



Abschlussprüfung Winter 2023/2024

Fachinformatiker für Anwendungsentwicklung

Dokumentation zur betrieblichen Projektarbeit

## Webapplikation „Perspektiv Dialog“

Laravel-Applikation zur Erhebung und Auswertung von  
„Perspektiv Dialog“-Umfragen.

Abgabedatum: Kiel, den 30.11.2023

**Prüfungsbewerber:**

Werschinin Elena

Ilssahl 9

24536 Neumünster

**Ausbildungsbetrieb:**

SWN Stadtwerke Neumünster Beteiligungen

GmbH

Bismarckstraße 51, 24534 Neumünster

**Ausbildungsstätte:**

ABS IT UG (haftungsbeschränkt)

Sophienblatt 82 - 86

24114 Kiel



## Inhaltsverzeichnis

1	Einleitung .....	1
1.1	Projektumfeld .....	1
1.2	Projektziel .....	2
1.3	Projektbegründung .....	2
1.4	Projektschnittstellen .....	3
2	Projektplanung .....	4
2.1	Projektphasen .....	4
2.2	Ressourcenplanung .....	5
2.3	Entwicklungsprozess.....	5
3	Analyse5 .....	
3.1	Ist-Zustand .....	5
3.2	Wirtschaftlichkeitsanalyse.....	5
3.2.1	„Make or Buy“-Entscheidung.....	6
3.2.2	Projektkosten .....	6
3.3	Nutzwertanalyse .....	7
3.4	Anwendungsfälle.....	7
3.5	Lastenheft / User Storys .....	8
4	Entwurfsphase .....	8
4.1	Zielplattform .....	8
4.2	Architekturdesign.....	8
4.3	Entwurf der Benutzeroberfläche .....	9
4.4	Datenmodell .....	10
5	Implementierungsphase .....	11
5.1	Implementierung der Datenstrukturen.....	11
5.2	Implementierung der Benutzeroberfläche .....	15
6	Abnahmephase.....	16
6.1	Tests.....	16
7	Einführungsphase .....	16
8	Fazit .....	17
8.1	Soll-/Ist-Vergleich .....	17
8.2	Lessons Learned.....	17
8.3	Ausblick .....	18
Anhang 1	Gantt-Diagramm .....	1
Anhang 2	Flussdiagramm .....	2
Anhang 3	Work Items.....	3
Anhang 4	Ergebnisse .....	4
Anhang 5	Script.....	6
	Abbildungsverzeichnis.....	7

## Abkürzungen

Azure Cloud	Cloud-Computing-Plattform von Microsoft
CI/CD	Kontinuierliche Integration (englisch: Continuous Integration, kurz: CI) und kontinuierliche Auslieferung (englisch: Continuous Delivery, kurz: CD)
Deployment	Verteilung
GitHub	Onlinedienst zur Softwareentwicklung und Versionsverwaltung für Softwareprojekte
GUI	Grafische Benutzeroberfläche (englisch: Graphical User Interface)
HR	Personalwesen oder Personalmanagement (englisch: Human Resources)
IDE	Integrierte Entwicklungsumgebung (englisch: Integrated Development Environment)
IT	Informationstechnik (englisch: Information Technology)
MVC	Modell-Ansicht-Steuerung (englisch: Model View Controller)
Repository	verwaltetes Verzeichnis zur Speicherung und Beschreibung digitaler Objekte für ein digitales Archiv
Scrum	Rahmen für die Zusammenarbeit von Teams, insbesondere bei der agilen Softwareentwicklung
SSO	Einmalanmeldung (englisch: Single Sign-On)
SWN	Stadtwerke Neumünster

# 1 Einleitung

## 1.1 Projektumfeld

Das Projekt fand bei den Stadtwerken in Neumünster statt. „Die Stadtwerke Neumünster (SWN) sind ein regionales Energieversorgungs- und Dienstleistungsunternehmen in Neumünster.

Sie gehen auf die 1899 gegründete Baltische AG Licht-, Kraft- und Wasserwerke Neumünster zurück. Geschäftsfelder sind Energiedienstleistungen, die Versorgung mit Elektrizität, Gas, Fernwärme und Wasser sowie Telekommunikationsdienstleistungen.

Im Geschäftsfeld Energie fungiert SWN in der Stadt Neumünster als Grundversorger. Die Stromerzeugung erfolgt über eine in der Stadt gelegene thermische Ersatzbrennstoffverwertungsanlage, die zugleich zur Erzeugung von Fernwärme genutzt wird. Befeuert wird sie überwiegend mit Ersatzbrennstoff, der in der im südlichen Neumünster gelegenen mechanisch-biologischen Abfallbehandlungsanlage gewonnen wird. Die Abfallbehandlungsanlage ist eine der größten Einrichtungen dieser Art Deutschlands.

Neben der thermische Ersatzbrennstoffverwertungsanlage unterhalten die Stadtwerke auf dem Gelände an der Bismarckstraße ein Heizkraftwerk, das mit Steinkohle betrieben wird.

Verstärkt werden neuerdings energienahe Dienstleistungen wie die Vermittlung von Elektroautos und der Vertrieb von Wandladestationen für die Elektromobilität unter der Marke „nonox“ angeboten.

Im Jahr 2009 erweiterte sich das SWN-Leistungsangebot auf Telekommunikationsdienstleistungen. Unter dem Namen GIGA5 werden Telefonie-, Internet- und TV-Dienste über Glasfaser angeboten. SWN unterhält im Kerngebiet Neumünster und Umgebung ein eigenes Glasfasernetz. In anderen Teilen Schleswig-Holsteins (u. a. die Kreise Segeberg, Dithmarschen und Steinburg) hat das Unternehmen umfassend Netze für Breitband-Zweckverbände geplant und gebaut.

Seit Januar 2015 wird der Stadtbusverkehr in Eigenregie der SWN Verkehr GmbH durchgeführt. Mit 35 Niederflurbussen werden auf 21 Buslinien insgesamt 75 Haltestellen bedient. Rund 2,6 Mio. Fahrgäste jährlich nutzen den Service. Unter der Marke „hinundwech“ wird seit dem Jahr 2020 ein zusätzliches Mobilitätsangebot für stationsgebundenen Ruftaxidienst unterhalten, der über eine Mobile App geordert wird.“<sup>1</sup>

Das Projekt wurde im IT-Fachbereich der SWN in Neumünster durchgeführt. Zu den Aufgaben des IT-Fachbereichs zählt das Analysieren von IT-Projekten und die Entscheidung, diese entweder an einen Dienstleister oder in die Eigenentwicklung zu übergeben. In dieser Abteilung gibt es insgesamt 17 Personen. Der Auftraggeber ist die Personalabteilung der SWN. Die Hauptaufgaben dieser Abteilung sind die Arbeit mit dem Personal, die Erhebung statischer Daten zur Personaleffizienz, die Planung und Durchführung von Maßnahmen zur Verbesserung der Personaleffizienz und der Qualität des Kundenservice.

## **1.2 Projektziel**

Das Ziel des Projekts besteht darin, eine Webapplikation mit einer grafischen Benutzeroberfläche sowie eine Anbindung zu einer Datenbank zu entwickeln, um die eingegebene Information zu speichern und zu lesen. Diese Anwendung bietet den Mitarbeitern eine bequeme Möglichkeit, ihre beruflichen, kommunikativen und anderen Fähigkeiten zu bewerten, während gleichzeitig ihre Vorgesetzten die Möglichkeit haben, sie ebenfalls zu bewerten. Beide Ergebnisse werden in einer Datenbank gespeichert und können für Vergleiche abgerufen werden. Zusätzlich soll nach der Besprechung der Ergebnisse die Möglichkeit bestehen, bestimmte Maßnahmen in einer Datenbank zu hinterlegen.

## **1.3 Projektbegründung**

Die SWN-Personalabteilung ist in Sachen Digitalisierung so weit fortgeschritten, dass entschieden wurde, Personalgespräche nicht mehr auf herkömmliche Weise, sondern digital durchzuführen. Dieser Schritt markiert eine wichtige Weiterentwicklung in der Effizienz und Verwaltung der Personalprozesse.

---

<sup>1</sup> Quelle: [https://de.wikipedia.org/wiki/Stadtwerke\\_Neumünster](https://de.wikipedia.org/wiki/Stadtwerke_Neumünster), Stand: 22.06.2021

Die Digitalisierung von Personalgesprächen bietet zahlreiche Vorteile:

- a) **Klare Struktur:** Digitale Personalgespräche können eine einheitliche und klar strukturierte Vorlage bieten, die es den Teilnehmern erleichtert, die relevanten Informationen einzutragen.
- b) **Strukturierte und leicht auffindbare Daten:** Digitale Aufzeichnungen erleichtern die Strukturierung und das Auffinden von Daten. Dies erleichtert die Nachverfolgung von Gesprächen und die Überprüfung vergangener Maßnahmen oder Vereinbarungen.
- c) **Papier- und Platzersparnis:** Der Umstieg auf digitale Abfragen reduziert den Bedarf an Papier und spart Platz in den Schränken der Personalabteilung. Dies ist nicht nur umweltfreundlicher, sondern auch kosteneffizienter.
- d) **Leichte Analyse der Daten:** Digitale Abfragen können leichter analysiert werden. Dies ermöglicht es der Personalabteilung, Trends zu identifizieren, Schwachstellen zu erkennen und gezielte Maßnahmen zur Verbesserung der Mitarbeiterzufriedenheit und -entwicklung zu ergreifen.

Die Einführung digitaler Personalgespräche soll die Effizienz und Effektivität der Prozesse innerhalb der Personalabteilung der SWN steigern und zur Verbesserung der Mitarbeiterbetreuung und -entwicklung beitragen.

#### 1.4 Projektschnittstellen

Die Webanwendung wird in der Azure Cloud gehostet. Eine wichtige Funktionalität von Azure ist das Single Sign-On (SSO), das in Verbindung mit dem Active Directory zur Authentifizierung der Benutzer dient.

Die Personalabteilung, unter der Leitung von Frau XY<sup>2</sup> als Abteilungsleiterin, ist der Auftraggeber für dieses Projekt. Herr AB ist als technischer Betreuer aus der IT-Abteilung für das Projekt zuständig. Die Zielgruppe für die Anwendung sind sämtliche Mitarbeiter der SWN.

---

<sup>2</sup> Alle Namen sind anonymisiert.

## 2 Projektplanung

### 2.1 Projektphasen

Die Gesamtdauer des Projekts betrug 80 Stunden. Zur effizienten Umsetzung wurde das Projekt in sieben Teile unterteilt, wobei jeder dieser Teile weiter in kleinere Abschnitte aufgeteilt wurde. Die sieben Hauptteile des Projekts sind: Analyse, Entwicklung des Zielkonzepts, Projektplanung, Implementierung, Testphase, Abnahme und Deployment, Erstellung der Dokumentation.

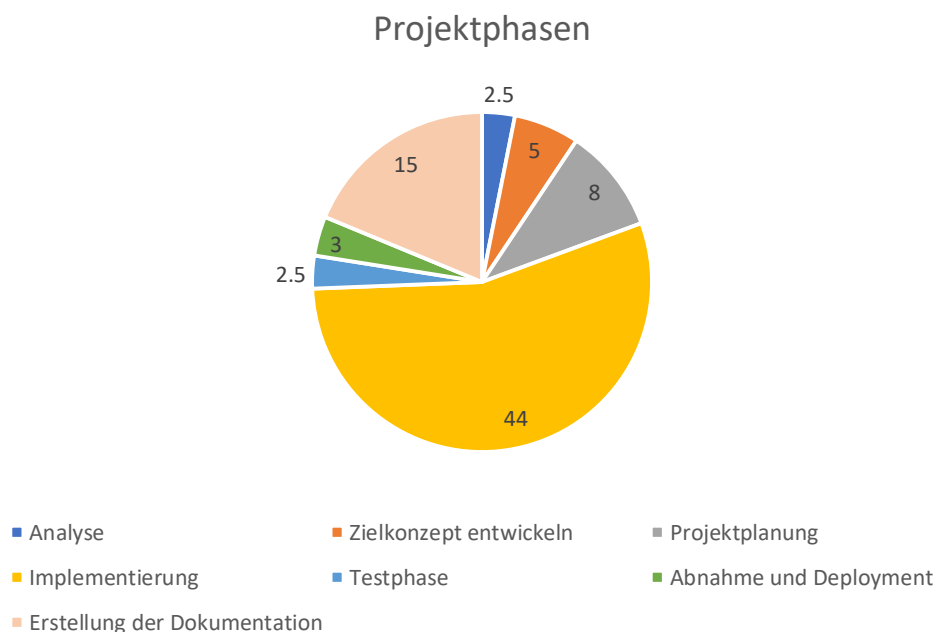


Abbildung 1: Projektphasen

Diese 7 Hauptteile des Projekts wurden in kleinere Teile unterteilt, um die Zeit für jeden Teil des Projekts genauer planen zu können. Im Anhang befindet sich ein Gantt-Diagramm, das alle Teilkomponenten des Projekts, sowie die dafür vorgesehene Zeit verdeutlicht (**Anhang 1**). Außerdem war die ursprünglich vorgesehene Zeit für die Abnahme zu lang und die Zeit für die Erstellung der Dokumentation zu kurz. Nach einer Umverteilung der Zeit wurden 3 Stunden für die Abnahme und weitere 15 Stunden für die Dokumentation vorgesehen.

## **2.2 Ressourcenplanung**

Um das Projekt durchzuführen, wurden Ressourcen benötigt, die sich in Software und Personal unterteilen lassen.

Für die Arbeit mit dem Projekt wurde Visual Studio Code als IDE, Laravel 10 als Webframework, GitHub als Versionskontrollsystem sowie die Azure Cloud als Hosting-Plattform gewählt. Für die Arbeit mit der Datenbank wurde das Tool PgAdmin 4, für die Kommunikation innerhalb des Projekts wurde Microsoft Teams benutzt.

Es wurden 9 Stunden für einen Betreuer innerhalb der IT-Abteilung eingeplant, der bei fachlichen Themen unterstützen konnte. Zusätzlich wurden 5 Stunden für den Auftraggeber einkalkuliert.

## **2.3 Entwicklungsprozess**

Innerhalb der SWN kommt ein an Scrum angelehntes Projektmanagement zum Einsatz, welches den Entwicklungsprozess flexibel gestaltet und dabei eine effiziente Dokumentation ermöglicht.

# **3 Analyse**

## **3.1 Ist-Zustand**

Die Personalabteilung verfügt derzeit nicht über eine spezifische Anwendung zur Feststellung und Verbesserung der Mitarbeitereffizienz und -qualität. Derzeit werden alle Personalgespräche mündlich geführt, die Daten werden digital nicht gespeichert, und es ist nicht einfach möglich, die Dynamik der Fähigkeiten der Mitarbeiter nachzuvollziehen.

## **3.2 Wirtschaftlichkeitsanalyse**

Aus Sicht des Auftraggebers wird das Projekt als vorteilhaft erachtet, da es ermöglicht, die Arbeitsqualität jedes Mitarbeiters zu analysieren, eine Strategie auszuwählen und Maßnahmen aufgrund dieser Analyse zu ergreifen. Dies trägt wiederum dazu bei, die Produktivität der Mitarbeiter zu erhöhen.



Es ist schwer, die genaue Wirkung der Anwendung jetzt in Zahlen abzuschätzen. Das Unternehmen betrachtet diese Anwendung jedoch als Mittel zur Verbesserung der Servicequalität und Arbeitseffizienz.

### 3.2.1 „Make or Buy“-Entscheidung

Die Personalabteilung hat gemeinsam mit der IT-Abteilung die verfügbare Software und Dienstleister auf dem Markt geprüft. Es wurde jedoch keine Software gefunden, die den Anforderungen der Personalabteilung entsprach. Aus diesem Grund wurde beschlossen, die Anwendung intern zu entwickeln.

### 3.2.2 Projektkosten

Die Kosten für das Projekt bestehen aus den Ausgaben für Azure-Dienste und den Stundenlöhnen der am Projekt beteiligten Mitarbeiter. In der Tabelle wurden keine Zahlungen für die Entwicklung der Anwendung durch den Praktikanten angegeben, da für die betriebliche Lernphase kein Geld gezahlt wurde. Eine Kostenaufstellung befindet sich in der **Tabelle 1**.

Kostenberechnung für den fachlichen Betreuer:

$$8 \text{ St / Tag} \cdot 220 \text{ Tage / Jahr} = 1.760 \text{ St / Jahr}$$

$$8.667 \text{ € / Monat} \cdot 13 \text{ Monate / Jahr} = 112.671 \text{ € / Jahr}$$

$$112.671 \text{ € / Jahr} \div 1.760 \text{ St / Jahr} = \underline{\underline{64 \text{ € / St}}}$$

Kostenberechnung für den Auftraggeber (Personalabteilung):

$$8 \text{ St / Tag} \cdot 220 \text{ Tage / Jahr} = 1.760 \text{ St / Jahr}$$

$$10.788 \text{ € / Monat} \cdot 13 \text{ Monate / Jahr} = 140.244 \text{ € / Jahr}$$

$$140.244 \text{ € / Jahr} \div 1.760 \text{ St / Jahr} = \underline{\underline{79.68 \text{ € / St}}}$$

Vorgang	Zeit (Stunden)	Kosten pro Stunden	Kosten
<b>Meeting mit Auftraggeber</b>	1 (Auftraggeber) + 1 (IT)	64 € + 79,68 €	143,68 €
<b>User Storys mit Personalabteilungsmitarbeiter erstellen</b>	2 (Auftraggeber)	79,68 * 2 €	159,36 €
<b>Entwicklungsprozess</b>	7 (IT)	64 € * 7	448 €
<b>Deploy</b>	1 (IT)	64 €	64 €
<b>Abnahme</b>	2 (Auftraggeber)	79,68 € * 2	159,36 €
<b>Insgesamt Personalkosten</b>			974,40 €
<b>Azure Cloud Service</b>			12 € / Monat

Tabelle 1: Kostenaufstellung

### 3.3 Nutzwertanalyse

Die nicht-monetären Vorteile dieses Projekts waren von Anfang an klar. Es wurde die Arbeit mit dem Personal vereinfacht, unnötige Schreibarbeit beseitigt und Prozesse digitalisiert. Der Kunde möchte ein Tool, welches ihm ermöglicht die berufliche und soziale Entwicklung der Mitarbeiter zu verfolgen.

Die vom Tool erhobenen Daten werden zu dem genutzt, um den Mitarbeiter bei seiner persönlichen Entwicklung zur Seite zu stehen.

Die Anwendung soll auch Abteilungsleitern helfen, ihre Mitarbeiter besser kennenzulernen. In Zukunft kann die Personalabteilung Daten zur Erstellung von Statistiken nutzen. Diese Datenbasis wird durch die Sammlung und Analyse von Informationen ermöglicht. Die Statistiken können dann als Grundlage zur Verbesserung von verschiedenen Prozessen verwendet werden.

### 3.4 Anwendungsfälle

Die Webanwendung wurde so konzipiert, dass ein Mitarbeiter und sein Vorgesetzter unabhängig voneinander ein Formular ausfüllen können, in dem der Mitarbeiter sowie der Vorgesetzter die beruflichen Qualitäten aus der eigenen Sicht heraus bewerten können. Im **Anhang 2** befindet sich ein Flussdiagramm, das den Prozess des Ausfüllens des Formulars veranschaulicht.

Des Weiteren können sowohl der Mitarbeiter als auch sein Vorgesetzter die von ihnen ausgefüllten Formulare jederzeit überprüfen. Nachdem der Mitarbeiter und sein Vorgesetzter den Fragebogen ausgefüllt haben, besprechen sie die Ergebnisse und wählen die notwendigen Maßnahmen zur Verbesserung der Arbeitsqualität des Mitarbeiters. Während dieser Besprechung kann der Vorgesetzte die Ergebnisse der beiden Umfragen anschauen und Maßnahmen eintragen.

Die Ergebnisse des Abschlussgesprächs und die Liste der erforderlichen Maßnahmen können sowohl vom Arbeitnehmer als auch vom Vorgesetzten eingesehen werden.

### **3.5 Lastenheft / User Storys**

Das Unternehmen entschied sich, die traditionelle Struktur von Lastenheft und Pflichtenheft abzulegen. Im Rahmen eines Treffens mit dem Kunden wurden stattdessen User Storys erstellt, auf deren Grundlage die Anwendung entwickelt wurde. Einige der User Storys sind im **Anhang 3** zu finden.

## **4 Entwurfsphase**

### **4.1 Zielplattform**

Für die Entwicklung nutzt SWN Microsoft-Produkte wie Microsoft Flow oder PowerApps. Für dieses Projekt wird das Laravel Framework verwendet. Ursprünglich war geplant, Laravel in Kombination mit Vue.js zu verwenden, doch später wurde entschieden, dass die Fähigkeiten von Laravel für dieses Projekt ausreichen. Für SWN waren mehrere Kriterien wichtig, wie die Möglichkeit des Hostings auf Azure, Geschwindigkeit von Apps, Update-Fähigkeit, Vorkenntnisse usw.

### **4.2 Architekturdesign**

Laravel ist ein Open-Source-PHP-Framework, mit dem Entwickler Webanwendungen erstellen können. Ein PHP-Framework stellt Entwicklern eine Bibliothek mit bereits programmierten Modulen zur Verfügung, um Webanwendungen schnell zu erstellen. Laravel basiert auf einer MVC-Architektur (Model View Controller), welche eine einfache Aufteilung der Programmbestandteile ermöglicht.

„MVC besteht aus drei Komponenten:

**Model:** Die zentrale Komponente, die die Daten und Regeln in einem Projekt definiert

**View:** Die Darstellung der durch das Modell definierten Daten und Logik

**Controller:** steuert die Funktionen, wandelt sie in nutzbare Formate um und übergibt die Daten an die View- und Modellkomponenten.“<sup>3</sup>

Die Interaktion zwischen Model, View und Controller ist in der folgenden **Abbildung 1** zu sehen.

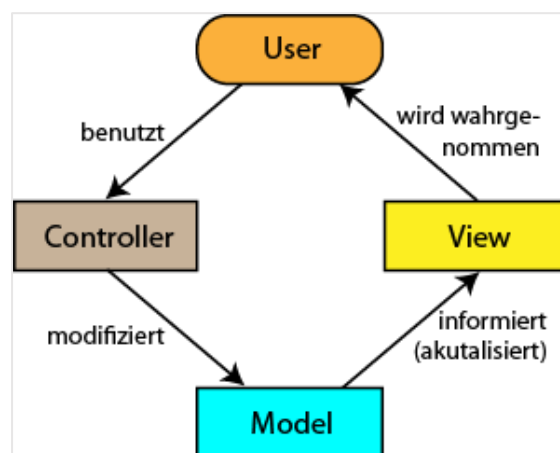


Abbildung 2: (Interaktion zwischen Model, View und Controller)

#### 4.3 Entwurf der Benutzeroberfläche

Für die Anwendung "Perspektiv Dialog" wurden besonderen Wert auf Benutzerfreundlichkeit gelegt, weshalb wir uns entschieden haben, eine grafische Oberfläche (GUI) zu nutzen. Als erster Schritt muss sich der Benutzer zuerst authentifizieren und autorisieren, danach können alle erforderlichen Felder ausgefüllt und gespeichert werden. Insgesamt wurden zur erleichterten Nutzung vier Kategorien von Kriterien für die Selbstbewertung (oder für die Bewertung der eigenen Mitarbeiter, wenn der Vorgesetzter die Anwendung nutzt) erstellt. Die Kategorien wurden als Dropdown-Akkordeon gestaltet. Für die Darstellung wurde das Firmenlogo und die Firmenfarbe gewählt. Für das Dropdown-Akkordeon wurde eine Palette von 4 Pastelltönen hinzugefügt.

<sup>3</sup> Quelle: <https://glossar.hs-augsburg.de/Model-View-Controller-Paradigma>

## 4.4 Datenmodell

Es wurde beschlossen, PostgreSQL als Datenbank für dieses Projekt zu verwenden.

In der folgenden **Abbildung 3** sieht man die Datenbanktabellen, die für das Projekt "Perspektiv Dialog" erstellt wurden, sowie die Beziehungen zwischen diesen Tabellen.

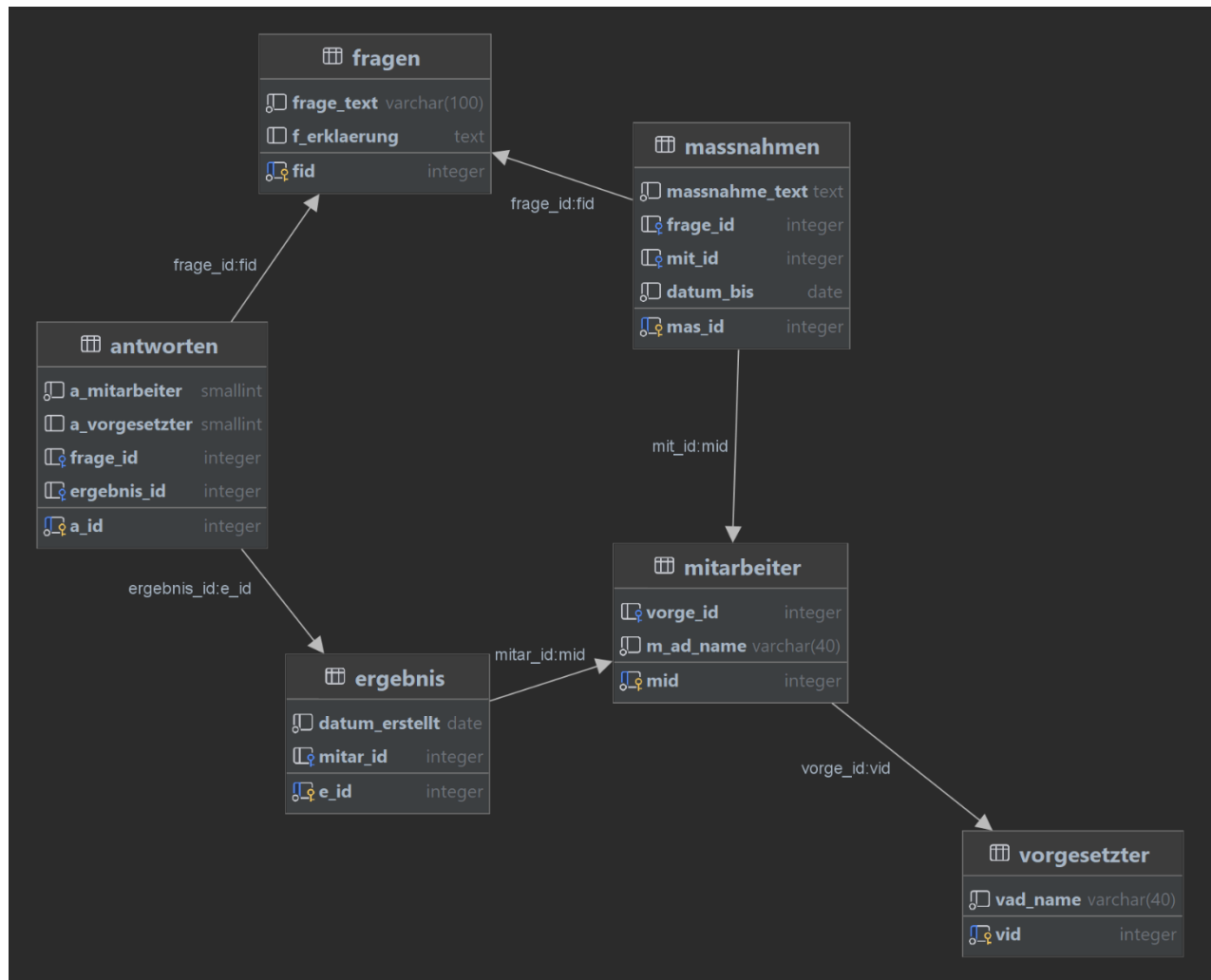


Abbildung 3: Datenbanktabellen

## 5 Implementierungsphase

### 5.1 Implementierung der Datenstrukturen

Migrationen wurden verwendet, um die Datenbankstruktur zu erstellen. Migrationen sind eine Möglichkeit, die Datenbankstruktur in Laravel zu verwalten. Die Migration wurde mit dem Befehl ``php artisan make:migration tabelle_name`` erstellt. Dadurch wurde eine Migrationsdatei im Verzeichnis ``database/migrations`` erstellt. In dieser Datei wurden die Felder definiert, die zur Speicherung der Benutzerdaten verwendet werden.

Code-Schnipsel:

```
class CreateUserDataTable extends Migration
{
    public function up()
    {
        Schema::create('antworten', function (Blueprint $table) {
            $table->a_id();
            $table->smallInteger('a_vorgesetzter');
            $table->smallInteger('a_mitarbeiter');
            $table->integer('frage_id')->unsigned();
            $table->integer('ergebnis_id')->unsigned();

            $table->foreign('frage_id')->references('id')->on('fragen');
            $table->foreign('ergebnis_id')->references('id')->on('ergebnis');
        });
    }
}
```

Die Migration wurde dann mit dem Befehl ``php artisan migrate`` durchgeführt, was zur Erstellung der Tabelle "antworten" in der PostgreSQL-Datenbank führte. Die Tabellen "ergebnis", "vorgesetzter", "mitarbeiter", "fragen" und "massnahmen" wurden ebenfalls erstellt.

Wenn Daten von einem Nutzer gesendet werden, werden die Informationen in einer Datenbank gespeichert. Dies wurde durch den Einsatz von Controllern erreicht, die die Abfragen verarbeiteten und Operationen im Zusammenhang mit der Speicherung der Antworten und der Erstellung von Datensätzen in Tabellen durchführen.

Auszug vom Code eines Controllers:

```
public function saveAntworten(Request $request)
{
    <!-- Abrufen der Antworten aus dem Formular-->
    $antwortenData = $request->all();

    <!-- Abspeichern der Antworten in der Tabelle "antworten"-->
    $antwort = new Antworten();
    $antwort->a_vorgesetzter = $antwortenData['vorgesetzter_id'];
    $antwort->a_mitarbeiter = $antwortenData['mitarbeiter_id'];
    $antwort->frage_id = $antwortenData['frage_id'];
    $antwort->ergebnis_id = $antwortenData['ergebnis_id'];
    $antwort->save();

    <!-- Erstellen eines Datensatzes in der Tabelle "ergebnis"-->
    $ergebnis = new Ergebnis();
    $ergebnis->datum = now();
    $ergebnis->mitarbeiter_id = $antwortenData['mitarbeiter_id'];
    $ergebnis->ergebnis_id = $antwort->id;
    $ergebnis->save();
}
```

Es wurden Routes konfiguriert, die eine Verbindung zwischen URLs und Controller-Methoden herstellten. Auf diese Weise können Benutzer Daten an den Server senden, in dem sie verarbeitet und in einer Datenbank gespeichert werden.

Für die Interaktion mit den Daten in der Tabelle wurde ein Modell erstellt. Ein Modell in Laravel ist ein Objekt, das den Datensätzen in einer Datenbanktabelle entspricht. Das Modell wurde mit dem Befehl `php artisan make:model model\_name` erstellt.

Als nächstes wurde das Modell `antworten` erstellt, um die Struktur der Tabelle widerzuspiegeln und Eigenschaftsdefinitionen wie den Tabellennamen und den Primärschlüssel zu enthalten.

Auszug aus dem Modellcode:

```
class Antworten extends Model
{
    protected $table = 'antworten';
    protected $primaryKey = 'a_id';
    public $timestamps = false;
}
public function fragen()
{
    return $this->belongsTo(Frage::class, 'frage_id', 'f_id');
}
public function ergebnis()
{
    return $this->belongsTo(Ergebnis::class, 'ergebnis_id', 'e_id');
}
```

Es wurden Beziehungen zwischen der Tabelle "antworten" und anderen Tabellen wie "fragen" und "ergebnis" definiert, um sicherzustellen, dass die Daten der Normalform entsprechen. Die Webanwendung kann mit den Daten interagieren, indem sie diese in der Datenbank speichert oder aus ihr abruft.

Es wurden drei Seiten erstellt, die mit den Daten interagieren. Auf der Startseite `home.blade.php` wurden die Daten in den Tabellen "ergebnis" und "antworten" gesammelt. Auf der Seite `ergebnis.blade.php` wurden die Daten aus den Tabellen "antworten", "ergebnis" und "frage" in einer Tabelle dargestellt. Außerdem wurden auf dieser Seite Dateneingaben realisiert, die in die Tabelle "massnahmen" geschrieben wurden.

In der letzten Entwicklungsphase wurde die Seite `ergebnis\_massnahmen.blade.php` hinzugefügt, die eine wichtige Ergänzung der Webanwendung darstellt. Diese Seite ermöglicht es den Benutzern, Daten aus den Tabellen "antworten", "ergebnis", "frage" und "massnahme" einzusehen. Es wurde eine Jahresfilterfunktion implementiert, die es dem Benutzer ermöglicht, einen bestimmten Zeitraum für die Anzeige der Daten auszuwählen.



## Code-Schnipsel:

```
<!-- Formular zur Auswahl des Jahres und zum Senden von Daten an den Server -->
<form method="get" action="{{ route('ergebnis.index') }}">
    <select name="year">
        @foreach ($availableYears as $year)
            <!-- Erstellen von Dropdown-Optionen basierend auf den verfügbaren Jahren, die im
            System unterstützt werden -->
            <option value="{{ $year }}" @if ($year == $selectedYear) selected @endif>{{ $year
        }}</option>
        @endforeach
    </select>
    <!-- Schaltfläche "Zeigen" zum Absenden des Formulars -->
    <button type="submit">Zeigen</button>
</form>

...
<!-- Tabelle zur Anzeige von Daten -->
<tbody>
    @foreach ($antworten as $antwort)
        <!-- Verwendung der @php-Direktive zur Ausführung von PHP-Code im Template -->
        @php
            $ergebnis = $ergebnis->find($antwort->ergebnis_id);
            // Suchen des entsprechenden Datensatzes in der "massnahmen"-Tabelle basierend
            auf dem "ergebnis"-Datum
            $massnahme = $massnahmen->where('massnahme_datum', $ergebnis->datum)-
            >first();
        @endphp
        <tr>
            <!-- Anzeige des Fragetextes -->
            <td>{{ $fragen->find($antwort->frage_id)->frage_text }}</td>
            <!-- Anzeigen der Benutzerantworten, z. B. "a_mitarbeiter" und "a_vorgesetzter" -->
            <td>{{ $antwort->a_mitarbeiter }}</td>
            <td>{{ $antwort->a_vorgesetzter }}</td>
            <!-- Anzeigen des Textes und des Datums der Maßnahme, die mit dem Ergebnis
            verknüpft ist -->
            <td>{{ $massnahme->massnahme_text }} durchführen bis {{ $massnahme-
            >massnahme_datum }}</td>
        </tr>
    @endforeach
</tbody>
```

## 5.2 Implementierung der Benutzeroberfläche

Die Benutzeroberfläche wurde mit HTML in Kombination mit CSS erstellt. Bootstrap wurde hinzugefügt, um die Erstellung verschiedener Optiken zu erleichtern. Heutzutage ist es wichtig, den Prinzipien des responsiven Designs zu folgen, daher wurde CSS-Flexbox verwendet.

Das Dropdown-Akkordeon wird benutzt, um den nicht genutzten Teil der Umfrage auszublenzen. Die Klasse `.collapse` verweist auf das zusammenklappbare Element, dies ist der Inhalt, der mit einem Klick auf eine Schaltfläche ein- oder ausgeblendet wird. Um den ausklappbaren Inhalt zu steuern (anzeigen/ausblenden), wurde das Attribut `data-bs-toggle="collapse"` zum `<button>`-Element hinzugefügt. Dann wurde das Attribut `data-bs-target="#collapseOne"` hinzugefügt, um die Schaltfläche mit dem eingeklappten Inhalt zu verknüpfen. Siehe **Abbildung 4**

```
51 <div class="container">
52   <div class="accordion" id="accordionExample">
53     <div class="accordion-item">
54       <h2 class="accordion-header" id="headingOne">
55         <button class="accordion-button erste" type="button" data-bs-toggle="collapse"
56           data-bs-target="#collapseOne" aria-expanded="true" aria-controls="collapseOne">
57           Fach- und Medienkompetenzen
58         </button>
59       </h2>
60       <div id="collapseOne" class="accordion-collapse collapse" aria-labelledby="headingOne"
61         data-bs-parent="#accordionExample">
62         <div class="accordion-body">
```

Abbildung 4: Dropdown-Akkordeon

Wie oben beschrieben, wurden insgesamt drei Seiten erstellt. Alle Seiten wurden im gleichen Unternehmens-Design gestaltet. Farben und Schriften wurden entsprechend den Unternehmensstandards verwendet. Darstellungen der Seiten finden sich in **Anhang 4**.

## 5.3 Geschäftslogik

Damit der Benutzer die Umfrage erfolgreich ausfüllen kann, ist es wichtig, dass der Benutzer für jedes Fragenelement eine Antwort auswählt. Um zu überprüfen, ob tatsächlich eine Option in jeder Radiobutton-Gruppe mit der Klasse `"btn-group"` ausgewählt wurde, wurde es ein Code in JavaScript implementiert, welcher den aktuellen Status der Fragenelementen überprüft.

Die Hauptfunktion `checkSelection()` iteriert über diese Gruppen und verwendet die Funktion `checkIfChecked()`, um festzustellen, ob alle Optionsfelder in einer Gruppe ausgewählt wurden. Ist dies nicht der Fall, wird die Ansicht auf das entsprechende Element verschoben und je nach Auswahl wird eine Meldung durch Aufruf der `displayMessage()`-Funktion angezeigt oder verborgen.

Zusätzlich gibt es die Funktion `checkAllSelected()`, die prüft, ob alle Radiobuttons in allen Gruppen ausgewählt sind und im Erfolgsfall die Daten an den Server sendet, welche die erhaltene Daten in eine Datenbank abspeichert.

Insgesamt bietet dieses Skript eine benutzerfreundliche Möglichkeit, die Auswahl von Radiobuttons zu überwachen und je nach Ergebnis Meldungen anzuzeigen.

Das Skript mit zusätzlichen Kommentaren befindet sich in **Anhang 5**.

## 6 Abnahmephase

### 6.1 Tests

Im Rahmen der Implementierungsphase wurden alle Funktionsweisen bereits getestet. In der Testphase wurden für jede Funktion Unit-Tests durchgeführt, bei denen die übergebenen Parameter und zurückgegebenen Werte geprüft wurden, um die korrekte Rückmeldung sicherzustellen. Aufgetretene Fehler wurden behoben. Nachdem alle Funktionen erfolgreich geprüft worden sind, führte der Auftraggeber auch einen Blackbox-Test für die Anwendung durch. Der Kunde war mit den Ergebnissen zufrieden.

## 7 Einführungsphase

Für das Projekt „Perspektiv Dialog“ wurde ein separates Repository auf GitHub eingerichtet, sowie eine Main- und Test-Branche erstellt, in das der gesamte Quellcode hochgeladen wurde. Um das Projekt automatisch auf der Azure Cloud bereitzustellen, wurde eine integrierte Bereitstellung (CI/CD) im GitHub-Repository konfiguriert. Jedes Mal, wenn Code auf GitHub in der Main-Branche aktualisiert wird, startet der Deployment Prozess auf Azure.

automatisch, wodurch eine kontinuierliche Integration gewährleistet wird. Dadurch wird garantiert, dass Änderungen am Projekt schnell und sicher bereitgestellt werden.

## **8 Fazit**

### **8.1 Soll-/Ist-Vergleich**

Folgende Ziele wurde erreicht:

- Alle in der Analysephase festgelegten Aufgaben sind abgeschlossen.
- Alle User Storys sind implementiert worden.
- Der Auftraggeber hat die Arbeit in vollem Umfang abgenommen.

Auf dieser Grundlage ist das Projekt als erfolgreich umgesetzt zu betrachten. Während der Umsetzung wurde versucht, sich an den Zeitplan zu halten, der in der Analysephase entwickelt wurde, was aber nicht immer gelungen ist. So musste beispielsweise für die Entwicklung des Datenbankkonzepts zwei Stunden mehr aufgewendet werden als geplant, für die Erstellung der Datenbank selbst wurden jedoch zwei Stunden weniger benötigt. Die Tests wurden kontinuierlich während der Entwicklungsphase durchgeführt. Im Großen und Ganzen gelang es jedoch, das Projekt innerhalb der vorgesehenen 80 Stunden zu erstellen. In einigen Phasen der Projektentwicklung gab es Schwierigkeiten, bei deren Bewältigung der technische Betreuer des Projekts geholfen hat.

### **8.2 Lessons Learned**

Während der Planung und Durchführung des Projekts hat die Praktikantin viel gelernt. Sie konnte ihre Kenntnisse in der praktischen Anwendung von User Storys vertiefen, was ihr aufzeigte, wie wichtig die Kommunikation mit den Kunden ist. Das ist ein wichtiger Ansatz in der Entwicklung, um die Bedürfnisse der Benutzer genauer zu verstehen und für die Entwickler in spezifische Aufgaben umzusetzen.

Zweitens konnten die Kenntnisse im Laravel-Framework vertieft werden. Dieses Tool hilft bei der Entwicklung von Web-Anwendungen, indem es Aufgaben wie Routing, Datenbankverwaltung und Vorlagenerstellung erleichtert.

Außerdem hat die Praktikantin die Grundsätze von MVC (Model-View-Controller) kennengelernt, die das Herzstück vieler moderner Web-Anwendungen bilden. Das Verständnis der Abstraktionsebenen in MVC half ihr, einen besseren und leichter wartbaren Code zu schreiben.

Zudem lernte sie, wie sie ihre Zeit besser planen kann. Eine wichtige Fertigkeit in der Entwicklung besteht darin, Projekte zu managen, Fristen einzuhalten und Ziele zu erreichen. Für die Anwendungsentwicklung benötigt man eine präzise Planung und ein gutes Zeitmanagement. In dieser Hinsicht ist sie organisierter und produktiver geworden.

Diese Fähigkeiten und Kenntnisse, die während dieser Phase erworben wurden, werden in Zukunft unentbehrlich sein und zu ihrem beruflichen Wachstum im Bereich Entwicklung und Projektmanagement beitragen.

### **8.3 Ausblick**

Der Auftraggeber des Projekts „Perspektiv Dialog“ plant bereits die weitere Entwicklung der Anwendung. Es ist geplant, die gesammelten Daten für Statistiken zu nutzen, die das Wachstum der beruflichen Fähigkeiten der Mitarbeiter, die Anzahl der Maßnahmen pro Mitarbeiter, ihre Effizienz usw. zu zeigen.

## Anhang

### Anhang 1

#### Gantt-Diagramm

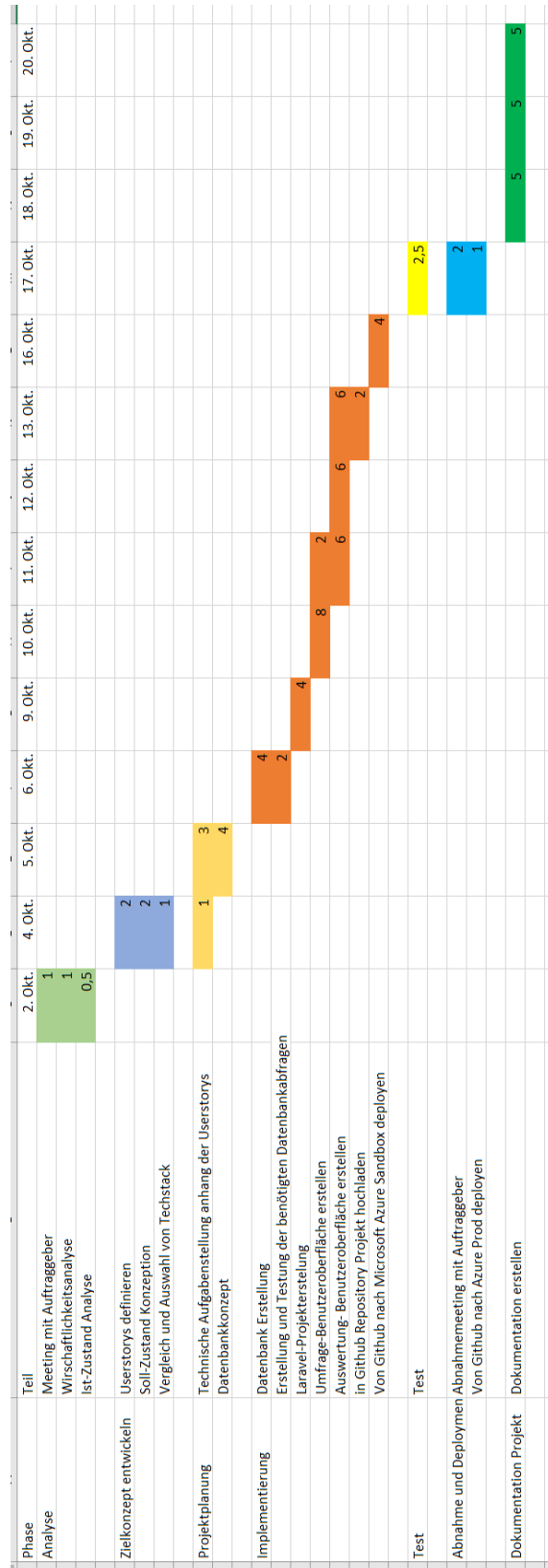
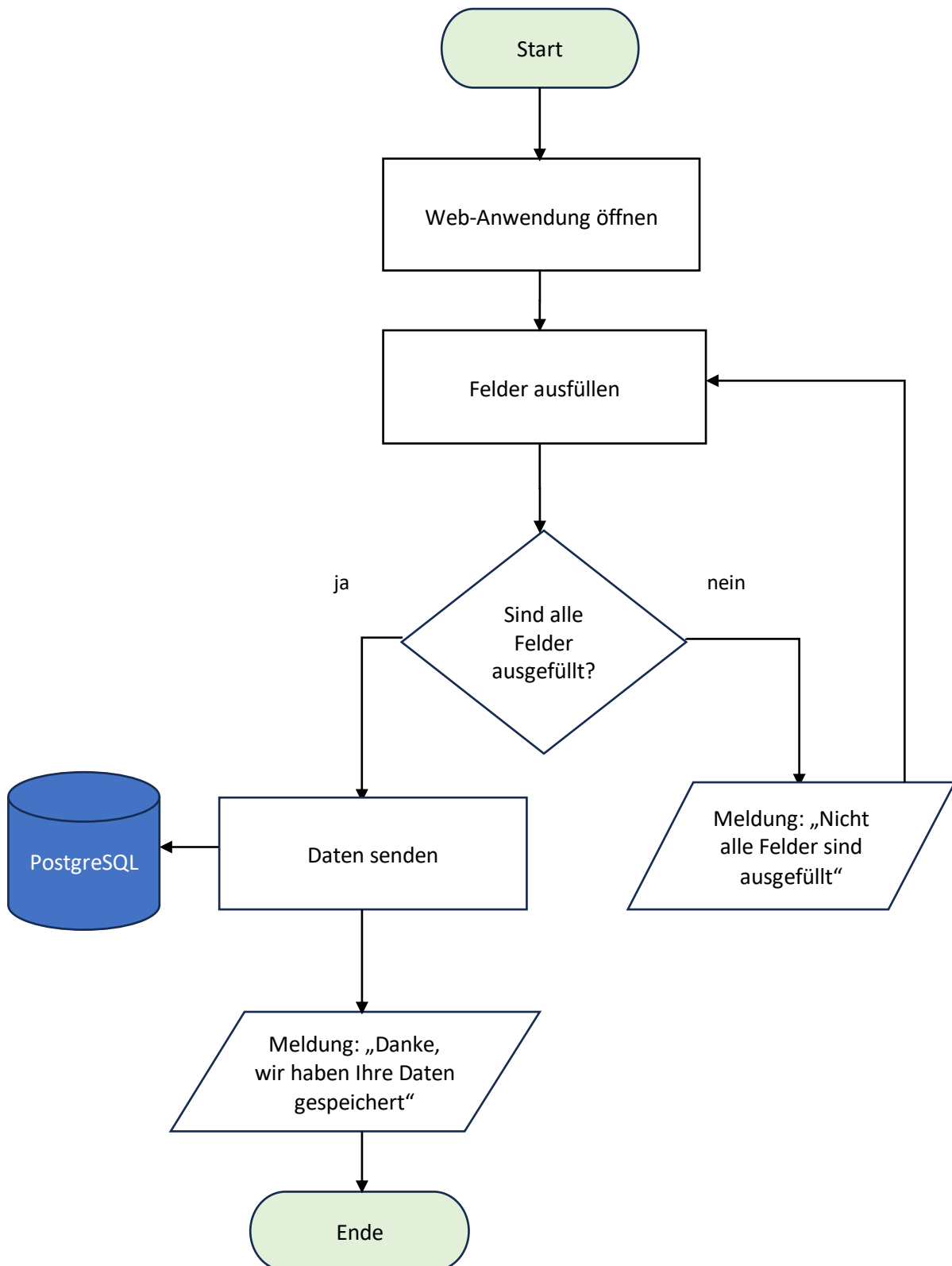


Abbildung 5: Gantt-Diagramm

## Anhang 2

### Flussdiagramm



## Anhang 3

### Work Items






















Work items			<a href="#">Recently updated</a> <a href="#">+ New Work Item</a> <a href="#">Open filtered view in Queries</a> <a href="#">Column Options</a> <a href="#">Import Work Items</a> <a href="#">Recycle Bin</a>			
Filter by keyword			<a href="#">Issue</a> <a href="#">Assigned to</a> <a href="#">States</a> <a href="#">Area</a> <a href="#">Tags</a>			
ID	Title	Assigned To				
127	 Ich als MA (und/oder VG) möchte meine Bewertung anschauen	 Werschinin Elena				
118	 Ich als MA (und/oder VG) möchte nach Abschluss der Bewertung eine Bestätigung erhalten, dass meine Antworten gespeichert wurden.	 Werschinin Elena				
116	 Als Mitarbeiter (MA) möchte ich in der Lage sein, an einer Selbstbewertung teilzunehmen.	 Werschinin Elena				
124	 Als Vorgesetzter möchte ich an Bewertung meiner Mitarbeitern teilzunehmen.	 Werschinin Elena				
121	 Ich als VG möchte Möglichkeit haben während des Gesprächs einige Maßnahmen in spezielle Feldern eintragen können.	 Werschinin Elena				
130	 Ich als MA (und/oder VG) möchte schreibgeschützte Ansicht für meine bereits freigegebenen ausgewählten Auswertungen haben.	 Werschinin Elena				
126	 Ich als VG möchte zu jeder Maßnahme ein Zieldatum eintragen können	 Werschinin Elena				
120	 Ich als VG möchte Ergebnissen während eines Gesprächs abrufen.	 Werschinin Elena				
122	 Ich als MA möchte Maßnahmen freigeben können.	 Werschinin Elena				
123	 Ich als MA (und/oder VG) möchte ich auf die Ergebnisse der letzten Jahre zugreifen können.	 Werschinin Elena				

Abbildung 6: Work Items



## Anhang 4

### Ergebnisse



# Perspektivdialog

Ergebnisse

**Liebe Mitarbeiterinnen und Mitarbeiter,**

Wir freuen uns, dass Sie an unserer Umfrage teilnehmen! Ihre Meinung und Ihre Erfahrungen sind uns äußerst wichtig, da sie uns dabei helfen, unser Unternehmen kontinuierlich zu verbessern und Ihnen ein besseres Arbeitsumfeld zu bieten. Ihre Teilnahme ist entscheidend, und wir schätzen Ihre Zeit und Mühe.

Herzlichen Dank und beste Grüße

Fach- und Medienkompetenzen

Personale Kompetenzen

Lernkompetenz

Führungskompetenz

Senden

PerspektivDialog designed by Elena Werschinin

Abbildung 7: Benutzeroberfläche für die Auswahl

Fach- und Medienkompetenzen

Personale Kompetenzen

Lernkompetenz

Lernbereitschaft

Der/die Mitarbeiter\*in ist bereit, fehlendes fachliches und methodisches Wissen, tätigkeitsspezifische Qualifikationen und Erfahrungen durch entsprechende Weiterbildungsmaßnahmen zu erwerben. Er/sie ist Neuem gegenüber aufgeschlossen und lernt aus Misserfolgen und Erfolgen.

1

2

3

4

Nicht relevant

1 ist die geringste, 4 ist die höchste Ausprägung.

Innovations- und Veränderungskompetenz

Der/die Mitarbeiter\*in kann neue Ideen, innovative Ansätze und Verbesserungen wahrnehmen, äußern, vorantreiben und diese dann in nutzbringende Prozesse, Produkte oder Dienstleistungen umsetzen. Er/sie sieht Veränderungen als Chancen für die Zukunft.

1

2

3

4

Nicht relevant

1 ist die geringste, 4 ist die höchste Ausprägung.

Führungskompetenz

Abbildung 8: Benutzeroberfläche für die Dateneingabe



## Perspektivdialog

Ergebnisse

Fähigkeit	Bewertung Mitarbeiter	Bewertung Vorgesetzter	Maßnahme
<b>Fach- und Medienkompetenzen</b>			
Arbeitsorganisation	1	nicht relevant	<input type="text"/> <div>durchführen bis</div> <input type="text" value="10-30-2023"/>
Kundenorientierung	nicht relevant	2	<input type="text"/> <div>durchführen bis</div> <input type="text"/>
Arbeitseffizienz	2	3	<input type="text"/> <div>durchführen bis</div> <input type="text"/>
Organisationsfähigkeit	2	2	<input type="text"/> <div>durchführen bis</div> <input type="text"/>

Abbildung 9: Benutzeroberfläche für das Zwischenergebnis, wo Maßnahmen eingefügt werden können



## Perspektivdialog

Ergebnisse

2023

Zeigen

Fähigkeit	Bewertung Mitarbeiter	Bewertung Vorgesetzter	Maßnahme
<b>Fach- und Medienkompetenzen</b>			
Arbeitsorganisation	1	nicht relevant	
Kundenorientierung	nicht relevant	2	Maßnahme Kundenorientierung durchführen bis 30.10.2023
Arbeitseffizienz	2	3	
Organisationsfähigkeit	2	2	
Arbeitssorgfalt	2	nicht relevant	
Fachkompetenz	4	2	Maßnahme Fachkompetenz durchführen bis 30.11.2023
Verhandlungskompetenz	2	nicht relevant	Maßnahme Verhandlungskompetenz durchführen bis 01.12.2023
<b>Personale Kompetenzen</b>			
Teamfähigkeit	2	4	
Kommunikationsfähigkeit	2	3	
Zuverlässigkeit	2	2	Maßnahme Zuverlässigkeit durchführen bis 30.10.2023
Kritikfähigkeit	2	nicht relevant	
Konfliktfähigkeit	2	2	
Fähigkeit zur Selbstreflexion	2	2	

Abbildung 10: Benutzeroberfläche für die Ergebnisanzeige

## Anhang 5

```
<script>
<!-- Überprüfen, ob alle Radio-Buttons in den Elementen mit Klasse "btn-group" ausgewählt sind-->
function checkSelection() {
    const collection = document.querySelectorAll(".btn-group");
    let allSelected = true;
    for (let i = 0; i < collection.length; i++) {
        if (!checkIfChecked(collection[i].querySelectorAll(".btn-check"))) {
            scrollToElement(collection[i]);
            allSelected = false;
            break;
        }
    }
    displayMessage(allSelected);
}
<!-- Überprüfen, ob mindestens ein Radio-Button ausgewählt ist-->
function checkIfChecked(radiobuttons) {
    for (let j = 0; j < radiobuttons.length; j++) {
        if (radiobuttons[j].checked) {
            return true;
        }
    }
    return false;
}
<!-- Element ins Sichtfeld scrollen-->
function scrollToElement(element) {
    element.scrollIntoView(false);
}
<!-- Nachricht basierend auf der Auswahl anzeigen oder verstecken-->
function displayMessage(allSelected) {
    const warningMessage = document.getElementById("warning");
    const okMessage = document.getElementById("ok");

    if (!allSelected) {
        warningMessage.style.display = "block";
        okMessage.style.display = "none";
    } else {
        warningMessage.style.display = "none";
        okMessage.style.display = "block";
    }
}
<!-- Überprüfen, ob alle Radio-Buttons in allen Elementen mit Klasse "btn-group" ausgewählt sind-->
function checkAllSelected() {
    const collection = document.querySelectorAll(".btn-group");
    let allSelected = true;
    for (let i = 0; i < collection.length; i++) {
        if (!checkIfChecked(collection[i].querySelectorAll(".btn-check"))) {
            allSelected = false;
            break;
        }
    }
    if (allSelected) {
        document.getElementById("allSelectedMessage").style.display = "block";
    }
}
</script>
```

## Abbildungsverzeichnis:

Abbildung 1: Projektphasen .....	4
Abbildung 2: (Interaktion zwischen Model, View und Controller) .....	9
Abbildung 3: Datenbanktabellen .....	10
Abbildung 4: Dropdown-Akkordeon .....	15
Abbildung 5: Gantt-Diagramm .....	1
Abbildung 6: Work Items .....	3
Abbildung 7: Benutzeroberfläche für die Auswahl .....	4
Abbildung 8: Benutzeroberfläche für die Dateneingabe .....	4
Abbildung 9: Benutzeroberfläche für das Zwischenergebnis, wo Maßnahmen eingefügt werden können .....	5
Abbildung 10: Benutzeroberfläche für die Ergebnisanzeige .....	5

## Tabellenverzeichnis

Tabelle 1: Kostenaufstellung .....	7
------------------------------------	---