



# Quanto-Solutions

November 2023

<b>Name</b>	<b>Matrikelnummer</b>	<b>Githubaccount</b>
Chris Schröder	767384	Chris0297
Michele Pomarico	766583	Michele-92
Fabian Sonek	767903	e2Neo
Baran Kal	769042	Roplck
Daniel Ryssel	768960	danGitRys

# Contents

<b>1</b>	<b>Introduction</b>	<b>6</b>
<b>2</b>	<b>Kurzfassung</b>	<b>6</b>
<b>3</b>	<b>Zielgruppe/Problemstellung</b>	<b>7</b>
3.1	User Stories . . . . .	7
3.1.1	Nutzer (Mitarbeiter, Projekt Manager, Manager) . . . . .	7
3.1.2	Projekt Manager . . . . .	7
3.1.3	Manager . . . . .	7
3.2	Functional Requirements . . . . .	8
3.2.1	Projektanlage für Manager . . . . .	8
3.2.2	Projektmanagemnt-Befugnisse für Manager . . . . .	8
3.2.3	Tägliche Einsatzplanung für Projektmanager . . . . .	8
3.2.4	Zeitaufzeichnung und Reminder für Mitarbeiter . . . . .	8
3.2.5	Berechtigungsabhängige Funktionalitäten . . . . .	8
3.3	Nonfunctional Requirements . . . . .	8
<b>4</b>	<b>High Level Arichitektur Programm</b>	<b>9</b>
4.1	Struktursicht . . . . .	9
4.2	Verhaltenssicht . . . . .	10
4.3	Verteilungssicht . . . . .	11
<b>5</b>	<b>Architektur/ Technologie</b>	<b>12</b>
5.1	Github . . . . .	12
5.2	Genutzte IDEs . . . . .	12
5.3	Postman . . . . .	12
5.4	Figma . . . . .	12
5.5	Microsoft SQL Server Managment Studio . . . . .	13
5.6	Frontend: Vue.js . . . . .	13
5.7	Datenbank: Microsoft SQL Server . . . . .	13
5.8	Authentifizierung SAP . . . . .	13
5.9	Projektmanagment Jira . . . . .	13
<b>6</b>	<b>Aufwandsschätzung</b>	<b>15</b>
<b>7</b>	<b>Mockup</b>	<b>17</b>
7.1	LoginScreen . . . . .	17
7.2	Homescreen . . . . .	17
7.3	New Project . . . . .	18
7.4	Manage Project . . . . .	19
7.5	Sidebar . . . . .	20
7.6	Time Registration . . . . .	21
7.7	Working Times . . . . .	21
<b>8</b>	<b>Datenbank</b>	<b>23</b>
8.1	Tabellen . . . . .	24
8.1.1	Employee . . . . .	24
8.1.2	Team . . . . .	25
8.1.3	Project . . . . .	25
8.1.4	Position . . . . .	25
8.1.5	Assignment . . . . .	26
8.1.6	Booking . . . . .	26
8.1.7	Forecast . . . . .	26

<b>9 Schnittstellen Definition</b>	<b>28</b>
9.1 Datenbank Backend . . . . .	28
9.2 Backend Frontend . . . . .	28
<b>10 Protokoll</b>	<b>29</b>
10.1 1. Woche, Zeitraum 02.10.2023-08.10.2023 . . . . .	29
10.2 2. Woche Zeitraum 09.10.2023 -22.10.2023 . . . . .	29
10.3 3.Woche Zeitraum 16.10.2023 - 22.1.2023 . . . . .	29
10.4 4. Woche Zeitraum: 23.10.203 -29.10.2023 . . . . .	29
10.5 5. Woche Zeitraum: 30.10.2023 - 05.11.2023 . . . . .	30
10.6 6. Woche Zeitraum 06.11.2023-12.11.2023 . . . . .	31
10.7 7. Woche Zeitraum 13.11.2023-19.11.2023 . . . . .	31

## Abbildungsverzeichnis

1	Komponentendiagramm . . . . .	9
2	Ablaufdiagramm . . . . .	10
3	Verteilungsdiagramm . . . . .	11
4	Screenshot Github . . . . .	12
5	Screenshot VS-Code . . . . .	13
6	Screenshot Postman . . . . .	14
7	Screenshot Micorosoft SQL Server Managment Studio . . . . .	15
8	Login-Ansicht . . . . .	17
9	Home-Ansicht . . . . .	18
10	New Project Ansicht . . . . .	19
11	Position-Einfügen Ansicht . . . . .	19
12	Manage-Project-Ansicht . . . . .	20
13	Sidebar-Ansicht . . . . .	20
14	Time Registration Ansicht . . . . .	21
15	Working Times Ansicht . . . . .	22
16	Logische Sicht Datenbank . . . . .	23
17	Physisches Layout Datenbank . . . . .	24
18	Veranschaulichung ODBC Schnittstelle . . . . .	28
19	Veranschaulichung REST-API . . . . .	28

# **1 Introduction**

## **2 Kurzfassung**

Dieses Dokument behandelt die Dokumentation der Projektmanagement Software, welche im Rahmen des Moduls "Projekt Softwaretechnik und Medieninformatik" im Studiengang Softwaretechnik und Medieninformatik an der Hochschule Esslingen für die Firma Quanto-Solutions entwickelt wird.

### 3 Zielgruppe/Problemstellung

Zum aktuellen Zeitpunkt werden von der Firma Quanto-Solutions für die Projektverwaltung Excel-Tabellen verwendet, um die Mitarbeiter einem Projekt zuzuordnen und die Arbeitszeit zu erfassen. Dabei gibt es das Problem des Verwaltens der Zugriffsrechte auf die Excel Dateien, da aus Datenschutzgründen nicht jeder Mitarbeiter alle Informationen einsehen darf, was darin resultiert, dass nur die Managementebene in der Firma diese nutzen darf. Dadurch gestaltet sich das Eintragen der Arbeitsstunden als sehr Zeit aufwändig. Um diese Probleme zu lösen, wurden wir beauftragt, eine Projektmanagement-Software zu entwickeln, die für unsere Zielgruppe, alle Mitarbeiter von Quanto-Solutions, zugänglich ist und eine Möglichkeit bietet, IST-Aufwände schnell und komfortabel buchen zu können.

#### 3.1 User Stories

##### 3.1.1 Nutzer (Mitarbeiter, Projekt Manager, Manager)

- Als Nutzer möchte ich mich einloggen können, um Zugriff auf das Projektmanagement-Tool zu erhalten. (Login)
- Als Nutzer möchte ich meine Arbeitszeiten erfassen können, um diese auf ein Projekt zu buchen. (Time Registration)
- Als Nutzer möchte ich meine eingeplanten Arbeitszeiten sehen können, um einen Überblick über meine erfassten und geplanten Arbeitszeiten zu bekommen. (Working Times)
- Als Nutzer möchte ich meine Arbeitszeiten korrigieren können, um eine korrekte Erstellung der Rechnung zu gewährleisten. (Time Correction)
- Als Nutzer möchte ich mit Hilfe der Sidebar schnell die Ansichten wechseln können, um effektiver zu arbeiten. (Sidebar)
- Als Nutzer möchte ich in meinem Home Screen alle meine Projekte aufgelistet bekommen, um eine Übersicht des aktuellen Standes der Projekte zu erlangen. (Home)

##### 3.1.2 Projekt Manager

- Als Projekt Manager möchte ich Projekte verwalten können, um die Mitarbeiter auf Tagesebene einem Projekt in einer bestimmten Position zuzuordnen. (Manage Project)
- Als Projekt Manager möchte ich die Arbeitszeiten der Mitarbeiter sehen können und Sie manuell nachtragen, um die richtige Abrechnung der Positionen gewährleisten zu können. (Working Times+)

##### 3.1.3 Manager

- Als Manager möchte ich ein neues Projekt anlegen können mit folgenden Informationen: Kunde, Projektname, Zeitraum, Projektmanager, hinzufügen von Mitarbeitern, Positionen und Tagessätzen, um einen besseren Projektüberblick und damit eine strukturierte Planungsmöglichkeit zu erhalten. (New Project)
- Als Manager möchte ich Projekte verwalten können, um die Mitarbeiter auf Tagesebene einem Projekt in einer bestimmten Position zuzuordnen. (Manage Project)
- Als Manager möchte ich die Arbeitszeiten der Mitarbeiter sehen können und Sie manuell nachtragen, um die richtige Abrechnung der Positionen gewährleisten zu können. (Working Times+)

## **3.2 Functional Requirements**

### **3.2.1 Projektanlage für Manager**

Das System muss es Managern ermöglichen, neue Projekte anzulegen, wobei sie wichtige Informationen wie Kundenname, Projektbezeichnung und den Zeitraum festlegen können. Dies erleichtert die klare Definition und Überwachung der Projekte von ihrem Beginn an.

### **3.2.2 Projektmanagement-Befugnisse für Manager**

Manager sollten in der Lage sein, für jedes Projekt einen verantwortlichen Projektmanager (PM) zu bestimmen und Mitarbeiter für das Projekt zuzuordnen sowie den Mitarbeitern Positionen zuzuordnen zu können. Diese Funktion gewährleistet, dass die Zuständigkeiten klar verteilt sind und die Teams gut organisiert sind.

### **3.2.3 Tägliche Einsatzplanung für Projektmanager**

Projektmanager müssen auf Tagesebene eine detaillierte Einsatzplanung für die Mitarbeiter erstellen können. Dies ermöglicht eine präzise und effektive Zeiterfassung, um sicherzustellen, dass Projekte termingerecht abgeschlossen werden.

### **3.2.4 Zeitaufzeichnung und Reminder für Mitarbeiter**

Die Mitarbeiter sollten in der Lage sein, ihre "IST-Aufwände" zu erfassen, um den Arbeitsfortschritt und die Aufgabenverwaltung zu gewährleisten. Darüber hinaus sollte das System automatische Erinnerungen bereitstellen, um sicherzustellen, dass die Zeitaufzeichnung zeitnah und korrekt erfolgt.

### **3.2.5 Berechtigungsabhängige Funktionalitäten**

Die Funktionalitäten des Systems sollten abhängig von der Benutzerrolle variieren. Das bedeutet, dass Manager, Projektmanager und Mitarbeiter jeweils nur auf die Funktionen und Informationen zugreifen können, die für ihre jeweiligen Aufgaben relevant sind. Dies gewährleistet die Sicherheit von Daten und erhöht die Effizienz, da Benutzer nicht mit irrelevanten Informationen überlastet werden.

## **3.3 Nonfunctional Requirements**

Als Non-Functional Requirements in diesem Projekt ist wichtig, dass die Anwendung als Web-Anwendung läuft.



## 4 High Level Arichtektur Programm

### 4.1 Struktursicht

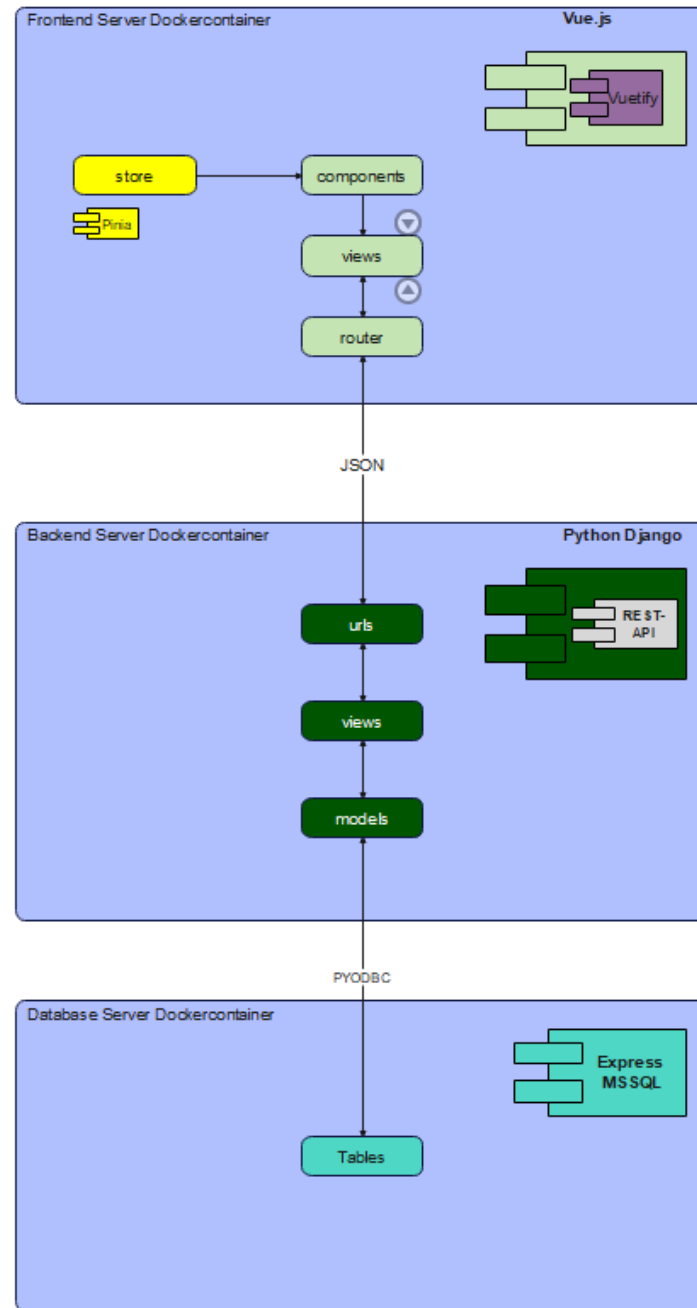


Figure 1: Komponentendiagramm

## 4.2 Verhaltenssicht

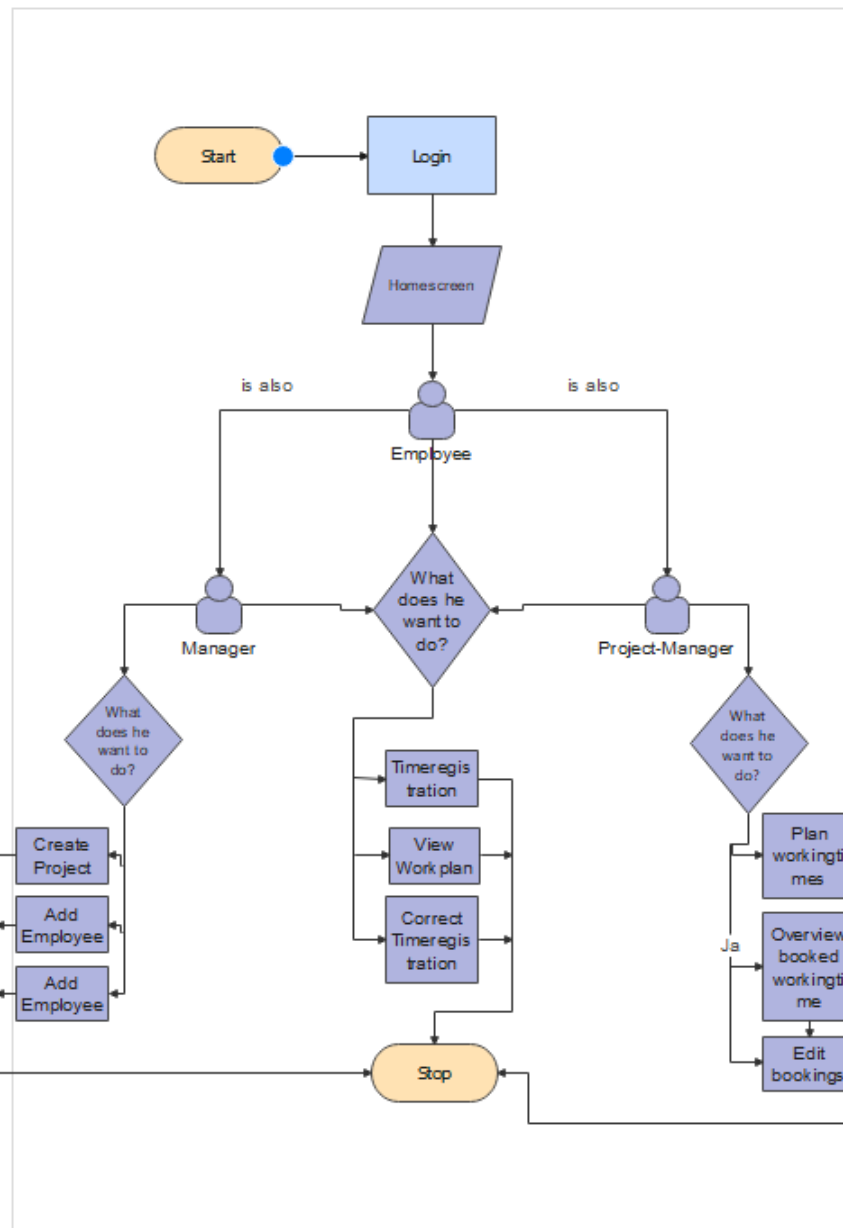


Figure 2: Ablaufdiagramm

### 4.3 Verteilungssicht

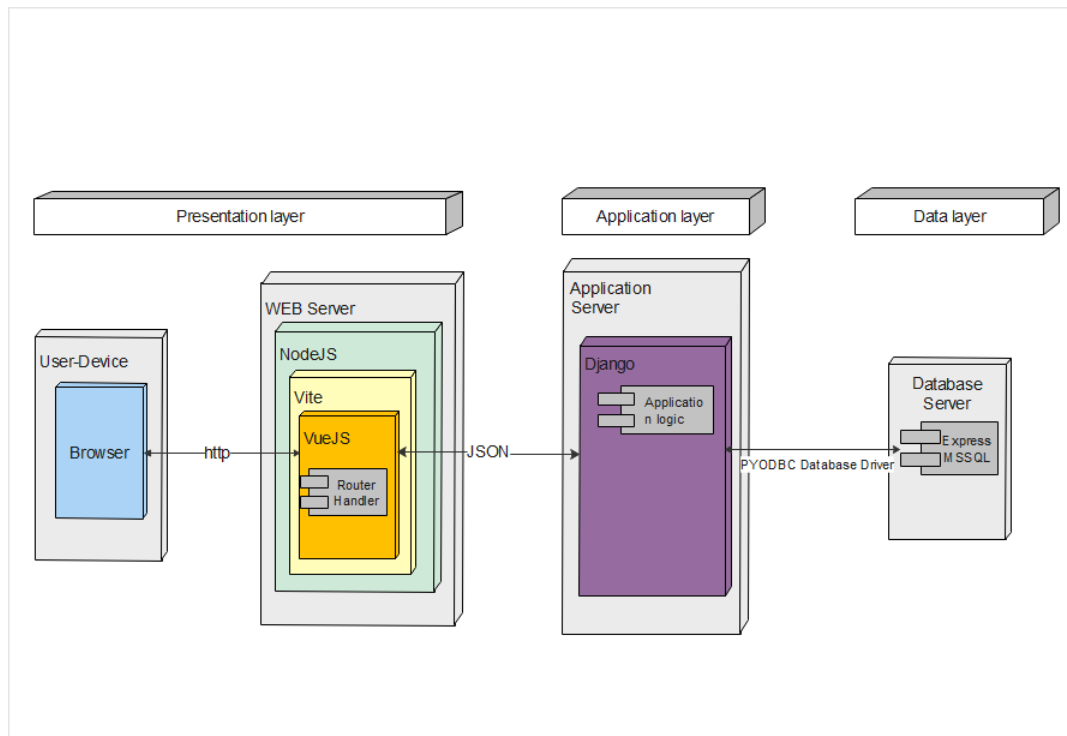


Figure 3: Verteilungsdiagramm

## 5 Architektur/ Technologie

### 5.1 Github

Zur Versionsverwaltung des Codes des Projektes wird Github verwendet. Darüber hinaus bietet GitHub Werkzeuge zur Fehlerverfolgung an, die es uns ermöglichen, Probleme und Verbesserungen in unserem Projekt zu dokumentieren und nachzuverfolgen. Die Verwendung von GitHub erleichtert die Zusammenarbeit und ermöglicht eine transparente und gut organisierte Entwicklungsumgebung für das Projekt.

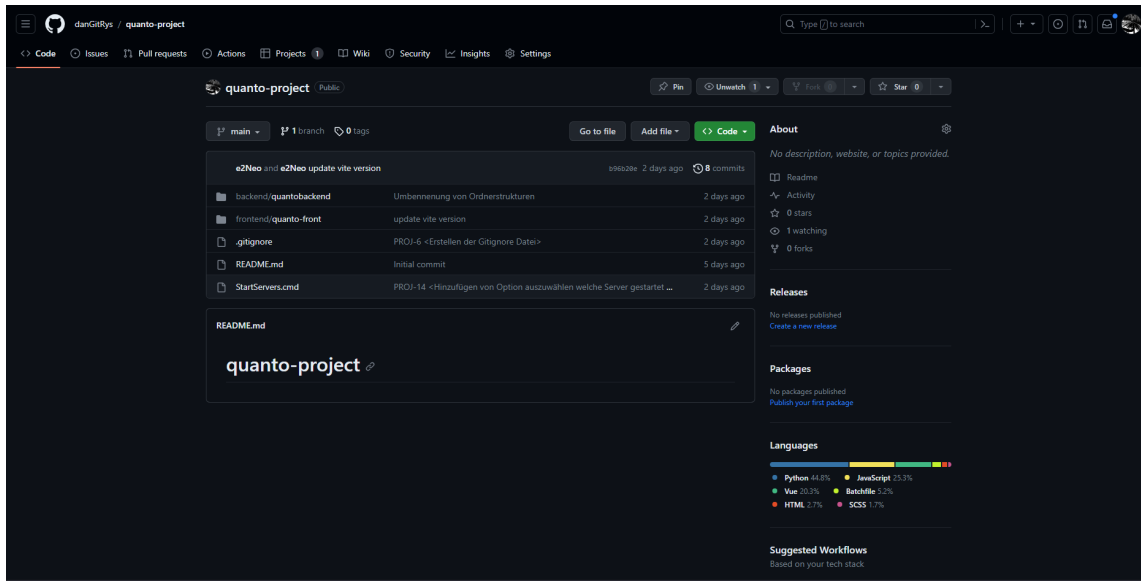


Figure 4: Screenshot Github

### 5.2 Genutzte IDEs

Als IDE wird VS-Code von Microsoft verwendet. Aufgrund der Konfigurierbarkeit von VS-Code wird es für die Entwicklung des Front- und Backend verwendet.

### 5.3 Postman

Zum Testen der Backend API wird Postman genutzt. Postman ist sowohl als lokale Anwendung nutzbar, als auch als Webanwendung oder als Plug-in VS-Code. Postman ermöglicht es dazu noch das man seine Collections in einem Team zusammen bearbeiten und teilen kann

### 5.4 Figma

Zum Erstellen unserer Mockups wird die Software Figma genutzt, da diese es ermöglicht, im Team gleichzeitig an Entwürfen zu arbeiten. Als erstes gab es die Entscheidung, welche Farbe für das Mockup genutzt werden sollte. Dabei hatten wir die Wahl zwischen einem orangen oder blauen Mockup, da diese sehr gut zum Stil von Quanto Solutions passen. Nach verschiedenen Tests haben wir uns entschieden, ein blaues Mockup zu erstellen. Die verschiedenen Screens werden im Folgenden gezeigt und näher erläutert. Der Login-Screen (Abbildung ), wo sich der Nutzer mit der Mitarbeiternummer und einem selbst gewählten Passwort einloggen kann, dient dazu, dass keine Person außerhalb der Firma Zugang erhält.

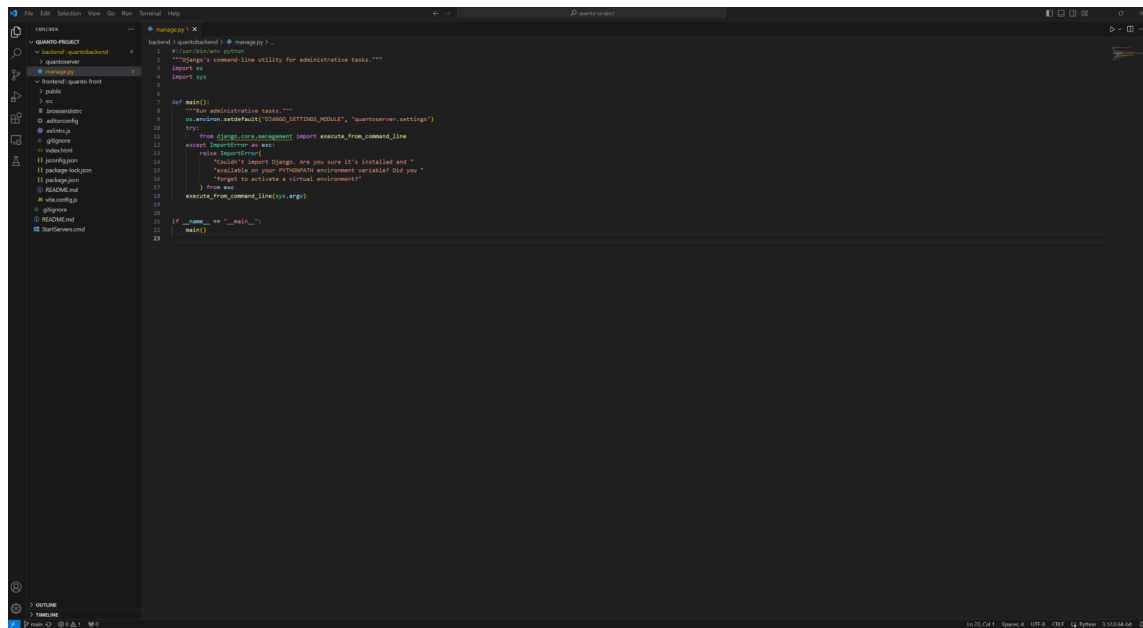


Figure 5: Screenshot VS-Code

## 5.5 Microsoft SQL Server Managment Studio

Zum Verwalten, Interagieren und Entwickeln unserer Datenbank nutzen wir Microsoft SQL Server Management Studio, da dieses Programm von Microsoft selbst stammt und wir sowohl die Datenbank als auch die Verwaltungssoftware für diese aus einer Hand haben.

## 5.6 Frontend: Vue.js

## 5.7 Datenbank: Microsoft SQL Server

## 5.8 Authentifizierung SAP

## 5.9 Projektmanagment Jira

Warum wir Jira als Kommunikationstool nutzen:

Wir setzen Jira als unser Kommunikationstool in unser Softwareprojekt ein, weil es sich als äußerst effizientes und vielseitiges Werkzeug für die interne und externe Kommunikation erwiesen hat. Wir nutzen Jira, um Informationen, Aufgaben, Fehler und Anforderungen in unseren Projekten zentral zu verwalten, was die Klarheit und Effizienz in der Projektarbeit fördert.

1. In Jira können wir in Echtzeit zusammenarbeiten und auf Aktualisierungen sowie Kommentare zugreifen, was eine schnelle und nahtlose Kommunikation unabhängig von den Standorten unserer Teammitglieder ermöglicht.
2. Die Anpassungsfähigkeit von Jira erlaubt es uns, Arbeitsabläufe und Prozesse exakt auf unsere Projektanforderungen zuzuschneiden und benutzerdefinierte Felder hinzuzufügen, um projektbezogene Informationen effizient zu verfolgen.
3. Die Verwendung von Jira steigert die Transparenz und Nachverfolgbarkeit in unseren Projekten, indem alle Teammitglieder einen klaren Überblick über den aktuellen Stand und den Fortschritt der Aufgaben haben.

Vorteile unsere Vorgehensweise mit Jira

1. Jira ermöglicht eine effiziente Planung und Priorisierung von Aufgaben und Anforderungen, indem wir Arbeitsabläufe und Sprints definieren und den Fortschritt leicht im Blick behalten.

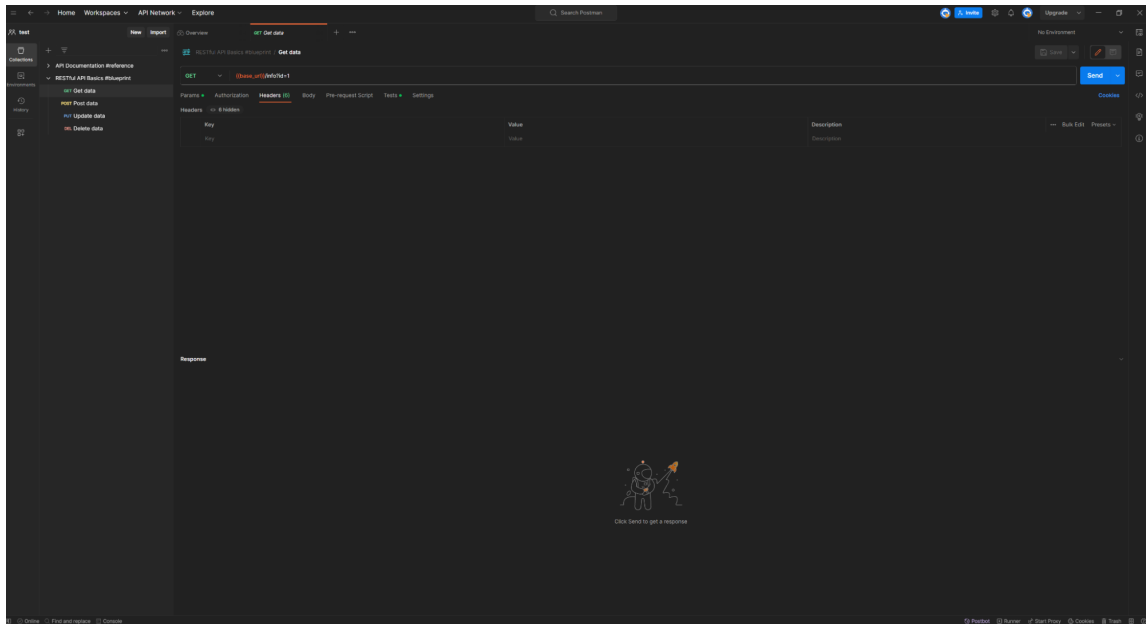


Figure 6: Screenshot Postman

2. Es erleichtert die detaillierte Meldung und Verfolgung von Softwarefehlern, was die Qualität unserer Software maßgeblich steigert.
3. Wir können Aufgaben klar zuweisen und Verantwortlichkeiten definieren, um sicherzustellen, dass jedes Teammitglied weiß, welche Aufgaben es zu erfüllen hat.
4. Jira fördert die Echtzeitkommunikation und den Austausch von Kommentaren zu Aufgaben und Anforderungen, was die Zusammenarbeit und den reibungslosen Informationsfluss im Team unterstützt.
5. Die umfassende Rückverfolgbarkeit von Anforderungen, Aufgaben und Problemen in Jira ist von großer Bedeutung in der Softwareentwicklung, um sicherzustellen, dass alle Anforderungen vollständig und termingerecht erfüllt werden.

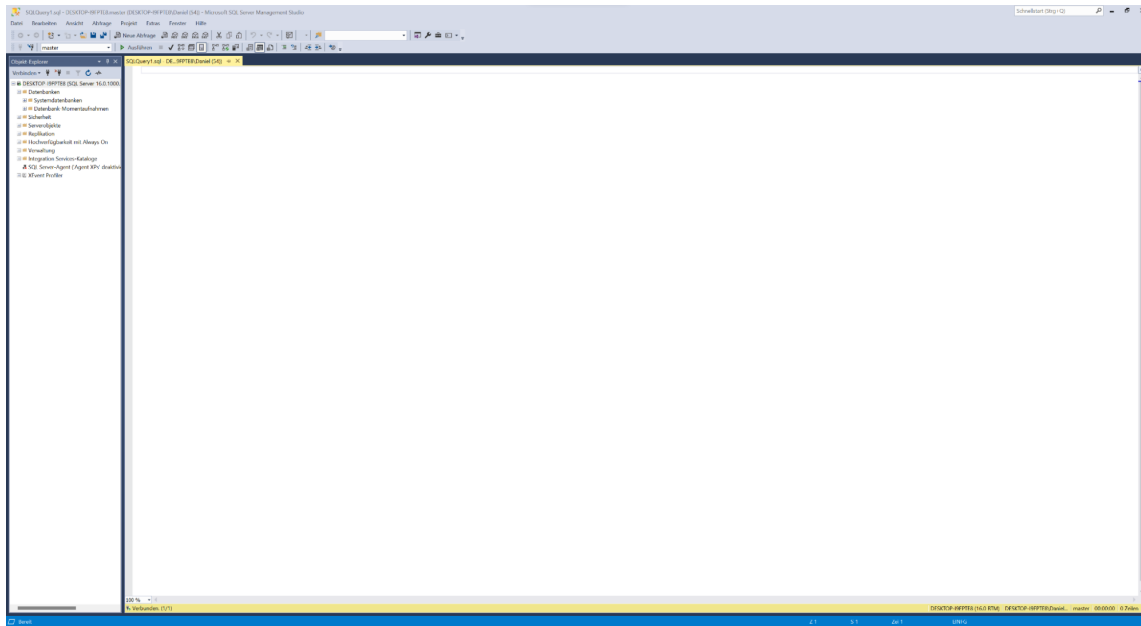


Figure 7: Screenshot Microsoft SQL Server Management Studio

## 6 Aufwandsschätzung

Da dieses Projekt im Rahmen des Moduls “Projekt Softwaretechnik und Medieninformatik” im Studiengang Softwaretechnik und Medieninformatik an der Hochschule Esslingen stattfindet, gibt es einen vorgeschriebenen Aufwand von 10 ECTS pro Person was 300 Stunden entspricht. Da das Team, welches dieses Projekt umsetzt, aus 5 Personen besteht, gibt es insgesamt 1500 Stunden, die in diesem Projekt verteilt werden können. Wir messen den Aufwand in Personentagen (PT), welche jeweils 8 Arbeitsstunden beinhalten. Dadurch kommen wir auf ein Gesamt Pensum von 187,5 PT, welches von der Hochschule vorgeschrieben ist. Bei der Methode der Aufwandsschätzung haben wir uns für die Bottom-Up-Methode entschieden. Hierbei arbeiten wir die Anforderungen des Projekts durch und schätzen für jede den entsprechenden Arbeitsaufwand.

Die Aufwandsschätzung verteilt sich bei diesem Projekt dabei folgendermaßen:

Um die einzelnen Anforderungen abzuarbeiten, brauchen wir erstmal ein Grundgerüst unserer Software, auf dem wir dann aufbauen können.

1. Frontend anlegen: 5PT
2. Backend anlegen: 5PT
3. Datenbank anlegen: 5PT
4. Prototyp eines Login-Systems implementieren und die Anforderung: “Die vorgesehenen Funktionalitäten für Manager PM und Mitarbeiter müssen jeweils nur für die jeweils nur für die jeweiligen Personen zur Verfügung stehen”: 5PT

Im folgenden werden die Anforderungen geschätzt die auf dem Grundgerüst aufbauen

1. SAP Login einrichten: 15PT
2. Manager müssen Projekte anlegen können (Kunde, Projektname, Budget, Zeitraum): 10PT
3. Manager müssen für Projekte Projektmanager (PM) festlegen und Mitarbeiter zuordnen können: 15PT
4. PM müssen auf Tagesebene eine zeitliche Einsatzplanung vornehmen und Mitarbeiter zuordnen können: 30PT

5. Mitarbeiter müssen ihre IST-Aufwände buchen können. Zudem sollen Mitarbeiter dafür einen Reminder erhalten: 30PT
6. Tagessätze Vor Ort und Remote unterscheiden sich! Jeweils eine Rechnungsposition: 5PT
7. Arbeitspensum wird jeweils für Vor-Ort und Remote getrennt festgelegt: 5PT
8. Es soll automatisiert eine Rechnung erzeugt werden und per E-Mail an die Rechnungsstellung geschickt werden: 10PT
9. Puffer zm flexibel einteilen: 27,5 PT



## 7 Mockup

### 7.1 LoginScreen

Über den Login-Screen kann der Nutzer sich mit seinem SAP-BTP Account anmelden. Es gibt 2 Eingabefelder, das eine fragt nach der E-Mail Adresse des Nutzer das andere nach dem Passwort. Wenn der Login erfolgreich war wird der Nutzer auf den Homescreen weitergeleitet. Falls nicht kommt ein Alert mit der Nachricht "Invalid Login" (siehe Abbildung 8).



Figure 8: Login-Ansicht

### 7.2 Homescreen

Nach dem Login wird man zum Homescreen (siehe Abbildung 9) weitergeleitet. Auf dem Homescreen werden alle Projekte angezeigt, in denen man involviert ist, sowie die jeweiligen Rollen in dem Projekt. Wenn man ein Projekt auswählt, in dem man Projektleiter ist, ändert sich die Sidebar und man bekommt mehr Zugriffsrechte und somit tiefere Einblicke in das Projekt. Dabei sieht die Sidebar je nach Rolle unterschiedlich aus. Der Manager, Projektleiter und Mitarbeiter haben jeweils unterschiedliche Zugriffsrechte. Um die Ansichten zu wechseln, kann man die Sidebar oder auch die orangenen Pfeile benutzen. Die Rollen des Managers, Projektleiters und des Mitarbeiters sehen im Homescreen den Projektnamen, den Projektleiter des Projektes sowie das Datum des Projektendes. Der Projektleiter und der Mitarbeiter sehen dazu noch die Rolle, die ihnen in den Projekten zugeteilt wurde. Während der Manager das Budget eines Projektes sieht, welches mit Hilfe der einzelnen Position der Mitarbeiter sowie den Personentagen errechnet wird.

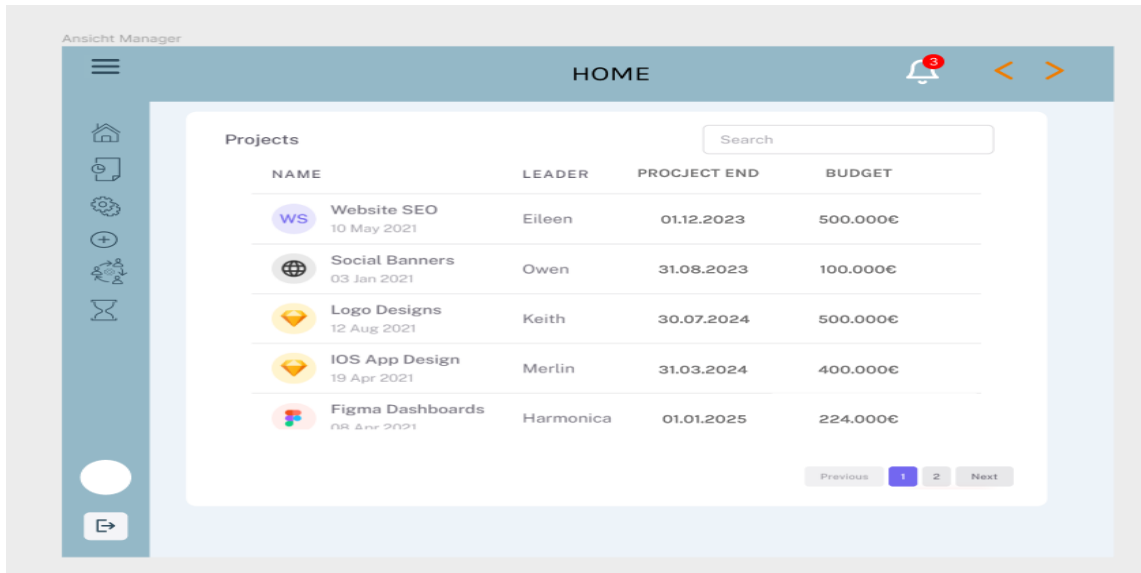


Figure 9: Home-Ansicht

### 7.3 New Project

Über die Eingabemaske New Project kann der Manager ein ganz neues Projekt anlegen. Alle Eingaben erfolgen über ein Eingabefeld. In der Maske kann er alle Daten, die für ein Projekt relevant sind, eingeben, die da wären: Projektnamen, Kunden-Namen, Projektleiter hinzufügen, Projektstart und Projektende. (Abbildung 24). Wenn er auf den “Add Employee” Button drückt, wird er auf die nächste Ansicht weitergeleitet. (Abbildung 10). In dieser Ansicht kann der Manager nun verschiedene Positionen händisch hinzufügen. Da diese von Projekt zu Projekt variieren können. Den jeweiligen Positionen kann er auch einen Tagessatz hinzufügen. Um einem Mitarbeiter die Positionen zuordnen zu können, wählt er zuerst einen Mitarbeiter aus, dann klickt er auf die Positionen, die er dem Mitarbeiter hinzufügen will, diese werden farblich hinterlegt. Dann kann er auf “Add Positions” klicken und die Positionen landen in der unteren Tabelle rechts neben dem Mitarbeiter. Wenn der Manager alle Positionen hinzugefügt hat, kann er auf “Submit” drücken und er landet wieder auf dem ersten Screen. Dort werden ihm nun nochmal alle Mitarbeiter angezeigt, die er zu einem Projekt hinzugefügt hat. Nun kann er auf den “Create” Button klicken und das Projekt wird erstellt. (Abbildung 11)

New Project

NEW PROJECT

Add Project Name:

Add Customer:

Add Project Manager:

Project Start:

Project End:

Add Employee 

Bernd Lustig  
Max Mustermann

Figure 10: New Project Ansicht

Hintergrund

Add Positon: 010 A Senior Developer Remote 010 B Senior Developer On-Site 020 A Junior Developer Remote ...

Add Daily Rates: 800 € 1000 € 500 € ...

Employee:	Positon:	PD:
Max Mustermann	010 A Senior Developer Remote	40
Bernd Lustig	020 A Junior Developer Remote	20

Figure 11: Position-Einfügen Ansicht

## 7.4 Manage Project

In der Ansicht "Manage Project" wird ein ausgewähltes Projekt in einem Kalender angezeigt. Diesen kann man je nach Wahl auf wöchentlich, monatlich oder auch jährlich setzen und auch durch das Dropdown-Menü ein anderes Projekt auswählen. In dem Kalender werden die Mitarbeiter auf Stunden-Ebene angezeigt. Auf der rechten Seite kann man neue Positionen erstellen. Dazu gibt man das Datum ein, wählt einen Mitarbeiter aus, die Position z.B. Senior Developer, die Start- und Endzeit und eine Beschreibung. Nach der Wahl des Mitarbeiters wird im "Free Time Slot" die noch freie Zeit des Mitarbeiters an diesem bestimmten Tag angezeigt. Auf der linken Seite werden die dann neu erstellten Positionen angezeigt. Dabei steht der Name des Mitarbeiters, die Position und die Zeit, die er noch übrig hat für das Projekt, in das er eingeteilt wurde. (Abbildung 11)

</

Figure 12: Manage-Project-Ansicht

## 7.5 Sidebar

Die Sidebar ist je nach Rolle unterschiedlich strukturiert. Der Mitarbeiter hat eine andere Ansicht als der Projektleiter. Jede Rolle hat nichtsdestotrotz die folgenden Buttons: Den "Home" Button, der wie zuvor gezeigt, den Homescreen anzeigt, den "Working-Times" Button, der die gearbeiteten Zeiten in Form eines Kalenders angezeigt, die Time Registration, in der man seine gearbeiteten Zeiten eingepflegt und die Einstellungen. Der Manager hat zudem noch einen New-Project-Button, mit dem neue Projekte erstellt werden können. (Abbildung 24). Der Projektleiter hat einen Manage-Project-Button, auf dieser Ansicht kann er die Mitarbeiter auf Stunden Ebene einem Projekt zuordnen (Abbildung 26). Mehr Details folgen bei den dazugehörigen Screenshots. (Abbildung Sidebar).



Figure 13: Sidebar-Ansicht

## 7.6 Time Registration

Über das Fenster "Time Registration" können die Mitarbeiter Ihre Arbeitszeiten erfassen. Als Erstes geben Sie die Startzeit ein, also um wie viel Uhr haben Sie angefangen zu arbeiten. Dann seine Pausenzeiten und Endzeit. Über ein DropDown Menü hat man noch die Möglichkeit, sich die Projekte anzeigen zu lassen, in denen man involviert ist und auf das man seine Zeit erfassen möchte. Wichtig ist auch noch, dass die richtige Position ausgewählt wird, da bei jeder Position ein anderer Verrechnungssatz hinterlegt ist.

Figure 14: Time Registration Ansicht

## 7.7 Working Times

In der Ansicht "Working Times" sieht der Mitarbeiter seine Arbeitszeiten in einer Übersicht aufgeschlüsselt. Die Zeiten werden in eingeplanten Arbeitsstunden pro Tag angezeigt. Es wird angezeigt, wie lange der Mitarbeiter tatsächlich gearbeitet hat, sowie seine Pausenzeiten. In der letzten Zeile der Tabelle sieht er die Summe der geplanten Arbeitsstunden und tatsächlichen Arbeitsstunden. Dadurch weiß der Mitarbeiter, wie weit er in dem Projekt ist und wie viel Zeit er an diesem Tag dafür verwendet hat. (Abbildung Working Times)

Working Times

<

Figure 15: Working Times Ansicht

## 8 Datenbank

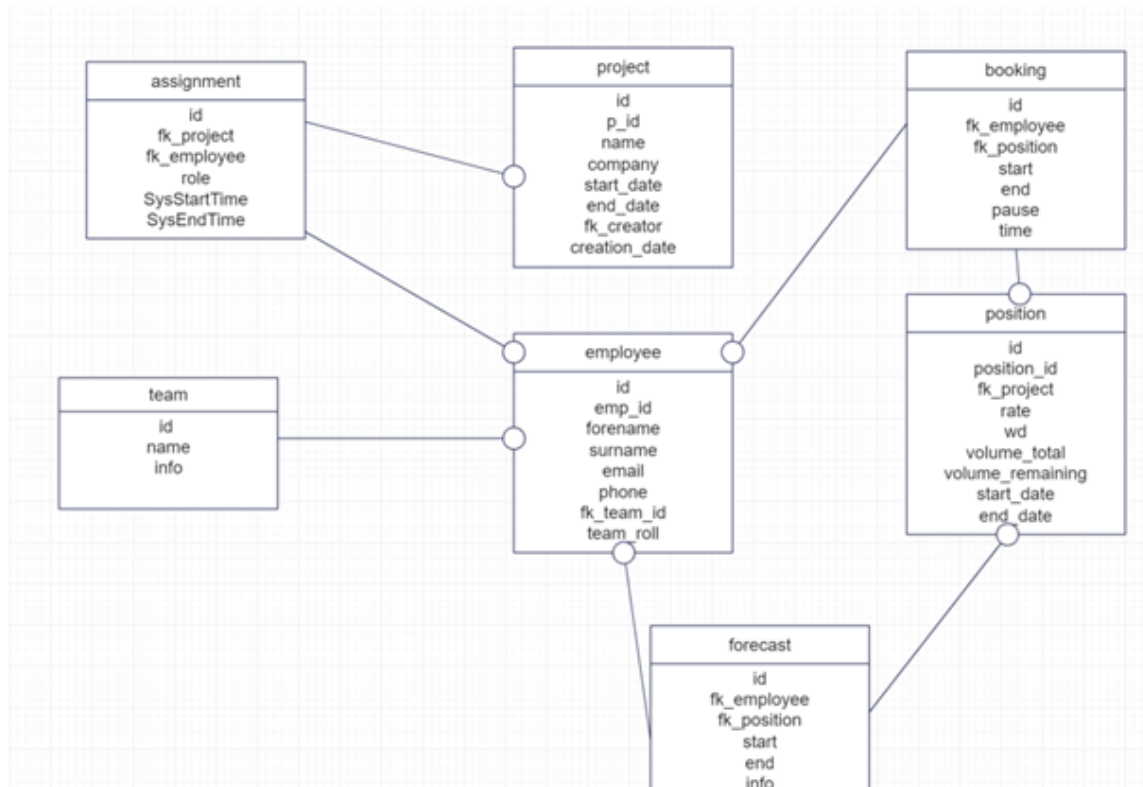


Figure 16: Logische Sicht Datenbank

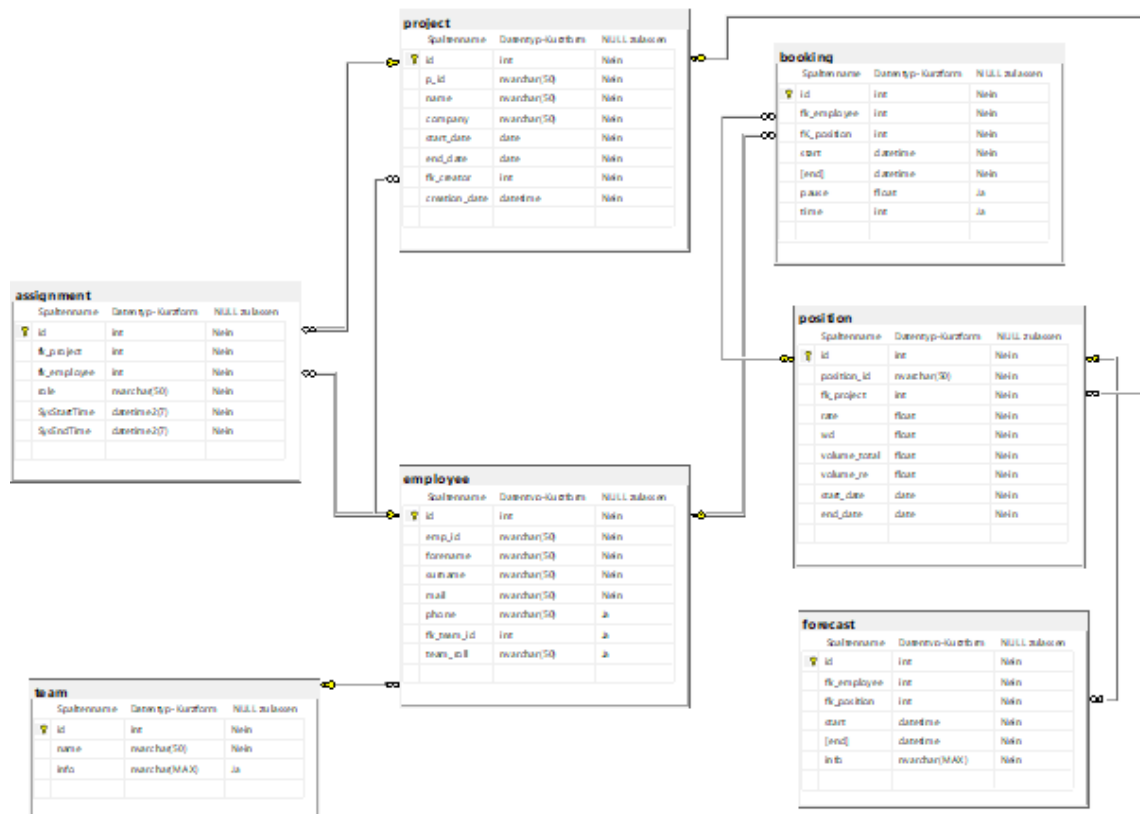


Figure 17: Physisches Layout Datenbank

## 8.1 Tabellen

### 8.1.1 Employee

Listing 1: Create Table Statement für Employee Table

```
CREATE TABLE [dbo].[employee] (
    [id] INT IDENTITY (1, 1) NOT NULL,
    [emp_id] NVARCHAR (50) NOT NULL,
    [forename] NVARCHAR (50) NOT NULL,
    [surname] NVARCHAR (50) NOT NULL,
    [mail] NVARCHAR (50) NOT NULL,
    [phone] NVARCHAR (50) NULL,
    [fk_team_id] INT NULL,
    [team_rol] NVARCHAR (50) NULL,
    CONSTRAINT [PK_employee] PRIMARY KEY CLUSTERED ([id] ASC),
    CONSTRAINT [FK_employee_team] FOREIGN KEY ([fk_team_id]) REFERENCES
    [dbo].[team] ([id])
);
```

In der Tabelle Employee werden die Mitarbeiter der Firma gespeichert die sowohl Projekten zugeordnet werden können, als auch diese administrieren.

**8.1.1.1 Id** Bei der Id handelt es sich um den Primärschlüssel der Tabelle welcher zugleich ein Autinkrement-Schlüssel ist, und dazu dient einen Employee eindeutig zu identifizieren.

**8.1.1.2 emp\_id** Die emp\_id ist eine Id welche einem Mitarbeiter zur Identifizierung zugeordnet wird. Bei dieser kann es sich um eine existierende aus dem Unternehmen handeln oder auch eine



neu implementierte Identifizierungsvariante handeln. Diese sollte dabei nicht mit der id Spalte in dieser Tabelle verwechselt werden.

**8.1.1.3 forename,surname und phone** Bei diesen Spalten handelt es sich bei dem Vornamen und Nachname um erforderliche Informationen und bei der Telefonnummer um eine optionale Informationen.

**8.1.1.4 mail** Der Mail-Adresse des Employees kommt eine besondere Wichtigkeit zu, da diese hinsichtlich des Logins mit SAP-BTP entscheidend ist. Die Mail Adresse sollte hierbei die gleiche sein, die auch in SAP verwendet wird, da die Zuordnung zwischen SAP und Datenbank über diese erfolgt. In Konsequenz davon sollten die Mail Adressen in der Datenbank einzigartig sein, da doppelte Mail Adressen zu Konflikten führen werden.

**8.1.1.5 fk\_team.id** Bei dem fk\_team.id handelt es sich um einen Foreign Key mit Bezug auf den Team Table. Durch die Implementierung des Foreign Keys an dieser Stelle kann ein Employee zu einem Zeitpunkt, nur einem Team zugeordnet sein.

**8.1.1.6 team.roll** Ein Employee hat in einem Projekt eine Rolle. Diese kann sein "Team-Leader" oder "Member".

## 8.1.2 Team

Listing 2: Create Table Statement für Team Table

```
CREATE TABLE [dbo].[team] (  
  [id] INT IDENTITY (1, 1) NOT NULL,  
  [name] NVARCHAR (50) NOT NULL,  
  [info] NVARCHAR (MAX) NULL,  
  CONSTRAINT [PK_team] PRIMARY KEY CLUSTERED ([id] ASC)  
);
```

## 8.1.3 Project

Listing 3: Create Table Statement für Project Table

```
CREATE TABLE [dbo].[project] (  
  [id] INT IDENTITY (1, 1) NOT NULL,  
  [p_id] NVARCHAR (50) NOT NULL,  
  [name] NVARCHAR (50) NOT NULL,  
  [company] NVARCHAR (50) NOT NULL,  
  [start_date] DATE NOT NULL,  
  [end_date] DATE NOT NULL,  
  [fk_creator] INT NOT NULL,  
  [creation_date] DATETIME NOT NULL,  
  CONSTRAINT [PK_project] PRIMARY KEY CLUSTERED ([id] ASC),  
  CONSTRAINT [FK_project_creator] FOREIGN KEY ([fk_creator])  
  REFERENCES [dbo].[employee] ([id])  
);
```

## 8.1.4 Position

Listing 4: Create Table Statement für Position Table

```
CREATE TABLE [dbo].[position] (  
  [id] INT IDENTITY (1, 1) NOT NULL,
```

```

[ position_id]          NVARCHAR (50) NOT NULL,
[ fk_project]           INT           NOT NULL,
[ rate]                 FLOAT (53)    NOT NULL,
[ wd]                   FLOAT (53)    NOT NULL,
[ volume_total]         FLOAT (53)    NOT NULL,
[ volume_remaining]     FLOAT (53)    NOT NULL,
[ start_date]           DATE          NOT NULL,
[ end_date]             DATE          NOT NULL,
CONSTRAINT [PK_position] PRIMARY KEY CLUSTERED ([id] ASC),
CONSTRAINT [FK_position_project] FOREIGN KEY ([fk_project])
REFERENCES [dbo].[project] ([id])
);

```

### 8.1.5 Assignment

Listing 5: Create Table Statement für Assignment Table

```

CREATE TABLE [dbo].[assignment] (
[id]          INT          IDENTITY (1, 1) NOT NULL,
[ fk_project] INT          NOT NULL,
[ fk_employee] INT         NOT NULL,
[ role]       NVARCHAR (50) NOT NULL,
CONSTRAINT [PK_assignment] PRIMARY KEY CLUSTERED ([id] ASC),
CONSTRAINT [FK_assignment_employee] FOREIGN KEY ([fk_employee]) REFERENCES
[dbo].[employee] ([id]),
CONSTRAINT [FK_assignment_project] FOREIGN KEY ([fk_project]) REFERENCES
[dbo].[project] ([id])
);
GO
CREATE UNIQUE NONCLUSTERED INDEX [Index_Assignment_Id]
ON [dbo].[assignment]([id] ASC);

```

### 8.1.6 Booking

Listing 6: Create Table Statement für Booking Table

```

CREATE TABLE [dbo].[booking] (
[id]          INT          IDENTITY (1, 1) NOT NULL,
[ fk_employee] INT         NOT NULL,
[ fk_position] INT         NOT NULL,
[ start]       DATETIME    NOT NULL,
[ end]         DATETIME    NOT NULL,
[ pause]       FLOAT (53)  NULL,
[ time]        INT         NULL,
CONSTRAINT [PK_tracking] PRIMARY KEY CLUSTERED ([id] ASC),
CONSTRAINT [FK_tracking_employee] FOREIGN KEY ([fk_employee]) REFERENCES
[dbo].[employee] ([id]),
CONSTRAINT [FK_tracking_position] FOREIGN KEY ([fk_position]) REFERENCES
[dbo].[position] ([id])
);

```

### 8.1.7 Forecast

Listing 7: Create Table Statement für Forecast Table

```
CREATE TABLE [dbo].[forecast] (  
  [id] INT IDENTITY (1, 1) NOT NULL,  
  [fk_employee] INT NOT NULL,  
  [fk_position] INT NOT NULL,  
  [start] DATETIME NOT NULL,  
  [end] DATETIME NOT NULL,  
  [info] NVARCHAR (MAX) NOT NULL,  
  CONSTRAINT [PK_plan] PRIMARY KEY CLUSTERED ([id] ASC),  
  CONSTRAINT [FK_plan_position] FOREIGN KEY ([fk_position]) REFERENCES  
  [dbo].[position] ([id])  
);
```

Der SQL Code zum erstellen der Datenbank inklusive der Tabellen, ist im Github-Repository des Backends zu finden.

## 9 Schnittstellen Definition

### 9.1 Datenbank Backend

Für die Schnittstelle zwischen der Datenbank und dem Backend haben wir uns für die Microsoft ODBC Schnittstelle entschieden. Diese kann wie folgt beschrieben werden.

”Die Microsoft Open Database Connectivity (ODBC)-Schnittstelle ist eine C-Programmiersprachenschnittstelle, mit der Anwendungen über eine Vielzahl von Datenbankverwaltungssystemen (DBMSs) auf Daten zugreifen können. ODBC ist eine schnittstelle mit niedriger Leistung, die speziell für relationale Datenspeicher entwickelt wurde.”[1].

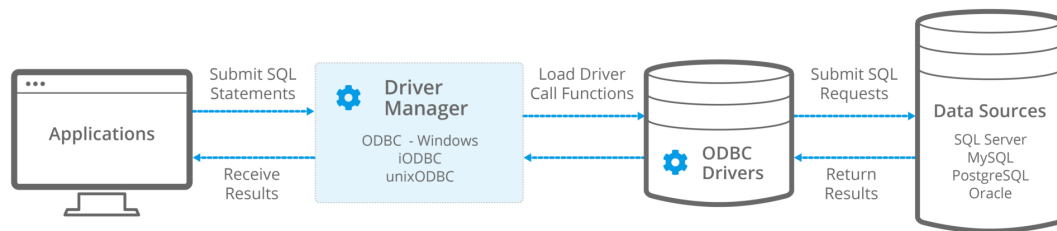


Figure 18: Veranschaulichung ODBC Schnittstelle  
[2]

Die ODBC Schnittstelle bietet dabei für uns mehrere Vorteile. Auch wenn in diesem Projekt als Datenbank Microsoft SQL Server verwendet wird, besteht die Möglichkeit das auch eine andere Datenbank-Technologie verwendet wird, solange diese ebenfalls die ODBC Schnittstelle unterstützt. Dadurch vergrößert sich die potenzielle Zielgruppe unseres Projekts, da für den Fall dass potenzielle Nutzer andere Datenbank-Systeme präferieren und verwenden wollen. Ein weiterer Vorteil der sich ergibt ist die Plattformunabhängigkeit, wodurch das Projekt auf allen gängigen Betriebssystemen laufen kann.

### 9.2 Backend Frontend

Als Schnittstelle zwischen dem Backend und Frontend wird eine REST-API verwendet. Dies ist die Konsequenz aus der Entscheidung für das Backend Framework, da es sich dabei um ein Framework handelt, welches eine REST-API implementiert, und wir daran nichts ändern, da die REST-API viele Vorteile für uns bietet.

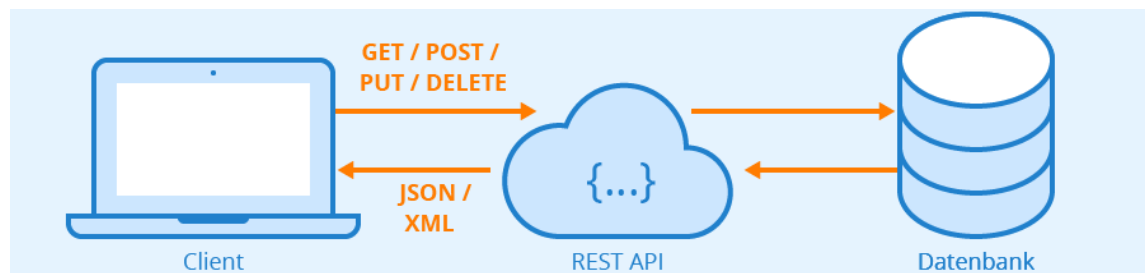


Figure 19: Veranschaulichung REST-API  
[3]

## **10 Protokoll**

### **10.1 1. Woche, Zeitraum 02.10.2023-08.10.2023**

Im Meeting am 06.10.2023 wurden folgenden Sachen hinsichtlich des Projektes festgelegt:

Quanto Solutions gab uns keine Vorgaben in Hinsicht auf Technologien und Programmiersprachen, mit der Einschränkung, dass es sich dabei um eine Webanwendung handeln muss, was sich jedoch auch aus den Vorgaben des SWTM-Moduls ergab. Somit wurde uns freie Hand gelassen im Aspekt auf die technischen Entscheidungen. Zudem wurde noch einmal genau auf die Anforderungen des Projektmanagement Tools eingegangen. Diese wurden unterteilt in Muss-, Soll- und Kann-Anforderungen. Des Weiteren wurde besprochen, wie ein Projekt aufgebaut ist und welche verschiedenen Rollen in diesem involviert sind. Dadurch wurde noch einmal deutlich gemacht, dass jede Rolle in einem Projekt verschiedene Zugriffsrechte hat. Bis zum nächsten Meeting sollten wir ein Mockup erstellen.

### **10.2 2. Woche Zeitraum 09.10.2023 -22.10.2023**

Im Meeting am 13.10 wurden folgende Sachen hinsichtlich des Projektes festgelegt.

Zur Zeiterfassung der Mitarbeiterzeiten wird die Start- und Endzeit des Arbeitstages inklusive Pausenzeiten, manuell von Hand eingetragen und nicht, wie im ersten Mockup, per Start- und Stop-Button gestoppt. Für das Eintragen der Arbeitszeiten wurde vereinbart, dass Mitarbeiter ihre Arbeitszeiten auf Stundenbasis eintragen können und nicht, wie bisher gedacht, nur ganze Tage buchen können. In Hinsicht auf das nachträgliche Ändern von Arbeitszeiten wurde festgelegt, dass die Arbeitszeiten bis zum Ende des Monats geändert werden können. Des Weiteren wurde festgehalten, dass Manager in Projekten alle Berechtigungen besitzen. Die Projektsprache hinsichtlich der User Interfaces wurde auf Englisch festgelegt, damit möglichst viele Mitarbeiter diese verwenden können, hinsichtlich der möglichen Akquisition von internationalen Firmen. Ebenfalls wurde festgelegt, dass die App-Sidebar auf jeder Seite der App sichtbar sein soll, um eine möglichst effektive und schnelle Navigation in der App zu gewährleisten. Hinsichtlich der Planung von Projekten in der Anwendung wurde festgehalten, dass Manager die Positionen erstellen. Der Projektleiter kann dann Mitarbeiter für diese Positionen zuordnen und auf Tagesbasis das Projekt planen.

### **10.3 3.Woche Zeitraum 16.10.2023 - 22.1.2023**

Im Meeting am 20.10 wurden folgende Sachen hinsichtlich des Projektes festgelegt.

Es wurde vom Kunden das blaue Farbschema für das Produkt gewünscht. Zudem wurde sich auch gewünscht, wenn möglich, die Positionen in "Add Employee" per Drag and Drop zu veranschaulichen. Doch da haben wir die endgültige Entscheidung, weswegen wir und auch für das Drag and Drop Feature entschieden haben. Bei "Manage Project" sollen alle Mitarbeiter angezeigt werden und diese eine begrenzung von 8 Stunden pro Tag haben. Dabei soll man diese Mitarbeiter direkt für mehrere Tage, Wochen und Monate einplanen können. Das soll man wiederholen, zum Beispiel per Checkbox und im Kalender auch rauslöschen können. Eine Tagessicht wurde ebenfalls vereinbart. In "Time Registration" haben wir beschlossen das Mitarbeiter am Ende des Monats ihre Zeiten bestätigen können und diese dann der Projektleiter sehen kann. Die Mitarbeiter können somit frei im Monat ihre Stunden anpassen. Nach dem bestätigen der Zeiten kann der Mitarbeiter eine Zeiten nicht mehr ändern, weswegen bei vergessen der Eintragung, die Zeiten händisch durch den Projektleiter erfolgen müssen.

### **10.4 4. Woche Zeitraum: 23.10.2023 -29.10.2023**

Im Meeting am 27.10 wurden folgende Sachen hinsichtlich des Projektes festgelegt.

In Bezug auf unser Kanban-Board wurde festgelegt die User Stories in kleinere Tasks zu zerlegen. Es wurde zudem festgelegt das wir 2 GitHub Repositories haben sollen, wobei die erste für das Frontend ist und die zweite für das Backend. Zur Code Dokumentation wurde gesagt das wir ein einheitlichen Kommentierstil haben sollen. In Bezug auf das Datenbankmodell sollten ein paar Änderungen vorgenommen werden, wie die Team Employee Tabelle zu entfernen und eine

TeamId anzulegen. Diese sollten angepasst werden. Es kam die Frage auf für den SQL Server einen Docker zu verwenden, welches bis zum nächsten Meeting von der Kundenseite beantwortet werden soll. Der Kalender in "Manage Project" ist zu viel des Guten und könnte auch durch eine einfache Tabelle dargestellt werden. Laut dem Kunden ist es unsere Entscheidung zu sagen ob wir ein Kalender oder eine Tabelle erstellen. Dabei spielt beim einplanen die Uhrzeit keine Rolle, sondern nur die 8 Stunden. Es wurde vereinbart bis nächste Woche Dienstag ein MockUp in Form einer Tabelle zu erstellen.

## **10.5 5. Woche Zeitraum: 30.10.2023 - 05.11.2023**

Im Meeting am 31.10 wurden folgende Sachen hinsichtlich des Projektes festgelegt. Zwischen dem Kalender und der Tabelle haben wir uns entschieden eine Tabelle zu machen und diese mit Vuetify Komponenten und Usability Features wie Drag and Drop zu erstellen. Dabei soll die Benutzerfreundlichkeit auf Stundenbasis auch verbessert werden. Hinzu kommt, dass wir bis Freitag, den 03.11, den Mockup abschließen sollen. Es wurde endgültig entschieden, beim Einplanen der Mitarbeiter per Tabelle, diese für bis zu 8 Stunden einplanen zu können. Als Zusatzfeature mit geringerer Priorität können wir eine Wiederholung der Planung erstellen, wo die Planung dann z.B. für jeden Montag der nächsten 2 Monate gilt. Beim Datenmodell wurden einige Tabellen geändert und es wurde festgelegt das Datenmodell bis zum nächsten Meeting als Bild bereitzustellen, damit der Kunde sich dazu Gedanken machen kann. Des Weiteren wurde festgehalten, Docker Container für jeweils das Frontend, Backend sowie für die Datenbank zu erstellen. Es wurde festgelegt einen zentralen Ort für die Protokolle zu haben und diese nach dem jedem Meeting mit dem Kunden zu teilen.

Im Meeting am 3.11 wurden folgenden Sachen hinsichtlich des Projektes festgelegt. Datenbank: Bei der Nutzung und Erstellung der Datenbank sollten keine Kosten entstehen, sie sollte lizenzfrei sein. Das Projekt ist ein Open-Source-Projekt. Dokumentation Datenbank: Die SQL Statements sollen ins Git hinzugefügt werden (Backend) Repository. In der Doku soll das Datenbankmodell hinzugefügt und erklärt werden. Datenbankmodell: Das Datenbankmodell muss noch einmal überarbeitet werden. In der Projekttable soll folgendes hinzugefügt werden: Projekt Ersteller, Datum wann das Projekt erstellt wurde, und wer der Projektmanager ist. In der Tabelle Employees soll das Attribut: team roll auf title umbenannt werden. Außerdem soll mit passenden Präfixen gearbeitet werden, zum Beispiel fk bei Foreign Keys. Um die Bezeichnung eindeutig zu machen, müssen noch alle Attribute, die auf Deutsch sind, auf Englisch umbenannt werden.

Mockup: Zusammen sind wir noch einmal das Mockup durchgegangen und haben letzte Anpassungen vorgenommen: Es wurde beschlossen, dass nichts dagegen spricht, dass der Projektleiter auch das Budget sehen kann für seine Projekte. Dem Manager soll bei der Projekterstellung auch das kalkulierte Budget angezeigt werden, bevor er das Projekt anlegt. Die Manage Project Ansicht: Soll eine Tabelle sein in der alle Mitarbeiter aufgelistet sind die diesem Projekt zugeteilt wurden: Folgendes soll in der Tabelle angezeigt werden: Eine Spalte mit bereits eingeplante Stunden über alle Projekte des Mitarbeiters dadurch kann man sehen wie viele Stunden man der Mitarbeiter an dem Tag noch einplanen könnte. In der nächsten Spalte kann man dann die Anzahl der Stunden eintragen, wie viele Stunden man den Mitarbeiter an diesem Tag einplanen möchte. Sowie eine Spalte Positionen, in dem man festlegt, auf welcher Position der Mitarbeiter an diesem Tag eingeplant werden soll. Diese Liste soll immer einen kompletten Monat umfassen und man kann per DropDown Menü bequem zwischen den Monaten hin und her wechseln. Die Working Times Ansicht: Soll auch nun auch in einer Tabelle dargestellt werden wie die Manage Project Ansicht, für ein besseres Look and Feel und eine bessere Übersichtlichkeit. Zudem soll es eine neue Extra-Ansicht geben, in dem der Mitarbeiter seine Zeiten sieht, die er noch nicht erfasst hat, obwohl er an diesen Tagen eingeplant war. Hier hat er noch die Möglichkeit, seine Zeiten manuell nachzutragen. Zudem kann der Mitarbeiter seine Zeiten am Monatsende bestätigen, nachdem er alles kontrolliert hat. Sobald er Sie bestätigt hat, kann er keine Änderungen mehr vornehmen. Sobald seine Arbeitszeiten bestätigt wurden, landen diese beim Projekt Manager, nur er kann jetzt noch Änderungen vornehmen. Bis zum nächsten Meeting sollten wir mit der Implementierung starten und schauen, dass wir das Frontend mit dem Backend und der Datenbank verbinden. Um erste Einträge in der Software vorzunehmen. (Nice to have: Mange Project Ansicht: Mitarbeiter

anklicken um Extra Infos zu erhalten, in welchen anderen Projekten der Mitarbeiter noch involviert ist.)

## **10.6 6. Woche Zeitraum 06.11.2023-12.11.2023**

Teambesprechung mit Andre: Protokoll vom 10.11.2023, 14:00 Uhr bis 15:00 Uhr

Die heutige Teambesprechung mit Andre fokussierte sich auf entscheidende Aspekte der Authentifizierung, Softwarearchitektur und Meilensteinplanung. Besonderes Augenmerk wurde auf die SAP-Integration gelegt, wobei unterschiedliche Berechtigungen je nach Rolle diskutiert wurden. Im Bereich der Softwarearchitektur standen die Planung von Meilensteinen sowie die Schnittstellenbeschreibung mittels OpenAPI SWAGGER im Vordergrund. Ein weiterer Schwerpunkt lag auf der technischen Durchführung und Containerisierung, einschließlich der Ankündigung von Meilensteinen und dem Backend-Zugriff auf Datenbanken. Des Weiteren wurde die Vorbereitung für Meilenstein 2 thematisiert, einschließlich der Inhalte und Terminvereinbarungen. Die Besprechung lieferte einen klaren Fahrplan für die Umsetzung der vereinbarten Maßnahmen bis zum nächsten Meeting. Teambesprechung mit Lisa: Protokoll vom 10.11.2023, 15:00 Uhr bis 16:00 Uhr

Im Anschluss an die Besprechung mit Andre stand die Teambesprechung mit Lisa im Fokus. Das Team setzte sich intensiv mit der Gestaltung einer neuen Tabelle in Figma auseinander. Dabei wurden das Modell und die Ansicht der Tabelle überprüft und diskutiert. Besondere Aufmerksamkeit galt der Filterung nach Personen, der Gestaltung der Positionsspalte und der Farbgebung basierend auf der Stundenanzahl. Die Diskussion über die unabhängige Funktionalität und die spätere Verbindung rundete die Besprechung ab. Diese Zusammenfassung bietet einen klaren Überblick über die behandelten Themen und dient als Grundlage für die Umsetzung der festgelegten Maßnahmen bis zum nächsten Meeting.

## **10.7 7. Woche Zeitraum 13.11.2023-19.11.2023**

## References

- [1] Microsoft. (2023) Microsoft open database connectivity (odbc). 14.11.2023. [Online]. Available: <https://learn.microsoft.com/de-de/sql/odbc/microsoft-open-database-connectivity-odbc?view=sql-server-ver16>
- [2] 14.11.2023. [Online]. Available: <https://www.codelessplatforms.com/docs/what-is-odbc/>
- [3] 14.11.2023. [Online]. Available: <https://www.seobility.net/de/wiki/images/f/f1/Rest-API.png>