |
| Szenario | Ein Benutzer möchte auf der Skill-Übersicht (Skillmatrix) nach dem Namen eines Mitarbeiters suchen können |
| Testvoraussetzung | * Ein Benutzer ist angemeldet * ein Mitarbeiter ist mit folgenden Eigenschaften erfasst   Name: «Lio»  Vorname: «Janosch»  Email: «janosch.lio@cmiag.ch»  IsAdmin: «false» |
| Testschritte | 1. Der Benutzer navigiert zur Page «Start» 2. Der Benutzer klickt in die Suchleiste über der Mitarbeitertabelle und gibt folgenden Text ein: «Lio» |
| Erwartetes Ergebnis | Der Benutzer sieht den Erfassten Mitarbeiter mit Namen «Lio» in der Mitarbeitertabelle |

Tabelle 32 Testfall 014

|  |  |
| --- | --- |
| Testfall-Nr | 015 |
| Anforderungsnummer | AN-07 |
| Szenario | Ein Benutzer möchte auf der Skill-Übersicht (Skillmatrix) nach Mitarbeiter filtrieren, die sich im Skill «Englisch» mit mindestens einer 2 eingeschätzt haben. |
| Testvoraussetzung | * ein Benutzer ist angemeldet * ein Mitarbeiter ist mit folgenden Eigenschaften erfasst   Name: «Lio»  Vorname: «Janosch»  Email: «janosch.lio@cmiag.ch»  IsAdmin: «false»   * eine Skillkategorie namens «Sprachen» ist vorhanden * ein Skill namens «Englisch» mit der Skillkategorie «Sprachen» ist vorhanden   Der Mitarbeiter hat sich in diesem Skill mit 3 eingeschätzt |
| Testschritte | 1. Der Benutzer navigiert zur Page «Start» 2. Der Benutzer klickt unter dem Title «Skillmatrix» auf den Button «Filter hinzufügen» 3. Der Benutzer wählt im Dropdown den Skill «Englisch» 4. Der Benutzer trägt im Nummernfeld «Min» die Zahl 2 ein 5. Der Benutzer klickt auf den Button «Suchen» |
| Erwartetes Ergebnis | Der Benutzer sieht den erfassten Mitarbeiter mit Namen «Lio» in der Mitarbeitertabelle |

Tabelle 33 Testfall 015

|  |  |
| --- | --- |
| Testfall-Nr | 016 |
| Anforderungsnummer | AN-08 |
| Szenario | Benutzer sollen mittels Microsoft Entra ID authentifiziert werden. |
| Testvoraussetzung | * Es existiert ein privates Office 365 Konto, das nicht im Tenant «CM Informatik AG» existiert |
| Testschritte | 1. Der Benutzer greift auf die Skillmatrix-Page zu 2. Der Benutzer gibt die Email-Adresse seines privaten Office 365 Kontos ein und klickt auf den Anmeldebutton |
| Erwartetes Ergebnis | Es erscheint ein Fenster von Microsoft, das sagt, dass die Email-Adresse des privaten Kontos nicht im Tenant «CM Informatik AG» vorhanden ist. |

Tabelle 34 Testfall 016

|  |  |
| --- | --- |
| Testfall-Nr | 017 |
| Anforderungsnummer | AN-09 |
| Szenario | Der erste Benutzer möchte auf die Skillmatrix zugreifen und Mitarbeiter erstellen |
| Testvoraussetzung | * Die Email-Adresse «janosch.lio@cmiag.ch» ist im Konfigurationsfile des Skillmatrix Projektes erfasst |
| Testschritte | 1. Der Benutzer greift auf die Skillmatrix-Page zu 2. Der Benutzer meldet sich mit seinem Microsoft Konto der CMI an |
| Erwartetes Ergebnis | Der Benutzer hat Administratorenrechte und kann jetzt beispielsweise neue Mitarbeiter erfassen. |

Tabelle 35 Testfall 017

|  |  |
| --- | --- |
| Testfall-Nr | 018 |
| Anforderungsnummer | AN-10 |
| Szenario | Der Administrator möchte einen Benutzer erstellen und vergisst die Eingabe eines Feldes |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet |
| Testschritte | 1. Der Administrator navigiert zur Page «Benutzerverwaltung» 2. Der Administrator klickt auf den Button «Mitarbeiter hinzufügen» 3. Der Administrator vergisst im Modal das Ausfüllen einer der Pflichtfelder 4. Der Administrator klickt auf den «OK» Button |
| Erwartetes Ergebnis | Der Administrator wird darauf hingewiesen, dass er alle Pflichtfelder ausfüllen soll. |

Tabelle 36 Testfall 018

|  |  |
| --- | --- |
| Testfall-Nr | 019 |
| Anforderungsnummer | AN-10 |
| Szenario | Der Administrator möchte eine neue Skillkategorie erstellen und vergisst die Eingabe des Feldes |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet |
| Testschritte | 1. Der Administrator navigiert zur Page «Skillsverwaltung» 2. Der Administrator klickt auf den Button «Skillkategorie Hinzufügen» 3. Der Administrator vergisst im Modal das Ausfüllen des Pflichtfeldes 4. Der Administrator klickt auf den «OK» Button |
| Erwartetes Ergebnis | Der Administrator wird darauf hingewiesen, dass er das Pflichtfeld ausfüllen soll. |

Tabelle 37 Testfall 019

|  |  |
| --- | --- |
| Testfall-Nr | 020 |
| Anforderungsnummer | AN-10 |
| Szenario | Der Administrator möchte einen neuen Skill erstellen und vergisst die Eingabe eines Feldes |
| Testvoraussetzung | * ein Benutzer mit Admin-Rechten ist angemeldet |
| Testschritte | 1. Der Administrator navigiert zur Page «Skillsverwaltung» 2. Der Administrator klickt auf den Button «Skill Hinzufügen» 3. Der Administrator vergisst im Modal das Ausfüllen eines Pflichtfeldes 4. Der Administrator klickt auf den «OK» Button |
| Erwartetes Ergebnis | Der Administrator wird darauf hingewiesen, dass er die Pflichtfelder ausfüllen soll. |

Tabelle 38 Testfall 020

# Entscheiden

Hier beginnt die Entscheidungsphase wo ich mich für verschiedene Lösungswege / Lösungsvarianten entscheiden werde.

## Blazor Hosting Modelle

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte BeschreibungEine Entscheidung, die ich treffen werde, ist, welches Hosting-Modell von Blazor verwendet werden soll. Hierfür habe ich eine Entscheidungsmatrix in Excel erstellt.

Abbildung 26 Entscheidung Blazor Hosting Modelle

### Kommentar zur Entscheidung des Hosting-Modells

Die beiden Hosting-Modelle wurden anhand fünf Kriterien verglichen. Jedes Kriterium besitzt eine Gewichtung von 1 bis 3. Punkte wurden von 1 bis 5 gegeben und das Endresultat ist die Multiplikation dieser beiden Zahlen. Um die richtige Wahl der beiden Hosting-Modelle zu treffen habe ich mir bei jedem Kriterium folgende Fragen gestellt.

1. Ladezeit beim ersten Zugriff
   1. Wie wichtig ist die Ladegeschwindigkeit des ersten Seitenladens für die Benutzererfahrung?
2. Serverlast
   1. Verfüge ich über ausreichende Serverressourcen, um die höhere Serverlast von Blazor Server zu bewältigen?
3. Offline-Fähigkeit
   1. Ist es notwendig, Funktionalitäten der Skillmatrix Offline-Fähig zu machen?
4. Sicherheit
   1. Wie sensibel sind die Daten, die in der Anwendung präsent sind?
5. Skalierbarkeit
   1. Wie entwickelt sich die Nutzerzahlen im Laufe der Zeit?
   2. Wie wichtig ist es, dass die Anwendung unter grosser Last reibungslos funktioniert?
6. Persönliche Erfahrung

Nach der Verteilung der Punkte wurde der Gewinner «Blazor Server» festgelegt.

## Blazor Komponenten Bibliothek

Ein Bild, das Text, Screenshot, Schrift, parallel enthält.

Automatisch generierte BeschreibungIn der Planung habe ich die Tabellen Komponente der beiden Komponenten Bibliotheken Blazor Bootstrap und Mudblazor verglichen. Ich werde auch hierzu eine Entscheidungsmatrix erstellen, um die Entscheidung zwischen den beiden zu erleichtern.

Abbildung 27 Entscheidung Blazor Komponenten Bibliothek

### Kommentar zur Entscheidung der Komponenten-Bibliothek

Auch hier habe ich verschiedene Kriterien aufgezählt und die beiden Bibliotheken anhand diesen verglichen. Hierbei wird für jedes Kriterium eine Gewichtung von 1 bis 3 gegeben und die Punkte von 1 bis 5. Das Endresultat ist die Multiplikation der beiden Zahlen. Ich habe mir bei den Kriterien folgende Fragen gestellt.

1. Design und Benutzerfreundlichkeit
   1. Ist das vorgefertigte Design der Komponenten ansprechend?
2. Komponentenvielfalt
   1. Bietet die Bibliothek alle Komponenten, die ich brauche?
   2. Bieten die Komponenten gut brauchbare integrierte Funktionalitäten?
3. Anpassungsfähigkeit
   1. Lassen sich die Komponente noch eigenständig anpassen?
4. Dokumentation und Community-Unterstützung
   1. Wie umfangreich ist die Dokumentation?
   2. Gibt es aktive Foren oder Plattformen, auf denen ich Antworten finde?
5. Performance und Effizienz
   1. Gibt es Performance-Probleme beim Benutzen dieser Bibliotheken?
6. Kompatibilität und Aktualität
   1. Werden die neusten Blazor und .Net Versionen unterstützt?
   2. Werden die Bibliotheken in der CMI schon verwendet?

Nach der Verteilung der Punkte wurde der Gewinner «MudBlazor» festgelegt.

## Entscheidungsgespräch

Ein Bild, das Text, Screenshot enthält.

Automatisch generierte BeschreibungAnschliessend dieser Entscheidungen habe ich mit Dominik ein Entscheidungsgespräch gemacht, indem wir meine Entscheidung nochmals besprochen haben. Ich erklärte ihm meine Vorgehensweise und die Kriterien, die ich für die Entscheidung der beiden Technologien verwendet habe.

Abbildung 28 Entscheidungsgespräch

# Realisieren

Hier beginnt der Realisierungsteil der Dokumentation.

## Erstellung des Blazor Projektes

Das Projekt wird im src Ordner des Repositories cmi-skillmatrix mit der CMI-Namenskonvention «CMI.Skillmatrix» benannt.

* Ich werde .NET 8.0 verwenden, damit sich das Projekt auf dem neusten Framework-Stand befindet.
* Es soll kein Authentifizierungstyp verwendet werden.
* Soll für HTTPS konfiguriert sein
* Blazor Server wird verwendet
* Die Interaktivität soll pro Page gesteuert werden
* Es sollen keine Sample Pages erstellt werden

### Installation MudBlazor

Um MudBlazor zu installieren habe ich der Manuellen Installation von MudBlazor gefolgt. Diese befindet sich hier:

<https://mudblazor.com/getting-started/installation#manual-install>

### Erstellung des MainLayout

Beim Aussehen der Skillmatrix möchte ich mich dem Aussehen der Kapplanung orientieren. Aus diesem Grund übernehme ich den Code des Files «MainLayout.razor» der Kapplanung.

Folgende drei Components, die in der Installationsanleitung von MudBlazor erwähnt wurden, habe ich in einer separaten Komponente, namens «MudProviders.razor», im neu erstellten Ordner «Shared» ausgelagert, die ich anschliessend in das MainLayout einfüge. Dies hat den Grund, dass das MainLayout so übersichtlicher ist:

Diese razor Komponente enthält das CMITheme im «MudThemeProvider» und die beiden anderen Provider, «MudDialogProvider» und «MudSnackbarProvider»

|  |  |
| --- | --- |
| Provider | Funktion |
| MudThemeProvider | Hier können eigene Farben und Layouts festgelegt werden. |
| MudDialogProvider | Ist dafür zuständig, dass das Abbilden von Dialogen / Modalen möglich ist. |
| MudSnackbarProvider | Ist dafür zuständig, dass kleine Nachrichten als «Snackbar» in der WebApplikation erscheinen können. Folgend ein Beispielbild: |

Tabelle 39 MudProviders

Ein Bild, das Text, Screenshot, Schrift, Design enthält.

Automatisch generierte BeschreibungNachdem diese Änderungen angepasst wurden, sieht die Webanwendung folgendermassen aus:

Abbildung 29 Webanwendung Grundgerüst

### Erstellung des Nav Menus

Im «Layout» Ordner wurde ein neues Razor File namens «NavMenu» erstellt, indem mit der «MudNavMenu» Komponente jedes NavItem mit einem «MudNavLink» erweitert wird.

## Datenbank erstellen

Als Datenbank werde ich MSSQL verwenden. Diese wird zum einen von Entity Framework unterstützt und zum anderen kenne ich mich mit diesem relationalen Datenbankmanagementsystem aus vorherigen Arbeiten im Geschäft aus.

Ich erstelle eine neue Datenbank mit dem Namen «CMI\_Skillmatrix» auf meinem lokalen SQL-Server.

### Datenbank verbinden

Damit meine Blazor Webanwendung mit der erstellten Datenbank arbeiten kann, erstelle ich im «appsettings.json» ein neues Objekt namens «ConnectionStrings». In diesem wird die Verbindungszeichenfolge, um sich mit der Datenbank zu verbinden, eingefügt.

Ein Bild, das Text, Screenshot, Schrift, Reihe enthält.

Automatisch generierte BeschreibungDiese Verbindungszeichenfolge wird anschliessend ins appsettings.json geschrieben.

Abbildung 30 Connection String

Im «Program.cs» File, schreiben wir den Connection String in eine Variable, die wir dann später bei der effektiven Verbindung verwenden werden.



Abbildung 31 Connection String Konfiguration

## Datenbankobjekte / Entities erstellen

Hier kommt das Entity Framework zum Zug. Es werden alle Objekte nach dem Datenbankdiagramm erstellt.

Anschliessend erstelle ich eine neue Klasse im «Data» Ordner namens «SkillmatrixDbContext». Dieser wir unser Datenbankcontext sein, der als Brücke zwischen der Anwendung und der Datenbank dient. Er bietet CRUD-Operationen an und konfiguriert das Datenmodell und Beziehungen zwischen den Tabellen.

Um diesen jedoch nutzen zu können, muss ich die NuGet-Packages des Entity Frameworks herunterladen.

Im DbContext werden alle Objekte in einem DbSet angegeben. Normalerweise kann in Entity Framework Core direkt in den Klassen der Objekte mit Anmerkungen angegeben werden, was ein Primary Key ist, oder was Required ist. Da wir jedoch mit dem Objekt «MitarbeiterSkill» ein Composite Key definieren, und diese Konfigurierung mit Anmerkungen nicht möglich ist, verwende ich Fluent API.

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte BeschreibungDie «OnModelCreating» Methode wird überschrieben. Hier werden die Primary Keys, assoziationen, usw. festgelegt.

Abbildung 32 Fluent API Assoziationen Konfiguration

Um den Composite Key zu definieren, erstellen wir eine Many-To-Many Beziehung zwischen Skills und Mitarbeiter und sagen dann, dass wir für diese Beziehung das MitarbeiterSkill-Objekt verwenden. Dies ist eine Many-to-Many Relation with class for join Entity.[[7]](#footnote-7)

Da wir nun den DbContext erstellt haben, können wir diesen im «Program.cs» angeben wo wir auch den vorherigen erstellten Connectionstring brauchen werden.

Abbildung 33 DbContext Konfiguration

### Migration durchführen

Ein Bild, das Text, Screenshot, Schrift, Zahl enthält.

Automatisch generierte BeschreibungDa wir jetzt die Entitäten erstellt haben und alle Beziehungen definiert haben können wir nun eine Migration mittels Command «Add-Migration» durchführen, die alle Tabellen und Relationen in einem Migrationsfile erstellt. Mittels Command «Update-Database» können wir diese Änderungen nun in der Datenbank vornehmen.

Abbildung 34 Migration MSSQL Tabellen

## Authorization und Bootstrapping

Ein Bild, das Text, Screenshot, Schrift, Reihe enthält.

Automatisch generierte BeschreibungDamit wir Azure Entra ID als Identity Provider nutzen können habe ich im File «appsettings.json» einen neuen Bereich «AzureEntraID» definiert. Die ClientID entspricht der Anwendungs-ID der Unternehmensanwendung «Kapplanung». Die TenantID entspricht der MandantenID der CM Informatik auf Azure.

Abbildung 35 Azure Entra ID Konfiguration

|  |  |
| --- | --- |
| Bezeichnung | Bedeutung |
| Instance | Dies ist die URL der Azure Entra ID-Instanz, bei der Authentifizierungsanfragen gestellt werden. Dies kann als Eintrittspunkt für den Azure Entra ID Authentifizierungs und Autorisierungsservice betrachtet werden |
| Domain | Dies ist die Domain, die für die Azure Entra ID Instanz spezifisch ist. |
| TenantId | Ist eine eindeutige ID, die den Azure Entra ID Mandanten repräsentiert. Ist einer Organisation, CM Informatik AG, zugeordnet. |
| ClientId | Ist eine eindeutige ID, die einer Anwendung auf Azure Entra ID zugewiesen wird. Die ID wird beim Authentifizierungsprozess verwendet, um die Anwendung zu identifizieren. |
| CallbackPath | Dies ist der Pfad, an den Azure Entra ID den Token der Authentifizierungsergebnisse nach dem Anmeldevorgang zusendet. |

Tabelle 40 Azure Entra ID Konfiguration Bezeichnung

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte BeschreibungDamit sich Mitarbeiter anmelden können muss ein Authentication Service dem «Program.cs» hinzugefügt werden. Diesen habe ich von der Kapplanung übernommen. Jedoch habe ich hier zusätzlich definiert, dass man seinen Microsoft Account bei jedem Login wählen muss. Hierfür müssen diese zusätzlichen Services als NuGet Paket installiert werden (Microsoft.AspNetCore.Authentication.OpenIdConnect und Microsoft.Identity.Web).

Abbildung 36 Azure Entra ID Konfiguration Program.cs

Diese Anpassung stellt die Option, sich mit seinem CMI-Microsoft Account einzuloggen, zur Verfügung. Damit sich Benutzer anmelden können, müssen wir im «\_Imports.razor» file noch zusätzlich ein Authorize-Attribut hinzufügen.

Ein Bild, das Text, Screenshot, Software, Webseite enthält.

Automatisch generierte BeschreibungNach dieser Änderung kommt beim Starten der Web Anwendung das Loginfenster.

Abbildung 37 Login Fenster

### Bootstrapping

Im File «appsettings.json» füge ich einen neuen Bereich «AdminUsers» ein, indem ich in einem Array verschiedene Emails definieren kann, die immer Admin-Rechte haben.

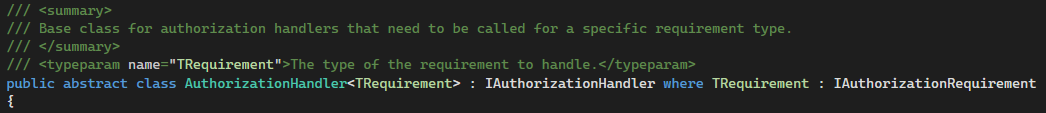
Abbildung 38 Bootstrapping Admin Users

### Authorization Handler

Damit wir sicherstellen können, dass nur Administrator Benutzer auf die Pages «Skillverwaltung» und «Mitarbeiterverwaltung» zugreifen können, erstellen wir ein eigenes Authorize-Attribut mit einem Authorization Handler.

Ich habe dafür ein neues File «RoleAttribute.cs» im Ordner «Attributes» erstellt. In diesem File sind zwei Klassen vorhanden. Einmal das AdminAuthorizationRequirement, das einen boolean Wert «IsAdminRequired» hat und das Interface «IAuthorizationRequiremen» implementiert. Diese Implementation ist für den «AdminAuthorizationHandler» wichtig, da dieser von einem AuthorizationHandler erbt, der nur Klassen mit dem «IAuthorizationRequirement» akzeptiert. Dies sehen wir hier in der de-kompilierten Klasse:

Abbildung 39 IAuthorizationHandler



Der «AdminAuthorizationHandler» überprüft, ob die Email des angemeldeten Benutzers Admin-Rechte hat indem wir auf die Mitarbeiter Tabelle zugreifen und schauen ob das Feld «IsAdmin» bei dieser Email «true» ist. Ebenfalls wird überprüft, ob die Email des angemeldeten Benutzers im Abschnitt «AdminUsers» des Files «appsettings.json» vorhanden ist.

Ein Bild, das Text, Screenshot, Schrift, Reihe enthält.

Automatisch generierte BeschreibungAnschliessend wird der AdminAuthorizationHandler im «program.cs» als Service hinzugefügt. Zusätzlich wird eine Authorization Policy erstellt, die das Requirement, also den «AdminAuthorizationRequirement», definiert.

Abbildung 40 Authorization Konfiguration

### NavMenu Visibilität

Im NavMenu sollen nur die Seiten angezeigt werden, auf die der angemeldete Benutzer auch Zugriff hat. Administratoren auf alle Pages, normale Benutzer nur auf die «Skillmatrix»-Page. Das Schöne an dem vorher erstellten «AdminAuthorizationHandler» ist, dass wir diesen nicht nur als «Authorize-Attribut» verwenden können, sondern auch allein um zu überprüfen, ob die angemeldeten Benutzer Administratoren sind.

Ich beschreibe das NavMenu genauer. Das NavMenu ist mit MudBlazor Komponenten aufgebaut. Das «MudNavMenu» definiert, dass hier das NavMenu ist. In diesem befindet sich der «Skillmatrix» Titel und darunter der «MudNavLink» für die Seite der Skillmatrix. Direkt darunter ist ein «if-statement», das überprüft, ob der boolean «IsAdmin» true ist. Falls dies so ist, werden die beiden weiteren Pages «Skillverwaltung» und «Benutzerverwaltung» angezeigt.

Im Code-Abschnitt befindet sich der boolean, der im «if-statement» verwendet wird. In der «OnInitializedAsync»-Methode wird hier nun ein «AdminRequirement» instanziert. Dieses ist das AdminRequirement, das der «AdminAuthorizationHandler» verwendet. Danach wird mit einem AuthorizationService mit der Methode «AuthorizeAsync» überprüft, ob der Angemeldete Benutzer (der erste Parameter der Methode) das «adminRequirement» (der dritte Parameter) erfüllt. Wie jedoch weiss er, ob er das erfüllt?

Ein Bild, das Text, Screenshot enthält.

Automatisch generierte BeschreibungASP.NET Core sucht nach einem «IAuthorizationHandler», das dieses Requirement bearbeiten kann. Dort findet er meinen «AdminAuthorizationHandler», der dann die Methode «HandleRequirementAsync» ausführt.

Abbildung 41 NavMenu Visibilität

### Zugriffsverweigerung

Wenn ein Benutzer auf die beiden Pages «Benutzerverwaltung» oder «Skillsverwaltung» ohne Administrator-Rechte zugreifen möchte, wird dieser auf den Pfad «Account/AccessDenied» weitergeleitet. Dort wird er darauf hingewiesen, dass er nicht die Rechte hat, diese Seite zu besuchen. Damit der Benutzer weitergeleitet wird, pass ich das «Program.cs» file mit einem «ConfigureApplicationCookie» an, wo der Pfad angegeben wird, an den der Benutzer weitergeleitet werden soll. Es wurde eine Razor-Page erstellt namens «AccessDenied», die dann angezeigt wird.

 Folgender Screenshot zeigt diese Konfiguration:

Abbildung 42 Zugriffsverweigerung Konfiguration

## Pages und Komponenten

Hier beginnt die Dokumentation aller Pages und die gebrauchten Konzepte.

### Bearbeiten von Einträgen

In diesem Abschnitt wird das Bearbeiten generell für Einträge in einer In-Line Edit Tabelle beschrieben, da das Prinzip in jeder Page das gleiche ist. Ich verwende dazu die Page «Benutzerverwaltung».

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte BeschreibungBeim Klick auf einen Eintrag wird ein «Backup» des aktuellen Objekts mit einem Mapping Prinzip erstellt und in einer Variable gespeichert. Wenn das Editieren abgebrochen wird, werden die Daten auf das Objekt, welches vorher gespeichert wurde, zurückgesetzt. Diese Methoden werden in den Parametern der MudBlazor Tabelle angegeben.

Abbildung 43 Backup und Reset

Wenn man den editierten Eintrag akzeptiert, wird die Methode «UpdateMitarbeiterAsync» ausgeführt. Hier wird der MitarbeiterService verwendet. Das aktuell bearbeitete Objekt wird in der Methode «UpdateMitarbeiterAsync» mitgegeben. Dort wird der bearbeitete Mitarbeiter mit der ID des mitgegebenen Objekts gesucht und anschliessend werden die Daten des Mitarbeiters mit den bearbeiteten Daten des mitgegebenen MitarbeiterDTOs zugewiesen und aktualisiert.

Ein Bild, das Text, Screenshot, Software, Multimedia-Software enthält.

Automatisch generierte Beschreibung

Abbildung 44 Update von Daten

### DTOs

Ein Bild, das Text, Screenshot, Software, Betriebssystem enthält.

Automatisch generierte BeschreibungDa ich das Prinzip «Separation of Concerns» in meiner IPA einhalten möchte, erstelle ich DTOs (Data transfer objects), die ich dann in Tabellen und Services als Nutzobjekte verwende. Damit ich diese verwenden kann wurden Mapper erstellt, die ein DTO zu einer Datenbank Entity und umgekehrt mappen. Folgend ein Beispiel zum Aufbau eines Mappers.

Abbildung 45 Skillkategorie Mapper

### Services

Um auf Daten von der Datenbank zugreifen zu können, erstelle ich pro Entity einen Service, der für das Holen, das Bearbeiten und Löschen der Daten zuständig ist. Diese Services verwenden den «SkillmatrixDbContext» um auf die Daten in der Datenbank zugreifen zu können und werden in der Page, in der sie gebraucht werden, mit einem «@inject» in die Pages injected. Es wird ein dbContext Instanziert, mit dem auf die Entities zugegriffen werden kann.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte BeschreibungHier ein Beispiel wie alle Skillkategorien mittels dem «SkillkategorieService» abgerufen werden.

Abbildung 46 SkillkategorieService

### Modale

Alle Modale befinden sich im Ordner «Modals» des Projekts. Die Modale haben im oberen Abschnitt einen «MudDialog», der alle Eingabe-Komponente, wie Textfelder, Checkboxen, usw. beinhaltet.

Ein Bild, das Text, Schrift, Software, Screenshot enthält.

Automatisch generierte BeschreibungDarunter befindet sich der Codeabschnitt. Hier können zusätzliche Parameter mit einem «Parameter» Attribut, hinzugefügt werden. Wenn ich einen Mitarbeiter löschen möchte, kann ich beispielsweise den Mitarbeiter Name und Vorname und die MitarbeiterId mitgeben, damit das Modal den Namen bei der Frage, ob der Mitarbeiter wirklich gelöscht werden soll, anzeigen kann. Im Modal selber wird die mitgegebene MitarbeiterId dann verwendet, um den diesen bei Bestätigung zu löschen.

Abbildung 47 Modal Mitarbeiter Löschen

### Skillsverwaltung Page

Auf der Skillverwaltung Page können Administrator-Benutzer neue Skillkategorien und Skills erstellen, bearbeiten und löschen. Dafür verwende ich die InLine-Edit Tabelle von MudBlazor.

Mittels einem MudGrid wird die Page in zwei Teile unterteilt, einmal für die Tabelle der Skillkategorien und für die Tabelle der Skills. Was man auf dieser Page beachten muss, ist, dass bei jedem neuen Skill, der hinzugefügt wird, ein neuer «MitarbeiterSkill» Eintrag erstellt werden muss, damit sich alle Mitarbeiter in diesem neuen Skill bewerten können. Ich beschreibe diesen Vorgang nun genauer.

Auf der «Skillverwaltung»-Page befindet sich eine «AddSkillAsync»-Methode, die das Dialog-Fenster «AddSkill» öffnet und auf ein erfolgreiches Hinzufügen wartet. Wenn das Hinzufügen funktioniert hat, bildet es eine Snackbar dar, dass der Skill erfolgreich hinzugefügt wurde.

Ein Bild, das Text, Screenshot, Software, Multimedia-Software enthält.

Automatisch generierte BeschreibungIm Dialog wird der neue Skill zuerst erstellt und der Datenbank hinzugefügt, damit der Skill eine eindeutige ID erhält.

Abbildung 48 Modal Neuer Skill

Ein Bild, das Text, Screenshot, Software enthält.

Automatisch generierte BeschreibungBeim Erstellen des Skills im Code bekommt dieser noch keine ID. Dies übernimmt die Datenbank. Die ID des hinzugefügten Skills wird anschliessend in einer Variable gespeichert. Danach rufen wir die Methode auf, die die neue Skill-ID verwendet, um für jeden Mitarbeiter einen neuen «MitarbeiterSkill» Eintrag zu erstellen.

Abbildung 49 SkillService Neue MitarbeiterSkill Einträge erstellen

Hier werden zuerst alle Mitarbeiter in eine Liste geschrieben, über die dann im foreach-Loop iteriert wird. Beim Iterieren wird ein neues MitarbeiterSkill-Objekt erstellt, das die MitarbeiterId, die SkillId (des neu erstellten Skills) und die Selbsteinschätzung standartmässig auf 0 setzt. Mit «context.SaveChangesAsync» werden die Einträge in der Datenbank vorgenommen.

### Benutzerverwaltung Page

Die Benutzerverwaltung Page ist ähnlich aufgebaut wie die Skillsverwaltungspage. Sie bietet eine InLine-Edit Tabelle, in der die Daten geändert werden können.

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte BeschreibungAuch hier müssen bei einem neuen Mitarbeiter alle Skills diesem zugewiesen werden, das heisst es müssen neue «MitarbeiterSkill» Einträge erstellt werden. Dies läuft ähnlich ab wie beim Hinzufügen eines neuen Skills. Der Unterschied liegt darin, dass wir alle Skills holen, und somit dann die neuen «MitarbeiterSkill» Einträge erstellen.

Abbildung 50 MitarbeiterSkillService Neue MitarbeiterSkill Einträge erstellen

### Skillmatrix Page

Hier beschreibe ich einige wichtige Vorgehen, die ich bei der Programmierung der Skillmatrix Page machte.

#### Datenabfrage für die Skillmatrix Page

Für die Skillmatrix Page erstelle ich zwei neue DTOs. Das erste DTO heisst «SkillkategorieMitSelbsteinschätzungDTO» das von SkillkategorieDTO erbt, damit dieser ebenfalls die Felder der SkillkategorieDTO enthält, und besitzt zusätzlich eine Liste vom zweiten DTO namens «SkillMitSelbsteinschaetzungDTO», das von «SkillDTO» erbt, und zusätzlich noch ein Feld für die Selbsteinschätzung beinhaltet. Somit können wir bei einem Klick auf den Mitarbeiter in der Mitarbeiter Tabelle die ID des Mitarbeiters verwenden um seine Skillkategorien mit Skills und der Selbsteinschätzung abrufen können. Dies geschieht im neu erstellten «MitarbeiterSkillService» mit der Methode «GetSkillkategorieWithSkillsWithSelbsteinschaetzungAsync». Hier werden zuerst alle mitarbeiterSkills mit der mitgegebenen MitarbeiterId geholt. Ich verwende hier Eager-Loading mit «.Include» damit keine zusätzlichen Abfragen für die Skills erstellt werden müssen. Hierbei wird über die Navigation Expression der Entities navigiert.

Ein Bild, das Text, Screenshot, Software enthält.

Automatisch generierte BeschreibungAnschliessend werden die Daten nach der Skillkategorie gruppiert. Danach werden mit dem Select die verschiedenen «SkillkategorieMitSelbsteinschaetzung» erstellt wo wir die Values mappen. Ebenfalls werden die dazugehörigen Skills mit der Selbsteinschätzung ebenfalls mit einem Select in die Liste geschrieben.

Abbildung 51 MitarbeiterSkillService Alle Skillkategorien und Skills mit Selbsteinschätzung

Diese Abfrage hat mich am meisten Zeit beim Erstellen der Skillmatrix gekostet. Deshalb bin ich am 14.03.2024 mit meiner Zeitplanung in Verzug geraten.

#### Editieren der Selbsteinschätzung

Ein Bild, das Text, Screenshot, Schrift, Software enthält.

Automatisch generierte BeschreibungIn der Anforderung AN-05 wird beschrieben, dass Administratoren die Selbsteinschätzung aller Mitarbeiter bearbeiten können. Mitarbeiter, die nicht das Admin-Recht haben, können nur sich selbst bewerten. Hierfür habe ich eine neue Methode «IsAllowedToEditAsync» erstellt, die überprüft, ob der Angemeldete Benutzer Admin ist oder ob die Email des Angeklickten Mitarbeiters der Email des angemeldete Benutzers entspricht. Ich gebe dieser Methode ein «MitarbeiterDTO» Objekt mit, da ich die Methode beim Anklicken des Mitarbeiters aufrufen werde. Je nachdem ob der Benutzer Admin ist oder seinen eigenen Mitarbeiter angeklickt hat, setzt ich den boolean «readOnly» auf false. Dieser boolean wert ist mit der Tabelle der Selbsteinschätzung «gebindet» und legt somit fest, ob die Tabelle readonly ist oder nicht.

Abbildung 52 Erlaubtes Editieren der Selbsteinschätzung

#### Filtrierung

Auf der Skillmatrix Page kann man mit einem Dynamischen Filter nach Mitarbeiter Suchen, die ein bestimmtes Skillkriterium erfüllen. Hierfür habe ich zwei neue Komponente erstellt. Ein «DynamicFilterContainer», der als Container für die Filter, welche dynamisch hinzugefügt, dient und ein «DynamicFilter», der eigentliche Filter. Ausserdem wurde eine Klasse «FilterModel» erstellt, das die Filterkriterien «SkillName», «MinRating» und «MaxRating» beinhaltet.

Der DynamicFilterContainer verwaltet die aktiven Filter und bietet die Funktionalität, Filter hinzuzufügen, zu entfernen und die Suchaktion auszuführen. Es beinhaltet eine Liste von «FilterModel», die mit «AddFilter» neue FilterModelle erhält und mit «RemoveFilter» FilterModelle entfernen kann.

Wenn der Benutzer auf den Suchbutton klickt wird der Callback «FiltersChanged» aufgerufen welcher die aktuelle Liste der Filter übergibt. Diese Liste brauchen wir in der Skillmatrix Page um nach den Filterkriterien zu suchen.

Der DynamicFilter stellt die Benutzeroberfläche für den Filter dar. Benutzer können hier mit einem Dropdown den Skill wählen und die minimale und maximale Bewertung eintragen.

Er beinhaltet eine Instanz von «FilterModel», der die spezifischen Daten für diesen Filter hält. In der Methode «OnRemove» wird ein EventCallback aufgerufen. Dieser löst die Methode «RemoveFilter» im «DynamicFilterContainer» auf.

Folgend ein Bild wie der «DynamicFilterContainer» in der Skillmatrix Page eingefügt wurde.

Abbildung 53 DynamicFilterComponent

Wenn nun ein Filter hinzugefügt oder gelöscht wird, führt dies die Methode «SearchUsersAsync» in der Skillmatrix Page aus. Dieser Methode wird die aktuelle Liste an FilterModel mitgegeben, um die Abfrage der Daten zu ermöglichen. Mittels «MitarbeiterSkillService» und der Methode «GetMitarbeiterBySkillAndRatingAsync» welche die Liste der aktuellen FilterModels erhält, habe ich eine LINQ-Abfrage erstellt.

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte BeschreibungZuerst werden alle Mitarbeiter in eine IQueryable Variable gespeichert. Die LINQ-Abfrage selber verwendet diese Variable, um anhand der MitarbeiterId mit einem «Join» die «MitarbeiterSkill» Einträge zu erhalten. Anschliessend werden mit einem zweiten Join alle Skills mittels der SkillId abgefragt. Mit einem «where» bestimme ich dann, dass ich nur die Einträge möchte, die dem Skillnamen des Filters entsprechen, wo die Selbsteinschätzung grösser oder gleich dem «MinRating» ist und die Selbsteinschätzund kleiner oder gleich dem «MaxRating» ist. Diese Abfrage wird für jeden Filter gemacht, was bedeutet, dass mehrmals die gleichen Mitarbeiter vorkommen könnten. Mit einem «Distinct» bestimme ich, dass ich keine Duplikate möchte.

Abbildung 54 Mitarbeiter Filtrierung

## Logging

Beim Logging wurde eigentlich angedacht, dass ich selbst einen Logging-Service erstelle und eine eigene Logging Tabelle mit Entity Framework Core erfasse und diese dann befülle. Nach einer Nachfrage mit Dominik Baltzer hat er mit Serilog vorgeschlagen, über das ich mich anschliessend informierte. Ich habe mich für Serilog entschieden. Das Entity «Log» wird also gelöscht.

Die beiden NuGet Packages «Serilog.AspNetCore» und «Serilog.Sinks.MSSqlServer» wurden installiert und die Konfiguration für Serilog wurde im Program.cs vorgenommen.

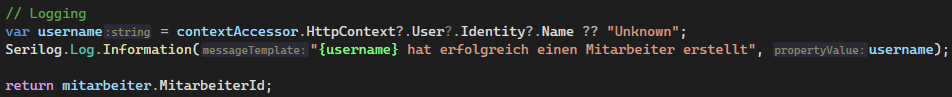
Logs werden immer dann erstellt, wenn sich Daten in der Datenbank verändern, also CRUD-Operationen. Diese werden immer mit dem aktuell angemeldeten Benutzer mit einem Property ergänzt, damit man weiss, wer die Operation durchgeführt hat. Die Logs werden automatisch in die Tabelle «Log» geschrieben. Folgendes Beispiel zeigt das Erstellen eines Logs beim Hinzufügen eines neuen Mitarbeiters.

Abbildung 55 Logging mit Serilog

Ein Bild, das Text, Screenshot, Schrift, Reihe enthält.

Automatisch generierte BeschreibungFolgend ein Bild der Log-Tabelle bei einem Testlauf.

Abbildung 56 Beispiel Logeinträge

## Technische Dokumentation

Auf dem Repository «cmi-skillmatrix» wurde das Markdown File «README.mc» nach einer CMI-Vorlage bearbeitet und angepasst. Es wurde ebenfalls eine Installationsanleitung verlinkt, der die Schritte beschreibt, die der Benutzer beim Starten beachten muss.

Ein Bild, das Text, Screenshot, Schrift, Reihe enthält.

Automatisch generierte BeschreibungZusätzlich wurde im «Program.cs» File eine Anpassung vorgenommen, die dafür sorgt, dass die Datenbank, falls noch nicht vorhanden, auf dem SQL-Server erstellt wird und die Migrationen ausgeführt werden und auf dem aktuellen Stand sind. Somit muss kein SQL-Script bei der Inbetriebnahme manuell ausgeführt werden.

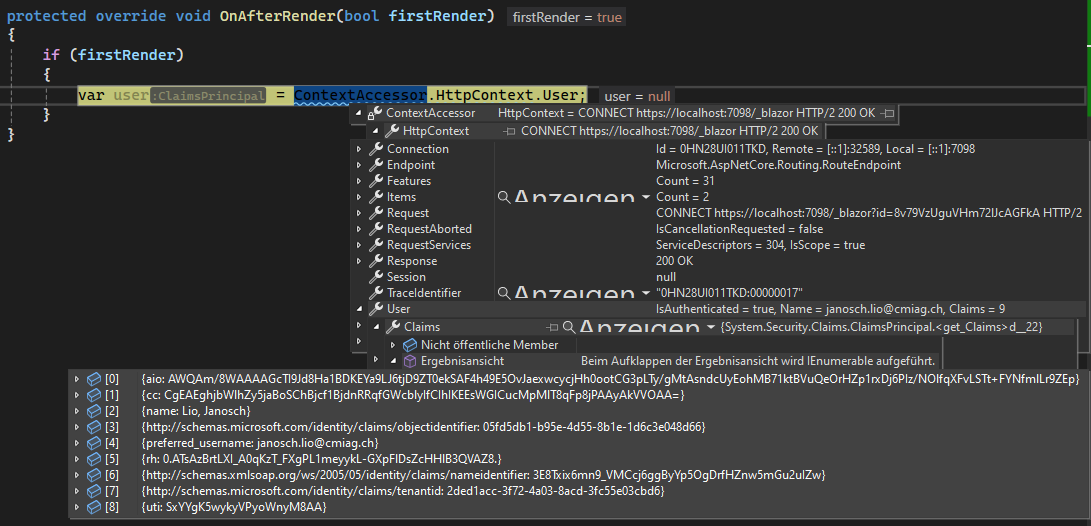
Abbildung 57 Migrate Konfiguration in Program.cs

## Arbeits- und Fachmethodik

Während der Realisationsphase habe ich hauptsächlich den Debugger in Visual Studio 2022 verwendet, um Fehler oder Unklarheiten zu beseitigen. Dieser Abschnitt dient dazu zu beweisen, dass ich die Fachmethodik korrekt anwenden kann. Ich verwende dabei die Page «Home», die einfach den Namen des aktuell angemeldeten Benutzers anzeigt. Ich möchte herausfinden, was der aktuell angemeldete Benutzer für Claims hat. Dafür muss ich den «IHttpContextAccessor» injecten.

Ich setze einen sogenannten «Breakpoint» in der «OnAfterRender» Methode. Dort habe ich eine Variable «user» erstellt, die den User des HttpContext beinhaltet. Wenn ich die Applikation nun starte und mich anmelde, komme ich in diesen Breakpoint rein. Das bedeutet die Applikation stoppt hier und ich kann einsehen, was Variablen oder Objekte beinhalten. Ich schaue nun in die Claims rein:

Abbildung 58 Fachmethodik Debugger



Mit dieser Fachmethodik konnte ich beispielsweise herausfinden, wie ich den Claim, der den Namen beinhaltet, verwende, um den Namen des aktuell angemeldeten Benutzers anzuzeigen.

# Kontrollieren

Hier beginnt die Dokumentation der Kontrollphase

## Testprotokoll

Folgen werden die Tests des Testkonzepts protokolliert.

|  |  |
| --- | --- |
| Testfall-ID | 001 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Die neu erstellte Skillkategorie ist in der Skillkategorie-Tabelle ersichtlich. |
| **Entspricht Erwartung** | Ja |

Tabelle 41 Testprotokoll 001

|  |  |
| --- | --- |
| Testfall-ID | 002 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der neu erstellte Skill «Englisch» mit der Skillkategorie «Sprachen» ist in der Skill-Tabelle ersichtlich |
| **Entspricht Erwartung** | Ja |

Tabelle 42 Testprotokoll 002

|  |  |
| --- | --- |
| Testfall-ID | 003 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Die Skillkategorie «Sprachen» ist in der Skillkategorie-Tabelle nicht mehr ersichtlich. Der dazugehörige Skill «Englisch» ist in der Skill-Tabelle nicht mehr ersichtlich |
| **Entspricht Erwartung** | Ja |

Tabelle 43 Testprotokoll 003

|  |  |
| --- | --- |
| Testfall-ID | 004 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Die Skill «Englisch» ist in der Skill-Tabelle nicht mehr ersichtlich |
| **Entspricht Erwartung** | Ja |

Tabelle 44 Testprotokoll 004

|  |  |
| --- | --- |
| Testfall-ID | 005 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Die Skill «Französisch» mit der Skillkategorie «Sprachen» ist in der Skill-Tabelle ersichtlich |
| **Entspricht Erwartung** | Ja |

Tabelle 45 Testprotokoll 005

|  |  |
| --- | --- |
| Testfall-ID | 006 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Die Skillkategorie «Languages» ist in der Skilllkategorie-Tabelle ersichtlich und der Skill «Englisch» hat neu die Skillkategorie «Languages» |
| **Entspricht Erwartung** | Ja |

Tabelle 46 Testprotokoll 006

|  |  |
| --- | --- |
| Testfall-ID | 007 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der neue Mitarbeiter mit folgenden Eigenschaften ist in der Mitarbeiter-Tabelle ersichtlich:  Name: «Lio»  Vorname: «Janosch»  Email: «janosch.lio@cmiag.ch»  IsAdmin: «true» |
| **Entspricht Erwartung** | Ja |

Tabelle 47 Testprotokoll 007

|  |  |
| --- | --- |
| Testfall-ID | 008 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der bearbeitete Mitarbeiter mit folgenden Eigenschaften ist in der Mitarbeiter-Tabelle ersichtlich:  Name: «Hans»  Vorname: «Alexander»  Email: «alexander.klee@cmiag.ch»  IsAdmin: «false» |
| **Entspricht Erwartung** | Ja |

Tabelle 48 Testprotokoll 008

|  |  |
| --- | --- |
| Testfall-ID | 009 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der Mitarbeiter mit folgenden Eigenschaften ist nicht mehr in der Mitarbeiter-Tabelle ersichtlich:  Name: «Klee»  Vorname: «Alexander»  Email: «alexander.klee@cmiag.ch»  IsAdmin: «false» |
| **Entspricht Erwartung** | Ja |

Tabelle 49 Testprotokoll 009

|  |  |
| --- | --- |
| Testfall-ID | 010 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der bearbeitete Mitarbeiter mit folgenden Eigenschaften ist in der Mitarbeiter-Tabelle ersichtlich:  Name: «Klee»  Vorname: «Alexander»  Email: «alexander.klee@cmiag.ch»  IsAdmin: «true» |
| **Entspricht Erwartung** | Ja |

Tabelle 50 Testprotokoll 010

|  |  |
| --- | --- |
| Testfall-ID | 011 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der Mitarbeiter «Janosch Lio» hat folgende Werte in der Skillmatrix:  Skillkategorie: «Sprachen»  Skillname: «Englisch»  Selbsteinschätzung: «3» |
| **Entspricht Erwartung** | Ja |

Tabelle 51 Testprotokoll 011

|  |  |
| --- | --- |
| Testfall-ID | 012 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der Mitarbeiter «Janosch Lio» hat nach der Bearbeitung eines Administrators folgende Werte in der Skillmatrix:  Skillkategorie: «Sprachen»  Skillname: «Englisch»  Selbsteinschätzung: «2» |
| **Entspricht Erwartung** | Ja |

Tabelle 52 Testprotokoll 012

|  |  |
| --- | --- |
| Testfall-ID | 013 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der Mitarbeiter «Janosch Lio» sieht die Skillmatrix und kann seine Skills anschauen |
| **Entspricht Erwartung** | Ja |

Tabelle 53 Testprotokoll 013

|  |  |
| --- | --- |
| Testfall-ID | 014 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der Benutzer sieht den erfassten Mitarbeiter «Janosch Lio» in der Mitarbeitertabelle |
| **Entspricht Erwartung** | Ja |

Tabelle 54 Testprotokoll 014

|  |  |
| --- | --- |
| Testfall-ID | 015 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der Benutzer kann nach dem Skill «Englisch» der Mindesteinschätzung «2» filtrieren und findet den Mitarbeiter «Janosch Lio» in der Mitarbeitertabelle |
| **Entspricht Erwartung** | Ja |

Tabelle 55 Testprotokoll 015

|  |  |
| --- | --- |
| Testfall-ID | 016 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der Benutzer kann sich mit seinem privaten Office 365 Account nicht anmelden und erhält folgende Nachricht:  «does not exist in tenant 'CM Informatik AG'» |
| **Entspricht Erwartung** | Ja |

Tabelle 56 Testprotokoll 016

|  |  |
| --- | --- |
| Testfall-ID | 017 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der Benutzer «janosch.lio@cmiag.ch» kann erfolgreich neue Mitarbeiter erfassen |
| **Entspricht Erwartung** | Ja |

Tabelle 57 Testprotokoll 017

|  |  |
| --- | --- |
| Testfall-ID | 018 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der Benutzer wird erfolgreich darauf hingewiesen, dass er alle Pflichtfelder ausfüllen muss |
| **Entspricht Erwartung** | Ja |

Tabelle 58 Testprotokoll 018

|  |  |
| --- | --- |
| Testfall-ID | 019 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der Benutzer wird erfolgreich darauf hingewiesen, dass er das Pflichtfeld ausfüllen muss |
| **Entspricht Erwartung** | Ja |

Tabelle 59 Testprotokoll 019

|  |  |
| --- | --- |
| Testfall-ID | 020 |
| **Tester** | Janosch Lio |
| **Datum Testdurchführung** | 15.03.2024 |
| **Testergebnis** | Der Benutzer wird erfolgreich darauf hingewiesen, dass er alle Pflichtfelder ausfüllen muss |
| **Entspricht Erwartung** | Ja |

Tabelle 60 Testprotokoll 020

## Testfazit

Alle Tests konnten erfolgreich durchgeführt werden was bedeutet, dass alle funktionalen Anforderungen erreicht wurden.

Während dem Testen sind mir keine Inkonsistente Verhalten oder unerwartete Fehler aufgefallen. Die Benutzeroberfläche ist benutzerfreundlich und beim Editieren, Erstellen oder Löschen von Daten wird man mit einer kleinen Nachricht unten links in der Ecke informiert.

Folgende Aspekte wurden bewusst nicht getestet:

|  |  |
| --- | --- |
| Test | Begründung |
| Leistung und Skalierbarkeit unter extremen Bedingungen | Die Skillmatrix wird nur als Nebentool und sehr wahrscheinlich nicht von viele Benutzern gleichzeitig verwendet |
| Kompatibilität mit vielen / älteren Browsern | Ich gehe davon aus, dass die Mitarbeiter der CM Informatik moderne Technologie benutzt |
| Verschiedene Netzwerkbedingungen | Tests unter verschiedenen Netzwerkbediungungen wie beispielsweise schwache Verbindung oder hohe Latenz wird ausgelassen, da das Tool hauptsächlich in stabilen Netzwerken verwendet wird |

Tabelle 61 Nicht-getestete Aspekte

# Auswerten

In dieser Phase wird das ganze Projekt nach der Verifizierung aller Testfälle ausgewertet. Hierfür ist die letzte Phase von IPERKA, «Auswerten» gedacht.

Ich finde, dass das Projekt gut gelungen ist. Der Zeitplan hat mir immer eine Vorlage gegeben, was heute zu tun ist und wo ich stehe. Bis auf einen Tag war ich immer im Plan. Dadurch, dass ich jeden Tag zwei Stunden für die Dokumentation eingeplant hatte, bin ich nicht in Stress geraten. Es wurden alle Meilensteine erreicht.

In der Realisationsphase hatte ich den Zeitaufwand für die Skillmatrix Page etwas unterschätzt. Da ich aber einen Realisationspuffer einbaute, glich sich das am nächsten Tag wieder aus.

IPERKA hat mich durch die IPA geleitet. Ich habe mich durch die Informationsphase, Planungsphase und Entscheidungsphase in der Realisationsphase sicher gefühlt und wusste was zu tun ist.

In der Kontrollier Phase habe ich alle vorher spezifizierten Testfälle getestet und ausgewertet. Jeder Testfall ist erfolgreich durchgelaufen.

Wo ich noch Verbesserungspotential sehe, ist bei dem Anklicken von Mitarbeitern in der Skillmatrix während man einen Skill noch am Bearbeiten ist. Wenn ein Skill noch in bearbeitung ist und der Benutzer den Mitarbeiter wechselt, persistieren die Skills in der Skilltabelle. Man sieht erst wieder die Skills eines anderen Mitarbeiters, wenn das Bearbeiten abgeschlossen oder abgebrochen wurde.

## Ausbaumöglichkeiten

Währen dem Programmieren und dem Konzeptionieren der Skillmatrix sind mir einige Ausbaumöglichkeiten in den Sinn gekommen, für die ich während der IPA nicht genügen Zeit hatte, die ich jedoch nach der IPA noch implementieren könnte.

### Hinzufügen eines Skills

Damit die Skillmatrix immer auf dem neusten Stand ist und als verlässliche Mitarbeitersuche in einem Bestimmten Bereich verwendet werden kann, könnte eine Mail an alle Mitarbeiter verschickt werden, wenn ein neuer Skill hinzugefügt wird. Anschliessend könnten sich alle Mitarbeiter in diesem Skill bewerten.

### Übersicht

Auf einer Übersichtseite oder Profilseite könnten verschiedene Graphen dargestellt werden. Zum einen eine Zusammenfassung aller Selbsteinschätzungen und Graphen, die Zeigen in welcher Kategorie man sehr gut ist und in welcher man sich noch verbessern könnte.

### Sortierung der Skillkategorien festlegen

In der alten Skillmatrix Excel-Datei waren die Skillkategorien in einer bestimmten Reihenfolge sortiert. Man könnte jetzt nun in der neuen Skillmatrix ein weiteres Feld «Sortierung» dem Entity «Skillkategorie» hinzufügen. Beim Erstellen oder Editieren einer Skillkategorie könnte man mit einer Nummer festlegen wie die Skillkategorien aufgelistet sind.

## Soll / Ist Vergleich

Für die IPA wurden insgesamt 80 Stunden eingeplant. Effektiv wurden auch 80 Stunden für die Arbeit aufgewendet und es mussten keine Überstunden geleistet werden.

Abbildung 59 Soll / Ist Vergleich

Wie man am Diagramm erkennen kann, habe ich in der Realisationsphase etwas mehr Aufwand gehabt als geplant. Dies ist auf den Längeren Aufwand bei der Skillmatrix zurückzuführen. Dafür hatte ich in der Kontrollier Phase weniger Aufwand, was das ganze wieder ausgleicht.

# Schlusswort

Meine Lehre als Informatiker bei der PMI für drei Jahre und bei der CM Informatik im letzten Jahr haben mich persönlich sehr geprägt. Die ersten drei Jahre meiner Lehrzeit habe ich mit Access und VBA verbracht, was mir persönlich nicht sehr gefallen hat und mir in der Schule leider auch nicht weitergeholfen hat. Mein Wissen, das ich in der Schule bzw. in der PMI angeeignet habe, konnte ich in beiden Bereichen nicht anwenden. In meinem letzten Jahr bei der CMI hingegen durfte ich mich mit C# auseinandersetzten.

Mein Objektorientiertes Wissen war anfangs noch nicht sehr gut, wurde jedoch mit der Zeit in einem wunderbaren Scrum Team (Scrum Team 1), das mir immer geholfen hat und auf das ich immer zugehen konnte, besser.

Trotz der starken Nervosität, die ich am Anfang der IPA hatte, verlief sie sehr gut. Ich bin mit allem durchgekommen und hatte keine grösseren Blockaden, die mich stark in Verzug brachten. Bei Fragen oder Unklarheiten konnte ich mich immer bei Dominik melden.

Die grössten Schwierigkeiten hatte ich beim Dokumentieren, da ich gemäss meinen Erfahrungen nicht so gut im Schreiben bzw. Ausformulieren bin. Grundsätzlich bin ich jedoch zufrieden und auch etwas stolz mit meiner Leistung und dem Ergebnis der IPA. Ich freue mich schon und bin gespannt darauf, die Rückmeldungen der Mitarbeiter zu erhalten und das Produkt in Gebrauch zu sehen.

Ich freue mich auch das Produkt weiterzuentwickeln, da mir das Entwickeln mit Blazor eigentlich sehr spass macht.

Ich möchte abschliessend noch bei folgenden Personen ein grosses Dankeschön ausrichten:

* Der Verantwortlichen Fachkraft Dominik Baltzer, welcher mich unterstützte und bei Fragen immer zur Verfügung stand.
* Meinem Arbeitskollegen Joel Weiss, der meine Dokumentation auf Satzstellung und Rechtschreibung prüfte
* Der Hauptexpertin Michelle Schär und dem Nebenexperten Dirk Decher, welche meine Arbeit bewerten
* Der CM Informatik, welche mir diese Ausbildung ermöglichte

# Glossar

A

**Azure Entra ID**· *Ein Cloud-Dienst für Identitäts- und Zugriffsmanagement, der Benutzeridentitäten und Zugriffsrechte für Cloud-Anwendungen verwaltet.*

B

**Blazor**· *Ein Open-Source-Webframework, das es ermöglicht, interaktive Webanwendungen mit C# statt JavaScript zu erstellen.*

**Blazor Server**· *Ein Hosting-Modell für Blazor-Anwendungen, bei dem die Anwendung auf dem Server ausgeführt und Benutzerinteraktionen über SignalR übertragen werden.*

**Blazor WebAssembly**· *Ein Hosting-Modell für Blazor, bei dem die Anwendung vollständig im Browser des Benutzers ausgeführt wird.*

D

**DTO (Data Transfer Object)**· *Wie es der Name schon sagt, überträgt Daten und wird als Nutzobjekt in der Applikation verwendet*

G

**GitHub**· *Eine Plattform, auf der Code gespeichert und versioniert werden kann.*

I

**Interface**· *Schnittstelle, welche von anderen Klassen implementiert oder auch verwendet*

K

**Kapplanung**· *Die Kapazitätsplanung der CM Informatik welche mit Blazor erstellt wurde.*

M

**MS SQL**· *Steht für Microsoft SQL Server. Dies ist ein relationales Datenbankmanagementsystem von Microsoft.*

N

**Nuget Package**· *Ein Software Packet, welches auf der Nuget Plattform hochgeladen wurde und verwendbar ist.*

S

**Skillmatrix**· *Eine Tabelle, die die Fähigkeiten, Kenntnisse oder Kompetenzen von Mitarbeitern in verschiedenen Bereichen auflistet*

**SeriLog** · *Eine .NET-Logging-Bibliothek, die es ermöglicht, detaillierte Logs zu erstellen und zu verwalten.*

V

**Visual Studio** · Programm von Microsoft mit dem C# Applikationen realisiert werden können*.*

*werden kann. Kann als «Vertrag» betrachtet werden*

**IPA**· *Individuelle praktische Arbeit*

**IPERKA**· *Projektmanagementmethode*

# Verzeichnisse

Folgend das Tabellenverzeichnis und das Abbildungsverzeichnis.

## Tabellenverzeichnis

[Tabelle 1 Legende Ordnerstruktur 11](#_Toc161908380)

[Tabelle 2 Beteiligte Person Janosch Lio 13](#_Toc161908381)

[Tabelle 3 Beteiligte Person Dominik Baltzer 13](#_Toc161908382)

[Tabelle 4 Beteiligte Person Michelle Schär 13](#_Toc161908383)

[Tabelle 5 Beteiligte Person Dirk Decher 13](#_Toc161908384)

[Tabelle 6 IPERKA Phasen 15](#_Toc161908385)

[Tabelle 7 Legende Zeitplan 17](#_Toc161908386)

[Tabelle 8 Meilensteine 17](#_Toc161908387)

[Tabelle 9 Skillmatrix Levels 33](#_Toc161908388)

[Tabelle 10 Funktionale Anforderungen 35](#_Toc161908389)

[Tabelle 11 Entity Framework Core 37](#_Toc161908390)

[Tabelle 12 Blazor Frontend Components 45](#_Toc161908391)

[Tabelle 13 Mitarbeiter Entity 47](#_Toc161908392)

[Tabelle 14 MitarbeiterSkill Entity 47](#_Toc161908393)

[Tabelle 15 Skill Entity 47](#_Toc161908394)

[Tabelle 16 Skillkategorie Entity 47](#_Toc161908395)

[Tabelle 17 Log Entity 48](#_Toc161908396)

[Tabelle 18 Testumgebung 49](#_Toc161908397)

[Tabelle 19 Testfall 001 49](#_Toc161908398)

[Tabelle 20 Testfall 002 50](#_Toc161908399)

[Tabelle 21 Testfall 003 50](#_Toc161908400)

[Tabelle 22 Testfall 004 51](#_Toc161908401)

[Tabelle 23 Testfall 005 51](#_Toc161908402)

[Tabelle 24 Testfall 006 52](#_Toc161908403)

[Tabelle 25 Testfall 007 52](#_Toc161908404)

[Tabelle 26 Testfall 008 53](#_Toc161908405)

[Tabelle 27 Testfall 009 53](#_Toc161908406)

[Tabelle 28 Testfall 010 54](#_Toc161908407)

[Tabelle 29 Testfall 011 55](#_Toc161908408)

[Tabelle 30 Testfall 012 55](#_Toc161908409)

[Tabelle 31 Testfall 013 56](#_Toc161908410)

[Tabelle 32 Testfall 014 56](#_Toc161908411)

[Tabelle 33 Testfall 015 57](#_Toc161908412)

[Tabelle 34 Testfall 016 57](#_Toc161908413)

[Tabelle 35 Testfall 017 58](#_Toc161908414)

[Tabelle 36 Testfall 018 58](#_Toc161908415)

[Tabelle 37 Testfall 019 58](#_Toc161908416)

[Tabelle 38 Testfall 020 59](#_Toc161908417)

[Tabelle 39 MudProviders 63](#_Toc161908418)

[Tabelle 40 Azure Entra ID Konfiguration Bezeichnung 67](#_Toc161908419)

[Tabelle 41 Testprotokoll 001 79](#_Toc161908420)

[Tabelle 42 Testprotokoll 002 79](#_Toc161908421)

[Tabelle 43 Testprotokoll 003 79](#_Toc161908422)

[Tabelle 44 Testprotokoll 004 79](#_Toc161908423)

[Tabelle 45 Testprotokoll 005 80](#_Toc161908424)

[Tabelle 46 Testprotokoll 006 80](#_Toc161908425)

[Tabelle 47 Testprotokoll 007 80](#_Toc161908426)

[Tabelle 48 Testprotokoll 008 80](#_Toc161908427)

[Tabelle 49 Testprotokoll 009 81](#_Toc161908428)

[Tabelle 50 Testprotokoll 010 81](#_Toc161908429)

[Tabelle 51 Testprotokoll 011 81](#_Toc161908430)

[Tabelle 52 Testprotokoll 012 82](#_Toc161908431)

[Tabelle 53 Testprotokoll 013 82](#_Toc161908432)

[Tabelle 54 Testprotokoll 014 82](#_Toc161908433)

[Tabelle 55 Testprotokoll 015 82](#_Toc161908434)

[Tabelle 56 Testprotokoll 016 83](#_Toc161908435)

[Tabelle 57 Testprotokoll 017 83](#_Toc161908436)

[Tabelle 58 Testprotokoll 018 83](#_Toc161908437)

[Tabelle 59 Testprotokoll 019 83](#_Toc161908438)

[Tabelle 60 Testprotokoll 020 83](#_Toc161908439)

[Tabelle 61 Nicht-getestete Aspekte 84](#_Toc161908440)

## Abbildungsverzeichnis

[Abbildung 1 Versionsverwaltung GitHub 11](#_Toc161908441)

[Abbildung 2 Ordnerstruktur 11](#_Toc161908442)

[Abbildung 3 Backup Strategie 12](#_Toc161908443)

[Abbildung 4 Arbeitsplatz 12](#_Toc161908444)

[Abbildung 5 Organigramm 14](#_Toc161908445)

[Abbildung 6 IPERKA Methode 14](#_Toc161908446)

[Abbildung 7 IPERKA Phasen 16](#_Toc161908447)

[Abbildung 8 Zeitplan 18](#_Toc161908448)

[Abbildung 9 Excel-Skillmatrix 34](#_Toc161908449)

[Abbildung 10 Blazor Server 36](#_Toc161908450)

[Abbildung 11 Blazor WebAssembly 37](#_Toc161908451)

[Abbildung 12 Mockup Skillmatrix 38](#_Toc161908452)

[Abbildung 13 Mockup Skillverwaltung 39](#_Toc161908453)

[Abbildung 14 Mockup Neue Skillkategorie 40](#_Toc161908454)

[Abbildung 15 Mockup Skillkategorie löschen 40](#_Toc161908455)

[Abbildung 16 Mockup Neuer Skill 41](#_Toc161908456)

[Abbildung 17 Mockup Skill löschen 41](#_Toc161908457)

[Abbildung 18 Mockup Benutzerverwaltung 42](#_Toc161908458)

[Abbildung 19 Mockup Neuer Mitarbeiter 43](#_Toc161908459)

[Abbildung 20 Mockup Mitarbeiter Löschen 43](#_Toc161908460)

[Abbildung 21 Tabelle Blazor Bootstrap 44](#_Toc161908461)

[Abbildung 22 Tabelle MudBlazor 45](#_Toc161908462)

[Abbildung 23 Use Case Diagramm 46](#_Toc161908463)

[Abbildung 24 Datenbankdiagramm 46](#_Toc161908464)

[Abbildung 25 Gesamtsystem 48](#_Toc161908465)

[Abbildung 26 Entscheidung Blazor Hosting Modelle 60](#_Toc161908466)

[Abbildung 27 Entscheidung Blazor Komponenten Bibliothek 61](#_Toc161908467)

[Abbildung 28 Entscheidungsgespräch 62](#_Toc161908468)

[Abbildung 29 Webanwendung Grundgerüst 64](#_Toc161908469)

[Abbildung 30 Connection String 64](#_Toc161908470)

[Abbildung 31 Connection String Konfiguration 64](#_Toc161908471)

[Abbildung 32 Fluent API Assoziationen Konfiguration 65](#_Toc161908472)

[Abbildung 33 DbContext Konfiguration 65](#_Toc161908473)

[Abbildung 34 Migration MSSQL Tabellen 66](#_Toc161908474)

[Abbildung 35 Azure Entra ID Konfiguration 67](#_Toc161908475)

[Abbildung 36 Azure Entra ID Konfiguration Program.cs 67](#_Toc161908476)

[Abbildung 37 Login Fenster 68](#_Toc161908477)

[Abbildung 38 Bootstrapping Admin Users 68](#_Toc161908478)

[Abbildung 39 IAuthorizationHandler 68](#_Toc161908479)

[Abbildung 40 Authorization Konfiguration 69](#_Toc161908480)

[Abbildung 41 NavMenu Visibilität 69](#_Toc161908481)

[Abbildung 42 Zugriffsverweigerung Konfiguration 70](#_Toc161908482)

[Abbildung 43 Backup und Reset 70](#_Toc161908483)

[Abbildung 44 Update von Daten 71](#_Toc161908484)

[Abbildung 45 Skillkategorie Mapper 71](#_Toc161908485)

[Abbildung 46 SkillkategorieService 72](#_Toc161908486)

[Abbildung 47 Modal Mitarbeiter Löschen 72](#_Toc161908487)

[Abbildung 48 Modal Neuer Skill 73](#_Toc161908488)

[Abbildung 49 SkillService Neue MitarbeiterSkill Einträge erstellen 73](#_Toc161908489)

[Abbildung 50 MitarbeiterSkillService Neue MitarbeiterSkill Einträge erstellen 74](#_Toc161908490)

[Abbildung 51 MitarbeiterSkillService Alle Skillkategorien und Skills mit Selbsteinschätzung 75](#_Toc161908491)

[Abbildung 52 Erlaubtes Editieren der Selbsteinschätzung 75](#_Toc161908492)

[Abbildung 53 DynamicFilterComponent 76](#_Toc161908493)

[Abbildung 54 Mitarbeiter Filtrierung 76](#_Toc161908494)

[Abbildung 55 Logging mit Serilog 77](#_Toc161908495)

[Abbildung 56 Beispiel Logeinträge 77](#_Toc161908496)

[Abbildung 57 Migrate Konfiguration in Program.cs 77](#_Toc161908497)

[Abbildung 58 Fachmethodik Debugger 78](#_Toc161908498)

[Abbildung 59 Soll / Ist Vergleich 86](#_Toc161908499)

# Angang

Der Anhang wird als Zip-File bei den IPA-Anhängen abgegeben. Das File beinhaltet einen Code Ordner mit dem Selbsterstellten Code. Alles, bis auf den Migration Ordner, ist selbst erstellt oder angepasst. Ein Ordner «IPA-Dokumente» existiert ebenfalls, indem sich alle Dokumente (Diagramme, Zeitplan, usw.) befindet.

Die Technische Dokumentation befindet sich im Ordner «Technische Dokumentation».

Die Selbstständigkeitserklärung wird in die History von PkOrg hochgeladen.

Die IPA-Dokumentation wird als PDF in der Abgabe «IPA Bericht» abgegeben.

1. Projekte-leicht-gemacht.de

   https://projekte-leicht-gemacht.de/blog/projektmanagement/klassisch/projektplanung/definition-meilensteine/ [↑](#footnote-ref-1)
2. learn.microsoft.com

   https://learn.microsoft.com/de-de/entra/fundamentals/whatis [↑](#footnote-ref-2)
3. bbv.ch

   https://bbv.ch/insights/blog/blazor/ [↑](#footnote-ref-3)
4. learn.microsoft.com

   https://learn.microsoft.com/de-de/ef/core/ [↑](#footnote-ref-4)
5. https://mudblazor.com/components/table#default-table [↑](#footnote-ref-5)
6. https://demos.blazorbootstrap.com/grid [↑](#footnote-ref-6)
7. learn.microsoft.com

   https://learn.microsoft.com/en-us/ef/core/modeling/relationships/many-to-many [↑](#footnote-ref-7)